MEE 390 OER



Nicholas Pohlman
Northern Illinois University



Experimental Design and Evaluation: A structured outline

This text is disseminated via the Open Education Resource (OER) LibreTexts Project (https://LibreTexts.org) and like the thousands of other texts available within this powerful platform, it is freely available for reading, printing, and "consuming."

The LibreTexts mission is to bring together students, faculty, and scholars in a collaborative effort to provide an accessible, and comprehensive platform that empowers our community to develop, curate, adapt, and adopt openly licensed resources and technologies; through these efforts we can reduce the financial burden born from traditional educational resource costs, ensuring education is more accessible for students and communities worldwide.

Most, but not all, pages in the library have licenses that may allow individuals to make changes, save, and print this book. Carefully consult the applicable license(s) before pursuing such effects. Instructors can adopt existing LibreTexts texts or Remix them to quickly build course-specific resources to meet the needs of their students. Unlike traditional textbooks, LibreTexts' web based origins allow powerful integration of advanced features and new technologies to support learning.



LibreTexts is the adaptable, user-friendly non-profit open education resource platform that educators trust for creating, customizing, and sharing accessible, interactive textbooks, adaptive homework, and ancillary materials. We collaborate with individuals and organizations to champion open education initiatives, support institutional publishing programs, drive curriculum development projects, and more.

The LibreTexts libraries are Powered by NICE CXone Expert and was supported by the Department of Education Open Textbook Pilot Project, the California Education Learning Lab, the UC Davis Office of the Provost, the UC Davis Library, the California State University Affordable Learning Solutions Program, and Merlot. This material is based upon work supported by the National Science Foundation under Grant No. 1246120, 1525057, and 1413739.

Any opinions, findings, and conclusions or recommendations expressed in this material are those of the author(s) and do not necessarily reflect the views of the National Science Foundation nor the US Department of Education.

Have questions or comments? For information about adoptions or adaptions contact info@LibreTexts.org or visit our main website at https://LibreTexts.org.

This text was compiled on 05/15/2026

TABLE OF CONTENTS

Preface and Licensing

Introduction

1: Making Measurement Connections

2: Periodic Signals and Data Acquisition

- 2.1: Features and Fundamental Equations
- 2.2: Sampling Experimental Data

3: Uncertainty Analysis

- 3.1: Component Uncertainty
- 3.2: Combining Uncertainty in Function

4: Technical Writing

5: Calibration and Dynamic Response

- 5.1: Fitting with Weighted Least Squares
- 5.2: Dynamic Response from Linear ODEs

6: Fourier Analysis

- 6.1: Descriptions
- 6.2: Conversion of Function
- 6.3: Discrete data conversion and filtering

7: Probability and Statistics

- 7.1: Contextual background resulting in histogram
- 7.2: Central moments of probability density function
- 7.3: Probability from Gaussian distribution
- 7.4: Analysis with finite statistics

8: Laboratory Exercises

- 8.1: Digital data sampling
- 8.2: Stress and strain
- 8.3: Temperature and Heat Transfer
- 8.4: Fluid Flow and Forces

9: Supplementary Material Section

- 9.1: Chapter 1 Supplementary Reading Materials
- 9.2: Chapter 2 Supplementary Reading Materials
  - 9.2.1: Discrete-Time Signal Processing
    - 9.2.1.1: Introduction to Digital Signal Processing

- 9.2.1.2: The Sampling Theorem
- 9.2.1.3: Amplitude Quantization
- 9.2.1.4: Discrete -Time Signals and Systems
- 9.2.2: Analog-to-Digital-to-Analog Conversion
  - 9.2.2.1: Introduction to Analog to Digital to Analog Conversion
  - 9.2.2.2: The Sampling Theorem
  - 9.2.2.3: Resolution and Sampling Rate
  - 9.2.2.4: Digital-to-Analog Conversion Techniques
  - 9.2.2.5: Analog-to-Digital Conversion
  - 9.2.2.6: Extended Topic - Digital Signal Processing
  - 9.2.2.7: Summary
  - 9.2.2.8: Problems
- 9.2.3: Sampling and Reconstruction
  - 9.2.3.1: Signal Sampling
  - 9.2.3.2: Sampling Theorem
  - 9.2.3.3: Signal Reconstruction
  - 9.2.3.4: Perfect Reconstruction
  - 9.2.3.5: Aliasing Phenomena
  - 9.2.3.6: Anti-Aliasing Filters
  - 9.2.3.7: Discrete Time Processing of Continuous Time Signals
- 9.3: Chapter 3 Supplementary Reading Materials
  - 9.3.1: Importance of Statistics and Experimental Measurements
  - 9.3.2: Total Uncertainty, Bias Error, and Propagation of Error
- 9.4: Chapter 4 Supplementary Reading Materials
- 9.5: Chapter 5 Supplementary Reading Materials
  - 9.5.1: Undamped Free Vibrations
  - 9.5.2: Viscous Damped Free Vibrations
  - 9.5.3: Friction (Coulomb) Damped Free Vibrations
  - 9.5.4: Undamped Harmonic Forced Vibrations
  - 9.5.5: Viscous Damped Harmonic Forced Vibrations
- 9.6: Chapter 6 Supplementary Reading Materials
  - 9.6.1: Introduction to the Frequency Domain
  - 9.6.2: Complex Fourier Series
  - 9.6.3: Classic Fourier Series
  - 9.6.4: Fourier Series Approximation of Signals
  - 9.6.5: Fourier Transform
  - 9.6.6: Sampling and Reconstruction
    - 9.6.6.1: Signal Sampling
    - 9.6.6.2: Sampling Theorem
    - 9.6.6.3: Signal Reconstruction
    - 9.6.6.4: Perfect Reconstruction
    - 9.6.6.5: Aliasing Phenomena
    - 9.6.6.6: Anti-Aliasing Filters
    - 9.6.6.7: Discrete Time Processing of Continuous Time Signals
- 9.7: Chapter 7 Supplementary Reading Materials
  - 9.7.1: Design of Experiments
    - 9.7.1.1: Design of Experiments via Taguchi Methods - Orthogonal Arrays
    - 9.7.1.2: Design of experiments via factorial designs

- 9.7.1.3: Design of Experiments via Random Design
- 9.7.1.4: Summary- Summary on Control Architectures' philosophies, advantages, and disadvantages.
- 9.8: Chapter 8 Supplementary Reading Materials

Index

Glossary

Detailed Licensing

Detailed Licensing

Preface and Licensing

About this Open Educational Resource (OER)

This book was created by Nicholas A. Pohlman at Northern Illinois University for *Experimental Methods in Mechanical Engineering* and was made possible through an Illinois Secretary of State/Illinois State Library (ISL) OER Grant awarded to Northern Illinois University Libraries' project: *Huskies Unbound: Developing OER for Undergraduate Success in High-Impact Courses*. Funded by a one-time $3 million allocation to the Illinois Secretary of State's Budget in Fiscal Year 24, and facilitated by the Consortium of Academic and Research Libraries in Illinois (CARLI), the ISL OER Grant Program was created to support the development of textbooks and ancillaries as open educational resources to reduce students' costs, encourage student success, and allow faculty to adapt materials to meet local needs.

I wish to express my gratitude to Illinois Governor JB Pritzker, Illinois Secretary of State Alexander Giannoulias, and Greg McCormick, Director of the Illinois State Library, for their commitment to Open Education.

Acknowledgements

The creation of this OER was made possible through the collaborative efforts of educators, librarians, and staff from Northern Illinois University. In particular, I acknowledge grant project members Jaime Schumacher, Missy Clapp, Cynthia Paralejas, Tracy Miller, and Lindsay Vreeland, whose time and expertise were invaluable to its development. I also extend my gratitude to Larissa Garcia and Dee Anna Phares, as co-directors of the grant, for their instrumental contributions in making this project possible and their management throughout the process.

In addition, I offer my appreciation for the training and support provided by Michele Leigh, Elizabeth Clarage, and Nicole Swanson and other CARLI staff members.

Finally, I wish to thank the 32 students enrolled in the Fall 2025 semester course at NIU who served as initial users and editors who helped improve the product for other undergraduates to use as a resource.

Peer Review

The Peer Reviewer process was completed by the following reviewer(s):

- Full name, Rank, Department or Field, Institution

Customization/Terms of Use

This textbook is licensed under a Creative Commons Attribution 4.0 International (CC-BY) license unless otherwise indicated, which means that you are free to:

- SHARE – copy and redistribute the material in any medium or format
- ADAPT – remix, transform, and build upon the material for any purpose

Licensing Terms

The licensor cannot revoke these freedoms as long as you follow the license terms.

- Attribution: You must give appropriate credit, provide a link to the license, and indicate if any changes were made. You may do so in any reasonable manner, but not in any way that suggests the licensor endorses you or your use.
- No Additional Restrictions: You may not apply legal terms or technological measures that legally restrict others from doing anything the license permits.
- Notice: You do not have to comply with the license for elements of the material in the public domain or where your use is permitted by an applicable United States copyright exception or limitation.
- No Warranties are Given: The license may not give you all of the permissions necessary for your intended use. For example, other rights such as publicity, privacy, or moral rights may limit how you use the material.

Attribution

Some of the content for this textbook may have been adapted from other open educational resources. For specific reference information about what was used and/or changed in this adaptation, please refer to the references at the end of each section of the book.

Suggested Attribution Statement for this Book

Content that is not adapted from other OER should include the following attribution statement:

Pohlman, Nicholas A. (2026). *Experimental Design and Evaluation: A Structured Outline.* Illinois Secretary of State/Illinois State Library OER Grant: Northern Illinois University. Licensed under CC BY 4.0.

References

This Preface is a derivative of Labatte, Jessica and Larissa Garcia. (2024). Illinois SCOERs. *Fundamentals of Photography and Creative Practice* by Consortium of Academic and Research Libraries in Illinois licensed under CC BY 4.0.

*A detailed breakdown of this resource's licensing can be found in **Back Matter/Detailed Licensing**.*

Introduction

Contextual Background

The author of this text has served as instructor for the Northern Illinois University course *Experimental Methods in Mechanical Engineering* numerous times over two decades. In the previous publishing environment, there was a reliance upon foundational texts for students to use as their reference resource:

- *Theory and Design for Mechanical Measurements* by Richard S. Figliola and Donald E. Beasley (current edition)
- *Experimental Methods for Engineers* by Jack Holmann (pdf download)
- *Introduction to Engineering Experimentation* by Anthony J. Wheeler and Ahmad R. Ganji (latest edition)
- *Measurement and Data Analysis for Engineering and Science* by Patrick F. Dunn (with Michael P. Davis contributing to the 4th edition) (full edition history)

Despite having the full background for derivation and application to numerous examples from the textbooks, students often found the lecture-based outline notes as a more efficient resource when analyzing engineering problems within the lab, through homework practice, and during open-resource examinations. In a lecture delivery environment, the hand-written notes that were built by the instructor were transcribed by the students at a frenetic pace; experience indicates, however, that copying and learning are not necessarily parallel processes. Through the pivot to online instruction, the paper-based notes were converted to static, portable document formats that proved to serve as the effective text resource for students. To that end, the outline in this Open Education Resource is provided as an aid to a broader audience.

How to Use the Outline

The text that follows is ordered in a manner that helps usher students through the foundational building blocks of experimental data collection and subsequent analysis. Limited prose is provided so that terminology or importance of elements are not hidden within sentence structures. Furthermore, the applicable forms of equations for use is provided; the full background of derivations from fundamental mathematical principles are hyperlinked to other external resources. The intent for the structure is to keep focus on the *How to use* ... process of the engineer while still having connections to building the knowledge from first principles.

As an example, the strain gage is a tool for measuring the local elongation of a solid sample material. In order to load and deflect the mechanical bonds, the understanding of static force reactions, mechanics of materials, mathematical principles of area moment of inertia, and resistive electronics all need to be connected. As each of those knowledge elements are connected (most curricula have those elements as four separate prerequisite courses), the resource links are provided to remind the reader of the prerequisite content that should exist in their knowledge base. If there is not an immediate recall, then the reader can start again from the first principles to build to application in the laboratory experiments.

A standard outline structure would follow a sequence of numerical and alphabetical formats: Roman numerical -> Capital letter -> Arabic numeral -> Lower case letter -> numerical value, and so forth. An old example from handwritten notes would be:

- VI. Periodic Signal Analysis, C. Special cases of Fourier Series, 3. Odd functions, b. thus all $A_n = 0$

Given the LibreText platform, readers will find the Chapter numbers now replace the roman numeral format. Therefore, when advancing from pages that are organized by the decimal values, the sequence of capital letters will be a continuum within the chapter itself. Using the outline format hopefully enables more efficient cross referencing both within the text as well as when readers may seek clarification from instructors on the meaning or applicability (e.g., "Can you explain what is meant by 4-quadrant inverse tangent of 2.B.b when describing phase shift?")

Organizational Order

Students tossed into a laboratory environment can sometimes feel overwhelmed.

- How does this connect to any sort of theory or equation that has not be "taught" to me recently?
- How should I organize planning, recording, and reporting?
- How do I convert a blank page into a formatted technical document ready for the right audience?

Therefore the sequence attempts to build from Blooms Taxonomy of Learning: readers will be reminded of their prior understanding, apply the principle in a (potentially) new analytical method through some completed examples, and then be evaluated via instructor-derived assessments of examinations or laboratory reports. The overarching sequence of content follows:

1. Organizing the experimental plan; useful questions for independent, learner-initiated projects
2. Sampling and storing time-dependent signals
3. Reporting uncertainty for specific sensors and combinations of multiple measurements
4. Organizing a writing process for technical documents
5. Characterizing linear differential equations in the laboratory
6. Signal analysis via Fourier Series and conversion to frequency domain
7. Fundamental principles of infinite and finite statistics through probability density

The theoretical background to conduct experiments related to many mechanical systems is provided in Chapter 8. These and other experiments can be implemented in any order to emphasize particular sections above as a parallel in-the-lab learning activity.

The Appendix in Chapter 9 concludes with the page contents from other OER where greater expansion on topics is available without interrupting the outline structure. Each section may have different Creative Commons license depending on authorship.

Finally, an advance thank you to all who read, find the work helpful in their learning, and provide feedback for continued revision and improvement.

1: Making Measurement Connections

Learning Objectives

- Willingness to ask questions (often repeatedly) to increase knowledge and understanding

A. Why conduct experiments?

Exercise 1.1

Students suggest their own reasons to conduct experiments:

Answer

- Test
- Evaluate
- Explore
- Confirm quality
- Discover

B. Why here in conjunction with Capstone project? Why as a junior or senior in engineering curriculum?

Exercise 1.2

Students indicate connectivity to their current status

Answer

- Sufficient foundation
- Confirmation of knowledge
- Preliminary project management
- Practice communication

C. How should the process of experiment be sequenced? There are many questions in each phase:

1. Planning
   - Does variation alone create an experiment (i.e., changing parameters in simulation)?
   - What outcomes will make for a successful experiment (regardless of data collected)?
   - Should the approach include redundancy, significant precision, testing outside nominal conditions, etc.?
2. Identifying system components
   - What is the fundamental principle and how is it observable?
   - Can the sensors reach the experimental region? AND survive the chemical/thermal/vibrational/structural environment?
   - How will the signals reach the signal conditioner to display/record digital values?
   - How will the presence of sensors affect the physical principle?
3. Assessing performance
   - What is the correlation of the independent variable to the dependent outcome and measurand?
   - To what standard is the device calibrated?
   - Is uncertainty based on component, combination, and/or statistics?
4. Setting signal sampling
   - Is the signal voltage, current, displacement, luminosity?
   - What makes for a sufficient data set? Consider both individual trials as well as overall system.

- What sampling rate has sufficient accuracy without excessive storage requirements?
- How does the combination of frequencies (input, output, sampling) influence data?

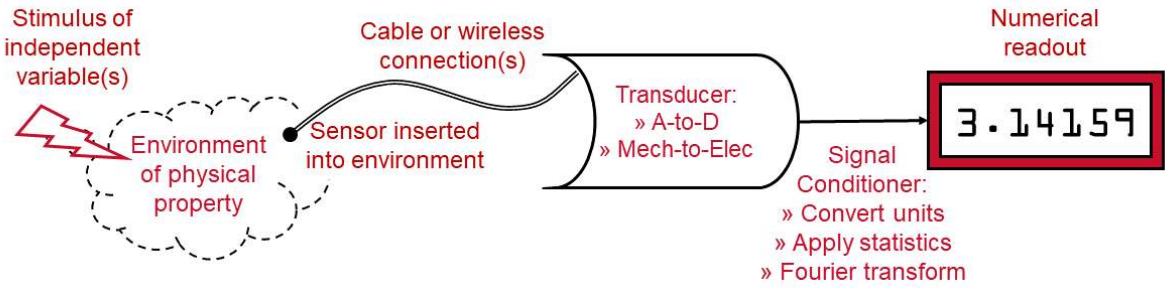
5. Analyze results
   - What analytical equations characterize the results?
   - How do the statistical approaches impact the confidence in the reported result?
   - Does the trend of input-to-output correlate based on the functional relationship?
   - What is the comparison with theory and/or other experiments?
   - Is uncertainty constant throughout or measurement dependent?
   - What are criteria for rejecting spurious data? Is the method of rejection reported?
   - If a hypothesis is proposed before experimentation, does the result confirm or reject the hypothesis?
6. Report (written and verbal)
   - What audience will review the experiment?
   - What level of detail is necessary for each section to:
     1. demonstrate the learning outcome?
     2. prove capability of the sensing device?
     3. repeat experiment for the non-expert?
     4. convince decision-maker of next steps?
   - What are the proper numbers, units, and uncertainty expressions?
   - How are you organizing the content to tell the story effectively?
   - Will it be reviewed in a static written form or in a dynamic oral presentation?

Answers from any question can *iteratively update* prior ones.

D. Useful definitions

1. Experimental Classification
   a. Variational - quantify functional relationship of input changes to output observations; calibrating a standard
   b. Validational - test hypothesis to validate or refute; improve existing theory
   c. Pedagogical - teach/demonstrate established concept
   d. Explorational - pursue first steps of idea/theory
2. Carefully crafted terms
   a. Parameters - fixed values for whole experiment
   b. Variable - something that undergoes change
   c. *Independent* variable - term that is manipulated by the experimenter
   d. *Dependent* variable - physical response of some sort
   e. **Measurand** - only those variables directly observed by a sensor

E. General Measurement Model



Exercise 1.3

Describe the use of the Extech HD350 Pitot Probe Anemometer in context of the General Measurement Model



Answer

Physical environment is the smaller cross section area of the wind tunnel (as compared to a large inlet area).

Numerous independent **variables** could be modified:

- Wind tunnel speed setting
- Spatial location of the probe
- Shape or position of upstream object deflecting air flow

A pitot probe is **mounted within the test section** through a small opening.

Black and white **tubes connect** the stagnation and static pressure ports to the handheld transducer; sometimes may be wires for electrical-based sensor.

Pressure differences **converted by transducer** to a numerical value.

Options for **signal conditioning**:

- Select units: meters per second, miles per hour, knots, etc.
- Volume flow rate: input cross-sectional area
- Sampling rate: real-time digital update, moving average of N data points

Numerical readout can be stored in handwritten notebook or digital data acquisition file.

F. Units of measure

1. Quantity is represented by three properties
   a. Dimension (fundamental physical description)
   b. Unit (from accepted standard, i.e. SI, Imperial)
   c. Magnitude (actual numerical value)
2. Fundamental dimensions are based on system of units

Order	Name	Representation	Typical Units
1.	Mass	[M]	kg, lb_m
2.	Length	[L]	meters, feet, miles
3.	Time	[T]	seconds, days
4.	Temperature	[$\mathcal{T}$]	Kelvin, Rankine
5.	Current	[I]	Ampere
6.	Light	[C]	candela, lumen
7.	Matter	[n]	mol

- Fundamentals are used in combination to create secondary dimensions (e.g., torque is equivalent to mass x length^2 x time^{-2}).

3. How to maintain universal consistency? (systems of units)
   a. Originally based on artifacts Historical context
   b. Now scientifically repeatable
   c. Standards
      - Length: meter - distance light travels in vacuum in $\frac{1}{299,792,258}$ seconds
      - Temperature: Kelvin - $\frac{1}{273.16}$ of triple point of water
      - Mass: WAS an artifact The kg is dead
      - Time: second - 9,912,631,700 cycles of Cesium radiation
4. Hierarchy of standards
   a. Traceable path relating unit to standard
   b. Allows trading of commodities with consistent magnitudes of units
   c. Calibration standard is typically 1 part-per-billion
      - Thickness of 2 sheets of paper to a football field
5. Convenient or unique conversion factors
   - 25.4 mm = 1 inch
   - 1 kW = 1.34102 hP
   - 5,280 feet = 1 mile
   - 1 smoot = 1.70 meter
   - 1 sydharb = 500 gigalitres
   - 1 beard second = 1 angstrom

1: Making Measurement Connections is shared under a CC BY-NC 4.0 license and was authored, remixed, and/or curated by LibreTexts.

CHAPTER OVERVIEW

2: Periodic Signals and Data Acquisition

The fundamental features of periodic signals are introduced along with combination of mechanisms to characterize. The associated discrete step sampling is introduced along with mechanism to avoid common errors in observation.

Learning Objectives

- Convert sine and cosine with appropriate frequency, amplitude, and phase
- Calculate preliminary statistics of the range of data
- Prepare to collect future data recognizing frequency aliasing, analog-to-digital conversion, and other elements

2.1: Features and Fundamental Equations
2.2: Sampling Experimental Data

For additional material from sources external to NIU, please visit the Appendix.

2: Periodic Signals and Data Acquisition is shared under a CC BY-NC 4.0 license and was authored, remixed, and/or curated by LibreTexts.

2.1: Features and Fundamental Equations

A. Classification

1. Deterministic: written functional form of independent variable t

$$x(t) = \alpha t^n \mathrm{e}^{-t/\beta} \tag{2.1.1}$$

2. *Adjective* Periodic: repeats at some fundamental frequency

   a. **Simple**: implies only a single frequency

   b. **Complex:** multiple frequencies are possible including through summation:

$$y(t) = D_0 + \sum_i C_i \cos(2\pi f_i t + \phi_i) \tag{2.1.2}$$

B. Combining periodics

1. Parameters of general simple period equation

$$y(t) = D_0 + C\sin(\omega t + \phi) \tag{2.1.3}$$

   a. D_0 is the mean offset from zero

   b. C is the amplitude of oscillation

   c. ω circular frequency [rad/s]

   d. f cyclic frequency [rev/time] ($f = \frac{\omega}{2\pi}$)

   e. T period is the inverse of cyclic frequency [time] ($T = 1/f$)

   f. ϕ phase offset (left-right shift)

   g. Huskie Commons Link to unscripted video with instruction and student interpretation

2. Combining sines and cosines into one

$$A\cos(\omega t) + B\sin(\omega t) + D_0 = C\sin(\omega t + \phi^*) + D_0 = C\cos(\omega t - \phi) + D_0 \tag{2.1.4}$$

   a. C is the norm of the amplitudes: $C = \sqrt{A^2 + B^2}$

   b. Phase is a 4-quadrant inverse tangent

$$\phi = \tan^{-1}\left(\frac{B}{A}\right) \tag{2.1.5}$$

$$\phi^* = \tan^{-1}\left(\frac{A}{B}\right) \tag{2.1.6}$$

   *sign of numerator and denominator are crucial to get values beyond $\frac{-\pi}{2} \le \phi \le \frac{\pi}{2}$

   c. D_0 is the mean offset (up or down depending on positive or negative sign)

C. Statistical features

1. Mean: average value over specific time or data range

   a. Continuous form starting at t_1 and finishing at t_2

$$\bar{x} = \frac{\int_{t_1}^{t_2} x(t)dt}{\int_{t_1}^{t_2} dt} \tag{2.1.7}$$

   b. Discrete is the standard version from list of x_i values

$$\bar{x} = \frac{1}{N}\sum_{i=1}^{N} x_i \tag{2.1.8}$$

   c. Be wary of the time range on dynamic signal (major part of some laboratory assignments)

2. Root mean square (RMS): indicates variation of the mean

   a. Continuous Signal

      i. The square $x(t)^2$

ii. Of which a mean is calculated $\frac{\int_{t_1}^{t_2} x(t)^2 dt}{\int_{t_1}^{t_2} dt}$

iii. Then rooted $\sqrt{\frac{\int_{t_1}^{t_2} x(t)^2 dt}{\int_{t_1}^{t_2} dt}}$

iv. In the simplified form

$$x_{rms} = \sqrt{\frac{1}{t_2 - t_1} \int_{t_1}^{t_2} x(t)^2 dt} \tag{2.1.9}$$

b. Similar in discrete form

$$x_{rms} = \sqrt{\frac{1}{N} \sum_{i=1}^{N} x_i^2} \tag{2.1.10}$$

c. RMS is NOT the standard deviation when $\bar{x} \neq 0$

i. Standard deviation is *relative* to the mean $\bar{x}$

$$x_{std} = \sqrt{\frac{1}{t_2 - t_1} \int_{t_1}^{t_2} \left[x(t) - \bar{x}\right]^2 dt} = \sqrt{\frac{1}{N} \sum_{i=1}^{N} (x_i - \bar{x})^2} \tag{2.1.11}$$

D. Some Example Problems

? Exercise 2.1.1

1. Convert the following into a cosine only and a sine only functions with the appropriate phase:

$$x(t) = -5\sin(10\pi t) + 8\cos(10\pi t)$$

Produce a plot of at least three periods using all three functions at different timesteps or markers to prove each are equivalent.

Answer

The amplitude will be the norm of the two amplitudes $C = \sqrt{(-5)^2 + 8^2} = 9.43$.

The phase will depend on whether it is cosine or sine.

$\phi = \tan^{-1}(\frac{-5}{8}) = -0.559$ and $\phi^* = \tan^{-1}(\frac{8}{-5}) = 2.13$

which results in

$$x(t) = 9.43\cos(10\pi t + 0.559) = 9.43\sin(10\pi t + 2.13)$$

With cyclic frequency of 5~Hz, the time range for the plot would need to be$0 \leq t \leq \frac{3}{5}$

? Exercise 2.1.2

Using the function$x(t)$in the prior Problem, calculate the mean for analog (integrate) and discrete (summation) methods for the following time intervals where$T$is the period of the original function:

- $0 < t < 3T$~sec
- $1.5T < t < 4T$~sec
- $0 < t < 1.1T$~sec

Interpret how the mean value changes as a function of the portion of the period included in the integration/summation range.

Answer

- $0 < t < 3T$ sec: The period$T = \frac{2\pi}{\omega} = \frac{2\pi}{10\pi} = 0.2$. The continuous integral will be

$$\bar{x} = \frac{1}{0.6-0}\int_0^{0.6} -5\sin(10\pi t) + 8\cos(10\pi t)dt \tag{2.1.12}$$

$$\bar{x} = \frac{5}{0.6(10\pi)}\cos\left(10\pi t|_0^{0.6} + \frac{8}{0.6(10\pi)}\sin\left(10\pi t|_0^{0.6}\right.\right. \tag{2.1.13}$$

$$\bar{x} = \frac{5}{2\pi}(1-1) \tag{2.1.14}$$

$$\bar{x} = 0$$

The discrete mean is straightforward, but will depend on whether you include the final timestep at$t_{max} = 0.2$. If using the functional values to the end and$\delta t = 0.001$~sec up yields a mean value$\bar{x} = 0.0398$whereas using only up to$t_{max} = 0.199$~s makes$\bar{x} = 9.23 \times 10^{-16}$(which is machine precision \ldots). The zero value is because the range was over a full period of the original signal.

- $1.5T < t < 4T$ sec: In continuous form

$$\bar{x} = \frac{1}{0.8-0.3}\int_{0.3}^{0.8} -5\sin(10\pi t) + 8\cos(10\pi t)dt = 0.637$$

and in discrete form$\bar{x} = 0.621$. Note that the fewer periods means that the partial period will shift the mean slightly.

- $0 < t < 1.1T$ sec: In continuous form

$$\bar{x} = \frac{1}{0.22-0}\int_0^{0.22} -5\sin(10\pi t) + 8\cos(10\pi t)dt = 0.542$$

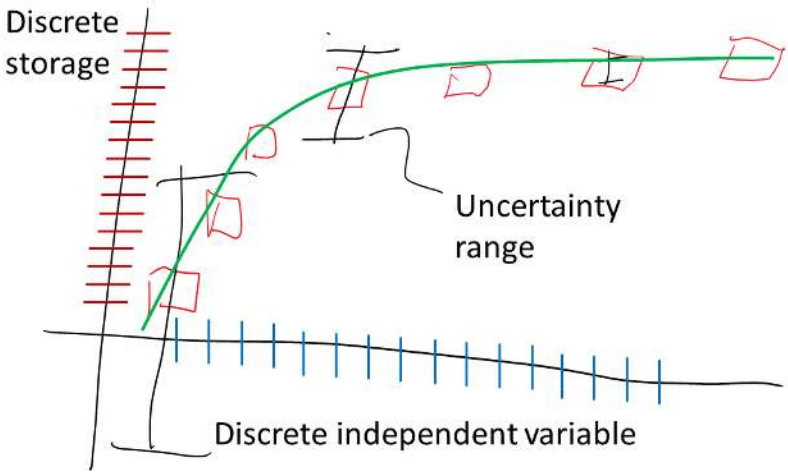
and in discrete form$\bar{x} = 0.552$. The very narrow window and discrete result creates the numerical difference. IfΔt were decreased it would more closely match the continuous form.

2.1: Features and Fundamental Equations is shared under a CC BY-NC 4.0 license and was authored, remixed, and/or curated by LibreTexts.

2.2: Sampling Experimental Data

E. Digital Data Acquisition (DAQ)

1. Digital form has inherent discrete, quantization of analog result
   a. Sampling frequency
   b. Resolution of digital conversion
   c. Sketch of continuous form of signal (green), but where DAQ *stores* particular data points (red markers)



2. Under sampling errors
   a. Sampling frequency $f_s = \frac{1}{\delta t}$
      i. $\downarrow f_s$ yields less data over time
      ii. Misrepresents form of the data (see Frequency_Sampling.m Matlab script)
   b. Shannon's sampling theorem (longer video description)
      i. Sample rate must be at least twice ($2\times$) the maximum expected frequency
      ii. Mathematically $f_s \geq 2f_{max}$ or in terms of time step $\delta t \leq \frac{1}{2f_{max}}$
   c. Aliasing frequencies occur from undersampling
      i. When Shannon's sampling theorem is not satisfied, higher frequencies appear as lower frequency signals.
      ii. Consider simple periodic form without DC offset

$$y(t) = C\sin(2\pi ft + \phi) \tag{2.2.1}$$

      with signal stored by DAQ at δt intervals

$$t = [0, \delta t, 2\delta t, \ldots, r\delta t] \tag{2.2.2}$$

      iii. Discrete form of the data at $r\delta t$ time step:

$$y(r\delta t) = C\sin(2\pi fr\delta t + \phi) \tag{2.2.3}$$

      where r is just integer number of step ($r = [0, 1, 2, ...]$)
      iv. Assume an integer multiplier of 2π in the form of mr is added to argument

$$y(r\delta t) = C\sin(2\pi fr\delta t + \phi + mr2\pi) \tag{2.2.4}$$

      output will not change due to periodicity.
      v. Collect the common terms for factoring

$$y(r\delta t) = C\sin\left(2\pi r\delta t\left(f + \frac{m}{\delta t}\right) + \phi\right) \tag{2.2.5}$$

since $2\pi r\delta t$ is common in both equations, the fundamental frequency f will be undistinguishable from alias frequency $f_a = f + \frac{m}{\delta t}$

vi. Folding point of aliasing is Nyquist frequency

- $f_N = \frac{f_s}{2} = \frac{1}{2\delta t}$
- Any original frequency f of the signal greater than Nyquist frequency $f > f_N$ will appear as lower frequency information.
- Nyquist folding diagram
- Other OER available resource

? Exercise 2.2.1

AC voltage oscillates at 60~Hz but sampled at 90~Hz. What are the Nyquist and aliasing frequencies?

Answer

Determine Nyquist frequency: $f_N = \frac{f_2}{2} = \frac{90}{2} = 45$ Hz

Is $f > f_N$)?: Yes, aliasing possible

Determine ratio of input to Nyquist: $\frac{f}{f_N} = \frac{60}{45} = 1.33$

Use ratio on folding diagram: $1.33 f_N$ drops to $0.67 f_N$ on bottom line

Calculate aliasing frequency: $f_a = 0.67(45) = 30$ Hz

Methodology can repeat for any samples up to 120 Hz when Nyquist frequency is satisfied

F. Analog to Digital Conversion error

1. Resolution of complex function is the result of bit resolution in analog-to-DC.
2. Resolution error is the smallest change to least-significant-bit:

$$Q = \frac{E_{fsr}}{2^m} \tag{2.2.6}$$

where E_{fsr} is the full range from minimum to maximum of signal and m is number of bits used in digital form.

3. Greater bits increases resolution at the cost of increasing hardware.

? Exercise 2.2.2

An A/D converter with 7-bit register on a full-scale output of 0.0-3.90 V converts analog values of $\sqrt{3}$ and $\sqrt{11}$ to a digital form. What is the binary number that the computer stores in its memory for both of these digital conversions? What is the actual decimal output with associated uncertainty of resolution?

Answer

The size of the least significant bit step is $\delta v = \frac{3.9-0.0}{2^7} = 0.03046875$~V.

For $\sqrt{3}$, the integer number of steps will be *floor*($\sqrt{3}/\delta v$) = 56 steps. Therefore, the binary number of the integer will be 56 = 011 1000. The stored value of the 7-bit computer will be $\sqrt{3} = 1.7062 \pm 0.0152$

For comparison to $\sqrt{11}$,

- *floor*($\sqrt{11}/\delta v$) = 108 steps
- 108 = 110 1100 in binary
- $\sqrt{11} = 3.290 \pm 0.0152$ with 7 bits with same uncertainty resolution

4. Examples of cost comparison of DAQ devices

a. National Instruments
b. LabJack
c. Measurement Computing
d. Contec
e. Note price variance depending on number of channels, sample rates, bit resolution, and connection type

---

2.2: Sampling Experimental Data is shared under a CC BY-NC 4.0 license and was authored, remixed, and/or curated by LibreTexts.

CHAPTER OVERVIEW

3: Uncertainty Analysis

Every measurement collected has some combination of true value within and some associated uncertainty. The methodology for building from resolution of individual data points, through instruments used, and then integration of multiple measurements into a total result are covered in this chapter. The subsequent methodology provides the mechanism to put bounds (i.e. error bars) on all future data results.

Learning Objectives

- Recognize measurement uncertainty is $\pm\frac{1}{2}$ the resolution of the sensor reading
- Integrate resolution uncertainty with combination of instrument uncertainty(ies) through root sum square
- Calculate sensitivity coefficient of functional relationship via the partial derivative to present overall uncertainty of a result

3.1: Component Uncertainty
3.2: Combining Uncertainty in Function

For additional material from sources external to NIU, please visit Appendix

3: Uncertainty Analysis is shared under a CC BY-NC 4.0 license and was authored, remixed, and/or curated by LibreTexts.

3.1: Component Uncertainty

A. Errors are properties of all measurements

1. $\hat{x}_{\text{gage}} = x_{\text{actual}} + \epsilon$, where ϵ arises from contributing error sources
2. Results should be presented as value ± uncertainty range

B. Categories of measurement error

1. Systematic error – constant offset to the true value
2. Random errors – variations about the measurand value
3. Comparison of types:

Systematic		Random
Not determined statistically		Scatter of data measurements allows statistics
Estimated via comparison methods		Affected by repetition and resolution
1. Calibration		
2. Comparison to other references		
3. Engineering experience		

C. Measurand uncertainty analysis

▼ Details

1. Identifies the ± range for individual sensor and resulting measurand
2. Like magnitude of a rank $\mathbb{R}$ vector, magnitude determined by root-sum-square:

$$u_r = \pm\sqrt{u_0^2 + u_1^2 + u_2^2 + ... + u_J^2} \tag{3.1.1}$$

   a. Since it's a summation → all units must match (e.g., cannot add ± 0.03 cm and 3% of measurement)
   b. Assumes unit sensitivity where none are more important than others
   c. Built from two types: *zero-order* and *instrument* uncertainty
3. Zero-order uncertainty
   a. Based on resolution (physical divisions of transducer stage)
   b. Variations create random scatter above/below scale value
   c. Equation: $u_0 = \pm 0.5 \times \text{resolution scale}$
   d. Explains importance of significant figures in science
4. Instrument errors
   a. Determined through calibration methods or manufacturer specs
   b. Types:
      i. Hysteresis: approaching value from increasing or decreasing rate
      ii. Linearity: consistency of slope
      iii. Sensitivity: impact of small and large disturbances
      iv. Repeatability: reach same value over and over again?
      v. Zero-shift: positive or negative bias near the lowest numerical values
      vi. Stability: consistency when independent variable is constant
      vii. Thermal drift: temperature dependence of sensor
   c. Equation:

$$u_I = \sqrt{u_H^2 + u_S^2 + ...} \tag{3.1.2}$$

5. Total resulting uncertainty:

$$u_r = \pm\sqrt{u_0^2 + u_I^2} \tag{3.1.3}$$

? Exercise 3.1.1

6. A vehicle speedometer is graduated in 5-mph increments. The OEM indicates instrument accuracy of ±4% of the reading.

   If the driver claims operating the vehicle at 60 mph, what is the associated uncertainty?

   Answer

   The resolution uncertainty is half the increment size: $u_0 = \pm\frac{1}{2}5 = \pm 2.5$ mph

   The instrument uncertainty is a calculation: $u_I = \pm 0.04(60) = \pm 2.4$ mph; since no other instrument uncertainties present, there is no need to find a magnitude via the root-sum-square.

   The resulting uncertainty is the combination of both: $u_r = \pm\sqrt{2.5^2 + 2.4^2} = \pm 3.5$.

   ... I swear officer, I could not be going anything more than 63.5 miles per hour

7. **Important distinctions** (when reading words):
   a. % *of the measurement* → multiply result by percentage
   b. % *FSO* (full scale output) → use full sensor range

3.1: Component Uncertainty is shared under a CC BY-NC 4.0 license and was authored, remixed, and/or curated by LibreTexts.

3.2: Combining Uncertainty in Function

D. Total combined uncertainty

1. Many results are a function of multiple measurands and variables with variety of units:

$$r = f(x_1, x_2, x_3, \dots, x_n) \tag{3.2.1}$$

2. Example scenarios:
   a. Strain: $\varepsilon = \frac{P}{AE_m}$
   b. Density: $\rho = \frac{m}{\pi R^2 L}$
   c. Natural frequency: $f = \sqrt{\frac{L}{g}}$
3. Cannot combine dissimilar units directly via root sum square:

$$u_\varepsilon \neq \pm\sqrt{u_P^2 + u_A^2 + u_{E_m}^2} \tag{3.2.2}$$

   since the measurands each have unique dimensions
4. Variance of result is defined:

$$\sigma_r^2 = \lim_{N\to\infty}\left(\frac{1}{N}\sum_{i=1}^{N}[r_i - r']^2\right) \tag{3.2.3}$$

5. Use Taylor Series to expand arguments:

$$r_i - r' = \sum(x_{k,i} - x_k')\frac{\partial r}{\partial x_k} + \sum(x_{k,i} - x_k')^2\frac{\partial^2 r}{\partial x_k^2}\dots \tag{3.2.4}$$

6. Assuming first first order of expansion and substitute variance definitions:

$$\sigma_{x_k}^2 = \lim_{N\to\infty}\frac{1}{N}\sum(x_{k,i} - x_k')^2 \tag{3.2.5}$$

$$\text{Covariance: } \sigma_{x_1x_2} = \lim_{N\to\infty}\frac{1}{N}\sum(x_{1,i} - x_1')(x_{2,i} - x_2') \tag{3.2.6}$$

7. Combined result variance:

$$\sigma_r^2 = \sum\left(\frac{\partial r}{\partial x_i}\right)^2\sigma_{x_i}^2 + 2\sum\frac{\partial r}{\partial x_i}\frac{\partial r}{\partial x_j}\sigma_{x_ix_j} \tag{3.2.7}$$

8. Replacing variance with uncertainty:

$$u_r^2 = \sum\left(\frac{\partial r}{\partial x_i}\right)^2 u_{x_i}^2 + 2\sum\frac{\partial r}{\partial x_i}\frac{\partial r}{\partial x_j}u_{x_i}u_{x_j} \tag{3.2.8}$$

9. **Total Combined Uncertainty:**

$$u_r^2 = \sum_{i=1}^{J}\theta_i^2u_i^2 + 2\sum_{i=1}^{J-1}\sum_{j=i+1}^{J}\theta_i\theta_ju_iu_j \tag{3.2.9}$$

   Where senstivity coefficient is $\theta_i = \frac{\partial r}{\partial x_i}$ in order to scale the measurand to the dimension of the result r
10. Magnitude of contribution is product of both θ_iu_i
11. When independent sensors are used, no covariance exists making product of uncertainties: $u_iu_j = 0$

? Exercise 3.2.1

E. A project team was estimating major head loss along a pipe length using the fundamental equation:

$$h_L = \left(\mathcal{F}\frac{L}{D}\right)\frac{v^2}{2g} \tag{3.2.10}$$

where the velocity was measured from a volumetric flow meter $Q = v\frac{\pi D^2}{4}$. The following measurements and resolutions were collected:

- Diameter $D = 0.75$ in with $\frac{1}{16}$inch resolution
- Volume flow rate of $Q = 65$ L/min with resolution of 5 L/min (necessary conversion 1 L = 61in^3)
- Gravity is estimated with resolution at 32.2 ft/sec^2
- Pipe length was $L = 40.25$ in with same $\frac{1}{16}$ inch resolution
- Reynolds limit was reached so friction coefficient $\mathcal{F} = 0.012$ was assumed a constant with no uncertainty

Determine the head loss in inches and an estimate of the uncertainty. Which measurand has the least impact on the uncertainty?

Answer

One could substitute for velocity right away or calculate uncertainty of Q separately; either way both will result in the same conclusion:

$$h_L = \left(\mathcal{F}\frac{L}{D}\right)\frac{8Q^2}{\pi^2 D^4 g} \tag{3.2.11}$$

The uncertainty from each of the measurands is only the resolution:

- $u_D = u_L = \pm 1/32$ in
- $u_Q = \pm 2.5$ L/min
- $u_g = \pm 0.05$ ft/sec^2

The partial derivatives are then:

- $\frac{\partial h_L}{\partial L} = \left(\mathcal{F}\frac{1}{D^5}\right)\frac{8Q^2}{\pi^2 g} = \frac{h_L}{L}$
- $\frac{\partial h_L}{\partial D} = -5\left(\mathcal{F}\frac{L}{D^6}\right)\frac{8Q^2}{\pi^2 g} = 5\frac{h_L}{D}$
- $\frac{\partial h_L}{\partial Q} = 2\left(\mathcal{F}\frac{L}{D^5}\right)\frac{8Q}{\pi^2 g} = 2\frac{h_L}{Q}$
- $\frac{\partial h_L}{\partial g} = -\left(\mathcal{F}\frac{L}{D^5}\right)\frac{8Q^2}{\pi^2 g^2} = -\frac{h_L}{g}$

After some conversion from liters to inches, etc, the total combined uncertainty then would be

$$u_{h_L} = \sqrt{\left(\frac{h_L}{L}u_L\right)^2 + \left(5\frac{h_L}{D}u_D\right)^2 + \left(2\frac{h_L}{Q}u_Q\right)^2 + \left(-\frac{h_L}{g}u_g\right)^2} \tag{3.2.12}$$

with a numerical calculation of

$$u_{h_L} = \sqrt{(0.014476)^2 + (3.8845)^2 + (1.4343)^2 + (-0.028953)^2} \tag{3.2.13}$$

with a final result of $h_L = 18.6 \pm 4.14$ in.

3.2: Combining Uncertainty in Function is shared under a CC BY-NC 4.0 license and was authored, remixed, and/or curated by LibreTexts.

4: Technical Writing

Engineering can sometimes feel that calculating solutions to problems is the primary challenge of producing, collecting, and reporting data. While those can sometimes feel like substantial hurdles, one of the biggest obstacles to engineering effectiveness is communication.

Learning Objectives

- How to *plan* to write
- Building the content including figures and organizing content into appropriate sections

A. *Pedagogy of writing*: What is the process by which a writer produces their work?

a. Environmental factors will influence content and effectiveness:

1. Place: personal or collaborative space; background noise; lighting; posture as a result of comfort
2. Time: before or after meal/sleep; hours or days before deadline; own rate of production
3. Format: sketching on whiteboard or notepad; building digital images; typing in particular platform
4. Mindset: enthusiasm for conveying content; pressure of approaching deadline; trust in collaborators

b. Code-switching for audience

a. Snap to friend: *HW #3 tuf*
b. AI-inspired fluff: *I found the challenges of Homework #3 insurmountable in the timeframe allotted...*
c. Conversational: *Phooey (f*@<), HW #3 can't get done in time*

c. Process in small steps

a. Writing does not require the sequential nature of reading
b. The hardest work may be a short statement
c. **Time Spent ≠ Word Count**

d. Utilize prompts to formulate thesis/framework

B. Templates and collaboration tools

1. Templates provide preset structure and formatting references
2. Examples:
   - ASME two-column format
   - Word 365 built-in report formats
   - Overleaf for LaTeX typesetting
3. Indexing tools for dynamic connections:
   - Use Word's cross-reference tool
     - Embed image and caption within same text box to group elements
     - Helps prevents dynamic movement when additional materials added prior
   - Direct coding within LaTeX:
     - **\label{eq:descriptor}** somewhere within the figure, equation, or table environment
     - **\ ref{eq:descriptor}** when cross referencing within the paragraph of the document

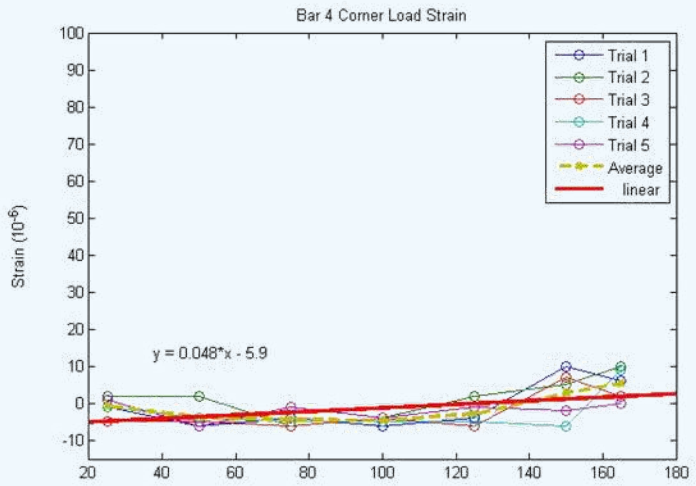
C. Presenting data

1. Graph forms (public domain examples found within Energy Information Administration):
   - Rectilinear: linear scale both directions
   - Semilog: rapid growth on one (or more) axis
   - Pie/Distribution: relative % outcomes
   - Stair step/bar: grouped instances of histogram
   - Polar: parametric equations
   - Contour: 3D spatial representation
2. Figure guidelines (modified list originally from Dunn & Davis 4th Edition: https://doi.org/10.1201/b22182)

2. Figure guidelines (modified list originally from Dunn & Davis 4th Edition. https://doi.org/10.1201/b22182)
   a. All axes (horizontal, vertical, radial, whatever) should have descriptive variable labels with units in brackets or parenthesis (author-defined style).
   b. Range and significant figures of tick values should be appropriate for data being presented (e.g., eliminate unnecessary zeros after decimal or empty space on chart).
   c. Background gridlines are suggested; at minimum, tick marks should be internal and large enough to indicate location of value; numbers on the tick/gridlines should be outside the plotting area.
   d. Experimental results should have individual markers for each data point but unique markers for different data sets.
   e. Some indication of error bars should be provided via uncertainty analysis or statistical methods; single error bar indicates uniform value across a data set, individual error bars on markers indicate variability of uncertainty.
   f. Analytical results with functional equations should have solid curves.
   g. Numerical results as a result of statistical trendline should have dashed curves.
   h. A legend should be present if multiple results are displayed within single graph.
   i. The overall figure should have an ***indexed*** caption that describes content separate from repeating axis labels; a title on a figure is redundant when caption is present.

? Exercise 4.1

3. Evaluate effectiveness of student-produced graphs

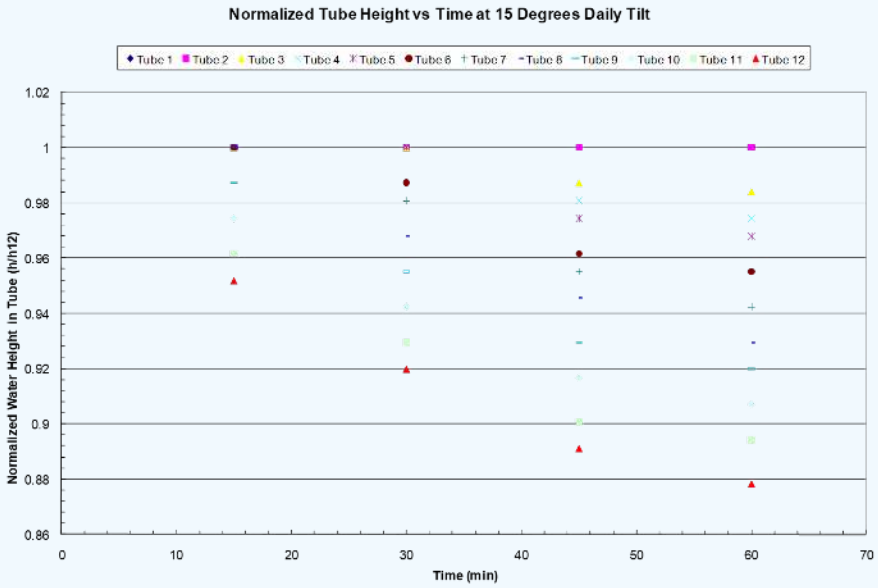


Answer

Range is extremely large relative to data displayed

Connected discrete data as if there were continuum between discrete independent variables

Exercise 4.2



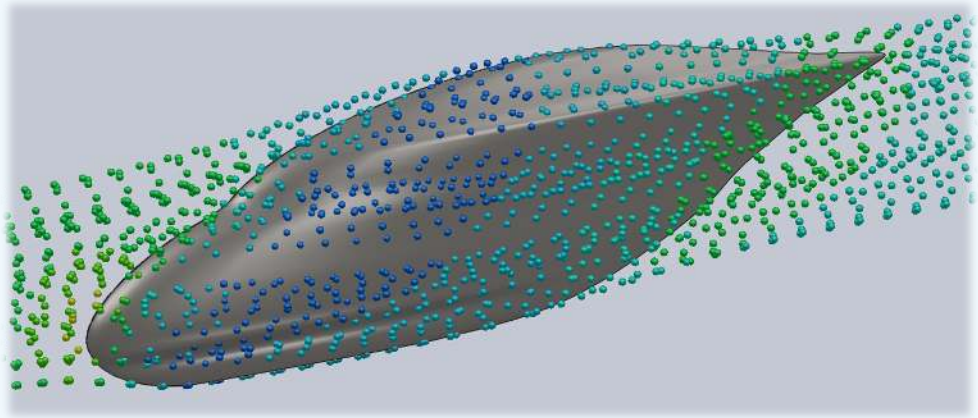
Answer

Markers far too small and indistinct

Title is redundant with the axis labels

Numerical value of *Tube* has no bearing on variable modified

Exercise 4.3

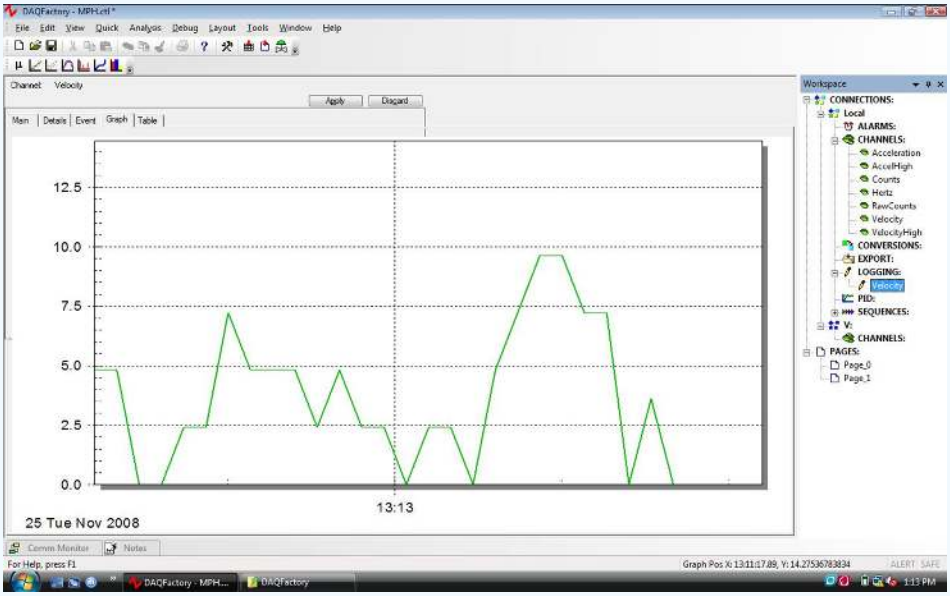


Answer

Beautiful artistic picture without any

- sense of length scales
- reason for individual spheres
- meaning of the color

Exercise 4.4



Answer

Screen capture without any useful axis labels

D. Writing Building Blocks (i.e. what sections to compose)

1. **Methodology and Approach**
   - Sufficient detail for replication, not step-by-step standard operating procedure manual.
   - Use schematics, images, annotated diagrams to connect with the measurands, variables, and results.
   - Example: Design-stage uncertainty of particular sensor: $u_{device} = \pm\sqrt{u_0^2 + u_I^2}$
   - Use narrative prose in paragraphs; lists would exist in tables that are described.
2. **Results and Discussion**
   - Only include pertinent tables and graphs; efficiency of content rather than quantity (i.e., multiple data sets on single chart).
   - Do not repeat data in multiple formats in order to expand page count.
   - Each figure should be explicitly described with an introduction, trend observed, and why it aligns with physical principle being explored.
   - Break separate ideas into paragraphs; avoid single-sentence paragraphs.
   - Scientific justification is the demonstration of learning, not the simple presence or style of the data.
3. **Abstracts** (typically lead all technical papers; individual submission in MEE 390 instruction)
   - Standalone summary: < 1 page
   - Should give sufficient overview to judge relevance
   - Primary components: title, authors, affiliation, funding, intro, methodology, results, conclusion
   - Results: be specific but not quantitative
   - Conclusion: summarize significance and next steps
4. Uncertainty Section
   - Combination of theory and experimental procedures
   - Identify key uncertainty contributors that affects results
   - Reporting should include sensitivity coefficient functions
   - MathCAD pages or Matlab scripts are insufficient as stand-alone content; provide interpretation
5. **Formatting Features**

- Include page numbers
- No title page unless collaborative abstract submitted
- Auto-generated TOC, List of Figures/Tables
- Captions: before tables (leads the reader to the numbers), after figures (to help interpret the picture they viewed). Do not *simply repeat* axis labels or table headings.

6. Future sections incorporated following formulative practices
7. **Theoretical Development**
    - Build equations step-by-step, like textbooks.
    - Define variables and cross-reference numbered equations when making substitutions.
    - Clarify independent/dependent/measured variables that will influence the original forms.
8. **Introduction**
    - Contextualize the topic's importance to developing engineering knowledge beyond assigned task
    - Include external references and real-world examples
    - Organize ideas into logical paragraph flow

E. Receiving and responding to feedback

1. Avoid tears and self-doubt: the learning occurs from the **practice**.
2. Note the elements or behaviors that may repeat even if only mentioned once or twice.
3. Incorporate the changes as improvement in practice.

F. Full text OER resource

4: Technical Writing is shared under a CC BY-NC 4.0 license and was authored, remixed, and/or curated by LibreTexts.

CHAPTER OVERVIEW

5: Calibration and Dynamic Response

The collected experimental results should connect in some manner to physical principles for which some theoretical background is known. In some cases, the best fit to a known functional form can be applied. Depending on the associated level of confidence (i.e., uncertainty), the emphasis on some data can be elevated via a weighted least-squares approach. Otherwise, mechanical systems can often respond similarly to ordinary differential equations for which constant coefficients indicate the type of response to step and periodic inputs. Subsections are applied to the experimental procedures to calibrate transfer functions or predict types of outcomes from step and periodic inputs.

Learning Objectives

- Line of best fit - create functional estimate $\hat{y}(x)$ of dependent response
  - Typically sources cover only uniform least-squares regression.
  - The approaches used in the *polyfit* function from Matlab or *Add Trendline* tools in Excel assume equal impact of each data point.
  - Sometimes data is more or less important so weighting is required (i.e., touchdowns carry more points than field goals; penalty kicks are the same as standard goals in soccer).
- Dynamic response of signal - review of Diff Eq. lite with specific input–output conditions
  - Forms of the analytical solutions will be provided.
  - Task will be sorting inputs to determine ***where*** variables should be inserted.
  - Decision tree of order of ODE and input type will define path to solution.
  - The engineering then comes in the interpretation of the solution.

5.1: Fitting with Weighted Least Squares
5.2: Dynamic Response from Linear ODEs

For additional material from sources external to NIU, please visit Appendix The additional subsections provide additional detail and description of one-degree-of-freedom systems with different differential order but oscillating inputs.

5: Calibration and Dynamic Response is shared under a CC BY-NC 4.0 license and was authored, remixed, and/or curated by LibreTexts.

5.1: Fitting with Weighted Least Squares

A. Least squares regression

1. Using discrete data, one can obtain a best estimate of functional relationship of data
2. Assume $\hat{y} = f(x)$ with polynomial form:

$$\hat{y}(x) = a_0 + a_1 x + a_2 x^2 + \ldots a_m x^m = \sum_{i=0}^{m} a_i x^i \tag{5.1.1}$$

where subscript identifies explicit coefficient multiplying continuous value of x raised to similar power

3. n different values of x and y will solve m coefficients if $m \leq n-1$
4. Deviation of the data to fit function over all data points

$$D = \sum_{i=1}^{N} \left[y_i - \hat{y}\left(x_i\right) \right]^2 \tag{5.1.2}$$

$$D = \sum_{i=1}^{N} \left[y_i - \left(a_0 + a_1 x + a_2 x^2 + \ldots a_m x^m \right) \right]^2 \tag{5.1.3}$$

How to minimize the value D? **Thru derivative of course**

$$dD = \frac{\partial D}{\partial a_0} da_0 + \ldots + \frac{\partial D}{\partial a_m} da_m \tag{5.1.4}$$

where full differential is applied.

5. In order for $dD \rightarrow 0$, each partial derivative should approach zero

$$\frac{\partial D}{\partial a_0} = 0 = \frac{\partial}{\partial a_0} \left\{ \sum_{i=1}^{N} \left[y_i - \left(a_0 + a_1 x + a_2 x^2 + \ldots a_m x^m \right) \right]^2 \right\} \tag{5.1.5}$$

$$\frac{\partial D}{\partial a_m} = 0 = \frac{\partial}{\partial a_m} \left\{ \sum_{i=1}^{N} \left[y_i - \left(a_0 + a_1 x + a_2 x^2 + \ldots a_m x^m \right) \right]^2 \right\} \tag{5.1.6}$$

which yields $m+1$ simultaneous equations to solve $a_0 \Rightarrow a_m$

6. How to set it up?
   a. Recall the polynomial form, but in terms of each i^{th} data point:

$$y_i = a_0 x_i^0 + a_1 x_i^1 + a_2 x_i^2 + \ldots a_m x_i^m \tag{5.1.7}$$

   where x_i is independent variable of trial and y_i is dependent measurand

   b. Need to determine coefficients, so set-up linear algebra equation:

$$\begin{bmatrix} 1 & x_1 & x_1^2 & \ldots & x_1^m \\ 1 & x_2 & x_2^2 & \ldots & x_2^m \\ 1 & x_3 & x_3^2 & \ldots & x_3^m \\ & & \vdots & & \\ 1 & x_n & x_n^2 & \ldots & x_n^m \end{bmatrix} \begin{bmatrix} a_0 \\ a_1 \\ a_2 \\ \vdots \\ a_m \end{bmatrix} = \begin{bmatrix} y_1 \\ y_2 \\ y_3 \\ \vdots \\ y_n \end{bmatrix} \tag{5.1.8}$$

   which can be written more succinctly as:

$$M\vec{a} = \vec{y} \tag{5.1.9}$$

   where:

   i. M is matrix of n rows (total trials) and $m+1$ columns (m^{th} order of the polynomial)
   ii. $\vec{a}$ is vector of $m+1$ unknown coefficients
   iii. $\vec{y}$ is vector list of n dependent measurands
   iv. Rewriting with dimensional sizes indicated:

$$M_{n\times m+1}\ \vec{a}_{m+1\times 1} = \vec{y}_{n\times 1} \tag{5.1.10}$$

c. How to isolate $\vec{a}$? *Use the inverse of course ...*

d. But it is impossible to invert a non-square matrix

7. Time to use pseudoinverse

a. Use the transpose to force a square form on left hand side

$$M^T\ M\ \vec{a} = M^T\ \vec{y} \tag{5.1.11}$$

where transpose is flip of rows and columns across diagonal:

$$N = \begin{bmatrix} 1 & 2 & 3 & 4 \\ 1 & 4 & 9 & 16 \\ 1 & -1 & -2 & 1 \end{bmatrix} \tag{5.1.12}$$

$$N^T = \begin{bmatrix} 1 & 1 & 1 \\ 2 & 4 & -1 \\ 3 & 9 & -2 \\ 4 & 16 & -1 \end{bmatrix} \tag{5.1.13}$$

b. Resulting left side is now square but same action on right side:

$$(M^T M)_{m+1\times m+1}\ \vec{a}_{m+1\times 1} = M^T_{m+1\times n}\ \vec{y}_{n\times 1} \tag{5.1.14}$$

c. Matrix multiplication is more efficient manner of composing numerous sums of input data

d. Inverse is applied recalling matrix multiplication is NOT commutative:

$$\vec{a} = (M^T M)^{-1} M^T\ \vec{y} \tag{5.1.15}$$

Example 5.1.1

e. Prove same results achieved when Add trendline or polyfit function used when compared to pseudoinverse

- independent data has the form $x = \{0.4, 1.1, 1.9, 3, 5, 6\}$
- dependent data results in $y = \{2.7, 3.6, 4.4, 5.2, 9.2, 12.1\}$
- What are number of clicks or code updates to change order of polynomial?

Solution

Add example text here.

B. Weighted least squares

1. Knowledge of pseudoinverse allows emphasis of data
2. Diagonal matrix of values can increase/decrease impact of data points

$$W^{1/2} M\ \vec{a} = W^{1/2}\ \vec{y} \tag{5.1.16}$$

where W has something like the form:

$$W = \begin{bmatrix} 10 & 0 & 0 & 0 \\ 0 & 10^3 & 0 & 0 \\ 0 & 0 & 2 & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix} \tag{5.1.17}$$

where x_3 is twice as important as x_4, x_1 is 5 times more important then x_3, and x_2 is 2 orders of magnitude more important than x_1

3. How to choose values to populate W?

a. always use positive values (hence the $^{1/2}$ power)

b. larger values for those with greater confidence/importance

c. lower values to reduce influence of spurious points

d. Reasons for choosing:

i. Size of uncertainty relative to measurement

ii. Spread of standard deviation statistics

iii. Trustworthiness of collaborative teammate

4. Solution follows similar pseudoinverse format

$$\vec{a} = (M^T W^{1/2} M)^{-1} M^T W^{1/2} \vec{y} \tag{5.1.18}$$

Example 5.1.2

5. How does changing W affect the polynomials and proximity of fit of example above? What happens to curve when emphasizing one or more data points?

Solution

Add example text here.

C. Standard error of the fit S_{yx}

1. Quantify deviation between y_i and $\hat{y}(x)$

$$S_{yx} = \frac{\sqrt{\sum_{i=1}^{N} (y_i - \hat{y}(x_i))^2}}{\nu} \tag{5.1.19}$$

where $\nu = N - (m+1)$ (number of data points subtracted by order of the polynomial fit

a. Provides "goodness of fit" like the R value

b. Higher m will often make S_{yx} decrease

c. Underlying physical principle should define the m^{th} order of fit; otherwise it adds unnecessary inflection points in $\hat{y}(x)$

5.1: Fitting with Weighted Least Squares is shared under a CC BY-NC 4.0 license and was authored, remixed, and/or curated by LibreTexts.

5.2: Dynamic Response from Linear ODEs

D. General Measurement Model (pivot to Diff EQ-lite)

1. Form of the most general linear differential equation:

$$a_n \frac{d^n y}{dt^n} + ... + a_2 \frac{d^2 y}{dt^2} + a_1 \frac{dy}{dt} + a_0 y(t) = F(t) \tag{5.2.1}$$

2. Left-hand-side of equation describes physics of inertia, dampers, springs, resistors, capacitors, etc.
3. Right-hand-side is arbitrary forcing function (i.e. impulse, step, ramps)
   a. $F(t) = b_m \frac{d^m x}{dt^m} + ... b_1 \frac{dx}{dt} + b_0 x(t)$
   b. Need to have $m \leq n$
   c. Solution forms for set $n = \{0, 1, 2\}$ will be given for variety of $F(t)$
   d. Each case of n builds to the next level
   e. Given the solution to D.E. $\Rightarrow$ just place terms in correct parameters (language and vocabulary becomes *critical!*)

E. Zero-order system

▼ Details

1. Direct, instantaneous response to input
2. General case for $n = 0$

$$a_0 y(t) = F(t) \tag{5.2.2}$$

3. Solution (simple algebraic step)

$$y(t) = kF(t) \tag{5.2.3}$$

where

   a. Static sensitivity $k = \frac{1}{a_0}$
   b. Appropriate for static calibration but scaling of input applies to higher orders of n
   c. k is calibration slope $k = \frac{dy}{dx}$

? Exercise 5.2.1

4. A pressure sensor responds with sensitivity of 25.0 mV/kPa; can absolute atmospheric pressure be measured on a multimeter with 0--5 V range?

   What is lowest range of atmospheric pressures that can be measured when smallest resolution is 2 mV?

   Answer

   Static sensitivity

$$k = 25.0 \tag{5.2.4}$$

   mV/kPa

$$y_{max} = 5 \tag{5.2.5}$$

   V

$$F_1(t) = 101.325 \tag{5.2.6}$$

   kPa

$$y = kF(t) = 25.0(101.325) = 2.53 \tag{5.2.7}$$

   V

   Yes, it can measure 1~atm

$$F(t)_{max} = \frac{y_{max}}{k} = \frac{5}{25.0 \times 10^{-3}} = 1.97 \tag{5.2.8}$$

atm

$$F(t)_{min} = \frac{y_{min}}{k} = \frac{2}{25.0} = 80 \tag{5.2.9}$$

Pa

Increasing multimeter range y_{max} increases pressure range

Decreasing k *also* increases pressure range if $y_{max} = 5$ V.

F. First Order System (cases when $n = 1$ in general form)

▼ Details

1. Storage elements (potential energy) eliminate instantaneous system response
2. General form of equation:

$$a_1 \frac{dy}{dt} + a_0 y(t) = F(t) \tag{5.2.10}$$

$$a_1 \dot{y}(t) + a_0 y(t) = F(t) \tag{5.2.11}$$

$$a_1 y'(x) + a_0 y(x) = F(x) \tag{5.2.12}$$

3. Rewritten in parameterized form:

$$\tau \dot{y}(t) + y(t) = kF(t) \tag{5.2.13}$$

where

a. $\tau = \frac{a_1}{a_0}$ is the time constant; multiplies the rate of response $\dot{y}(t)$
b. $k = \frac{1}{a_0}$ is still static sensitivity as zero-order

4. Homogeneous solution will always have the exponential decay $y_h \approx \mathbf{e}^{-t/\tau}$
5. *Particular* solution depends on input form; choice from **F.6** or **F.7**
6. Step function input

a. Discontinuous functional form:

$$F(t) = \begin{cases} 0 & t < 0 \\ AU(t) & t \geq 0 \end{cases} \tag{5.2.14}$$

i. A is amplitude; size of step that contains units
ii. $U(t)$ is unit step; value of *one*; no units

b. Makes the DE have the form

$$\tau \dot{y}(t) + y(t) = kAU(t) \tag{5.2.15}$$

with $y_0 = y(0)$ for the initial condition

c. System response for $t \geq 0$:

$$y(t) = kA + (y_0 - kA)\mathbf{e}^{\frac{-t}{\tau}} \tag{5.2.16}$$

where first portion is steady state response with addition of exponential part as transient function with magnitude

i. Evaluate initial condition: $y(t = 0) =$??
ii. Evaluate end condition: $y(t \to \infty) =$??
iii. Sketch of general form of the response

d. *Speed* of transient controlled by time constant τ

- show examples in different colors of larger and smaller τ

e. Exponential represents **dynamic error** between current point $y(t)$ and end condition y_∞ approaching kA:

$$\delta_f(t) = \mathbf{e}^{\frac{-t}{\tau}} = \frac{y(t) - y_\infty}{y_0 - y_\infty} \tag{5.2.17}$$

f. Whereas percent of response from starting point is **magnitude ratio**:

$$M(t) = 1 - \delta_f(t) = 1 - \mathbf{e}^{\frac{-t}{\tau}} = \frac{y(t) - y_0}{y_\infty - y_0} \tag{5.2.18}$$

g. Special cases of δ_f and M:

i. One time constant $t = \tau$

$$\delta_f(\tau) = \mathbf{e}^{-\tau/\tau} = \mathbf{e}^{-1} = 0.368 \tag{5.2.19}$$

$$M(\tau) = 1 - \mathbf{e}^{-\tau/\tau} = 0.632 \tag{5.2.20}$$

ii. Rise time: when at 90% response (or 10% error)

$$0.1 = \delta_f(t_{rise}) = \mathbf{e}^{-t/\tau} \tag{5.2.21}$$

$$0.9 = M(t_{rise}) = 1 - \mathbf{e}^{-t/\tau} \tag{5.2.22}$$

which can be solved for $t_{rise} = 2.30\tau$

7. Sine function input [an alternative, periodic forcing function $F(t)$]

a. Input has form $F(t) = A\sin(\omega t)$ where

i. A is amplitude of input (where units are)

ii. ω is circular frequency (rad/s)

iii. $f = \frac{\omega}{2\pi}$ when cyclic frequency is given in Hertz

iv. No vertical offset is included (would be part of *step function* above)

v. No horizontal phase shift of start: $F(t = 0) = A\sin(\omega \times 0) = 0$

b. D.E. in parameterized form

$$\tau\dot{y}(t) + y(t) = kA\sin(\omega t) \tag{5.2.23}$$

c. Solution in full form:

$$y(t) = \frac{kA}{\sqrt{\omega^2\tau^2 + 1}}\sin\left(\omega t - \tan^{-1}\left[\omega\tau\right]\right) + \left(y_0 + \frac{\omega\tau kA}{\omega^2\tau^2 + 1}\right)\mathbf{e}^{\frac{-t}{\tau}} \tag{5.2.24}$$

where the first part is oscillating steady-state response while the second portion is a transient response that decays with exponential.

d. Using the simplified form:

$$y(t) = kAM(\omega)\sin\left(\omega t + \phi\left(\omega\right)\right) + C\mathbf{e}^{\frac{-t}{\tau}} \tag{5.2.25}$$

where

i. Magnitude ratio is frequency dependent $M(\omega) = \frac{1}{\sqrt{\omega^2\tau^2+1}}$

ii. $M(\omega)$ represents ratio of output amplitude to input amplitude

iii. Uses same form for dynamic error that is now *frequency dependent* $\delta(\omega) = 1 - M(\omega)$

iv. Phase lag causes shift of $\phi(\omega) = -\tan^{-1}(\omega\tau)$ in radians

v. Convert to time lag of signal response thru input frequency $\beta = \frac{\phi}{\omega} = \frac{-\tan^{-1}(\omega\tau)}{\omega}$ in seconds

vi. Decay coefficient C depends on full combination of

- initial conditions y_0
- physics of the model: sensitivity k and time constant τ
- input parameters: amplitude A and frequency ω

e. $M(\omega)$ sometimes expressed in logarithmic sense

i. Consider ratios of power output to input:

$$bel = \log_{10}\left(\frac{P_{out}}{P_{in}}\right) \tag{5.2.26}$$

ii. Scale using prefix (i.e. like 1 decimeter = 10 meter) \[***deci*** bel = 10 \log_{10}\left(\frac{P_{out}}{P_{in}}\right) \]

iii. Power is squared quantity of measurands

$$dB = 10\log_{10}\left(\frac{\text{Amplitude}_{out}}{\text{Amplitude}_{in}}\right)^2 \tag{5.2.27}$$

$$dB = 20\log_{10}\left(M(\omega)\right) \tag{5.2.28}$$

8. Summary of 1st order systems

a. Total solution still follows form

$$y(t) = y_{homogeneous}(t) + y_{particular}(t) \tag{5.2.29}$$

where y_h is transient response with exponential terms and y_p will depend on the type of input used (step of section **F.6** or sinusoid of section **F.7**)

b. Given the general form of solutions, fit the provided data into appropriate coefficients.

? Exercise 5.2.2

c. During a step function calibration, a first-order vane anemometer with 10 fan blades is exposed to a step change from 12 mph initially to 88 mph. The transducer counts the number of fan blades that pass a magnetic sensor, producing a static sensitivity of $k = 22$ clicks/mph. After 143 milliseconds, the instrument output indicates 777 clicks; after 477 milliseconds the output is 1,444 clicks. Estimate the time constant of the first-order sensor.

What time after the step change will the dynamic error be less than 2.6%?

Answer

The system is first-order with a step change output making

$$y(t) = kA + (y_0 - kA)\mathrm{e}^{-t/\tau} \tag{5.2.30}$$

The amplitude at the end of the step is 88 mph which would become 1936 clicks (kA) as time increases; the value at the start $y_0 = 264$ clicks. One could solve for the exponential function or sometimes called the error fraction:

$$\delta_f(t) = \frac{y(t) - y_\infty}{y_0 - y_\infty} = \mathrm{e}^{-t/\tau} \tag{5.2.31}$$

$$\delta_f(0.143) = \frac{777 - 1936}{264 - 1936} = \mathrm{e}^{-0.143/\tau} \tag{5.2.32}$$

which can algebraically be solved for τ to determine that $\tau = 390$ milliseconds.

If the error function is used, the associated time of 2.6% error can be calculated:

$$0.026 = \mathrm{e}^{-t/0.390} \tag{5.2.33}$$

$$t = -0.390\ln(0.026) \tag{5.2.34}$$

where $t = 1.42$ sec.

G. Second order systems (cases when $n = 2$)

▼ Details

1. Systems with inertia in the physical description that affect acceleration and not rate of response
2. General form of differential equation

$$a_2\ddot{y} + a_1\dot{y} + a_0 y(t) = F(t) \tag{5.2.35}$$

3. Rewritten in more common form

$$\frac{1}{\omega_n^2}\ddot{y}+\frac{2\zeta}{\omega_n}\dot{y}+y(t)=kF(t) \tag{5.2.36}$$

where

a. Natural frequency from coefficients $\omega_n=\sqrt{\frac{a_0}{a_2}}$

b. Dimensionless damping ratio $\zeta=\frac{a_1}{2\sqrt{a_2 a_0}}$

4. Homogeneous solution form is dependent on ζ

a. Under damped scenario $0\leq\zeta<1$ -- oscillation is present

b. Critically damped $\zeta=1$ -- extremely rare scenario

c. Overdamped $\zeta>1$ -- no oscillation occurs

5. Particular solution for step input

a. Overdamped is complicated hyperbolic functions

$$y(t) = kA\left\{1-\mathbf{e}^{-\zeta\omega_n t}\left[\cosh\left(\omega_n t\sqrt{\zeta^2-1}\right)+\frac{\zeta}{\sqrt{\zeta^2-1}}\sinh\left(\omega_n t\sqrt{\zeta^2-1}\right)\right]\right\} \tag{5.2.37}$$

-- will be avoided from MEE-standpoint

b. Critically damped appears more like 1st order system with exponential decay

$$y(t)=kA\left\{1-\mathbf{e}^{-\omega_n t}\left(1+\omega_n t\right)\right\} \tag{5.2.38}$$

c. Underdamped has solution form

$$y(t)=kA\left\{1-\mathbf{e}^{-\zeta\omega_n t}\left[\frac{1}{\sqrt{1-\zeta^2}}\sin\left(\omega_n t\sqrt{1-\zeta^2}+\sin^{-1}\left[\sqrt{1-\zeta^2}\right]\right)\right]\right\} \tag{5.2.39}$$

d. Develop sketch of the response plots comparing three forms

e. Since no input frequency to follow, a **ringing** (or sometimes called *damped*) frequency is observed

i. Scaled version of natural frequency $\omega_d=\omega_n\sqrt{1-\zeta^2}$

ii. Period of oscillation $T_d=\frac{1}{f_d}=\frac{2\pi}{\omega_d}$

iii. Rise time is threshold of first achieving 90% response (e.g., like when $\delta(t_{rise})<0.1$)

iv. Settling time is when oscillations are less than ±10 around end condition

6. Solution for sinusoidal input of $F(t)$

$$y(t)=y_h+\frac{KA}{\sqrt{\left[1-\left(\frac{\omega}{\omega_n}\right)^2\right]^2+\left[2\frac{\zeta\omega}{\omega_n}\right]^2}}\sin\left(\omega t+\phi(\omega)\right) \tag{5.2.40}$$

a. Phase shift is frequency dependent

$$\phi(\omega)=\tan^{-1}\left(\frac{-2\frac{\zeta\omega}{\omega_n}}{1-\left(\frac{\omega}{\omega_n}\right)^2}\right) \tag{5.2.41}$$

where sign of numerator and denominator necessary for 4-quadrant inverse

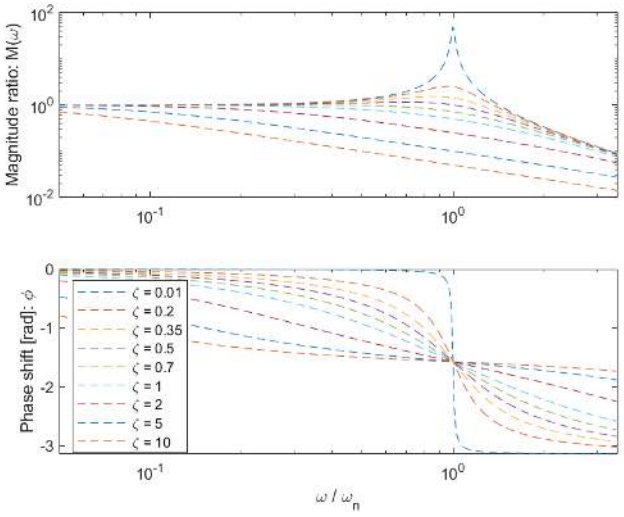
b. Amplitude of steady-state condition still has magnitude ratio

$$M(\omega)=\frac{\text{Amplitude}_{\text{output}}}{\text{Amplitude}_{\text{input}}}=\frac{\cancel{kA}}{\sqrt{\left[1-\left(\frac{\omega}{\omega_n}\right)^2\right]^2+\left[2\frac{\zeta\omega}{\omega_n}\right]^2}}\frac{1}{\cancel{kA}} \tag{5.2.42}$$

$$M(\omega)=\frac{1}{\sqrt{\left[1-\left(\frac{\omega}{\omega_n}\right)^2\right]^2+\left[2\frac{\zeta\omega}{\omega_n}\right]^2}} \tag{5.2.43}$$

produces bode plots for cases around $\omega = \omega_n$

c. Other characteristics of periodically forced 2nd order systems shown in figure



i. Resonance $\Rightarrow$ very small ζ but $\frac{\omega}{\omega_n} \to 1$
- Output becomes unstable form

ii. Small frequency ratio $\frac{\omega}{\omega_n}$ makes $M(\omega)$ nearly unity with small phase lag
- Best input to output transmission

iii. Large frequency ratio $\frac{\omega}{\omega_n}$ makes $M(\omega)$ small with delayed phase
- Potential to eliminate unwanted frequency data

7. Summary of 2nd order systems

a. Types of homogeneous solutions become dependent on scale of constant coefficients

b. Particular solutions consider the long-term outcome of the response rather than instant change

c. Sorting by ODE order and type of forcing aligns with terms defined in each of the 4 categories

? Exercise 5.2.3

d. A dynamic system is evaluated by multiple input types. When zero initial displacement $s(0) = 0$ and velocity $\dot{s}(0) = 0$ are then actuated with a step input $U(t)$ with the following differential form:

$$3.6\dot{s} + 15s + 0.6\ddot{s} = 52.5U(t) \tag{5.2.44}$$

Determine the following:

- full equation of the response $s(t)$
- damping ratio
- ringing (damped) frequency
- natural frequency

If the same system were excited by a periodic input, determine the range of frequencies where the output amplitude would exceed the input amplitude. At the maximum frequency on that range, calculate the phase lag and time delay of the response.

Answer

The coefficients of $a_2 = 0.6$, $a_1 = 3.6$ and $a_0 = 15$ determine the terms as

$$\omega_n = \sqrt{a_0/a_2} = \sqrt{15/0.6} = 5 \text{ rad/s} \tag{5.2.45}$$

$$\zeta = a_1/2\sqrt{a_0 a_2} = 3.6/2\sqrt{15(0.6)} = 0.6 \text{ (underdamped)} \tag{5.2.46}$$

$$\omega_d = \omega_n\sqrt{1-\zeta^2} = 4 \text{ rad/s} \tag{5.2.47}$$

With the amplitude of the input being 52.5, the product $KA = 3.5$ therefore the response is

$$y(t) = KA\left\{1 - \mathrm{e}^{-\zeta\omega_n t}\left[\frac{\sin(\omega_d t + \phi)}{\sqrt{1-\zeta^2}}\right]\right\} \tag{5.2.48}$$

$$y(t) = 3.5\left\{1 - 1.25\mathrm{e}^{-3t}\sin(4t + 0.927)\right\} \tag{5.2.49}$$

where the phase was found by $\phi = \sin^{-1}(\sqrt{1-.6^2})$.

Exercise 5.2.4

e. A step input of 16 N on a 2nd order system yields the following form of the voltage response of the sensor:

$$y(t) = 8.00 - 8.09\mathrm{e}^{-6.3t}\sin\left[41.53\,(t + 0.0342)\right] \tag{5.2.50}$$

Determine the parameters of the dynamic equation including damping ratio, static sensitivity, and natural frequency. Write the original form of the differential equation using the correct values for a_2, a_1, and a_0 with the forcing of 16 N shown on the right side.

Answer

Multiple simultaneous equations can be set-up:

$$A = 16 \tag{5.2.51}$$

$$kA = 8.00 \tag{5.2.52}$$

$$\frac{KA}{\sqrt{1-\zeta^2}} = 8.09 \tag{5.2.53}$$

$$-\zeta\omega_n = -6.3 \tag{5.2.54}$$

$$\omega_n\sqrt{1-\zeta^2} = 41.53 \tag{5.2.55}$$

$$\omega_n\sqrt{1-\zeta^2}\sin^{-1}\left[\sqrt{1-\zeta^2}\right] = 41.53 \times 0.0342 \tag{5.2.56}$$

Which results in static sensitivity $k = 0.5$ V/N, damping ratio $\zeta = 0.148$, natural frequency $\omega_n = 42.35$ rad/sec. The original dynamic equation would have the form:

$$0.00111\backslash\mathrm{Ddot}y + 0.014\dot{y} + 2y(t) = 16U(t) \tag{5.2.57}$$

Exercise 5.2.5

f. Assume the input to the device from Exercise 5.2.5 were to change to a sinusoid of range ± 33 N at a frequency of 25 rad/sec. Determine the amplitude of the response in millivolts along with the phase lag in seconds.

Answer

The slight changes would be $A = 33$ and $\omega = 25$. The amplitude of the response is:

$$\mathrm{Amp}_{out} = \frac{kA}{\sqrt{\left[1 - \left(\frac{\omega}{\omega_n}\right)^2\right]^2 + \left[2\frac{\zeta\omega}{\omega_n}\right]^2}} \tag{5.2.58}$$

where the inputs result in the amplitude of the output as 24.46 V with a phase of $\beta = -0.0105$ sec since $\beta = \phi/\omega$.

5.2: Dynamic Response from Linear ODEs is shared under a CC BY-NC 4.0 license and was authored, remixed, and/or curated by LibreTexts.

CHAPTER OVERVIEW

6: Fourier Analysis

Periodic signals can become more complex than just single frequency and amplitudes. Attempts can be made to approximate discontinuous and filter discrete signals through Fourier Series.

Learning Objectives

- Calculate coefficients that work in combination to increase representation of given functions
- Use frequency domain to filter data within signals including significant random noise

6.1: Descriptions
6.2: Conversion of Function
6.3: Discrete data conversion and filtering

For additional material from sources external to NIU, please visit the Appendix

6: Fourier Analysis is shared under a CC BY-NC 4.0 license and was authored, remixed, and/or curated by LibreTexts.

6.1: Descriptions

A. Classification

1. Stochastic - random elements that incorporate uncertainty of measurement to functional form
2. Deterministic - something for which written functional form can be produced
3. Periodic - repeats the trend/behavior at some frequency
    a. Recall that simple periodics have single frequency:

$$x(t) = A\cos(\omega t) \tag{6.1.1}$$

    b. Complex periodic functions have multiple combinations of characteristics that can be summed together:

$$y(t) = A_0 + \sum_i C_i \cos(2\pi f_i t + \phi_i) \tag{6.1.2}$$

        i. Subscript i indicates a unique combination of amplitude, frequency, and phase shift
        ii. Time t is the continuous variable
        iii. Adding more combinations $i \to \infty$ gets closer to representing observations
4. Observed results can be a combination of all three features
    a. Isolation of terms within deterministic functions are already covered in weighted-least-square regression
    b. Periodic representation can be determined from Fourier coefficients of C_i (section 6.2)
    c. Stochastic features can be removed through statistical or filtering mechanisms (section 6.3)

6.1: Descriptions is shared under a CC BY-NC 4.0 license and was authored, remixed, and/or curated by LibreTexts.

6.2: Conversion of Function

B. Representing Complex Periodic Waveform

? Exercise 6.2.1

1. If $y(t)$ is given, the how could one determine amplitudes C_i and phases ϕ_i in the summation?

   Answer

   Calculate the **Fourier Coefficients**

2. Assume that $y(t)$ is periodic
   a. $y(t) = y(t+T)$: repeats at fundamental period T
   b. Period relates to cyclic and circular frequency: $T_1 = \frac{1}{f_1} = \frac{2\pi}{\omega_1}$
3. Rather than trying to calculate phase, separate amplitude coefficient into trigonometric series

$$y(t) = \frac{A_0}{2} + \sum_{n=1}^{\infty} \left(A_n \cos\left(n\omega t\right) + B_n \sin\left(n\omega t\right)\right) \tag{6.2.1}$$

$$y(t) = \frac{A_0}{2} + \sum_{n=1}^{\infty} \left(A_n \cos(n\frac{2\pi}{T}t) + B_n \sin(n\frac{2\pi}{T}t)\right) \tag{6.2.2}$$

   where subscripts of coefficients A_i and B_i correspond to integer values of $n = \{1, 2, 3, ...\}$. [In other words, there is *no such thing* as $A_{1.42}$]

4. Integrating the original periodic signal over its fundamental period provides the mean value

$$A_0 = \frac{2}{T}\int_{-T/2}^{T/2} y(t)dt \tag{6.2.3}$$

   or

$$A_0 = \frac{2}{T}\int_{0}^{T} y(t)dt \tag{6.2.4}$$

   where the user gets to define which period range is easier to integrate

5. If the original function $y(t)$ is multiplied by orthogonal periodic form and integrated over the fundamental period:

$$\int_0^T y(t)\cos(m\omega t)dt = A_0\int_0^T \cos(m\omega t)dt + \sum_n A_n \int_0^T \cos(n\omega t)\cos(m\omega t)dt + B_n\int_0^T \sin(n\omega t)\cos(m\omega t)dt \tag{6.2.5}$$

   a. First integral is multiplying the mean value A_0 by integration of full period; hence result becomes zero
   b. Last term is multiple of $\sin$ and $\cos$ which are out of phase with one another AND a full period; also goes to zero value
   c. Middle term becomes zero value when frequencies don't match: $m \neq n$
   d. Remaining result becomes $\cos^2$ when $m = n$ which eventually integrates to simple value of $\frac{T}{2}$
   e. Allows for direct calculation of discrete coefficients

$$A_n = \frac{2}{T}\int_0^T y(t)\cos(n\omega t)dt \tag{6.2.6}$$

$$A_n = \frac{2}{T}\int_{-T/2}^{T/2} y(t)\cos(n\frac{2\pi}{T}t)dt \tag{6.2.7}$$

   where user selects range of period or argument form that is easiest to integrate

   f. Similar form applies to the $\sin$ function to get B_n

$$B_n = \frac{2}{T}\int_0^T y(t)\sin(n\frac{2\pi}{T}t)dt \tag{6.2.8}$$

$$B_n = \frac{2}{T}\int_{-T/2}^{T/2} y(t)\sin(n\omega t)dt \tag{6.2.9}$$

g. usually will be given $y(t)$ or build from graph (including piecewise function) to then get values of A_n and B_n that includes n as TBD integer value

6. Important frequency terms

a. *fundamental* harmonic is the $n = 1$; represents the maximum full period T of the original signal

b. increasing n are subsequent *harmonics* that increase frequency by integer multiples and decrease amplitude

C. Special Cases of Fourier Series

1. *Even* functions (will be cosine only)

a. definition of even function is $g(t) = g(-t)$

b. same as when considering $\cos(-n\omega t) = \cos(n\omega t)$

c. like bouncing away from mean value in same direction from $t = 0$

d. all $B_n = 0$

e. thus, simplified form

$$y(t) = A_0 + \sum_{n=1}^{\infty} A_n \cos(n\omega t) \tag{6.2.10}$$

2. Odd functions (will be sine only)

a. definition of odd function is $-h(t) = h(-t)$

b. same as when considering $\sin(-n\omega t) = -\sin(n\omega t)$

c. function moves in *opposite* directions from $t = 0$

d. all $A_n = 0$

e. thus, simplified form

$$y(t) = A_0 + \sum_{n=1}^{\infty} B_n \sin(n\omega t) \tag{6.2.11}$$

3. If *neither* is even or odd, then Fourier Series can contain both A_n and B_n

a. Yet still a way to solve more simply ... through phase shift

b. To make sine or cosine only

$$y(t) = A_0 + \sum_{n=1}^{\infty} C_n \cos(n\omega t - \phi_n) \tag{6.2.12}$$

$$y(t) = A_0 + \sum_{n=1}^{\infty} C_n \sin(n\omega t + \phi_n^*) \tag{6.2.13}$$

c. Same trig identity rules apply

$$C_n = \sqrt{A_n^2 + B_n^2} \tag{6.2.14}$$

$$\phi_n = \tan^{-1}(\frac{B_n}{A_n}) \tag{6.2.15}$$

$$\phi_n^* = \tan^{-1}(\frac{A_n}{B_n}) \tag{6.2.16}$$

d. but impossible to know ϕ_n in advance

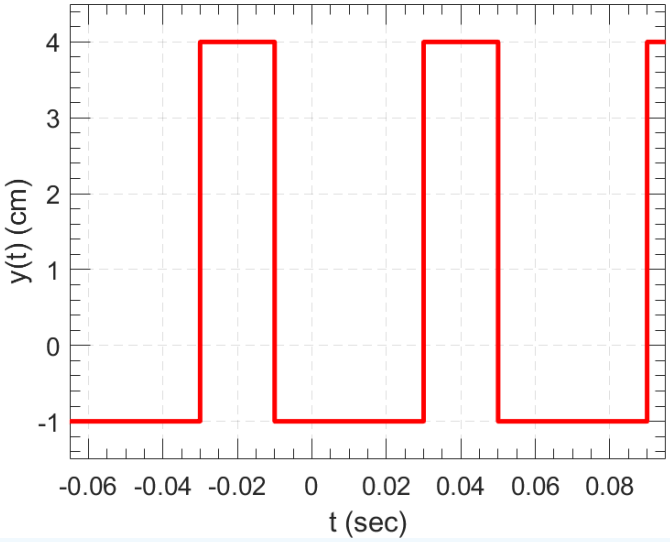
e. UNLESS use the phase shift in time

$$C_n = \frac{2}{T}\int_0^T y(t \pm t_0)\cos\left(n\omega\left[t \pm t_0\right]\right)dt \tag{6.2.17}$$

$$\phi_n = \pm n\omega t_0 \tag{6.2.18}$$

Exercise 6.2.1

Consider a 33% duty cycle signal with phase offset shown below. Determine the Fourier coefficients and then show a plot with the signal produced up to the 4^{th} and 10^{th} harmonic.



Answer

The piecewise function will have the following form:

$$y(t) = \begin{cases} -1 & -0.01 \le t < 0.03 \\ 4 & 0.03 \le t < 0.05 \end{cases} \tag{6.2.19}$$

The mean value A_0 is the average over the period $T = 0.06$ sec which would be

$$A_0 = \frac{2}{0.06}\left(\int_{-0.01}^{0.03} -1dt + \int_{0.03}^{0.05} 4dt\right) \tag{6.2.20}$$

$$A_0 = \frac{2}{0.06}(-1(0.03+0.01) + 4(0.05-0.03)) = 1.33 \tag{6.2.21}$$

The coefficients for the sine and cosine terms will be

$$A_n = \frac{2}{0.06}\left(\int_{-0.01}^{0.03} -1\cos\left(\frac{2\pi nt}{0.06}\right)dt + \int_{0.03}^{0.05} 4\cos\left(\frac{2\pi nt}{0.06}\right)dt\right) \tag{6.2.22}$$

$$A_n = \frac{1}{\pi n}\left(-\sin\left(\frac{2\pi nt}{0.06}\Big|_{-0.01}^{0.03}\right) + 4\sin\left(\frac{2\pi nt}{0.06}\Big|_{0.03}^{0.05}\right)\right) \tag{6.2.23}$$

$$A_n = \frac{1}{\pi n}\left(-5\sin(\pi n) + \sin(-\pi n/3) + 4\sin\left(\frac{10\pi n}{6}\right)\right) \tag{6.2.24}$$

$$B_n = \frac{2}{0.06}\left(\int_{-0.01}^{0.03} -1\sin\left(\frac{2\pi nt}{0.06}\right)dt + \int_{0.03}^{0.05} 4\sin\left(\frac{2\pi nt}{0.06}\right)dt\right) \tag{6.2.25}$$

$$B_n = \frac{1}{\pi n}\left(\cos\left(\frac{2\pi nt}{0.06}\Big|_{-0.01}^{0.03}\right) - 4\cos\left(\frac{2\pi nt}{0.06}\Big|_{0.03}^{0.05}\right)\right) \tag{6.2.26}$$

$$B_n = \frac{1}{\pi n}\left(5\cos(\pi n) - \cos(-\pi n/3) - 4\cos\left(\frac{10\pi n}{6}\right)\right) \tag{6.2.27}$$

The Matlab code for recreating the Fourier series is included in the in-class examples.

Possibly include? Consider the other option of phase shifting the signal:

$$y(t-0.03) = \begin{cases} -1 & -0.04 \le (t-0.03) < 0.0 \\ 4 & 0.0 \le (t-0.03) < 0.02 \end{cases}$$

The integration then becomes a little easier:

$$\begin{aligned} C_n &= \frac{2}{0.06}\left(\int_{-0.04}^{0.0} -1\cos\left(\frac{2\pi nt}{0.06}\right)dt + \int_{0.0}^{0.02} 4\cos\left(\frac{2\pi nt}{0.06}\right)dt\right) \\ C_n &= \frac{1}{\pi n}\left(-\sin\left(\frac{2\pi nt}{0.06}\Big|_{-0.04}^{0.0} + 4\sin\left(\frac{2\pi nt}{0.06}\Big|_{0.0}^{0.02}\right)\right) \\ C_n &= \frac{1}{\pi n}\left[0 + \sin\left(2\pi n\left(-\frac{2}{3}\right)\right) + 4\sin\left(2\pi n\left(\frac{1}{3}\right)\right) - 0\right] \end{aligned}$$

where the unique phase will be $\phi_n = \frac{2\pi n}{0.06}(-0.03)$ which would recreate the correct A_n and B_n coefficients

Example Matlab Code

To reproduce Fourier Series displaying the updated signal as harmonic n increases:

```
clear all; close all;

% Defining signal over period with display
t = [0,3,3,5,5,6]/100;
T = max(t)-min(t);
V = [-1,-1,4,4,-1,-1];
plot(t,V,'r-','LineWidth',2.5)
hold on;
set(gca,'FontSize',14,'XTick',[-8:2:8]/100,'TickLength',[0.035,0.025],'XMinorTick','on','YMinorTick','on','GridLineStyle','--');
xlabel('t (sec)');
ylabel('y(t) (cm)');
axis([-T*1-.5/100,1.5*T+.5/100,-1.5,4.5])
grid on
for i = [-2,-1,1,2]
plot(t+i*T,V,'r-','LineWidth',2.5);
end
pause;

% Create time domain array of the estimating signal
ts = [-8:.01:10]/100;
yt = 2/3*ones(size(ts));
h = plot(ts,yt,'k','Linewidth',2.0);

% Repeat calculation of coefficients at higher harmonics
for i = 1:1:30
BB = 1/(pi*i)*(5*cos(pi*i)-cos(-pi*i/3)-4*cos(10*pi*i/6))
AA = 1/(pi*i)*(-5*sin(pi*i)+sin(-pi*i/3)+4*sin(10*pi*i/6))
% sum with prior signal
yt = yt + AA*cos(2*pi*i*ts/T)+BB*sin(2*pi*i*ts/T);
% display updating result
set(h,'Ydata',yt);
title(['Harmonic n = ',num2str(i)])
drawnow;
if(i==4 | i==10)
pause
```

```
else
pause(0.1)
end
end
```

6.2: Conversion of Function is shared under a CC BY-NC 4.0 license and was authored, remixed, and/or curated by LibreTexts.

6.3: Discrete data conversion and filtering

D. Usefulness with discrete data

1. Also possible to use complex exponential and Euler formula

$$e^{\pm i\theta} = \cos\theta \pm i\sin\theta \tag{6.3.1}$$

$$\cos\theta = \frac{[e^{i\theta} + e^{-i\theta}]}{2} \tag{6.3.2}$$

$$\sin\theta = \frac{[e^{i\theta} - e^{-i\theta}]}{2} \tag{6.3.3}$$

2. Complex formulation simplifies multiplication of real and imaginary components through complex conjugates
3. When the integration is expanded beyond single period to ∞, then conversion of data from time domain to frequency domain possible

$$Y(\omega) = \int_{-\infty}^{+\infty} y(t)e^{-i\omega t}dt \tag{6.3.4}$$

? Exercise 6.3.1

a. What does plot of simple periodic appear in frequency domain?

Answer

Show single stem plot.

? Exercise 6.3.2

b. What happens if multiple frequencies are present?

Answer

Image of multiple stems of multi-frequency image.

4. Tool for discrete data is fast Fourier transform (FFT)
   a. developed in 1940s when computation was laborious
   b. divides data in half and sums over $N/2$ steps instead of N^2 process
   c. Simple *fft* call in Matlab

E. Using the FFT

1. Input is the real data array collected of length N points
2. Output is equal length complex (real + imaginary) array
   a. similar to A_n and B_n of Fourier coefficients when $y(t)$ was known
   b. BUT exists at ALL frequency points, not just harmonics of integer frequency value $n\omega$
3. Displaying the data on plot
   a. Horizontal axis is now cyclic frequency (f)
   b. Lowest frequency: at $f_{min} = 0$ the peak represents mean value
   c. Largest frequency is resolution of smallest time step $f_{max} = 1/\delta t$
   d. Frequency increment is total time of collection

$$df = \frac{1}{N\delta t} = \frac{1}{t_{end} - t_0} \tag{6.3.5}$$

   e. Which part to plot, real or imaginary?
      - Can use both through absolute value:

- Can use both through absolute value:

$$z = |x + yi| = \sqrt{x^2 + y^2} \tag{6.3.6}$$

$$C = |Y(\omega)| = \sqrt{\Re(Y(\omega))^2 + \Im(Y(\omega))^2} \tag{6.3.7}$$

f. Since no data are known *between* the df, acts as a comb function (*stem* plot in Matlab) rather than continuous curve

g. Note that the FFT causes a folding symmetry about the $N/2$ point

h. Hence, magnitude of stem is *half of amplitude* at that particular frequency

4. Introduces potential for amplitude ambiguity

a. When full period is collected, single peak appears AT particular frequency

b. Portions of periods cause spread of peak to adjacent frequencies

? Exercise 6.3.3

Figures need to be incorporated through example

Answer

Add texts here. Do not delete this text first.

c. Acts similarly to the statistics functions from Section 2.1 where time range affected value

d. How to reduce ambiguity? Increase total time to make ambiguity a smaller percentage of overall data collected (i.e., have MORE total periods).

e. What about signals with discontinuities (i.e., ∞ frequencies)? There will always be spreading.

F. Filtering in frequency domain

1. With raw data in temporal or spatial domain, the Fourier Transform converts to a frequency

2. Recall types of filtering from MEE 321 (vibrations) and/or MEE 322 (dynamic systems and control)

a. Describe a band-pass filter in terms of low-pass versus high-pass conditions

b. Analogy of window of visibility into a room

3. Using the FFT data, user may select either:

a. *particular* frequencies that stand out (include side shoulders for amplitude ambiguity described from Section 6.2.E)

b. *amplitudes* above some noise level

4. Practical considerations that must be implemented:

a. ***Must*** utilize FFT symmetry though

i. every f_r retained, must also include $f_{max} - f_r$

ii. other frequency domain values made zero in both real and imaginary domains

b. Use the inverse fast Fourier transform (*ifft*) to return to time domain

- Output will retain complex number elements, but only the real portion has value (imaginary should be $\rightarrow 0$)

c. Plotting noisy versus filtered data in Matlab examples

Matlab script

```
clear all; close all;
Input_freq = 78; % Hz
num_cycs = 5;
mth_order = 10;
omega = Input_freq*2*pi;
T = 1/Input_freq;
final_time = num_cycs*T+0*T;
dt = final_time/(2^mth_order);
t = dt:dt:final_time;
```

```
N = length(t);
V_org = (sin(omega*t));
V = V_org+1.5*randn(1,N);
plot(t,[V;V_org],'.');
pause; % all of the previous stuff is just building data;
% Yours will come from the digital data acquisition

vbar = mean(V); % simple statistics
sigma_vbar = std(V);

X = 1/N*fft(V); % Converting to frequency domain
df = 1/final_time; % step sizes for frequency plotting
f = df*(0:N-1); % creating the frequency array
Xvals = abs(X); % Combining the real and imaginary components
figure(2);
stem(f,Xvals); % Plotting in frequency domain
hold on;

low_end = 0.4*Input_freq; % Defining range of frequencies to be kept
high_end = 1.8*Input_freq;
I = find(f<low_end | (f >high_end & f < df*N/2) | ...
   (f>(max(f)-low_end)& f>=df*N/2) | (f>=df*N/2 & f<(max(f)-high_end)));
% This selection of the frequencies is critical. You want values in the
% window both for range f<N/2 and the range f>N/2;
stem(f([I]),Xvals([I]),'g');
xlabel('Frequency (Hz)');
pause;
xlim([0,high_end*2.5]);
pause;

filtered_X = X;
filtered_X(I) = 0+0i;
filtered_V = real(N*ifft(filtered_X)); % inverting back to the time domain
figure(1); hold on;
plot(t,filtered_V,'ro'); % plotting the filtered data
ylim([-2,2]);
```

6.3: Discrete data conversion and filtering is shared under a CC BY-NC 4.0 license and was authored, remixed, and/or curated by LibreTexts.

CHAPTER OVERVIEW

7: Probability and Statistics

To further improve communication of experimental results, the combination of probability (expected outcomes) with statistics (methodology to reduce data) are covered.

Learning Objectives

- Understand the meaning of central moment of probability density functions through calculation
- Convert finite quantity of data to have indication of true value of observation

▼ A. Introduction of Terminology

1. Probability – likelihood of particular event occurrence
2. Statistics – post processing analysis tool for interpretation of probability
3. Samples – discrete data points of collection in statistics that indicate probability
4. Ensemble – Set of repeated sampling procedures where statistics are applied across trial versus time or position domain
5. 7.1: Contextual background resulting in histogram
   7.2: Central moments of probability density function
   7.3: Probability from Gaussian distribution
   7.4: Analysis with finite statistics

For additional material from sources external to NIU, please visit the Appendix.

7: Probability and Statistics is shared under a CC BY-NC 4.0 license and was authored, remixed, and/or curated by LibreTexts.

7.1: Contextual background resulting in histogram

B. Types of probability

1. Rudimentary version is percentage of occurrences of particular events relative to total possible events

- **Exercise 7.1.1**

  Probability of drawing 7 of spades

  Answer

  1 out of 52

Exercise 7.1.2

Probability of drawing 7 of ANY suit

Answer

4 of 52; equivalently 1 of 13

Exercise 7.1.3

Outcome for rolling of single die

Answer

1 of 6

Exercise 7.1.4

Probability of rolling doubles on pair of dice

Answer

6 of 36 or equivalently 1 of 6

Exercise 7.1.5

Likelihood of going to jail in game of Monopoly

Answer

(1 of 6)^3 or 1 of 216

2. Combination of sets of outcomes

   a. Union and intersection -- adding of new events that overlap ... but must eliminate repeat counts

   b. Conditional probability -- likelihood given prior event occurrence (i.e., quarantine following international travel)

   - Not applicable to independent random data

   c. Permutation - n different outcomes taken m at a time

      i. Use of factorial function !

$$P_m^n = \frac{n!}{(n-m)!} \tag{7.1.1}$$

ii. Order of events matter in this scenario

d. Combination - n different outcomes taken m *independently* ordered times

a. Reduces number of possibilities

$$C_m^n = \frac{n!}{m!(n-m)!} \tag{7.1.2}$$

b.

? Exercise 7.1.6

How many ways can one create a sandwich combination?

- List of original options: 2 patties, 1 cheese, lettuce, tomato, special sauce, middle bread ...
- Pairing options together: proteins, veggies, condiments, containment, other

Answer

Add texts here. Do not delete this text first.

C. Representing statistical information

1. Sample versus population

a. Population is the whole/complete collection (i.e., Census 2020)

b. Sample is subset of the selected elements of the population

c. Creates different variables with unique identifiers

Definition: True versus sample

True Value	Sample statistics (with *inference* of "true")
x' = true mean of whole population	$\bar{x}$ = calculated mean from sample dataset
σ^2 = true variance	S_x^2 = sample variance

2. Histogram representation

a. Scatter of data counts as function of another variable

i. Uniform bin widths standard for statistical analysis

ii. non-uniform bins show equal probability, but changing ranges

iii. number of bins are user specified for representation

b. Uniform width histogram procedure

i. Determine total range $x_{range} = x_{max} - x_{min}$

ii. Select k number of intervals

iii. Define interval width $\Delta x = x_{range}/k$

iv. count occurrences n_j of data in each Δx

v. Verify sum of n_j equals N samples

vi. Plot n_j versus the midpoint of each Δx section

c. Change which aspect for uniform height? (leave for homework assignment)

3. Frequency distribution

a. Same as histogram but scale n_j/N

b. Creates dimensionless term rather than integer count

c. As $N \rightarrow \infty$ and \(\Delta x \rightarrow 0), the discrete device changes to continuum of probability density function (pdf)

d. Can use continuous form $p(x)$ (probability as function of value or range) for further characterization

D. Probability Density Function

1. Distribution of values for a random variable x

2. Probability that a specific value x^* occurs is $p(x^*)$
   a. Positive value $p(x) > 0$ everywhere
   b. Sum of all probability is unity

$$\int_{-\infty}^{+\infty} p(x)dx = 1 \tag{7.1.3}$$

   indicating that *area* under the pdf must equal 1
3. Relates to probability *distribution* function by integrating to limits
   a. Summing probability over a range

$$P\left[x_0 \leq x \leq x^*\right] = \int_{x_0}^{x^*} p(x)dx \tag{7.1.4}$$

   Recall that x_0 and x^* are specific fixed values while x is the variable of integration
   b. From plotting, the $p(x)$ will always be positive, less than 1, but its slope will have positive and negative rates
   c. Plots of $P\left[x_0 \leq x \leq x^*\right]$ always have positive slope but *bounded* by 0 and 1

7.1: Contextual background resulting in histogram is shared under a CC BY-NC 4.0 license and was authored, remixed, and/or curated by LibreTexts.

7.2: Central moments of probability density function

E. Calculating the Central Moment of pdf

1. Expected value function $E[x]$
   a. Similar feature to indicate mean value for the variable
      i. What is the expected value of 3? $E[3] = 3$
      ii. How would you do that for a random variable x?
         - $E[x]$ best guess would be the mean value
      iii. Allows the argument to be multiplied by the pdf

$$E[f(x)] = \int_{-\infty}^{+\infty} f(x)p(x)dx \tag{7.2.1}$$

         - $f(x)$ is the value person is adding up
         - $p(x)$ is fraction of cases when $f(x)$ occurs

   b. Expectation function is a linear operator:
      i. Distribution law applies: $E[a+b] = E[a] + E[b]$
      ii. Multiply by a constant: $E[7c] = 7E[c]$ when c is random variable
   c. Used to find the central moments of pdf μ_m through

$$\mu_m = E[(x-x')^m] = \int_{-\infty}^{+\infty} (x-x')^m p(x)dx \tag{7.2.2}$$

   where $f(x) = (x-x')^m$ as the argument of the expected value function

2. Zero$^{\text{th}}$ moment ($m = 0$)

$$\mu_0 = E[(x-x')^0] = \int_{-\infty}^{+\infty} (x-x')^0 p(x)dx \tag{7.2.3}$$

$$\mu_0 = E[1] = \int_{-\infty}^{+\infty} 1p(x)dx = 1 \tag{7.2.4}$$

   - Implies that area under pdf must have value of 1

3. First moment ($m = 1$)
   a. Using expectation function first

$$\mu_1 = E[(x-x')^1] = E[x] - E[x'] \tag{7.2.5}$$

   Which of those is constant? If expectation is like mean of random variable

$$\mu_1 = E[x] - x' = \mathbf{0} \tag{7.2.6}$$

   b. Using in pdf form instead

$$\mu_1 = \int_{-\infty}^{+\infty} (x-x')^1 p(x)dx \tag{7.2.7}$$

$$\mu_1 = \int_{-\infty}^{+\infty} xp(x)dx - x' \int_{-\infty}^{+\infty} p(x)dx = 0 \tag{7.2.8}$$

   c. In order for pdf version to work, must define true mean as

$$x' \equiv \int_{-\infty}^{+\infty} xp(x)dx \tag{7.2.9}$$

   d. If $m = 0$ defined area as 1, then x' acts as horizontal centroid of the area

4. Second central moment ($m = 2$)
   a. Using the linearity of expectation function again

a. Using the linearity of expectation function again

$$\mu_2 = E[(x - x')^2] = E[x^2 - 2xx' + x'^2] \tag{7.2.10}$$

$$\mu_2 = E[x^2] - 2x'E[x] + x'^2 \tag{7.2.11}$$

$$\mu_2 = E[x^2] - x'^2 \equiv \sigma^2 \tag{7.2.12}$$

where σ^2 is the variance of the random data

b. Same expansion could be applied to pdf

$$\mu_2 = \int_{-\infty}^{+\infty} (x - x')^2 p(x) dx = \sigma^2 \tag{7.2.13}$$

c. Note the different terms:

i. *variance* σ^2 is square of units relative to the mean

ii. *standard deviation* σ is in the **same** units of random variable

iii. *mean square* is the non-centered square of the random variable

$$\psi^2 = \int_{-\infty}^{+\infty} x^2 p(x) dx = \sigma^2 + x'^2 \tag{7.2.14}$$

5. Third central moment ($m = 3$)

$$\mu_3 = E[(x - x')^3] = \int_{-\infty}^{+\infty} (x - x')^3 p(x) dx \tag{7.2.15}$$

a. Defines the *skewness* or asymmetry of the pdf

b. $Sk = \frac{\mu_3}{\sigma^3}$ making it dimensionless and signed value

c. Right shifted (tail stretches to the right) $Sk > 0$

d. Left shifted when $Sk < 0$

e. Balanced when $Sk = 0$

6. Fourth central moment ($m = 4$)

$$\mu_4 = E[(x - x')^4] = \int_{-\infty}^{+\infty} (x - x')^4 p(x) dx \tag{7.2.16}$$

a. Defines the *kurtosis* or flatness versus peakedness of the pdf

b. $Ku = \frac{\mu_4}{\sigma^4}$ since scaled is again dimensionless

c. $Ku = 3$ for bell curve (normal distribution; next section)

d. $Ku > 3$ when sharp top peak (high derivative over the top)

e. $0 < Ku < 3$ when more flat on top

7. Drawn graphically as part of lecture

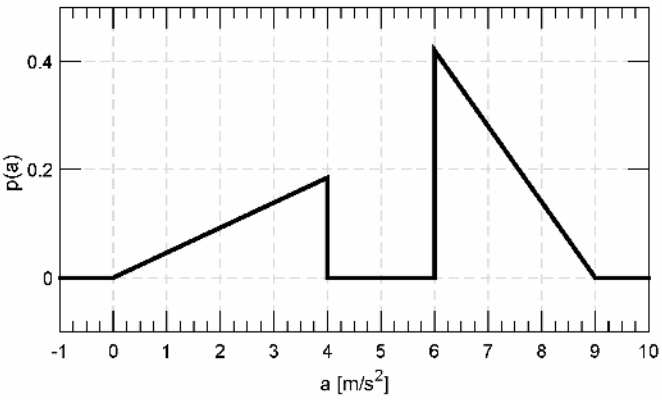
8. All moments are independent of one another except for the true mean x' since that is embedded within $m = \{2, 3, 4\}$

? Exercise 7.2.1

Consider a sailboat-like probability density function for acceleration:

$$p(a) = \begin{cases} 0 & a < 0 \\ ma & 0 \le a < 4 \\ 0 & 4 \le a < 6 \\ -0.14a + 1.26 & 6 \le a < 9 \\ 0 & a \ge 9 \end{cases} \tag{7.2.17}$$

shown in the figure:



1. What value of m is necessary for the slope to have a valid probability density?
2. Determine the true mean value a'
3. Calculate the probability acceleration falls between the range of $2.5 < a < 6.5$ m/s\(^{2}).
4. Determine the numerical values of variance, skewness, and kurtosis of the probability density function; justify the magnitude and scale of calculated results.

Answer

1. Area under curve has to add to unity:

$$\int_{-\infty}^{+\infty} p(a)da = 1 \tag{7.2.18}$$

setting up integration or area calculation of triangles

$$\int_0^4 ma\ da + \int_6^9 (-0.14a + 1.26)\ da = 1 \tag{7.2.19}$$

completing integration yields the result $m = 0.04625$

2. The distance of variable a now multiplied by probability

$$a' = \int ap(a)da = \int_0^4 0.04625a^2\ da + \int_6^9 \left(-0.14a^2 + 1.26a\right)\ da \tag{7.2.20}$$

which is equivalent to the centroid calculation of right triangles. The integration results in $a' = 5.397$ m/s^2

3. Calculate area under limited range of the function

$$\$P(2.5 < a < 6.5) = \int_{2.5}^4 0.04625a\ da + \int_6^{6.5} (-0.14a + 1.26)\ da \tag{7.2.21}$$

produces probability $P = 42\%$

4. First, the variance will need to be determined:

$$\sigma^2 = \int_0^4 ma^3\ da + \int_6^9 \left(-0.14a^3 + 1.26a^2\right)\ da - a'^2 \tag{7.2.22}$$

making $\sigma^2 = 5.021$ m^2/s^4

5. The value of skewness is

$$Sk = \left[\int_0^4 (a - a')^3 ma\ da + \int_6^9 (a - a')^3(-0.14a + 1.26)\ da\right]/\sigma^3 = -0.547 \tag{7.2.23}$$

indicating a left shift due to $Sk < 0$ (longer stretch to the left relative to 5.397)

6. The value of kurtosis is

$$Ku = \left[\int_0^4 (a - a')^4 ma \; da + \int_6^9 (a - a')^4 (-0.14a + 1.26) \; da \right] / \sigma^4 = 1.91 \tag{7.2.24}$$

indicating somewhat "flat" form near the mean region of the pdf.

7.2: Central moments of probability density function is shared under a CC BY-NC 4.0 license and was authored, remixed, and/or curated by LibreTexts.

7.3: Probability from Gaussian distribution

F. Using standard probability density functions

1. Some unique forms of pdfs
   a. Bernoulli -- only two f possible outcomes, but probability can be unequal
   b. Binomial -- multiple outcomes are possible, but distribution is uniform
   c. Poisson -- number of event opportunities becomes so large that likelihood is extremely small (i.e., earthquake → tidal wave → coastal nuclear power plant → sea wall that is too short \ldots)
   d. Weibull -- time to failure of component (i.e., 99.999999% success of devices on spacecraft)
   e. Normal (Gaussian) -- standard bell curve
   f. Student t -- like normal, but with finite number of data; sample instead of continuous population
2. Normal distribution function (also referred to as Gaussian)
   a. If assuming an infinite number of possibilities ($-\infty < x < \infty$ AND infinite number of decimal points), then function is

$$p(x) = \frac{1}{\sigma\sqrt{2\pi}} \mathrm{e}^{\frac{-(x-x')^2}{2\sigma^2}} \tag{7.3.1}$$

   where x' is the true mean, σ is the standard deviation, and π is an irrational but cool number NOT always related to circles
   b. Sketch is a bell curve with 1st central moment around x'
   c. Follows Central Limit Theorem: when a random process results from the summation of many elementary random processes, the result tends toward a Gaussian distribution. Examples include:
      - Noise of raindrops on a roof
      - Size distribution of grains on sandpaper
      - Static from analog radio/TV signals
3. Using a normalized Gaussian distribution:
   a. Clarify *normalized* relative to *normal* distribution: like dividing $p(x)$ by the magnitude to get a unit vector
   b. Instead of using units of x, define the variable as *number of standard deviations* x is from the mean

$$\beta = \frac{x - x'}{\sigma} \tag{7.3.2}$$

   c. Seeking a specific value through z substitution:

$$z_1 = \frac{x_1 - x'}{\sigma} \tag{7.3.3}$$

   d. Calculate the probability of x from the functional form of the probability density function (i.e., integrate $p(x)$)

$$P(x' - \delta x \le x \le x' + \delta x) = \int_{x'-\delta x}^{x'+\delta x} p(x)dx \tag{7.3.4}$$

   e. Instead of using range δx, just define as specific values: $+x_1 = x' + \delta x$ and $-x_1 = x' - \delta x$

$$P(-x_1 \le x \le +x_1) = \int_{-x_1}^{+x_1} p(x)dx \tag{7.3.5}$$

   f. But can also use the β and z_1 if solving for the x terms

$$x = \beta\sigma + x' \tag{7.3.6}$$

$$x_1 = z_1\sigma + x' \tag{7.3.7}$$

$$dx = \sigma d\beta \tag{7.3.8}$$

$$P(-z_1\sigma + x' \le \beta\sigma + x' \le z_1\sigma + x') = \int_{-z_1\sigma+x'}^{z_1\sigma+x'} \frac{1}{\sigma\sqrt{2\pi}} \mathrm{e}^{\frac{-\beta^2}{2}} \sigma d\beta \tag{7.3.9}$$

g. Since x' everywhere as a shift, it can be removed

h. Additionally, there is a symmetry to the function to make it one-sided

$$P(0 \leq \beta \leq z_1) = 2\int_0^{z_1} \frac{1}{\sqrt{2\pi}} \mathrm{e}^{-\beta^2/2} d\beta \tag{7.3.10}$$

where integral is calculated as the symmetry of the cumulative distribution function (*normcdf* call in Matlab)

- Tables of values for integration to standard deviations below and above the mean
- Conversion is also possible through the similarity to error function

$$\frac{1}{\sqrt{2\pi}} \int_{-\infty}^{z_1} \mathrm{e}^{-\beta^2/2} d\beta = \frac{1}{2}\left[1 + \mathrm{erf}\left(\frac{z_1}{\sqrt{2}}\right)\right] \tag{7.3.11}$$

where the integral output is an available function call in Matlab (*erf*)

- To make it single-sided, half of the distribution is considered

$$\frac{1}{\sqrt{2\pi}} \int_{0}^{z_1} \mathrm{e}^{-\beta^2/2} d\beta = \frac{1}{2}\left[\mathrm{erf}\left(\frac{z_1}{\sqrt{2}}\right)\right] \tag{7.3.12}$$

? Exercise 7.3.1

i. Electronic capacitors have Gaussian distribution with mean of 145 μF and standard deviation of 33 μF. Compute the probability that the capacitance is:

- greater than 189 μF
- less than 120 μF
- between 121 and 167 μF

Answer

For the first scenario $z_1 = \frac{189-145}{33} = \frac{4}{3}$. Since seeking probability greater than value, the complementary cumulative is needed. From the table lookup at 1.33 decimal precision, probability is 9.17%.

Scenario #2: $z_1 = \frac{120-145}{33} = \frac{-25}{33} = -0.76$. The cumulative integration less than -0.76 standard deviations is 22.36%.

The third scenario has a top and bottom boundary for the integration:

- $z_{low} = \frac{121-145}{33} = \frac{-8}{11} = -0.73$ for the left boundary
- $z_{high} = \frac{167-145}{33} = \frac{2}{3} = 0.67$ for the right boundary

Multiple approaches could be considered:

1. Using one-sided cumulative boundary function ($\int_0^{z^*} p(x)dx$) the absolute values of the standard deviations can be used: $|z_{low}| \rightarrow 26.73\%$ probability and $|z_{high}| \rightarrow 24.86\%$ for a total of 51.59% probability
2. The difference of the full integration can be done: ($\int_{-\infty}^{z_{high}} - \int_{-\infty}^{z_{low}}$) where the lookup tables would have 74.86 - 23.27 = 51.59%
3. Use the edges of the integration subtracted from unity ($1 - \int_{-\infty}^{z_{low}} - \int_{z_{high}}^{\infty}$) where the last term is the complementary cumulative: 100 - 23.27 - 25.14 = 51.59%

7.3: Probability from Gaussian distribution is shared under a CC BY-NC 4.0 license and was authored, remixed, and/or curated by LibreTexts.

7.4: Analysis with finite statistics

G. Finite Statistics with Student t distribution

1. Normal distribution assumes an infinite population, but only limited samples are available for calculating the mean and variance of a sample set.
   a. Sample mean

$$\bar{x} = \frac{1}{N}\sum_{i=1}^{N} x_i \tag{7.4.1}$$

$$\bar{x} \neq x' \text{ explicitly} \tag{7.4.2}$$

   b. Sample variance

$$S_x^2 = \frac{1}{N-1}\sum_{i=1}^{N} (x_i - \bar{x})^2 \tag{7.4.3}$$

   c. Since another statistical term was used in S_x, the degree of freedom is considered

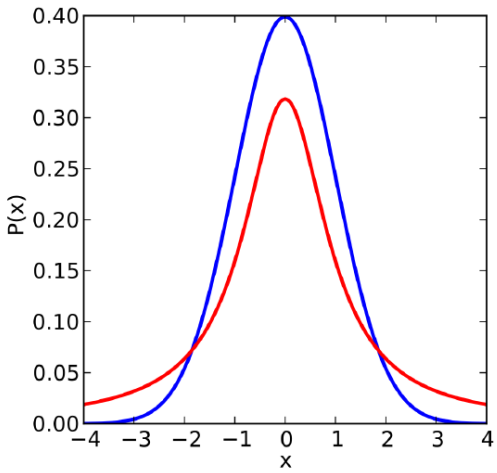
$$\nu = N - c \tag{7.4.4}$$

   where N is the number of samples and c is the number of statistical constraints used in calculations.

2. If accidentally substituting sample statistics into Normal distribution

$$p(x) = \frac{1}{S_x\sqrt{2\pi}} \mathrm{e}^{\frac{-(x-\bar{x})^2}{2S_x^2}} \tag{7.4.5}$$

   then
   a. Comparison of the distribution of normal (blue) and finite statistics (red)



   b. An overestimate of probability near sample mean
   c. Underestimates the probability far from the tail
   d. More complex function so two-sided Table format is provided

3. Normalizing the t variable (similar to z_1)
   a. Non-dimensional term

$$t = \frac{x - \bar{x}}{S_x} \tag{7.4.6}$$

   depends on two parameters
      i. ν degree of freedom (i.e., # of samples)

ii. P confidence interval of probability

b. Method of prediction of *next* selection of random variable

$$x_{N+1} = \bar{x} \pm t_{\nu,P} S_x \ (\%\ P) \tag{7.4.7}$$

i. Same confidence interval $\%\ P$ but fewer samples increases range of uncertainty

ii. Same sample size but smaller $\%\ P$ reduces range of uncertainty

iii. As $N \to \infty$, then $t_{\nu,P} \to z_p$ since the sample statistics approach infinite terms

4. Example

Exercise 7.4.1

Example of *finite* version: Consider the same capacitors but mean ($\bar{x} = 145\ \mu$F) and standard deviation ($S_x = 33\ \mu$F) statistics are determined from 31 samples.

- What is expected value of the *next* capacitor with 95% confidence?
- How would fewer data points $N < 31$ affect that range for the prediction?

Answer

From the set of $\nu = N - 1 = 30$ and confidence interval, the $t_{30,95\%} = 2.042$. Therefore

$$x_{32} = 145 \pm 2.042(33) = 77.6 \longleftrightarrow 212.4\ \mu\text{F} \tag{7.4.8}$$

If the same mean and standard deviation statistics were collected from $N = 24$ data points, $t_{23,95\%} = 2.069$ then the results would indicate

$$76.7 < x_{25} < 213.3\ \mu\text{F} \tag{7.4.9}$$

5. Using the Standard Deviation of the Means

a. The sample statistics $\bar{x}$ and S_x provide insight on the true properties x' and σ, respectively.

b. Using the N samples, the central limit theorem offers a range on the prediction of the true mean:

$$x' = \bar{x} \pm t_{\nu,P} S_{\bar{x}} \tag{7.4.10}$$

where

$$S_{\bar{x}} = \frac{S_x}{\sqrt{N}} \tag{7.4.11}$$

c. Note the difference between the next measurement versus and indication of the true mean value with the confidence interval.

d. Offers an uncertainty of x' from the statistics instead of the sensor

Exercise 7.4.2

e. From the prior example, how large is uncertainty of the true mean value from 24 samples at 99.5\% confidence?

Answer

The t multiplier is found from $t_{23,99.5\%} = 3.104$. Hence, the range of the true mean would be

$$x' = 145 \pm 3.104 \left(\frac{33}{\sqrt{24}} \right) = 145 \pm 20.91\ \mu\text{F} \tag{7.4.12}$$

The width of the uncertainty decreases as the confidence level $\%P$ similarly decreases.

H. Design of Experiments

1. Specific field of statistical methods separate from the iterative design-process questions in Chapter 1
2. Numerous methods define the mechanism to choose range of multi-variable experiments
3. Approaches available from NIST
4. Additional LibreText content available via the Appendix

7.4: Analysis with finite statistics is shared under a CC BY-NC 4.0 license and was authored, remixed, and/or curated by LibreTexts.

CHAPTER OVERVIEW

8: Laboratory Exercises

The following sections contain examples of providing fundamental background theory integrated with experimental measurands and equipment. There is no need to follow the sections sequentially as the scale and scope of the laboratory activity can align with topics in Chapters 1–7.

8.1: Digital data sampling

8.2: Stress and strain

8.3: Temperature and Heat Transfer

8.4: Fluid Flow and Forces

8: Laboratory Exercises is shared under a CC BY-NC 4.0 license and was authored, remixed, and/or curated by LibreTexts.

8.1: Digital data sampling

A. Physical property

1. Any recorded value has a limited resolution
    a. Number of digits on digital readout
    b. Number of bits for decimal resolution
    c. Handwritten recorded values
2. Storage in a digital environment is correlated:
    a. How dynamic is the input?
    b. What is the sampling and display rate of tool?
    c. Saturation of a sensor where input exceeds limit?
3. Digital data acquisition offers control controls for many features
    a. Sampling frequency: $f_s = \frac{1}{\delta t}$ defining the time interval between data points; spatial equivalent is image resolution
    b. Range of storage: if property exceeds the minima or maxima, then only storage capability is range extrema
    c. Analog-to-digital conversion: indicates number of storage divisions over range

B. Waveforms and potential digital sampling errors

1. Sinusoid: $f(t) = A\sin(\omega t)$
    a. May not locate the peak value of amplitude
        i. Sampling may not occur AT the peak
        ii. Bit resolution may not approach resolution input A
        iii. Saturation limits if storage range is less than magnitude
    b. Sampling may not store same value each cycle affecting ω calculation
        i. Consider statistics of multiple one-period frequencies
        ii. Increase number of oscillations nT where $n > 1$ to improve accuracy
2. Sawtooth: $f(t) = mt + D$ with slope m and vertical offset D
    a. Peak-to-peak range may not record at specific time at magnitude of extrema
    b. Instantaneous transition from maximum to minimum will appear with non-infinite slope over δt
3. Ramp: like sawtooth but linear ramp in both up and down direction
    a. Non-differentiable transition occurs at extrema values
4. Square wave:
    a. Constant values at extrema over some portion of oscillating period
    b. Duty cycle definition: Portion of period at maximum relative to total period

C. Activities for exploration

1. Select fixed DAQ parameters (sampling frequency, resolution, range) while adjusting waveform:
    a. Signal type
    b. Peak-to-peak range
    c. Oscillating frequency
    d. Mean offset
    e. Ramp-rates or duty cycles
    f. Combinations of multiple signals together
2. Select fixed signal type while adjusting DAQ:
    a. Slope of instantaneous changes as function of sampling frequency
    b. Extrema values while adjusting range (i.e., ± 5 V compared to 0--25 V) and resolution (8-bit versus 10–16 bit storage)
3. Experimental planning
    a. Number of scenarios to consider within a range

b. Naming convention of files (with corresponding notes for inputs)
c. Plotting data:
    i. Full time sets
    ii. Portions or multiple periods of results

8.1: Digital data sampling is shared under a CC BY-NC 4.0 license and was authored, remixed, and/or curated by LibreTexts.

8.2: Stress and strain

A. Overview with Terms

1. Stress – Internal material response as shear or normal force F per unit area A_c:
   $\sigma = \frac{F}{A_c}$
2. Strain – Elongation ∂L of material relative to unloaded length L:
   $\varepsilon = \frac{\partial L}{L}$
3. Stress–strain relationship - when in elastic regime with linear slope that does not permanently deform material:
   $\sigma = E_m \varepsilon$
   where E_m is Young's Modulus.

B. Single Strain Gage

1. Local elongation measurable with strain gage
2. Transduction via resistance in a wire: rate of flow of electrons through conductive wire:

$$R = \frac{\rho_e L}{A} \tag{8.2.1}$$

   where R is total resistance, ρ_e is resistivity based on material property, L is length of wire, and A indicates the cross sectional area. Applying the full differential:

$$\frac{dR}{R} = \frac{d\rho_e}{\rho_e} + \frac{dL}{L} - \frac{dA}{A} \tag{8.2.2}$$

   which can then incorporate Poisson's ratio ν_p for relative decrease in area due to elongation

$$\frac{dR}{R} = \frac{d\rho_e}{\rho_e} + \frac{dL}{L}(1 + 2\nu_p) \tag{8.2.3}$$

3. Gage factor (GF): reduce complexity to simple scaling factor instead of combination of material properties:
   $GF = \frac{\partial R/R}{\varepsilon}$
4. Localized measurement: strain is a local feature based on the local stress; if local stress changes due to force or area changes, corresponding changes to ε will be observed.

C. Resistance Measurement in Electronics

1. Wheatstone Bridge: tool to connect gages in mechanism that one can calculate input voltage E_i and output voltage E_o relative to changes in resistance $R_{location}$


Wheatstone Bridge

$$E_o = E_i \left(\frac{R_1}{R_1 + R_2} - \frac{R_3}{R_3 + R_4} \right) \tag{8.2.4}$$

2. Single resistor change: assuming an originally balanced bridge $R_1 = R_2 = R_3 = R_4$, the incremental output voltage δE_o is a function of small change in resistance δR:

$$\frac{\delta E_o}{E_i} = \frac{\delta R/R}{4 + 2\delta R/R} \tag{8.2.5}$$

? Exercise 8.2.1

3. Given a 1 cm x 3 cm bar with Youngs Modulus of 200×10^6 kN/m³ has an axial force of 35 kN applied.

   The strain gage has nominal resistance of 120 Ω and gage factor of 2.15. When input voltage is 9 V determine the output voltage E_o

   Answer

   Cross sectional area of axial load: $A_c = l \times w = 0.01 \times 0.03 = 3 \times 10^{-4}$ m²

   Stress calculation: $\sigma = \frac{F}{A_c} = \frac{35 \times 10^3}{3 \times 10^{-4}} = 1.167 \times 10^8$ N/m³

Strain via the Elastic Modulus: $\varepsilon = \frac{\sigma}{E_m} = 5.83 \times 10^{-4}$

Incremental change in resistance: $\delta R = GF\ R\ \varepsilon = 2.15(120)5.83 \times 10^{-4} = 0.1505\Omega$

Output voltage: $\delta E_o = E_i \left(\frac{GF\ \varepsilon}{4+2GF\ \varepsilon}\right) = 2.82$ mV

D. Bending Stress on a Beam

1. Shear and moment review

a. Shear force from distributed load:

$$V(x) = -\int W(x)dx \tag{8.2.6}$$

b. Bending moment:

$$M(x) = \int V(x)dx \tag{8.2.7}$$

c. Bending stress:

$$\sigma = \frac{-M}{I}z \tag{8.2.8}$$

where z is distance from centroid.

2. Examples of Shear-Moment Diagrams described within lecture

- Cantilever beam with load P
- Uniform load W over span $0.6H$
- Linearly decreasing load over $2B$
- Four-point and three-point bending examples

8.2: Stress and strain is shared under a CC BY-NC 4.0 license and was authored, remixed, and/or curated by LibreTexts.

8.3: Temperature and Heat Transfer

A. Physical property

1. Kinetic energy of particles
2. How is this energy transferred? Heat!
   → 0^{th} Law of Thermodynamics: material in same environment eventually attains the same temperature
3. Units - relates scales to known transition points
   a. Scale requirements
      i. Reference point - a repeatable condition (e.g., phase transition)
      ii. Size of degree - number of divisions between reference points
      iii. Interpolation mechanism - polynomial form tends to be linear
   b. Standard scales:
      i. Kelvin
         a. Reference points: Triple point of water and absolute zero
         b. Size of degrees: 273.16 K down to 0 K
      ii. Celsius
         a. Reference points: Triple point of water and boiling point at 1 atm
         b. Size of degrees: 0°C to 100°C
      iii. Fahrenheit (Some fun history)
         a. Reference points: Coldest water-salt brine to underarm temperature
         b. Size of degrees: 0°F to 96°F

B. Temperature sensors

1. Thermal expansion
   a. Bulb thermometer: liquid expands, capillary tube shows precision
   b. Bimetallic strip: different thermal expansion rates cause bending
2. Electrical Resistance (source)
   a. RTD: Resistance-temperature function:

$$R(T) = R_0 \left[1 + \alpha(T - T_0) + \beta(T - T_0)^2 ...\right] \tag{8.3.1}$$

   b. Thermistors: Exponential resistance function, rugged, fast response

$$R(T) = R_0 e^{\beta\left(\frac{1}{T} - \frac{1}{T_0}\right)} \tag{8.3.2}$$

3. Thermocouples (source)
   a. Seebeck effect: EMF generated by temperature difference in bonded dissimilar metals
   b. Key Laws:
      i. Homogeneous materials: No EMF from identical metals
      ii. Intermediate materials: No added EMF if junctions same temperature
      iii. Intermediate temperatures: EMFs add up
         - $T_1 - T_2 \Rightarrow V_{12}$
         - $T_2 - T_3 \Rightarrow V_{23}$
         - $T_1 - T_3 \Rightarrow V_{13}$
         - $V_{13} = V_{12} + V_{23}$
   c. Examples use T-type, K-type, E-type thermocouples
4. Radiative measurements (source)
   a. Non-contact, fast, infrared-based
   b. Accuracy depends on material emissivity

C. Thermodynamics and heat transfer

1. First law of thermodynamics: heat (Q) and work (W) are equivalent to

$$Q - W = \Delta U + \Delta KE + \Delta PE \tag{8.3.3}$$

Internal energy (U), kinetic (KE), and potential energy (PE)

2. Internal energy from temperature:

$$\Delta U = mC_p\Delta T \tag{8.3.4}$$

   a. Used for power ratings ($\frac{d\Delta U}{dt}$)
   b. Temperature rises with constant mass/material
   c. Response modeled like first-order system

3. Example experiments:

   a. Heat water bath to calibrate thermocouple
   b. Add ice to change temperature and mass
   c. Conductive heating of aluminum prism

8.3: Temperature and Heat Transfer is shared under a CC BY-NC 4.0 license and was authored, remixed, and/or curated by LibreTexts.

8.4: Fluid Flow and Forces

A. Velocity and Force Measurements in fluids

1. Physical properties

a. Pressure -- force per unit area

i. Result of collisions of fluid/gaseous particles with solid surfaces

ii. Pressure is always a *relative* term

$$p_{system} = p_{absolute} = p_{atmospheric} + p_{gage} \tag{8.4.1}$$

which can be drawn schematically (check lecture recording)

b. Volume flow rate -- quantity of material moving through time

i. Requires three length dimensions and time unit

ii. Material can subdivide through a control volume (or be additive if multiple sources)

iii. Has functional form

$$\dot{\underline{V}} = \int \vec{V} \cdot \hat{n} dA \tag{8.4.2}$$

where $\vec{V}$ is the velocity vector perpendicular to (hence dot product of $\hat{n}$) to local incremental area dA (confirm with dimensional check)

iv. If velocity is uniform over an area, then algebraic form of

$$\dot{\underline{V}}_{in} = -|\vec{V}|A \tag{8.4.3}$$

$$\dot{\underline{V}}_{out} = +|\vec{V}|A \tag{8.4.4}$$

c. Mass flow rate -- scale that incorporates fluid density ρ

$$\dot{m} = \int \rho \, \vec{V} \cdot \hat{n} \, dA \tag{8.4.5}$$

d. Reaction forces in fluid flow

i. Fluid must displace around solid object

ii. Aerodynamics uses high lift relative to drag to create flight

iii. Nature forms the most aerodynamic shape -- raindrop

iv. How does shape and velocity affect drag forces? Through dimensionless coefficient of drag:

$$c_{drag} = \frac{2F_{drag}}{\rho V^2 A} \tag{8.4.6}$$

2. Conservation of Mass

a. Tracking mass through Reynolds Transport Theorem

i. Since mass of a system cannot be created or destroyed $\frac{Dm_{sys}}{Dt} = 0$

ii. Full equation

$$0 = \frac{\partial}{\partial t} \int_{CV} \rho d\underline{V} + \oint_{CS} \rho \left(\vec{V} \cdot \hat{n} \right) dA \tag{8.4.7}$$

iii. Some noted features:

- $\int_{CV} \rho \, d\underline{V}$ is total mass in the control volume

- $\int_{CS} \rho \left(\vec{V} \cdot \hat{n}\right) \, dA$ is mass flow rate ($\dot{m}$) for area of interest
- dot product automatically manages the signs for inflow (negative) and outflow (positive)
- Total equation is only a scalar
- Like nodal analysis from electrical engineering: larger diameter wire can carry more current -- but charge of electron can be different due to changing density

b. Explicit assumptions/simplifications that must be stated (if used)

i. Density is constant over the volume and/or area: $\rho \int_{CV} d\underline{V}$ and $\rho \int_{CS} \left(\vec{V} \cdot \hat{n}\right) \, dA$; constant can come out of the integral

ii. Steady mass flow conditions $\Rightarrow \frac{\partial(\_)}{\partial t} = 0$; implies mass flow rate in equals mass flow rate out

iii. Uniform velocity over an area (comes out of integral): $\rho \left(\vec{V} \cdot \hat{n}\right) \int_{CS} dA = \rho \left(\vec{V} \cdot \hat{n}\right) A$

iv. Velocity projected onto control surface area: when $\vec{V} \parallel \hat{n}$ then $\left(\vec{V} \cdot \hat{n}\right) = \pm V$ with sign associated with out and in, respectively

v. Velocity components perpendicular to a control surface create no flux: $\vec{V} \perp \hat{n} = 0$

3. Conservation of Linear Momentum

a. Derivation:

i. If $B_{sys} = m\vec{V}$ (the momentum), then Reynolds Transport Theorem yields:

$$\frac{Dm\vec{V}}{Dt} = \frac{\partial}{\partial t} \int_{CV} \rho \vec{V} d\underline{V} + \oint_{CS} \rho \, \vec{V} \left(\vec{V}_{\text{rel}} \cdot \hat{n}\right) \, dA \tag{8.4.8}$$

ii. Recall Newton's Second Law of rigid body:

$$\sum \vec{F} = m\vec{a} = m\frac{d\vec{V}}{dt} \tag{8.4.9}$$

iii. Since the mass can change with time to affect momentum:

$$\sum \vec{F} = \frac{d(m\vec{V})}{dt} = m\frac{d\vec{V}}{dt} + \vec{V}\frac{dm}{dt} \tag{8.4.10}$$

iv. Thus forces can act on control volume and control surfaces:

$$\sum \vec{F} = \frac{\partial}{\partial t} \int_{CV} \rho \, \vec{V} \, d\underline{V} + \oint_{CS} \rho \, \vec{V} \, \left(\vec{V}_{\text{rel}} \cdot \hat{n}\right) \, dA \tag{8.4.11}$$

b. Notes about the equation:

i. Made of three parts:

Forces acting on drawn CV =	Rate of accumulation of momentum in CV +	Flux of momentum in and out of C

ii. Vector equation: all three directions embedded in single equation

- $\rho \vec{V} (\vec{V} \cdot \hat{n}) \Rightarrow$ Vector multiplied by 2 scalars
- Forces in x-direction are function of u-component of velocity

iii. Momentum IN is negative while momentum OUT is positive

- Sign of velocity vector is carried through though too

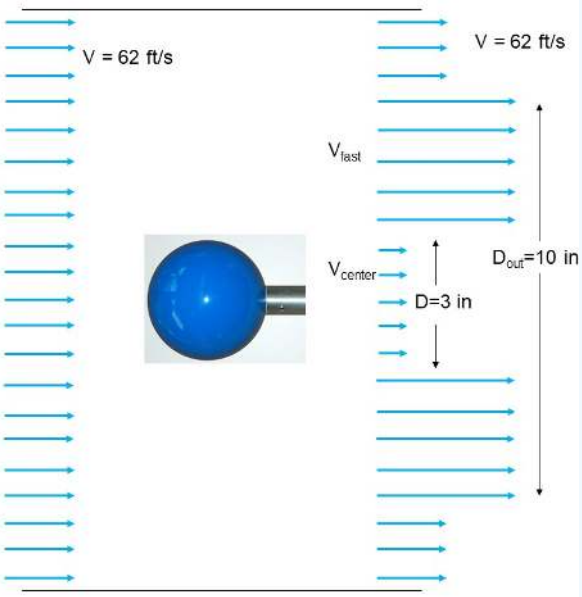
iv. Summation of forces are ALL acting on CV

- Body forces (weight of stuff inside CV)
- Surface forces (pressure normal to surface, viscosity parallel to surface, etc.)

v. All the assumptions from Conservation of Mass necessary to simplify to form $\vec{F} = \dot{m}\vec{V}$

? Exercise 8.4.1

c. The NIU wind tunnel has cross-sectional dimensions 18" by 18" and is calibrated by holding a 4" diameter sphere in the center of the test section. The incoming flow of air ($\rho = 2.3 \times 10^{-3}$ slug/ft^3) is made uniform with velocity of 62 ft/s. At a point 8 inches downstream of the sphere, velocity measurements are found to be uniform at particular radial locations. The magnitude V_{center} is 4.7 ft/s and occurs only in an area of diameter 3 in immediately behind the sphere. An annulus (i.e., donut shape) area has a uniform velocity V_{fast} that goes from the $D_{in} = 3$ in to $D_{out} = 10$ in. The remaining area of the wind tunnel's square cross-section still moves at 62 ft/s. The pressure and viscous forces are negligible and density is constant.



Determine the uniform velocities that should be measured in the annulus region along with the total drag force necessary to hold the sphere in place.

Answer

$V_{fast} = 67.67$ ft/s is the high speed flow in the annulus

$R_x = 58.658$ lb to hold the sphere in place

4. Conservation of energy (Bernoulli equation)

   a. Tracking energy through combination of static, dynamic, and hydrostatic pressure

   b. Constant equation along a streamline

$$p_{static} + \frac{1}{2}\rho V^2 + \rho g h = p_{total} \tag{8.4.12}$$

   c. Dimensional change possible through algebraic manipulation

$$\frac{P}{\rho g} + \frac{V^2}{2g} + Z = h^* = \text{Constant} \tag{8.4.13}$$

   when

   - fluid is incompressible (ρ is constant)
   - flow is steady: $\frac{\partial}{\partial t} = 0$
   - flow is frictionless ($\mu = 0$)
   - along a streamline
   - pressure and gravity are the only forces

   d. Since fluid cannot turn a corner, but pressure can, height difference in a *manometer* fluid can relate the pressure difference and associated velocity

$$V_{flow} = \sqrt{\frac{2}{\rho_{air}} \rho_{manometer} g \Delta h} \tag{8.4.14}$$

B. Sensors

1. Vane Anemometer
    a. Rotating blades serve as inverse to fan
    b. Convert rotational frequency to air speed
    c. Rated for minimum cut-in velocity
    d. Integrated temperature probe
2. Pitot Tube Anemometer
    a. Difference in stagnation versus static pressure leads to velocity
    b. Also observable in height difference of fluid (i.e., in of H_2O)
    c. Produce both analog and digital comparison
    d. May explore rotational dependence or velocity profile
3. Force gage
    a. Mount different shapes to explore drag coefficient

8.4: Fluid Flow and Forces is shared under a CC BY-NC 4.0 license and was authored, remixed, and/or curated by LibreTexts.

CHAPTER OVERVIEW

9: Supplementary Material Section

9.1: Chapter 1 Supplementary Reading Materials
9.2: Chapter 2 Supplementary Reading Materials
Discrete-Time Signal Processing
9.2.1.1: Introduction to Digital Signal Processing
9.2.1.2: The Sampling Theorem
9.2.1.3: Amplitude Quantization
9.2.1.4: Discrete -Time Signals and Systems
9.2.2: Analog-to-Digital-to-Analog Conversion
9.2.2.1: Introduction to Analog to Digital to Analog Conversion
9.2.2.2: The Sampling Theorem
9.2.2.3: Resolution and Sampling Rate
9.2.2.4: Digital-to-Analog Conversion Techniques
9.2.2.5: Analog-to-Digital Conversion
9.2.2.6: Extended Topic - Digital Signal Processing
9.2.2.7: Summary
9.2.2.8: Problems
9.2.3: Sampling and Reconstruction
9.2.3.1: Signal Sampling
9.2.3.2: Sampling Theorem
9.2.3.3: Signal Reconstruction
9.2.3.4: Perfect Reconstruction
9.2.3.5: Aliasing Phenomena
9.2.3.6: Anti-Aliasing Filters
9.2.3.7: Discrete Time Processing of Continuous Time Signals
9.3: Chapter 3 Supplementary Reading Materials
9.3.1: Importance of Statistics and Experimental Measurements
9.3.2: Total Uncertainty, Bias Error, and Propagation of Error
9.4: Chapter 4 Supplementary Reading Materials
9.5: Chapter 5 Supplementary Reading Materials
9.5.1: Undamped Free Vibrations
9.5.2: Viscous Damped Free Vibrations
9.5.3: Friction (Coulomb) Damped Free Vibrations
9.5.4: Undamped Harmonic Forced Vibrations
9.5.5: Viscous Damped Harmonic Forced Vibrations
9.6: Chapter 6 Supplementary Reading Materials
9.6.1: Introduction to the Frequency Domain
9.6.2: Complex Fourier Series
9.6.3: Classic Fourier Series
9.6.4: Fourier Series Approximation of Signals

9.6.5: Fourier Transform
9.6.6: Sampling and Reconstruction
9.6.6.1: Signal Sampling
9.6.6.2: Sampling Theorem
9.6.6.3: Signal Reconstruction
9.6.6.4: Perfect Reconstruction
9.6.6.5: Aliasing Phenomena
9.6.6.6: Anti-Aliasing Filters
9.6.6.7: Discrete Time Processing of Continuous Time Signals
9.7: Chapter 7 Supplementary Reading Materials
9.7.1: Design of Experiments
9.7.1.1: Design of Experiments via Taguchi Methods - Orthogonal Arrays
9.7.1.2: Design of experiments via factorial designs
9.7.1.3: Design of Experiments via Random Design
9.7.1.4: Summary- Summary on Control Architectures' philosophies, advantages, and disadvantages.
9.8: Chapter 8 Supplementary Reading Materials

9: Supplementary Material Section is shared under a not declared license and was authored, remixed, and/or curated by LibreTexts.

9.1: Chapter 1 Supplementary Reading Materials

9.1: Chapter 1 Supplementary Reading Materials is shared under a not declared license and was authored, remixed, and/or curated by LibreTexts.

SECTION OVERVIEW

9.2: Chapter 2 Supplementary Reading Materials

Resources developed by authors external to NIU with full prose and background materials:

Discrete-Time Signal Processing

9.2.1.1: Introduction to Digital Signal Processing
9.2.1.2: The Sampling Theorem
9.2.1.3: Amplitude Quantization
9.2.1.4: Discrete -Time Signals and Systems

9.2.2: Analog-to-Digital-to-Analog Conversion

9.2.2.1: Introduction to Analog to Digital to Analog Conversion
9.2.2.2: The Sampling Theorem
9.2.2.3: Resolution and Sampling Rate
9.2.2.4: Digital-to-Analog Conversion Techniques
9.2.2.5: Analog-to-Digital Conversion
9.2.2.6: Extended Topic - Digital Signal Processing
9.2.2.7: Summary
9.2.2.8: Problems

9.2.3: Sampling and Reconstruction

9.2.3.1: Signal Sampling
9.2.3.2: Sampling Theorem
9.2.3.3: Signal Reconstruction
9.2.3.4: Perfect Reconstruction
9.2.3.5: Aliasing Phenomena
9.2.3.6: Anti-Aliasing Filters
9.2.3.7: Discrete Time Processing of Continuous Time Signals

9.2: Chapter 2 Supplementary Reading Materials is shared under a CC BY-NC 4.0 license and was authored, remixed, and/or curated by LibreTexts.

TABLE OF CONTENTS

9.2.1.1: Introduction to Digital Signal Processing

9.2.1.2: The Sampling Theorem

9.2.1.3: Amplitude Quantization

9.2.1.4: Discrete -Time Signals and Systems

9.2.1: Discrete-Time Signal Processing is shared under a CC BY-NC 4.0 license and was authored, remixed, and/or curated by LibreTexts.

9.2.1.1: Introduction to Digital Signal Processing

Learning Objectives

- Segue for DSP chapter.

Not only do we have analog signals --- signals that are real- or complex-valued functions of a continuous variable such as time or space --- we can define **discrete/digital** ones as well. Digital signals are **sequences**, functions defined only for the integers. We thus use the notation **s(n)** to denote a discrete-time one-dimensional signal such as a digital music recording and **s(m,n)** for a discrete-"time" two-dimensional signal like a photo taken with a digital camera. Sequences are fundamentally different than continuous-time signals. For example, continuity has no meaning for sequences.

Despite such fundamental differences, the theory underlying digital signal processing mirrors that of analog signals: Fourier transforms, linear filtering and linear systems parallel what previous chapters described. These similarities make it easy to understand the definitions and why we need them, but the similarities should not be construed as "analog wannabes." We will discover that digital signal processing is **not** an approximation to analog processing. We must explicitly worry about the fidelity of converting analog signals into digital ones. The music stored on CDs, the speech sent over digital cellular telephones, and the video carried by digital television all evidence that analog signals can be accurately converted to digital ones and back again.

The key reason why digital signal processing systems have a technological advantage today is the **computer**: computations, like the Fourier transform, can be performed quickly enough to be calculated as the signal is produced, and programmability means that the signal processing system can be easily changed. This flexibility has obvious appeal and has been widely accepted in the marketplace. Programmability means that we can perform signal processing operations impossible with analog systems (circuits). We will also discover that digital systems enjoy an **algorithmic** advantage that contributes to rapid processing speeds: Computations can be restructured in non-obvious ways to speed the processing. This flexibility comes at a price, a consequence of how computers work. How do computers perform signal processing?

Note

Taking a systems viewpoint for the moment, a system that produces its output as rapidly as the input arises is said to be a **real-time** system. All analog systems operate in real-time; digital ones that depend on a computer to perform system computations may or may not work in real-time. Clearly, we need real-time signal processing systems. Only recently have computers become fast enough to meet real-time requirements while performing non-trivial signal processing.

Digital signal processing has been applied to many different areas including medicine, telecommunication, military, and scientific to mention a few. For example, the number of sun spots have been observed since the mid 1700's and analyzed to determine patterns of the frequency of their occurrence.

Example

The following is an example of digital signal processing applied to scientific data analysis. The sunspot data was obtained from the ***Solar Influences Data Analysis Center (SILSO)*** (opens in new window)CC BY-NC 4.0. Also, see the National Solar Observatory (NSO) for more information solar activity.

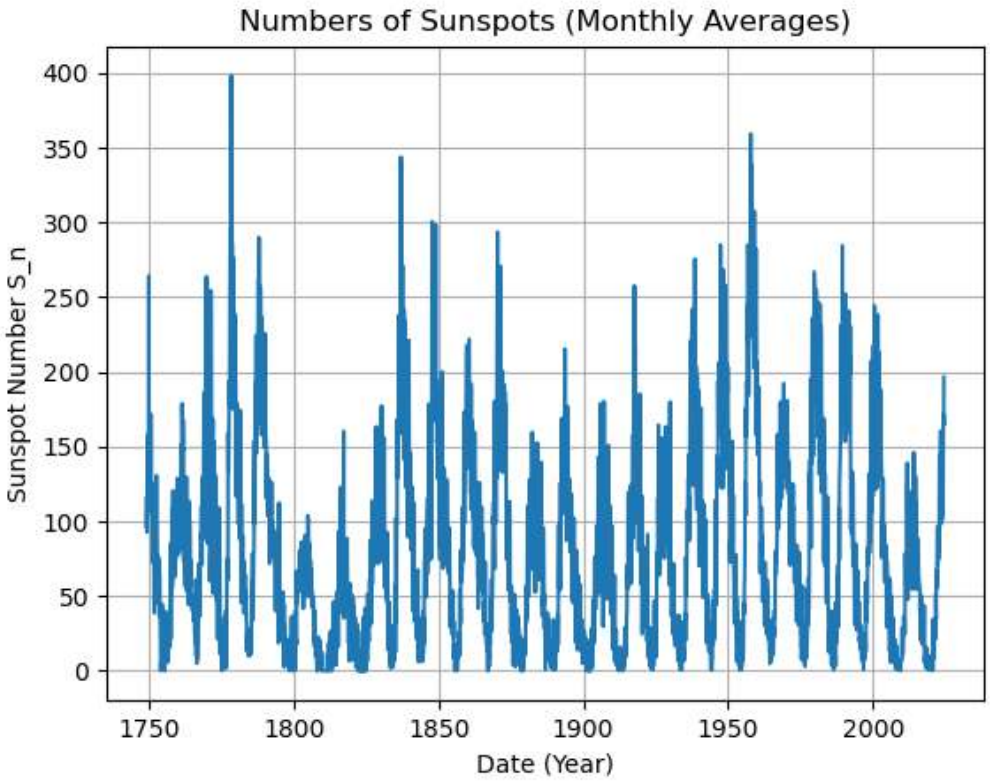


Figure 9.2.1.1.1: Sunspot Monthly Average Data. Late 1700's to early 1800's is know as the Dalton Minimum.

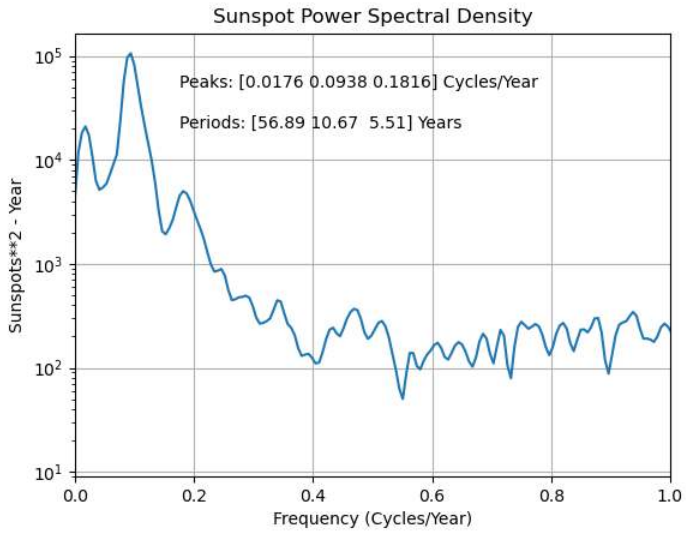


Figure 9.2.1.1.2: Power Spectral Density of Sunspot Data.

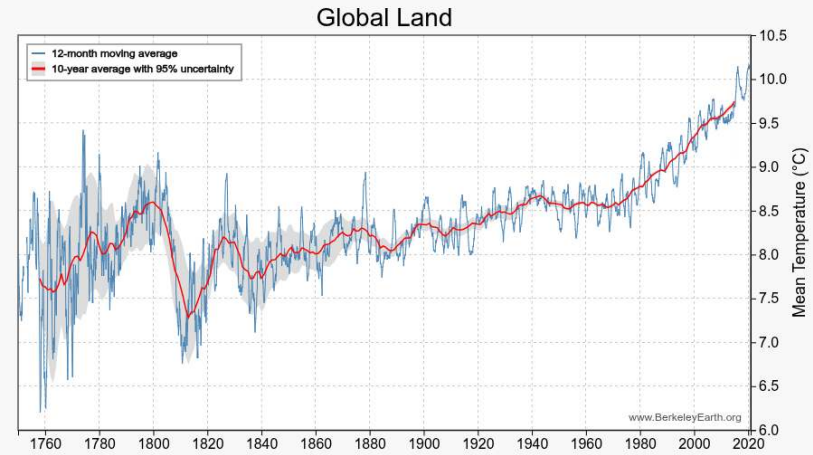


Figure 9.2.1.1.3: Global average land temperature. (Berkeley Earth Global temperature history [berkeleyearth.org])

The average number of sunspots that appeared on a monthly basis is displayed in the above graph, and the frequency analysis of these sunspots was obtained from power spectral density (PSD) analysis of this sunspot data as shown in the second graph. Not surprisingly, there was a peak in the PSD analysis at 0.0938 cycles/year for a period of 10.67 years. This corresponds to the observed cyclic variation of sunspots in the above graph. The PSD analysis also reveals that there are two additional peaks: one at 0.0176 cycles/year with a period of 56.89 years and another at 0.1816 cycles/year for a 5.51 year period. The ~57 year variation may be observed as the waxing and waning in the peak heights that occur every ~10.7 years corresponding to the completion of the sun's magnetic pole reversal. The higher frequency peak that occurs at ~5.5 year cycle, is not noticeable in the sunspot data in the above figure which is consistent with its diminished peak magnitude in the PSD analysis relative to the other two peaks.

Python Code

```
'''
    Sunspot Number - Power Spectral Density
    Download Sunspot Data and Estimate Power Spectral Density
    Solar Influences Data Analysis Center (WDC-SILSO) [https://www.sidc.be/SILSO/home]
    Royal Observatory of Belgium, Brussels
    % Input data Monthly mean total sunspot number (1/1749 - 8/2024)
    Columns separated by one or more spaces
        Column 1-2: Gregorian calendar date
            - Year
            - Month
        Column 3: Date in fraction of year.
        Column 4: Monthly mean total sunspot number.
        Column 5: Monthly mean standard deviation of the input sunspot numbers.
        Column 6: Number of observations used to compute the monthly mean total
                sunspot number.
        Column 7: Definitive/provisional marker. A blank indicates that the value is
                definitive. A '*' indicates that the value is still provisional.
'''

import pandas as pd
import numpy as np
import matplotlib.pyplot as plt
from scipy import signal
import sys
```

run | run all | restart | restart & run all

This page titled 9.2.1.1: Introduction to Digital Signal Processing is shared under a CC BY-NC 4.0 license and was authored, remixed, and/or curated by Don H. Johnson (Connections) .

- **5.1: Introduction to Digital Signal Processing** by Don H. Johnson is licensed CC BY 1.0. Original source: https://cnx.org/contents/d442r0wh9.72:g9deOnx519.

9.2.1.2: The Sampling Theorem

Learning Objectives

- Converting between a signal and numbers.

Analog-to-Digital Conversion

Because of the way computers are organized, signal must be represented by a finite number of bytes. This restriction means that **both** the time axis and the amplitude axis must be **quantized**: They must each be a multiple of the integers. [1] Quite surprisingly, the Sampling Theorem allows us to quantize the time axis **without error** for some signals. The signals that can be sampled without introducing error are interesting, and as described in the next section, we can make a signal "samplable" by filtering. In contrast, no one has found a way of performing the amplitude quantization step without introducing an unrecoverable error. Thus, a signal's value can no longer be any real number. Signals processed by digital computers must be **discrete-valued**: their values must be proportional to the integers. Consequently, **analog-to-digital conversion introduces error**.

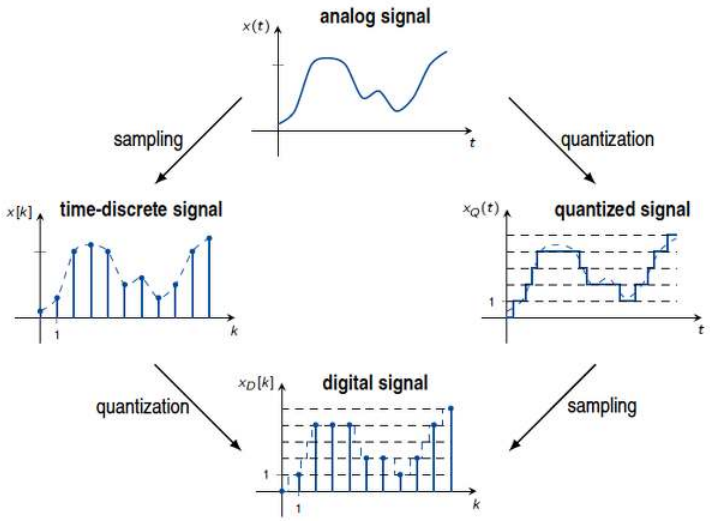


Figure 9.2.1.2.1: Analog to digital signal stages of signal representation. Sasche Spors, Digital Signal Processing - Theory and Computational Examples. Introduction [https://nbviewer.org/github/spatiala...duction.ipynb]

The Sampling Theorem

Digital transmission of information and digital signal processing all require signals to first be "acquired" by a computer. One of the most amazing and useful results in electrical engineering is that signals can be converted from a function of time into a sequence of numbers **without error**: We can convert the numbers back into the signal with (theoretically) **no** error. Harold Nyquist, a Bell Laboratories engineer, first derived this result, known as the Sampling Theorem, in the 1920s. It found no real application back then. Claude Shannon, also at Bell Laboratories, revived the result once computers were made public after World War II.

The sampled version of the analog signal **s(t)** is $\mathbf{s(nT_s)}$, with $\mathbf{T_S}$ known as the **sampling interval**. Clearly, the value of the original signal at the sampling times is preserved; the issue is how the signal values **between** the samples can be reconstructed since they are lost in the sampling process. To characterize sampling, we approximate it as the product:

$$x(t) = s(t)P_{T_s}(t)$$

with $\mathbf{P_{Ts}}$ being the periodic pulse signal. The resulting signal, as shown in Figure 5.3.1 has nonzero values only during the time intervals :

$$\left(nT_s - \frac{\Delta}{2}, nT_s + \frac{\Delta}{2}\right), n \in \{..., -1, 0, 1, ...\}$$

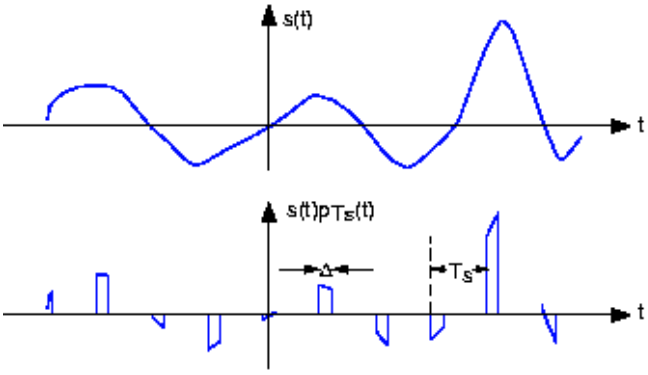


Figure 9.2.1.2.2: The waveform of an example signal is shown in the top plot with its sampled version in the bottom.

For our purposes here, we center the periodic pulse signal about the origin so that its Fourier series coefficients are real (the signal is even).

$$P_{T_s}(t) = \sum_{k=-\infty}^{\infty} c_k e^{\frac{i2\pi kt}{T_s}}$$

$$c_k = \frac{\sin\left(\frac{\pi k\Delta}{T_s}\right)}{\pi k}$$

If the properties of **s(t)** and the periodic pulse signal are chosen properly, we can recover **s(t)** from **x(t)** by filtering.

To understand how signal values between the samples can be "filled" in, we need to calculate the sampled signal's spectrum. Using the Fourier series representation of the periodic sampling signal,

$$x(t) = \sum_{k=-\infty}^{\infty} c_k e^{\frac{i2\pi kt}{T_s}} s(t)$$

Considering each term in the sum separately, we need to know the spectrum of the product of the complex exponential and the signal. Evaluating this transform directly is quite easy.

$$\int_{-\infty}^{\infty} s(t) e^{\frac{i2\pi kt}{T_s}} e^{i2\pi ft} dt = \int_{-\infty}^{\infty} s(t) e^{-\left(i2\pi\left(f-\frac{k}{T_s}\right)\right)} dt = S\left(f - \frac{k}{T_s}\right)$$

Thus, the spectrum of the sampled signal consists of weighted (by the coefficients $\mathbf{c_k}$) and delayed versions of the signal's spectrum Figure 5.3.2 below.

$$X(f) = \sum_{-\infty}^{\infty} c_k S\left(f - \frac{k}{T_s}\right)$$

In general, the terms in this sum overlap each other in the frequency domain, rendering recovery of the original signal impossible. This unpleasant phenomenon is known as **aliasing**.

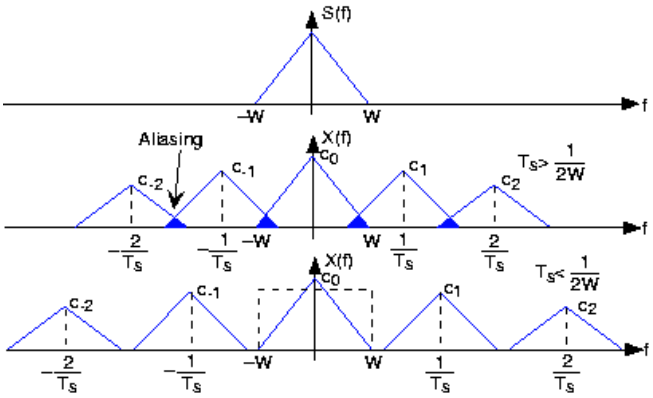


Figure 9.2.1.2.3: The spectrum of some bandlimited (to **W** Hz) signal is shown in the top plot. If the sampling interval $\mathbf{T_s}$ is chosen too large relative to the bandwidth **W**, aliasing will occur. In the bottom plot, the sampling interval is chosen sufficiently small to avoid aliasing. Note that if the signal were not bandlimited, the component spectra would **always** overlap.

If, however, we satisfy two conditions:

- The signal **s(t)** is **bandlimited**—has power in a restricted frequency range—to **W** Hz
- the sampling interval $\mathbf{T_S}$ is small enough so that the individual components in the sum do not overlap - **Ts < 1/2 W**

Aliasing will not occur. In this delightful case, we can recover the original signal by lowpass filtering **x(t)** with a filter having a cutoff frequency equal to **W** Hz. These two conditions ensure the ability to recover a bandlimited signal from its sampled version: We thus have the **Sampling Theorem**.

Example 9.2.1.2.1

The example below demonstrates the effect of under sampling a signal.

Solution

```
'''
    Aliasing Demonstration
'''
import numpy as np
import matplotlib.pyplot as plt

# Define the original signal
f_original = 5  # Frequency of the original signal (Hz)
fs = 200  # Sampling frequency (Hz)
t = np.arange(0, 1+1/fs, 1/fs)  # Time vector

# Generate the original signal
original_signal = np.sin(2 * np.pi * f_original * t)

# Undersample the signal
fs_undersampled = 4  # Undersampled sampling frequency (Hz)
t_undersampled = np.arange(0, 1+1/fs, 1/fs_undersampled)
undersampled_signal = np.sin(2 * np.pi * f_original * t_undersampled)

# Plot the signals
plt.figure(figsize=(10, 6))
plt.plot(t, original_signal, label=f'Original Signal ({f_original} Hz)')
plt.plot(t_undersampled, undersampled_signal, 'ro-',
         label=f'Undersampled Signal (fs = {fs_undersampled} Hz)')
```

run | run all | restart | restart & run all

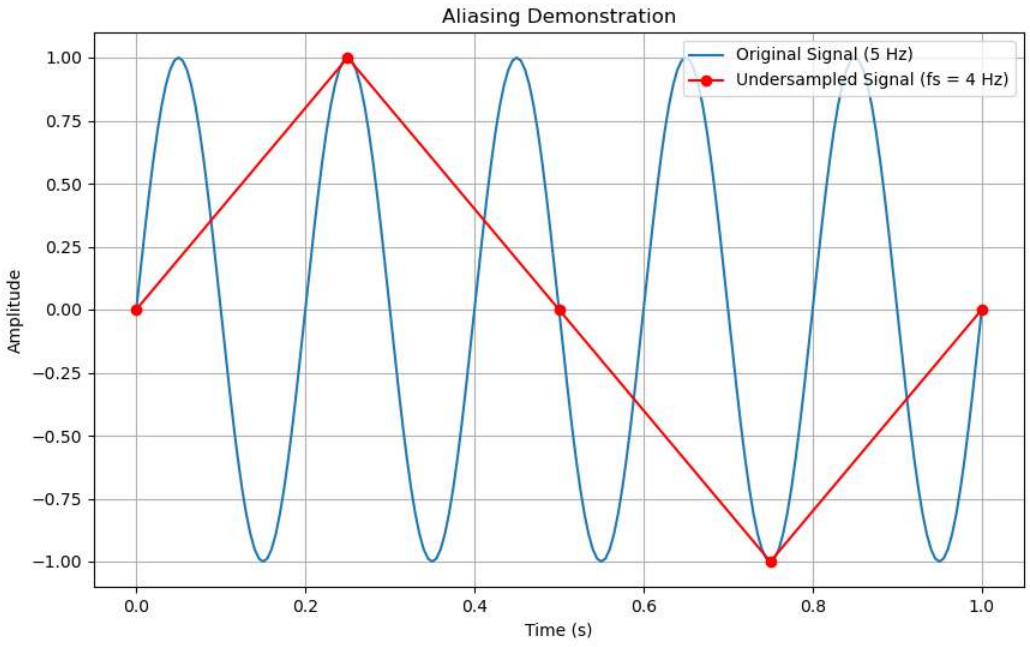


Figure 9.2.1.2.4: Aliasing example.

? Exercise 9.2.1.2.2

The Sampling Theorem (as stated) does not mention the pulse width **Δ**. What is the effect of this parameter on our ability to recover a signal from its samples (assuming the Sampling Theorem's two conditions are met)?

Solution

The only effect of pulse duration is to unequally weight the spectral repetitions. Because we are only concerned with the repetition centered about the origin, the pulse duration has no significant effect on recovering a signal from its samples.

The frequency $\mathbf{1/2T_s}$, known today as the **Nyquist frequency** and the **Shannon sampling frequency**, corresponds to the highest frequency at which a signal can contain energy and remain compatible with the Sampling Theorem. High-quality sampling systems ensure that no aliasing occurs by unceremoniously lowpass filtering the signal (cutoff frequency being slightly lower than the Nyquist frequency) before sampling. Such systems therefore vary the **anti-aliasing** filter's cutoff frequency as the sampling rate varies. Because such quality features cost money, many sound cards do **not** have anti-aliasing filters or, for that matter, post-sampling filters. They sample at high frequencies, 44.1 kHz for example, and hope the signal contains no frequencies above the Nyquist frequency (22.05 kHz in our example). If, however, the signal contains frequencies beyond the sound card's Nyquist frequency, the resulting aliasing can be impossible to remove.

If we satisfy the Sampling Theorem's conditions, the signal will change only slightly during each pulse. As we narrow the pulse, making

? Exercise 9.2.1.2.3

What is the simplest bandlimited signal? Using this signal, convince yourself that less than or equal two samples/period will not suffice to specify it. If the sampling rate $\mathbf{1/T_S}$ is not high enough, what signal would your resulting undersampled signal become? For example, let fs_undersampled = 10 (the Nyquist frequency) in Exercise 2.1.

Solution

The simplest bandlimited signal is the sine wave. At the Nyquist frequency, exactly two samples/period would occur. Reducing the sampling rate would result in fewer samples/period, and these samples would appear to have arisen from a lower frequency sinusoid.

Footnotes

1. We assume that we do not use floating-point A/D converters.

Continuous-Time Periodic Signals

A continuous-time period signal must meet the following requirement:

$$x(t) = x(t+T),\ -\infty < t < \infty \tag{9.2.1.2.1}$$

for example

$$x(t) = A\cos(2\pi F t\ +\ \theta),\ where\ T = 1/F \tag{9.2.1.2.2}$$

where F may be an irrational floating point number.

Discrete-Time Periodic Signals

A discrete-time period signal (or sequence) must meet the following requirement:

$$x(n) = x(n+N),\ -\infty < n < -\infty \tag{9.2.1.2.3}$$

where n and N are integers.

Since n and N must be integers, there an additional requirement for trigonometric functions.

$$x(n) = A\cos(\omega\, n) \tag{9.2.1.2.4}$$

is a discrete-time periodic signal if

$$\omega\, N = 2\pi\, k \tag{9.2.1.2.5}$$

where the period, N, and k are both integer, and the frequency, $f = \frac{k}{N}$, is a rational fraction.

Example 9.2.1.2.1

Is the following function periodic and if so what is the period?

$$x(n) = cos(0.6\pi n) \tag{9.2.1.2.6}$$

Solution

$$N = \frac{2\pi k}{\omega} = \frac{2\pi k}{0.6\pi} = \frac{10k}{3} \Rightarrow \frac{k}{N} = \frac{3}{10} \Rightarrow N = 10 \tag{9.2.1.2.7}$$

Note the $\frac{k}{N}$ fraction was reduced to the lowest common denominator.

```
'''
    Example of a Periodic function
'''
import numpy as np
import matplotlib.pyplot as plt

k, N = 3, 10
n = np.arange(0,2*N)

x = np.cos(2*np.pi*n*k/N)
plt.figure()
plt.stem(n,x)
plt.xlim(0,20)
plt.grid()
plt.xlabel('n')
```

```
plt.ylabel('x(n)')
plt.xticks(np.arange(0,2*N,2))
plt.show()
```

run | run all | restart | restart & run all

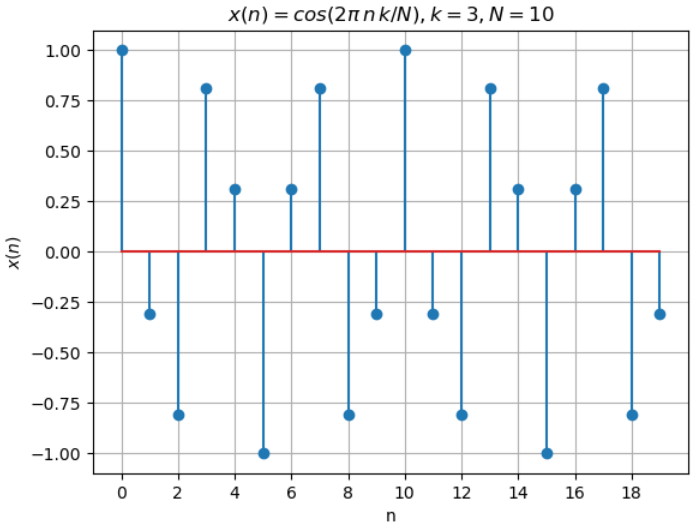


Figure 9.2.1.2.5: Example periodic sequence. Two periods.

This page titled 9.2.1.2: The Sampling Theorem is shared under a CC BY-NC 4.0 license and was authored, remixed, and/or curated by Don H. Johnson (Connections) via source content that was edited to the style and standards of the LibreTexts platform.

- **5.3: The Sampling Theorem** by Don H. Johnson is licensed CC BY 1.0. Original source: https://cnx.org/contents/d442r0wh9.72:g9deOnx519.

9.2.1.3: Amplitude Quantization

Learning Objectives

- Analog-to-digital conversion.

The Sampling Theorem says that if we sample a bandlimited signal **s(t)** fast enough, it can be recovered without error from its samples

$$s(nT_s), n \in \{..., -1, 0, 1, ...\}$$

Sampling is only the first phase of acquiring data into a computer: Computational processing further requires that the samples be **quantized**: analog values are converted into digital form. In short, we will have performed **analog-to-digital (A/D) conversion**.

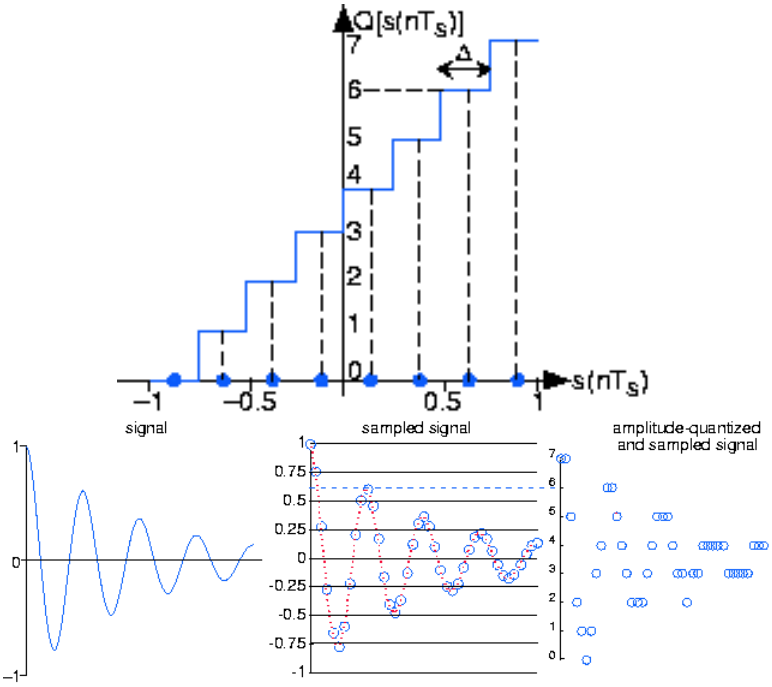


Figure 9.2.1.3.1: A three-bit A/D converter assigns voltage in the range ***[-1,1]*** *to one of eight integers between 0 and 7. For example, all inputs having values lying between 0.5 and 0.75 are assigned the integer value six, and, upon conversion back to an analog value, they all become 0.625. The width of a single quantization interval* ***Δ*** *equals* $2/2^B$. *The bottom panel shows a signal going through the analog-to-digital converter, where B is the number of bits used in the A/D conversion process (3 in the case depicted here). First, it is sampled, then amplitude-quantized to three bits. Note how the sampled signal waveform becomes distorted after amplitude quantization. For example, the two signal values between 0.5 and 0.75 become 0.625. This distortion is irreversible; it can be reduced (but not eliminated) by using more bits in the A/D converter.*

A phenomenon reminiscent of the errors incurred in representing numbers on a computer prevents signal amplitudes from being converted with no error into a binary number representation. In analog-to-digital conversion, the signal is assumed to lie within a predefined range. Assuming we can scale the signal without affecting the information it expresses, we'll define this range to be **[-1,1]**. Furthermore, the A/D converter assigns amplitude values in this range to a set of integers. A B-bit converter produces one of the integers

$$\{0, 1, ..., 2^B - 1\}$$

for each sampled input. Figure 5.4.1 shows how a three-bit A/D converter assigns input values to the integers. We define a **quantization interval** to be the range of values assigned to the same integer. Thus, for our example three-bit A/D converter, the quantization interval Δ is 0.25; in general, it is $2/2^B$.

? Exercise 9.2.1.3.1

Recalling the plot of average daily highs in this frequency domain problem, why is this plot so jagged? Interpret this effect in terms of analog-to-digital conversion.

Solution

The plotted temperatures were quantized to the nearest degree. Thus, the high temperature's amplitude was quantized as a form of A/D conversion.

Because values lying anywhere within a quantization interval are assigned the same value for computer processing, **the original amplitude value cannot be recovered without error**. Typically, the D/A converter, the device that converts integers to amplitudes, assigns an amplitude equal to the value lying halfway in the quantization interval. The integer 6 would be assigned to the amplitude 0.625 in this scheme. The error introduced by converting a signal from analog to digital form by sampling and amplitude quantization then back again would be half the quantization interval for each amplitude value. Thus, the so-called **A/D** error equals half the width of a quantization interval: $\mathbf{1/2^B}$. As we have fixed the input-amplitude range, the more bits available in the A/D converter, the smaller the quantization error.

To analyze the amplitude quantization error more deeply, we need to compute the **signal-to-noise** ratio, which equals the ratio of the signal power and the quantization error power. Assuming the signal is a sinusoid, the signal power is the square of the rms amplitude:

$$power(s) = \left(\frac{1}{\sqrt{2}}\right)^2 = \frac{1}{2}$$

The Figure 5.4.2 details a single quantization interval.

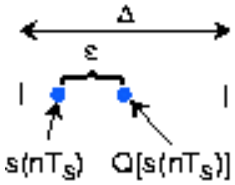


Figure 9.2.1.3.2: A single quantization interval is shown, along with a typical signal's value before amplitude quantization $\boldsymbol{s(nT_s)}$ *and after* $\boldsymbol{Q(s(nT_S))}$. $\boldsymbol{\varepsilon}$ *denotes the error thus incurred.*

Its width is Δ and the quantization error is denoted by ε. To find the power in the quantization error, we note that no matter into which quantization interval the signal's value falls, the error will have the same characteristics. To calculate the rms value, we must square the error and average it over the interval.

$$\begin{aligned} rms(\varepsilon) &= \sqrt{\frac{1}{\Delta}\int_{-\Delta}^{\Delta} \varepsilon^2 d\varepsilon} \\ &= \left(\frac{\Delta^2}{12}\right)^{\frac{1}{2}} \end{aligned}$$

Since the quantization interval width for a B-bit converter equals

$$\frac{2}{2^B} = 2^{-(B-1)}$$

we find that the signal-to-noise ratio for the analog-to-digital conversion process equals

$$SNR = \frac{\frac{1}{2}}{\frac{2^{-(2(B-1))}}{12}} = \frac{3}{2}2^{2B} = 6B + \log 1.5 dB$$

Thus, every bit increase in the A/D converter yields a 6 dB increase in the signal-to-noise ratio. The constant term **10 log1.5 equals 1.76**.

Exercise 9.2.1.3.2

This derivation assumed the signal's amplitude lay in the range **[-1,1]**. What would the amplitude quantization signal-to-noise ratio be if it lay in the range **[-A,A]**?

Solution

The signal-to-noise ratio does not depend on the signal amplitude. With an A/D range of **[-A,A]** the quantization interval

$$\Delta = \frac{2A}{2^B}$$

and the signal's rms value (again assuming it is a sinusoid) is

$$\frac{A}{\sqrt{2}}$$

Exercise 9.2.1.3.3

How many bits would be required in the A/D converter to ensure that the maximum amplitude quantization error was less than 60 db smaller than the signal's peak value?

Solution

Solving

$$2^{-B} = 0.001$$

results in

$$B = 10\ bits$$

Exercise 9.2.1.3.4

Music on a CD is stored to 16-bit accuracy. To what signal-to-noise ratio does this correspond?

Solution

A 16-bit A/D converter yields a SNR of

$$6 \times 16 + 10 \log 1.5 = 97.8dB$$

Once we have acquired signals with an A/D converter, we can process them using digital hardware or software. It can be shown that if the computer processing is linear, the result of sampling, computer processing, and unsampling is equivalent to some analog linear system. Why go to all the bother if the same function can be accomplished using analog techniques? Knowing when digital processing excels and when it does not is an important issue.

This page titled 9.2.1.3: Amplitude Quantization is shared under a CC BY-NC 4.0 license and was authored, remixed, and/or curated by Don H. Johnson (Connections) via source content that was edited to the style and standards of the LibreTexts platform.

- **5.4: Amplitude Quantization** by Don H. Johnson is licensed CC BY 1.0. Original source: https://cnx.org/contents/d442r0wh9.72:g9deOnx519.

9.2.1.4: Discrete -Time Signals and Systems

Learning Objectives

- Study of Discrete-Time signals.

Mathematically, analog signals are functions having as their independent variables continuous quantities, such as space and time. Discrete-time signals are functions defined on the integers; they are sequences. As with analog signals, we seek ways of decomposing discrete-time signals into simpler components. Because this approach leads to a better understanding of signal structure, we can exploit that structure to represent information (create ways of representing information with signals) and to extract information (retrieve the information thus represented). For symbolic-valued signals, the approach is different: We develop a common representation of all symbolic-valued signals so that we can embody the information they contain in a unified way. From an information representation perspective, the most important issue becomes, for both real-valued and symbolic-valued signals, efficiency: what is the most parsimonious and compact way to represent information so that it can be extracted later.

Real- and Complex-valued Signals

A discrete-time signal is represented symbolically as **s(n)** where **n={...,-1,0,1,...}**

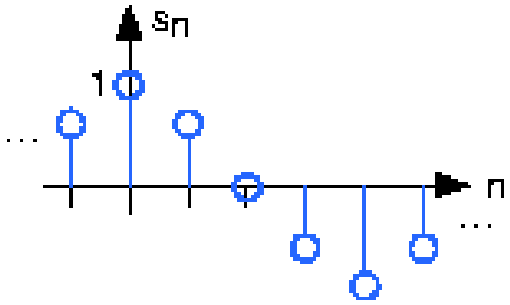


Figure 9.2.1.4.1: The discrete-time cosine signal is plotted as a stem plot. Can you find the formula for this signal?

We usually draw discrete-time signals as stem plots to emphasize the fact they are functions defined only on the integers. We can delay a discrete-time signal by an integer just as with analog ones. A signal delayed by **m** samples has the expression **s(n-m)**.

Complex Exponentials

The most important signal is, of course, the **complex exponential sequence**.

$$s(n) = e^{i2\pi fn}$$

Note that the frequency variable **f** is dimensionless and that adding an integer to the frequency of the discrete-time complex exponential has no effect on the signal's value.

$$e^{i2\pi(f+m)n} = e^{i2\pi fn}e^{i2\pi mn} = e^{i2\pi fn}$$

This derivation follows because the complex exponential evaluated at an integer multiple of

Sinusoids

Discrete-time sinusoids have the obvious form

$$s(n) = A\cos(2\pi fn + \varphi)$$

As opposed to analog complex exponentials and sinusoids that can have their frequencies be any real value, frequencies of their discrete-time counterparts yield unique waveforms **only** when **f** lies in the interval:

$$\left(-\frac{1}{2}, \frac{1}{2}\right]$$

This choice of frequency interval is arbitrary; we can also choose the frequency to lie in the interval **[0,1)**. How to choose a unit-length interval for a sinusoid's frequency will become evident later.

Unit Sample

The second-most important discrete-time signal is the **unit sample**, which is defined to be

$$\delta(n) = \begin{cases} 1 & \text{if } n = 0 \\ 0 & \text{if } otherwise \end{cases}$$

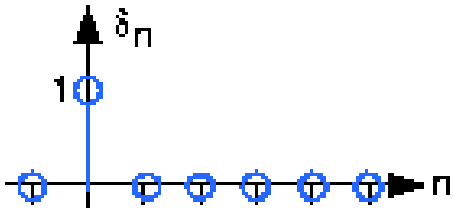


Figure 9.2.1.4.2: The unit sample.

Examination of a discrete-time signal's plot, like that of the cosine signal shown in Figure 5.5.2, reveals that all signals consist of a sequence of delayed and scaled unit samples. Because the value of a sequence at each integer **m** is denoted by **s(m)** and the unit sample delayed to occur at **m** is written **δ(n-m)**, we can decompose **any** signal as a sum of unit samples delayed to the appropriate location and scaled by the signal value.

$$s(n) = \sum_{m=-\infty}^{\infty} s(m)\delta(n-m)$$

This kind of decomposition is unique to discrete-time signals, and will prove useful subsequently.

Unit Step

The **unit step** in discrete-time is well-defined at the origin, as opposed to the situation with analog signals.

$$u(n) = \begin{cases} 1 & \text{if } n \geq 0 \\ 0 & \text{if } n < 0 \end{cases}$$

Symbolic Signals

An interesting aspect of discrete-time signals is that their values do not need to be real numbers. We do have real-valued discrete-time signals like the sinusoid, but we also have signals that denote the sequence of characters typed on the keyboard. Such characters certainly aren't real numbers, and as a collection of possible signal values, they have little mathematical structure other than that they are members of a set. More formally, each element of the **symbolic-valued** signal **s(n)** takes on one of the values **$\{a_1,\ldots,a_k\}$** which comprise the **alphabet A**. This technical terminology does not mean we restrict symbols to being members of the English or Greek alphabet. They could represent keyboard characters, bytes (8-bit quantities), integers that convey daily temperature. Whether controlled by software or not, discrete-time systems are ultimately constructed from digital circuits, which consist **entirely** of analog circuit elements. Furthermore, the transmission and reception of discrete-time signals, like e-mail, is accomplished with analog signals and systems. Understanding how discrete-time and analog signals and systems intertwine is perhaps the main goal of this course.

Discrete-Time Systems

Discrete-time systems can act on discrete-time signals in ways similar to those found in analog signals and systems. Because of the role of software in discrete-time systems, many more different systems can be envisioned and "constructed" with programs than can be with analog signals. In fact, a special class of analog signals can be converted into discrete-time signals, processed with software, and converted back into an analog signal, all without the incursion of error. For such signals, systems can be easily produced in software, with equivalent analog realizations difficult, if not impossible, to design.

Discrete-Time System Classifications

Linearity

A discrete-time system is linear if it satisfies the properties of homogeneity (scaling) and additivity

- Scaling: $\mathcal{T}[ax(n)] = a\mathcal{T}[x(n)]$
- Additivity: $\mathcal{T}[x_1(n) + x_2(n)] = \mathcal{T}[x_1(n)] + \mathcal{T}[x_2(n)]$

Time or Shift Invariant

A discrete-time systems is *shift invariant* if and only if

$y(n) = \mathcal{T}[x(n)] \Rightarrow y(n-k) = \mathcal{T}[x(n-k)]$

If the input to a system is delayed by k, the output should be the same output delayed by k.

Causality

A discrete-time system is causal if the output at any index n depends only the current and previous indices

$y(n) = \mathcal{T}[x(n), x(n-1), x(n-2), ...]$

A system that is a function of future values of the input in addition to the current and previous inputs is noncausal. A systems that is a function of only future inputs is anti-causal.

Stability

A discrete-time system is bounded input-bounded output (BIBO) stable if every bounded input produces a bounded output.

If $|x(n)| \leq K < \infty \Rightarrow |y(n)| \leq M < \infty \ \ \forall n$ then the system is BIBO stable.

Static vs Dynamic

A discrete-time system that is only dependent on the input at current index is *static* or a *memoryless* system. Systems that depend on previous or future values of the input are *dynamic* systems.

This page titled 9.2.1.4: Discrete -Time Signals and Systems is shared under a CC BY-NC 4.0 license and was authored, remixed, and/or curated by Don H. Johnson (Connections) via source content that was edited to the style and standards of the LibreTexts platform.

- **5.5: Discrete -Time Signals and Systems** by Don H. Johnson is licensed CC BY 1.0. Original source: https://cnx.org/contents/d442r0wh9.72:g9deOnx519.

SECTION OVERVIEW

9.2.2: Analog-to-Digital-to-Analog Conversion

Learning Objectives

After completing this chapter, you should be able to:

- Outline the concept of pulse code modulation.
- Detail the advantages and disadvantages of signal processing in the digital domain.
- Define the terms resolution, quantization, and Nyquist frequency.
- Define an alias, and detail how it is produced and subsequently avoided.
- Explain the operation of an $R/2R$ digital-to-analog converter.
- Explain the need for anti-alias and reconstruction filters.
- Explain the operation of a successive approximation analog-to-digital converter.
- Detail the need for and operation of a track-and-hold amplifier.
- Explain the operation of a flash analog-to-digital converter.
- Compare the different analog-to-digital converters in terms of speed, size and complexity, and detail typical applications for each.

9.2.2.1: Introduction to Analog to Digital to Analog Conversion

9.2.2.2: The Sampling Theorem

9.2.2.3: Resolution and Sampling Rate

9.2.2.4: Digital-to-Analog Conversion Techniques

9.2.2.5: Analog-to-Digital Conversion

9.2.2.6: Extended Topic - Digital Signal Processing

9.2.2.7: Summary

9.2.2.8: Problems

This page titled 9.2.2: Analog-to-Digital-to-Analog Conversion is shared under a CC BY-NC-SA 4.0 license and was authored, remixed, and/or curated by James M. Fiore via source content that was edited to the style and standards of the LibreTexts platform.

9.2.2.1: Introduction to Analog to Digital to Analog Conversion

Up to now, all of the circuits you have studied in this book were analog circuits. That is, the input waveforms were time-continuous and had infinite resolution along the time and amplitude axes. That is, you could discern increasingly smaller and finer changes as you examined a particular section. No matter whether the circuit was a simple amplifier, function synthesizer, integrator, filter, or what have you, the analog nature of the signal was always true. Fundamentally, the universe is analog in nature (at least as far as we can tell - until someone discovers a quantum time particle). Our only real deviation from the pure analog system was the use of the comparator. Although the input to the comparator was analog, the output was decidedly digital; its output was either a logic high $(+V_{sat})$ or a logic low $(-V_{sat})$. You can think of the comparator's output as having very low resolution, as only two states are possible. The comparator's output is still time-continuous in that a logic transition can occur at any time. This is in contrast to a pure digital system where transitions are time-discrete. This means that logic levels can only change at specific times, usually controlled by some form of master clock. A purely digital system then, is the antithesis of a pure analog system. An analog system is time-continuous and has infinite amplitude resolution. A digital system is time-discrete and has finite amplitude resolution (two states in our example).

As you have no doubt noticed in your parallel work, digital systems have certain advantages and benefits relative to analog systems. These advantages include noise immunity, storage capability, and available numeric processing power. It makes sense, then, that a combination of analog and digital systems could offer the best of both worlds. This chapter examines the processes of converting analog signals into a digital format and turning digital words into an analog signal. A few representative examples of processing the signal in the digital domain are presented as well. Some examples with which you might already be familiar include the stereo compact disk (CD) and the digital storage oscilloscope. We will break down this topic into two broad sections: analog-to-digital conversion (AD) and digital-to-analog conversion (DA). Since many AD systems require digital-to-analog converters, we will examine DA systems first.

12.1.1: The Advantages and Disadvantages of Working in the Digital Domain

Given enough time, an analog circuit may be designed and manufactured for virtually any application. Why, then, would anyone desire to work in the digital domain? Perhaps the major reason for working in the digital domain is the flexibility it offers. Once signals are represented in a digital form, they may be manipulated by various means, including software programs. You have probably discovered that replicating a computer program is far easier than replicating an analog circuit. What's more, a program is much easier to update and customize than a hardware circuit. Because of this, it is possible to manipulate a signal in many different ways with the same digital/computer hardware; all that needs to be altered is the manipulation instructions (i.e., the program). The analog circuit, in contrast, needs to be re-wired, and extra components need to be added or old portions removed. This can be far more costly and time-intensive than just updating software. By working in the digital domain, processing circuits do not exist per se; rather, a generic IC such as a CPU is used to create a “virtual circuit”. With a certain amount of intelligence in the system design, the virtual circuit may be able to alter its own performance in order to precisely adapt to various signals. This all boils down to the fact that a digital scheme may offer much greater flexibility for involved tasks and allows a streamlined, generic hardware solution for complex applications. Because of this attribute, the digital solution may wind up being significantly less expensive than its analog counterpart.

When an analog signal is transferred to the digital domain, it is represented as a series of numbers (usually, high/low binary logic levels). One nice property of this representation is that it is exactly repeatable. In other words, an infinite number of copies of the data may be generated, and no distortions or deviations from the original will appear. The last copy will be identical to the first. Compare this to a simple analog copy. For example, if you were to record a song with a cassette recorder and then make a copy of the tape, the second-generation copy would suffer from increased noise and distortion. A copy of the second copy would produce even worse results. Every time the signal is copied, some corruption occurs. It is for this reason that early long distance telephone calls were of such low quality. Modern communications systems employ digital techniques that allow much higher quality, even if one person is in New York and the other is in Australia, halfway around the planet.

Besides being a desirable mathematical attribute, repeatability also lends itself to the problem of long term storage. A storage medium for a binary signal only needs to resolve two levels, whereas the analog medium needs to resolve very fine changes in signal strength. As you might guess, deterioration of the analog medium is a serious problem and results in information loss. The

digital medium can theoretically survive a much higher level of deterioration without information loss. In a computerized system, data may be stored in a variety of formats including RAM (Random Access Memory) and magnetic tape or disk. For playback only (i.e., read only), data may be stored in ROM (Read Only Memory) or laser disk formats (such as DVD or audio CD).

As always, the benefits of the digital scheme arrive with specific disadvantages. First, for simpler applications, the cost of the digital approach is very high and cannot be justified. Second, the process of converting a signal between the analog and digital domains is an inexact one. Some information about the signal will be lost during the conversion. This is because the digital representation has finite resolution. This means that only signal changes larger than a certain minimum size (the resolution step size) are discernable, and therefore, some form of round-off error is inevitable. This characteristic helps determine the range of allowable signals, from the smallest detectable signal to the maximum signal before overload occurs. Third, analog systems are inherently faster than digital systems. Analog solutions can process input signals at much higher frequencies than digital schemes. Also, analog systems work in real-time, whereas digital systems might not. Digital systems can only perform in real-time if the input signal is not a very high frequency, if the processing task is not overly complex, or if specialized processing circuits are added. Not all applications require real-time performance, so this limitation is not always a problem. Also, because we can expect computing power to get less and less expensive in the coming years, cost-effective digital processing will undoubtedly expand into new areas.

This page titled 9.2.2.1: Introduction to Analog to Digital to Analog Conversion is shared under a CC BY-NC-SA 4.0 license and was authored, remixed, and/or curated by James M. Fiore via source content that was edited to the style and standards of the LibreTexts platform.

- **12.1: Introduction to Analog to Digital to Analog Conversion** by James M. Fiore is licensed CC BY-NC-SA 4.0. Original source: http://www.dissidents.com/resources/OperationalAmplifiersAndLinearICs_3E.pdf.

9.2.2.2: The Sampling Theorem

There are many different ways in which an analog signal may be turned into a digital form. This process is referred to as AD conversion or, more simply, as digitization. We will only examine the most popular method, called pulse code modulation, or PCM for short. The reverse of this process, or turning the digital information back into analog form, is called DA conversion. The two most important charracteristics in the conversion process are sampling frequency and amplitude resolution. Let's look at the basic idea behind PCM.

In essence, PCM measures and encodes the value of the input signal at specific points in time. Normally, the time spacing is constant, and several points are used over the length of one input cycle. In other words, the process involves taking representative samples of the input signal over time. This is shown graphically in Figure 9.2.2.2.1, and is referred to as sampling. The result of the sampling procedure is a list of times and corresponding amplitude values. A sequence might look something like: at $t = 1$ ms, $V_{in} = 23$ mV; at $t = 2$ ms, $V_{in} = 45$ mV; at $t = 3$ ms, $V_{in} = -15$ mV; etc. If the time interval between samples is held constant (i.e., constant rate of sampling), then all we need to know is a starting time and the sampling rate in order to reconstruct the actual sample times[1]. This is much more efficient than recording each sample time. The resulting amplitude values may be manipulated in a variety of ways since they are now in a numeric form. The ultimate accuracy of this conversion will depend on two primary factors: how often we sample the signal and the accuracy and resolution of the sample measurement. Theoretically, the conversion will never be 100% accurate; that is, once converted, a finite amount of information will be lost forever. Another way of stating this is that when the digital representation is converted back to analog, the result will not be identical to the original waveform. In practical terms, it is possible to reduce the error to such small values that it may be ignored in many applications. It is important then, that we investigate the implications of sample rate and accuracy/resolution on the quality of conversion.

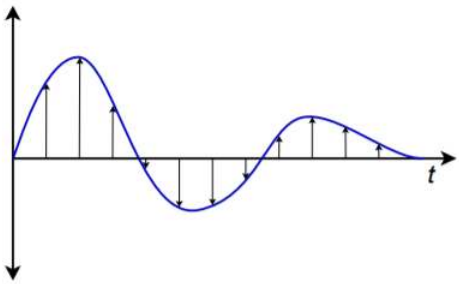


Figure 9.2.2.2.1: Sampling input signals over time.

References

[1]The terms sample rate and sample frequency are often used interchangeably, and are usually denoted by f_s.

This page titled 9.2.2.2: The Sampling Theorem is shared under a CC BY-NC-SA 4.0 license and was authored, remixed, and/or curated by James M. Fiore via source content that was edited to the style and standards of the LibreTexts platform.

- **12.2: The Sampling Theorem** by James M. Fiore is licensed CC BY-NC-SA 4.0. Original source: http://www.dissidents.com/resources/OperationalAmplifiersAndLinearICs_3E.pdf.

9.2.2.3: Resolution and Sampling Rate

Perhaps the most obvious source of error is the finite measurement accuracy of the individual signal levels. The major problem here is one of resolution. Resolution represents the finest discernible change in the signal and is often specified in terms of a number of bits, although a voltage specification is also possible. Because the signal level is represented with a binary number, it follows that a large number of bits are needed in order to achieve fine resolution. For example, if an eight-bit word is used, there will be 256 distinct values available. If the maximum peak-to-peak value of the input signal is 1 V, it works out that each step in the word represents about 3.9 mV (1 V/256). Under these conditions, it would be impossible to perfectly encode a value of 14 mV. The nearest values available would be binary 11, which yields 11.7 mV (3 times 3.9 mV), and binary 100, which yields 15.6 mV (4 times 3.9 mV). Obviously, the resulting round-off creates some error in the digital representation. This error can be reduced by increasing resolution so that finer steps may be detected. If 16 bits are used for the same 1 V range, a total of 65,536 values are available, with each step working out to 15.26 μV. Now, although we still may not be able to exactly represent the 14 mV level, we are guaranteed to be within $\pm$7.63 μV, instead of $\pm$1.95 mV as in the eight-bit case. Because the round-off errors tend to be random in magnitude and polarity, this effect may be viewed as a noise source. In other words, lower resolutions (i.e., fewer bits) produce noisier signals. The number of bits required for a particular application may vary from fewer than 6 for high speed video applications to more than 20 for high quality audio or measurement purposes.

It is important to note that once the size of the digital word is chosen and the peak amplitude fixed, the input signal must stay within specific bounds or gross distortion will occur. For example, if a peak input of 1 V produces the maximum numeric value, there is no way that the digital word can represent a level greater than 1 V. Likewise, if the step size is set at 1 mV, any signals less than 1 mV are lost. Also, low-level signals will suffer from reduced resolution. For best results, the signal peak should produce the maximum numeric value. If the peak is significantly less, the result is akin to using fewer bits in the representation. Finally, even if the resolution is adequate, the absolute accuracy of the conversion must be considered, as it is in any measurement device.

Example 9.2.2.3.1

A certain system uses a 12-bit word to represent the input signal. If the maximum peak-to-peak signal is set for 2 V, determine the resolution of the system and its dynamic range.

A 12-bit word means that 2^{12}, or 4096, levels are possible. As these levels are equally spaced across the 2 V range, each step is

$$\text{Step Size} = \frac{2V}{4096}$$
$$\text{Step Size} = 488\mu V$$

Therefore, the system can resolve changes as small as 488 μV.

Dynamic range represents the ratio of the largest value possible to the smallest.

$$\text{Dynamic Range} = \frac{2V}{488\mu V}$$
$$\text{Dynamic Range} = 4096$$
$$\text{Dynamic Range} = 20\log_{10} 4096$$
$$\text{Dynamic Range} = 72dB$$

In the preceding problem, note that the voltage range affects the step size, but does not affect the bit resolution. The actual number of discrete steps that may be resolved is set by the number of bits available. You may notice that each additional bit adds approximately 6 dB of range (a doubling of voltage). Consequently, the dynamic range calculation may be streamlined to

$$\text{Dynamic Range} \approx 6dB \times \text{ Number of Bits} \tag{9.2.2.3.1}$$

Example 9.2.2.3.2

Audio compact disks use a 16-bit representation of the music signal. Determine the dynamic range. Also, if the maximum output level is 0.775 V peak, determine the step size.

$$\text{Dynamic Range } \approx 6dB \times \text{ Number of Bits}$$
$$\text{Dynamic Range } \approx 6dB \times 16$$
$$\text{Dynamic Range } \approx 96dB$$

For 16 bits, the total number of steps is 2^{16}, or 65536. Assuming that the signal is bipolar, the total signal range will be from −0.775 V to +0.775 V, or 1.55 V.

$$\text{Step Size } = \frac{1.55V}{65536}$$
$$\text{Step Size } = 23.65\mu V$$

At this point, we must consider the effect of sampling rate on the quality of the signal. It should be intuitively obvious that higher sampling rates afford greater overall conversion accuracy. Of course, there is a trade-off associated with high sampling rates, and that is the accompanying high data rate. In other words, greater resources will be required to store and process the larger volume of digital information. The real question is, just how fast does the sampling rate need to be, for optimum efficiency? The Nyquist-Shannon sampling theorem states that at least two samples are needed per cycle for proper signal conversion. If the signal is not a sinusoid, then at least two samples are required per cycle of the highest frequency component. For example, if a range of signals up to 10 kHz needs to be digitized, then a sample rate of at least 20 kHz is required. Normally, a certain amount of “breathing room” is added to this figure. Another way of looking at this relationship is to state that the highest input-frequency component can be no more than one-half of the sampling rate. Because this is such an important parameter, the value of onehalf of the sampling rate is given the name Nyquist frequency.

$$\text{Nyquist frequency } = \frac{f_s}{2} \tag{9.2.2.3.2}$$

If an input frequency component is greater than the Nyquist frequency, a unique form of distortion called alias distortion, is produced. The resulting distortion product, called an alias, is a new signal at a frequency that is equal to the difference between the input and Nyquist frequencies. Normally, this new signal is not harmonically related to the input signal, and thus, is easily detected. The aliasing effect is shown graphically in Figure 9.2.2.3.1.

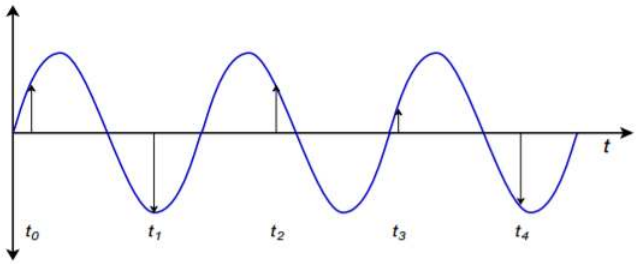


Figure 9.2.2.3.1: The aliasing effect (sampling rate too low).

Here we see a sampling rate that is only about 1.5 times the input frequency, rather than the required factor of 2 times minimum. In Figure 9.2.2.3.2 the sample points are redrawn and connected as simply as possible. Note that the resulting outline is that of a lower-frequency wave. What we notice here is that the data points produced in Figure 9.2.2.3.1 are identical to the points produced by a lower-frequency input wave. When these data points are converted back to analog form, the DA converter will produce this lower frequency wave. Oddly enough, the original waveform has completely disappeared; hence the term alias. Any signal component that is greater than the Nyquist frequency will produce aliases.

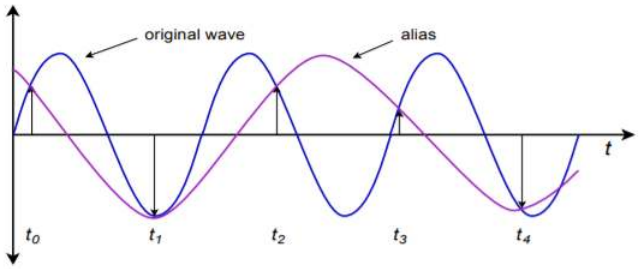


Figure 9.2.2.3.2: Alias production.

When considering the possibility of alias distortion, it is worth repeating that the components of the signal must be investigated, not just the base frequency. For example, a 1 kHz square wave has a 1 kHz fundamental and an infinite series of odd-numbered harmonics (3 kHz, 5 kHz, 7 kHz, 9 kHz, etc.). If a 12 kHz sampling rate is used to digitize this signal, a fair amount of alias distortion will be seen. In this example, the Nyquist rate is 6 kHz. All of the harmonics above 6 kHz will produce an alias. In order to prevent this, the input signal must be frequency-band limited. That is, a low-pass filter must be used to attenuate all components above the Nyquist frequency before AD conversion takes place. The amount of filtering required depends on the resolution of the conversion and the relative strength of the above-band signals. Because filters cannot roll off infinitely fast, as noted in Chapter Eleven, sampling rates are normally set more than twice as high as the highest needed input component. In this way, the Nyquist frequency will be somewhat greater than the maximum desired input frequency. The low-pass filter (often referred to as an anti-alias filter) will use this frequency range as its transition band. Even though the attenuation in the transition band is less than optimum, alias distortion will not be a problem. This is shown graphically in Figure 9.2.2.3.3. As you can see, high-roll-off rate filters are desirable in order to attenuate the out-of-band signals as quickly as possible. Very fast filter roll off rates mean that the sampling frequency need only be as little as 10% greater than the theoretical minimum in order to maintain sufficient alias rejection.

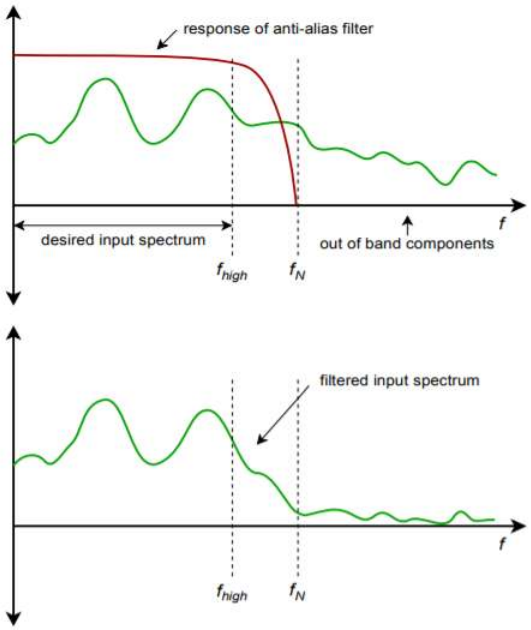


Figure 9.2.2.3.3: Filtering to eliminate aliasing effects.

Example 9.2.2.3.3

Suppose that you need to digitize telephone signals. Assuming that you would like to maintain a dynamic range of at least 50 dB, with an upper frequency limit of 3 kHz, determine the minimum acceptable sampling rate and number of bits of required. The minimum Nyquist rate is equal to the highest desired input frequency. In this case, that's 3 kHz. Because the sampling rate is twice the Nyquist frequency, the sampling rate must be at least 6 kHz. In reality, if input components above 3 kHz exist, an

anti-alias filter will be needed, and the sampling rate will have to be increased somewhat. Because dynamic range is set by the number of bits used, we find that

$$\begin{aligned} \text{Bits Required } &= \frac{\text{Dynamic Range}}{6dB} \\ \text{Bits Required } &= \frac{50dB}{6dB} \\ \text{Bits Required } &= 9 \end{aligned}$$

Because we cannot have a fractional bit, the value is rounded up. The final system specification is a minimum rate of 6 kHz with a 9-bit resolution. Note that this represents a data rate of 9 bits per sample times 6000 samples per second, or 54000 bits per second (6750 bytes per second).

This page titled 9.2.2.3: Resolution and Sampling Rate is shared under a CC BY-NC-SA 4.0 license and was authored, remixed, and/or curated by James M. Fiore via source content that was edited to the style and standards of the LibreTexts platform.

- **12.3: Resolution and Sampling Rate** by James M. Fiore is licensed CC BY-NC-SA 4.0. Original source: http://www.dissidents.com/resources/OperationalAmplifiersAndLinearICs_3E.pdf.

9.2.2.4: Digital-to-Analog Conversion Techniques

The basic digital-to-analog converter is little more than a weighted summing amplifier. Each successive bit in the digital word represents a level that is twice as large as the preceding bit. If each bit is taken as a given current or voltage, the increasing levels may be produced by using different gains in the summing inputs. A simple four-bit converter is shown in Figure 9.2.2.4.1.

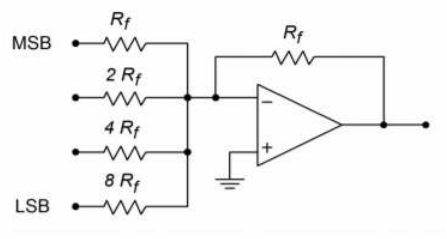


Figure 9.2.2.4.1: A simple 4-bit converter.

This system can represent 2^4, or 16, different levels. Each input is driven by a simple high/low logic level that represents a 1 or 0 for that particular bit. Note that the input resistors vary by factors of 2. The gain for the upper most path is R_f/R_f, or unity. This input is used for the most significant bit of the input word (MSB). The next input shows a gain of $R_f/(2R_f)$, or 0.5. The third input shows a gain of 0.25, and the final input shows a gain of 0.125. The final input has the lowest gain and is used for the least significant bit of the input word (LSB). If the input word had a higher resolution (i.e., more bits), extra channels would be added, each having half the gain of the preceding input. To better understand the conversion process, let's take a look at a few representative inputs and outputs.

The circuit of Figure 9.2.2.4.1 may be driven by simple 5 V TTL-type logic circuits. 5 V represents a logical high, whereas 0 V represents a logical low. What is the output level if the input word is 0100? Because a logical high represents 5 V, 5 V is being applied to the second input. All other inputs receive a logical low, or 0 V. The output is the summation of the input signals (remember, this is an inverting summer, so the final output should have its sign reversed).

$$V_{out} = -(V_{in1}A_1 + V_{in2}A_2 + V_{in3}A_3 + V_{in4}A_4)$$
$$V_{out} = -(0V \times 1 + 5V \times 0.5 + 0V \times 0.25 + 0V \times 0.125)$$
$$V_{out} = -2.5V$$

So, a value of 4 (binary 100) is equivalent to a potential of 2.5 V. If we increase the word value to 9 (binary 1001), we see

$$V_{out} = -(V_{in1}A_1 + V_{in2}A_2 + V_{in3}A_3 + V_{in4}A_4)$$
$$V_{out} = -(5V \times 1 + 0V \times 0.5 + 0V \times 0.25 + 5V \times 0.125)$$
$$V_{out} = -5.625V$$

The minimum output occurs at binary 0000, (0 V) and the maximum at binary 1111 (−9.375 V). The step size is equal to the logic level times the minimum gain; in this case that's 0.625 V. Notice that the output value may be found by simply multiplying the value of the input word by the minimum step size. Also, it is important to note that the output signal is unipolar (in this example, always negative).

A digital representation, of course, is made up of a sequence of words, not just one word. In reality, the logic circuits are constantly feeding the summing amplifier new words at a predetermined rate. Because of the changing inputs, the output of the converter is constantly changing as well. Using our previously calculated values, if the converter is fed the sequence $0000, 0100, 1001, 1111,$ the output will move from 0 V to −2.5 V, to −5.625 V, to a final value of −9.375 V. This output is graphed in Figure 9.2.2.4.2.

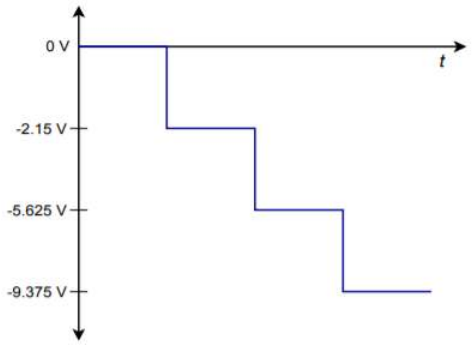


Figure 9.2.2.4.2: Output with four digital words.

If this sequence is repeated over and over, the waveform of Figure 9.2.2.4.3 is the result. Note that a “stair-step” type wave is created. You might also think of this as a very rough form of a ramp function. A better ramp would be produced if we used all of the available values for the input sequence, as in $0000, 0001, 0010, 0011, \ldots, 1111$. In order to remove the negative DC offset and make the signal bipolar, all we need to do is pass the signal through a coupling capacitor. The frequency of this waveform is controlled by the rate at which the words are fed to the converter. Note that by increasing the resolution and the number of words fed to the converter per cycle, a very close approximation to the ideal ramp function may be achieved. For that matter, by changing the input words to other sequences, we can create a wide variety of output wave shapes. This is the concept behind the digital arbitrary function generator. An arbitrary function generator allows you to create wave shapes beyond the simple sine/square/triangle found on the typical laboratory function generator. We'll take a closer look at this particular piece of test equipment a little later.

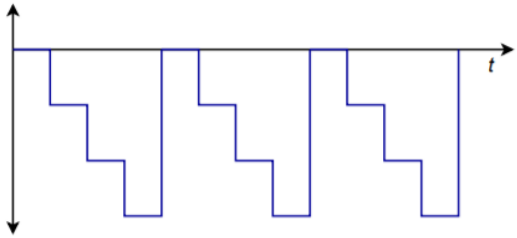


Figure 9.2.2.4.3: Cycled output.

In order to increase resolution, it appears that all you need to do to the summing amplifier is add extra channels with larger and larger resistors. Unfortunately, the resistor sizes soon become impractical and another approach is required. For example, a 16-bit system would require that the LSB resistor be equal to 65,536 R_f. One problem is that the resulting small input current may be dwarfed by input bias and offset currents. Also, high component accuracy is needed for the more significant inputs in terms of the input resistors and the drive signals. The excessively large resistors may also contribute added noise. The standard solution to this problem involves of the use of an $R/2R$ resistive divider network.

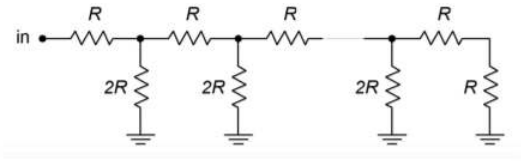


Figure 9.2.2.4.4: $R/2R$ ladder network

An $R/2R$ network is shown in Figure 9.2.2.4.4. This circuit exhibits the unique attribute of constant division by 2 for each stage. You may think of this as either a division of voltage at each successive node or a division of current in each successive leg. An example of a four stage (i.e., four-bit) network is shown in Figure 9.2.2.4.5.

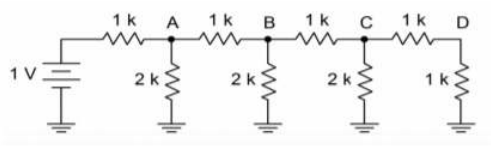


Figure 9.2.2.4.5: A 4-stage ladder.

In order to find the voltage at any given node, the loading effects of the following stages must be taken into account. This is much easier to do than it first appears. If we need to find the voltage at point A, we must first find the resistance in parallel with the initial 2 $k\Omega$ resistor. A quick inspection shows that each stage is loaded by the following stages, so it is easiest if we start at the last stage and work toward the input. The effective resistance to the right of node C is 1 $k\Omega$ in series with 1 $k\Omega$, or 2 $k\Omega$. This resistance is placed in parallel with the 2 $k\Omega$ resistor seen from node C to ground. The result is 1 $k\Omega$. In other words, from C to ground, we see 1 $k\Omega$. This creates a 2:1 voltage divider with the 1 $k\Omega$ resistor placed from B to C, so the voltage at C must be half the voltage at B. This also points up the fact that the current entering node C splits into two equal portions: one that travels towards point D, and the other that travels through the 2 $k\Omega$ resistor to ground. This is shown graphically in Figure 9.2.2.4.6.

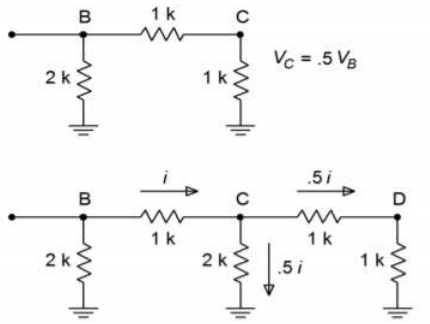


Figure 9.2.2.4.6: Ladder analysis. a. Equivalent circuit (top). b. Current division (bottom).

If you look at the equivalent circuit section of Figure 9.2.2.4.6, you will notice that this portion now looks exactly like the final portion of the original network. That is, every time a section is simplified and analyzed, the result will be a halving of voltage and current. It is already apparent that the voltage at D must be half the voltage at C, which in turn, must be half the voltage at B. As you can now prove, it follows that the voltage at B must be half the voltage at A. In a similar fashion, the current passing through each $2R$ leg is half the preceding current. (For current division, the final section is not used to derive a current since it will be equal to the value in the preceding stage.) The halving of current is just what is needed for the binary representation of the digital input word.

Adapting the $R/2R$ network to the DA converter is relatively easy. The network is fed from a stable current source, with each $2R$ element feeding into a summing amplifier. In series with each $2R$ element is a solid-state switch, which sets the appropriate logic level. This is shown in Figure 9.2.2.4.7, with the network effectively on its side. When a logical high is presented to a given bit, the switch is closed and current flows through the $2R$ element and into the op amp. Note that the right end of the resistor is effectively at ground, as the summing node of the op amp is a virtual ground. If a logical low is presented, the switch shunts the current to ground, bypassing the op amp. In this way, the appropriately weighted currents are summed and used to produce the output voltage.

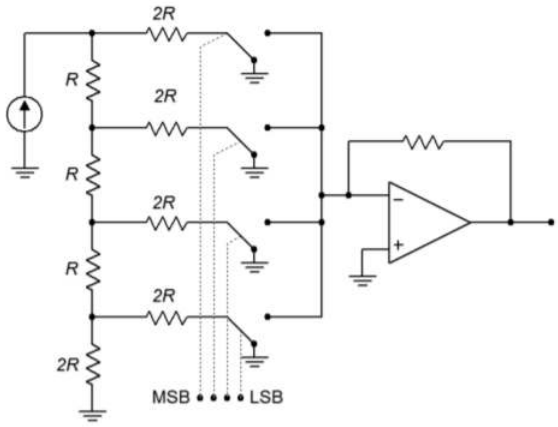


Figure 9.2.2.4.7: Converter with R/2R ladder.

This technique offers several advantages over the simpler weighted gain version. First, all branches are fed by one common current source. Because of this, there is no need for output level matching. Second, only two different values of resistors are required for any number of bits used, rather than the impractically wide range seen earlier. It is more economical to control the tolerance of just two different parts than 12 or 16. Note that small input currents are still generated for the least significant bits, so attention to input bias and offset currents remains important.

12.4.1: Practical Digital-to-Analog Converter Limits

Perhaps the most obvious limit associated with the DA converter is its speed. The op amp used in the DAC must be much faster than the final signals it is meant to produce. A given output waveform may contain several dozen individual sample points per cycle. The op amp must respond to each sample point. Consequently, wide bandwidth and high slew rates are required.

Integrated DAC spec sheets offer a few important parameters of which you should be aware. First of all, there is conversion speed. This figure tells how long it takes the DAC to turn the digital input word into a stable analog output voltage. This sets the maximum data rate. Next come the accuracy and resolution. Resolution indicates the number of discrete steps that may be produced at the output, and is set by the number of bits available. This is not the same as accuracy. Accuracy is actually comprised of several different factors including offset error, error, and nonlinearity. Offset error is normally measured by applying the all-zero input word and then measuring the output signal. Ideally, this signal will be zero volts. The deviation from zero is taken as the offset error. This has the effect of making all output levels inaccurate by a constant voltage. Offset error is relatively easy to compensate for in many applications by applying an equal offset of opposite polarity. Gain error is a deviation that affects each output level by a constant percentage. It is as if the signal were passed through a small amplifier or attenuator. This error may be compensated for by using an amplifier with a gain equal to the reciprocal of the error. The two gains will effectively cancel. The effect of offset and gain error are shown in Figures 9.2.2.4.8 and 9.2.2.4.9.

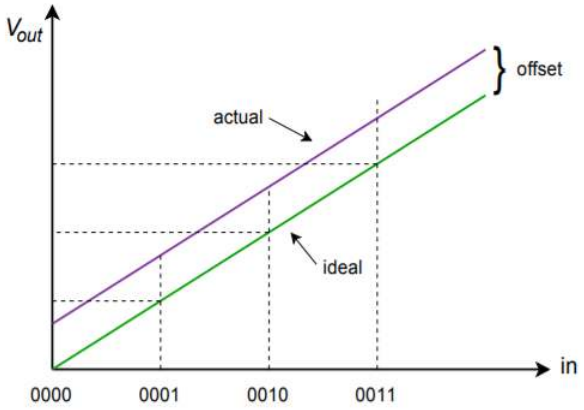


Figure 9.2.2.4.8: Offset error only.

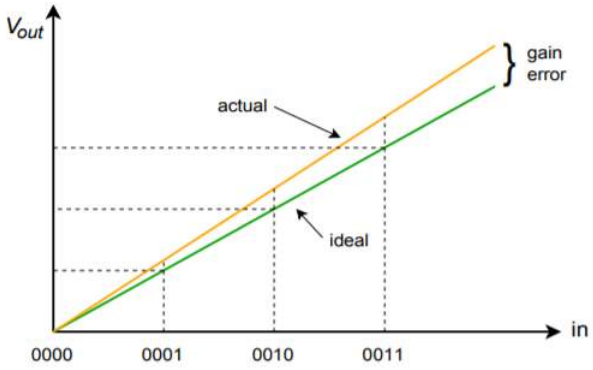


Figure 9.2.2.4.9: Gain error only.

Nonlinearity errors may be broken into two forms: integral nonlinearity and differential nonlinearity. Integral nonlinearity details the maximum offset between the ideal outputs and the actual outputs for all possible inputs. Differential nonlinearity details the maximum output deviation relative to one LSB caused by two adjacent input words. If differential nonlinearity is beyond ± 1 LSB, the system may be non-monotonic. In other words, a higher digital input word may actually produce a lower analog output voltage. These two forms of error are shown in Figure 9.2.2.4.10. Note that it is possible to have high integral nonlinearity and yet still have modest differential nonlinearity. This is the case in Figure 9.2.2.4.10*b*.

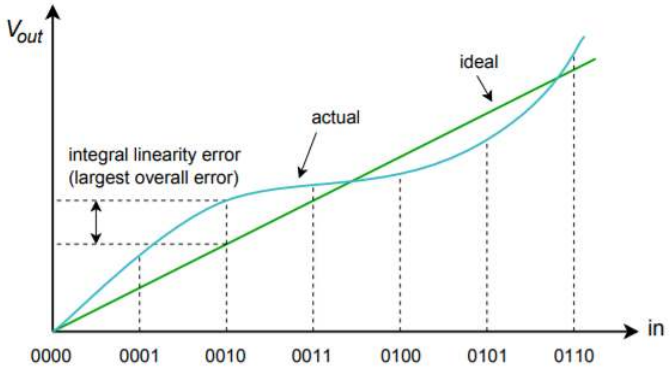


Figure 9.2.2.4.10*a*: Linearity error Integral linearity error.

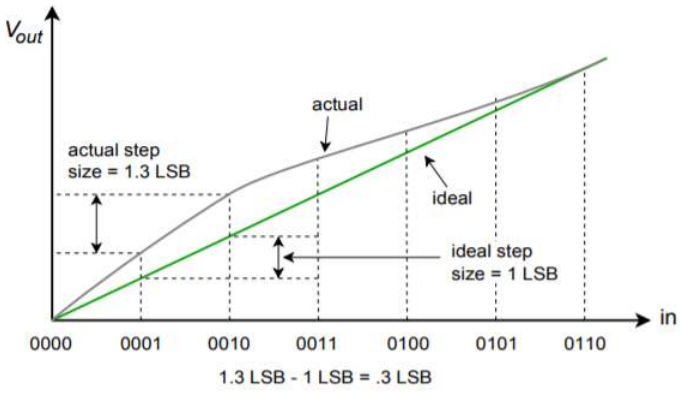


Figure 9.2.2.4.10*b*: Linearity error (continued) Differential linearity error (relative-adjacent error).

As you can see, accuracy is dependent on rather complex factors. In an effort to boil this down to a single number, some manufactures give an effective number of bits specification. For example, a 16-bit DAC may be specified as having 14-bit accuracy. This means that the 14 most significant bits behave in the idealized fashion, but the lowest 2 bits may be swamped out by linearity errors. Another spec that you will sometimes see is no missing codes. This means that for every increase in the input word, there will be an appropriate positive output level change.

In practice, the standard DA converter is used with an output filter. As you can see from the previous figures, the waveforms produced by the DAC contain a stair-step side effect. Generally, this is not desirable. The abrupt changes in output level indicate that higher frequency components are present. All components above the Nyquist rate should be filtered out with an appropriate low-pass filter. This filter is sometimes referred to as a reconstruction or smoothing filter. In an improperly designed system, the reconstruction filter will remove some of the highest in-band frequency components (i.e., components immediately below the Nyquist frequency). To compensate for this, logic levels are often latched to the DAC for shortened periods, thus creating a more spiked appearance, rather than the stair-step form. This effect is shown in Figure 9.2.2.4.11. Although this spiked waveform appears to be less desirable than the stair-step form, it creates higher levels for the uppermost components, and after filtering, the result is a smoother overall frequency response.

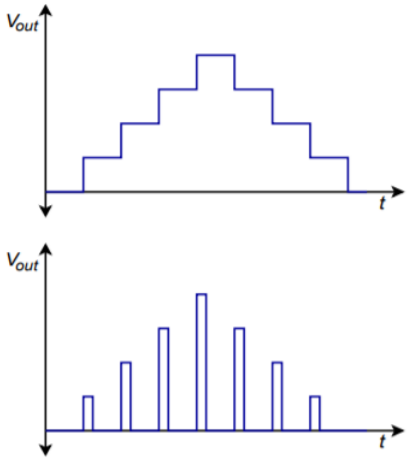


Figure 9.2.2.4.11: Output reconstruction. a. Full period latch (top). b. Partial period latch (bottom).

To further increase the quality of the output waveform, a technique known as oversampling is sometimes employed. The basic idea is to create new sample points in between the existing ones. The result is a much denser data rate, which hopefully, will yield more exacting results after filtering. Also, the higher data rate may loosen the requirements of the reconstruction filter. A typical system might use four times oversampling, meaning that the output data rate is four times the original. Therefore, for each input word, three new words have to be added. This effect is shown in Figure 9.2.2.4.12.

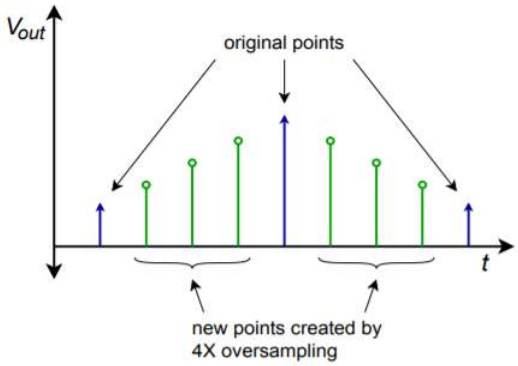


Figure 9.2.2.4.12: Oversampled output.

There are a number of ways to create the new sample points. The most obvious way is via simple interpolation, but this does not achieve the best results. Another technique involves initializing the new values to zero and then passing the data stream through a digital low-pass filter, which effectively calculates the proper values. An extension of the oversampling principle is the delta-sigma technique. In delta-sigma, very high rates of oversampling are used in conjunction with specialized digital filter algorithms. The algorithms essentially trade the higher data rate for a slower rate with increased resolution. The design and analysis of delta-sigma systems is fairly advanced and is beyond the scope of this text. Suffice to say that these techniques can increase the quality of the output signal and are widely used in applications such as high-quality audio CD and DVD players.

12.4.2: Digital-to-Analog Converter Integrated Circuits

There are many possible applications for digital-to-analog converters, and a number of different chips have evolved to meet specific needs. Generally, you can group these into specific classes, such as high speed, high resolution, or low cost. We will examine three representative types. The devices we will look at are the DAC0832; a basic 8-bit unit, the DAC7545; a microprocessor-compatible 12-bit unit, and the PCM1716; a 24-bit high-quality converter used in the audio industry.

DAC0832

This IC is a popular microprocessor-compatible 8-bit converter. The DAC0830 and DAC0831 are similar, but with somewhat reduced performance. It is a multiplying DAC. In other words, the output signal is a function of the digital input word and a reference input. In some applications, the reference input is not fixed, but rather, is a variable input signal. A feature list and pin-out are shown in Figure 9.2.2.4.13. Notable items are a settling time of only 1 μs, low power requirements, and high linearity. The DAC0832 may be used in either stand-alone mode or with a microprocessor. The switching waveforms are shown in Figure 9.2.2.4.14.

National Semiconductor

DAC0830/DAC0832
8-Bit µP Compatible, Double-Buffered D to A Converters

General Description

The DAC0830 is an advanced CMOS/Si-Cr 8-bit multiplying DAC designed to interface directly with the 8080, 8048, 8085, Z80®, and other popular microprocessors. A deposited silicon-chromium R-2R resistor ladder network divides the reference current and provides the circuit with excellent temperature tracking characteristics (0.05% of Full Scale Range maximum linearity error over temperature). The circuit uses CMOS current switches and control logic to achieve low power consumption and low output leakage current errors. Special circuitry provides TTL logic input voltage level compatibility.

Double buffering allows these DACs to output a voltage corresponding to one digital word while holding the next digital word. This permits the simultaneous updating of any number of DACs.

The DAC0830 series are the 8-bit members of a family of microprocessor-compatible DACs (MICRO-DAC™).

Features

- Double-buffered, single-buffered or flow-through digital data inputs
- Easy interchange and pin-compatible with 12-bit DAC1230 series
- Direct interface to all popular microprocessors
- Linearity specified with zero and full scale adjust only — NOT BEST STRAIGHT LINE FIT.
- Works with ±10V reference-full 4-quadrant multiplication
- Can be used in the voltage switching mode
- Logic inputs which meet TTL voltage level specs (1.4V logic threshold)
- Operates "STAND ALONE" (without µP) if desired
- Available in 20-pin small-outline or molded chip carrier package

Key Specifications

- Current settling time: 1 µs
- Resolution: 8 bits
- Linearity: 8, 9, or 10 bits (guaranteed over temp.)
- Gain Tempco: 0.0002% FS/°C
- Low power dissipation: 20 mW
- Single power supply: 5 to 15 V_{DC}

Typical Application

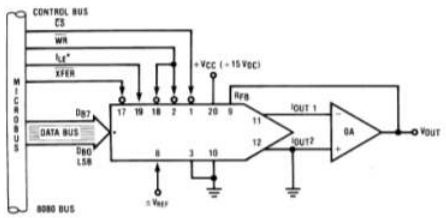


Figure 9.2.2.4.13: DAC0832. Reprinted courtesy of Texas Instrutment

https://eng.libretexts.org/@go/page/122597

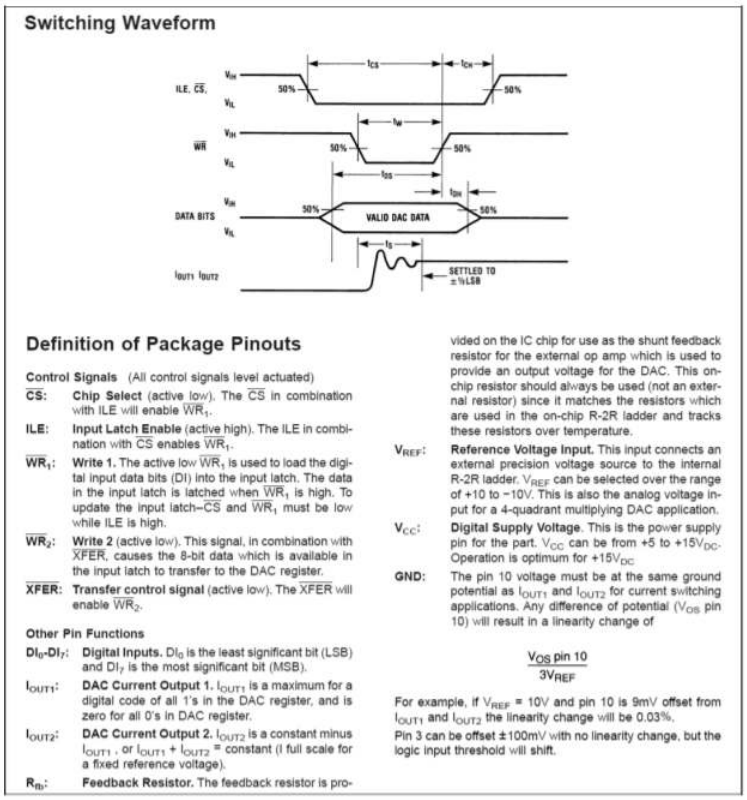


Figure 9.2.2.4.14: DAC0832 switching waveforms Reprinted courtesy of Texas Instrutments

An interesting application of the DAC0832 can be found in Figure 9.2.2.4.15. Basically, this is a digitally-controlled state variable filter.

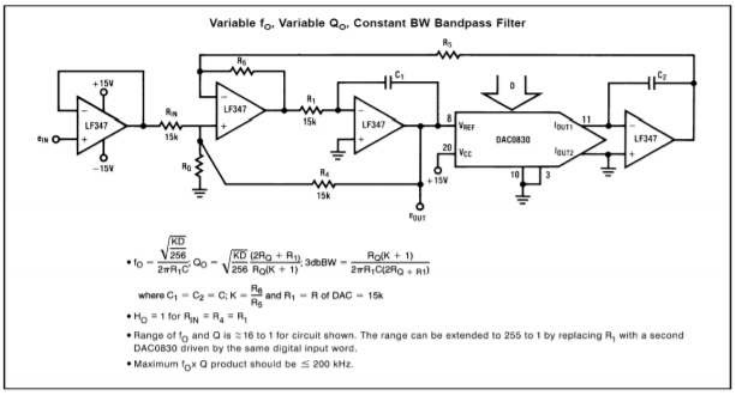
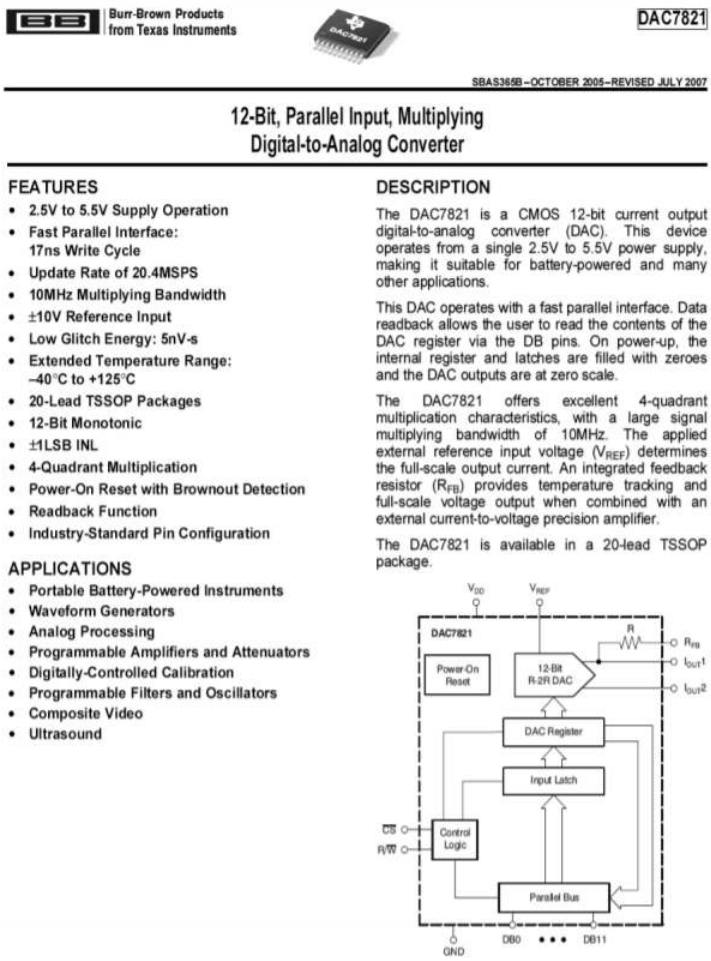


Figure 9.2.2.4.15: DAC0832 state variable filter application. Reprinted courtesy of Texas Instrutments

Note that the converter replaces the input resistor of the second integrator. Normally, that resistor would be used to convert the output voltage of the first integrator into an input current for the second integrator. This job is now handled by the DAC0832. The digital input word effectively sets the voltage-to-current conversion. Thus, a change in the input word alters the tuning frequency of the filter just as a potentiometer. Compare this circuit to the OTA-based voltage-controlled filter from Chapter Eleven. Conceptually, they are very similar.



Burr-Brown Products from Texas Instruments

DAC7821

SBAS365B – OCTOBER 2005 – REVISED JULY 2007

12-Bit, Parallel Input, Multiplying Digital-to-Analog Converter

FEATURES

- 2.5V to 5.5V Supply Operation
- Fast Parallel Interface: 17ns Write Cycle
- Update Rate of 20.4MSPS
- 10MHz Multiplying Bandwidth
- ±10V Reference Input
- Low Glitch Energy: 5nV-s
- Extended Temperature Range: –40°C to +125°C
- 20-Lead TSSOP Packages
- 12-Bit Monotonic
- ±1LSB INL
- 4-Quadrant Multiplication
- Power-On Reset with Brownout Detection
- Readback Function
- Industry-Standard Pin Configuration

APPLICATIONS

- Portable Battery-Powered Instruments
- Waveform Generators
- Analog Processing
- Programmable Amplifiers and Attenuators
- Digitally-Controlled Calibration
- Programmable Filters and Oscillators
- Composite Video
- Ultrasound

DESCRIPTION

The DAC7821 is a CMOS 12-bit current output digital-to-analog converter (DAC). This device operates from a single 2.5V to 5.5V power supply, making it suitable for battery-powered and many other applications.

This DAC operates with a fast parallel interface. Data readback allows the user to read the contents of the DAC register via the DB pins. On power-up, the internal register and latches are filled with zeroes and the DAC outputs are at zero scale.

The DAC7821 offers excellent 4-quadrant multiplication characteristics, with a large signal multiplying bandwidth of 10MHz. The applied external reference input voltage (V_{REF}) determines the full-scale output current. An integrated feedback resistor (R_{FB}) provides temperature tracking and full-scale voltage output when combined with an external current-to-voltage precision amplifier.

The DAC7821 is available in a 20-lead TSSOP package.

Figure 9.2.2.4.16: DAC7821. Reprinted courtesy of Texas Instruments

DAC7821

The DAC7821 is a fairly standard 12-bit linear converter and is shown in Figure 9.2.2.4.16. Its interesting aspects are that it is a multiplying converter and that it is microprocessor-compatible. The multiplying effect comes from the fact that a reference is used to drive the $R/2R$ ladder network. If the reference is changed, the output is effectively rescaled. Consequently, you can think of the output signal as equal to the reference value times the digital input word. You may also think of this as a form of “digital volume control”.

With the inclusion of a few extra logic lines, the IC has become microprocessor compatible. This means that the DAC7821 has chip select and read/write lines along with the 12 data input lines. This allows the converter to be connected directly to the microprocessor data bus. By using memory-mapped I/O, the microprocessor can write data to the converter just as it writes data to memory. A 16-bit microprocessor system can present the converter with all of the data it needs during one write cycle, however an 8-bit microprocessor will need two write cycles and some form of latch. One address may be used for the lower 8 bits, and another address for the remaining 4 bits. A simplified system is shown in Figure 9.2.2.4.17 using an 8-bit microprocessor.

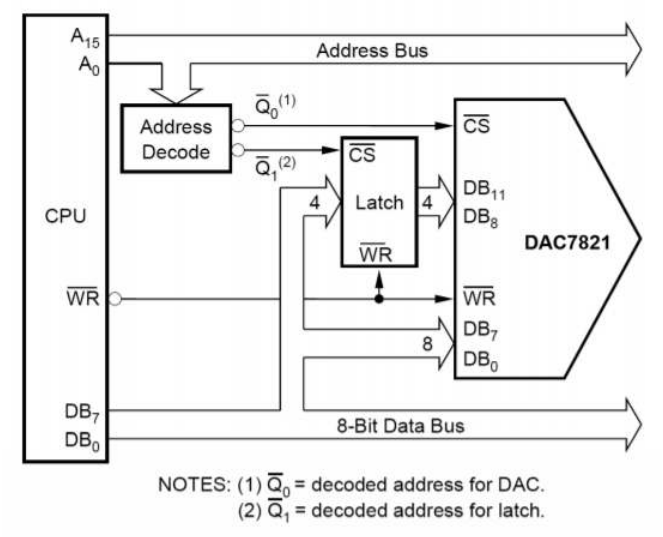
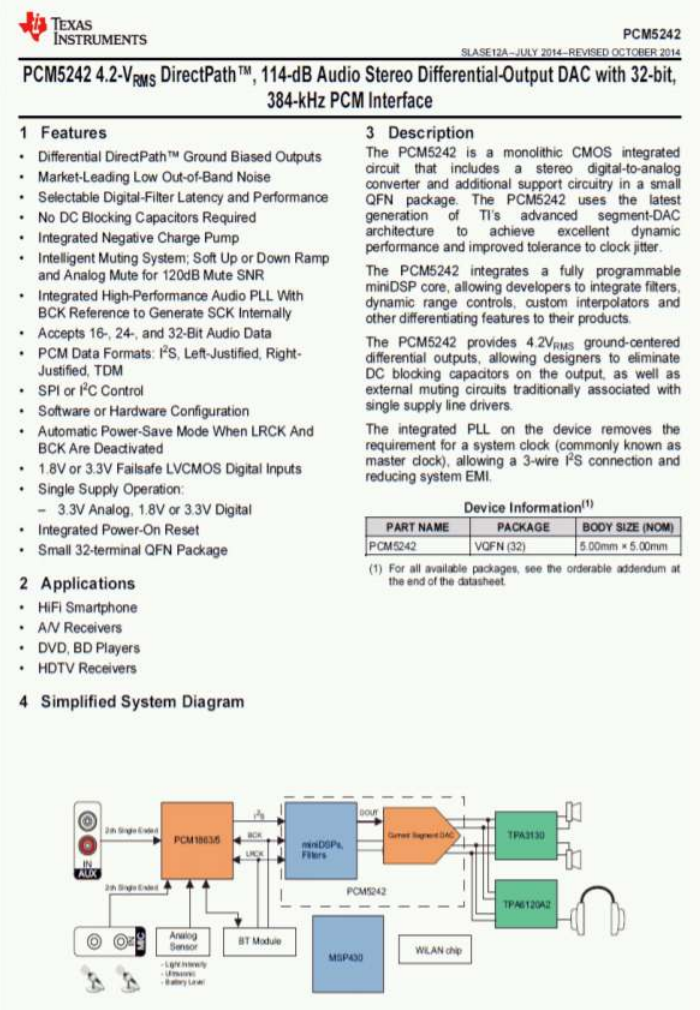


Figure 9.2.2.4.17: Microprocessor to DAC7821 interface. Reprinted courtesy of Texas Instruments

PCM5242

The PCM5242 is a stereo 24-bit converter designed specifically for high-quality digital audio applications. It comes in a VQFN (Very thin Quad Flat No-lead) package. A block diagram and feature list are shown in Figure 9.2.2.4.18*a*. Unlike the other converters, the PCM5242 features serial input of data, not parallel. It includes its own on-board serial conversion circuitry and logic. This technique helps to reduce system cost. It is also surprisingly convenient as many specialized digital signal processing ICs that might be used with the PCM5242 utilize a serial-type output. This may be fed directly into the PCM5242 in 16-, 24-, or 32-bit format.



TEXAS INSTRUMENTS

PCM5242

SLASE12A – JULY 2014 – REVISED OCTOBER 2014

PCM5242 4.2-V_{RMS} DirectPath™, 114-dB Audio Stereo Differential-Output DAC with 32-bit, 384-kHz PCM Interface

1 Features

- Differential DirectPath™ Ground Biased Outputs
- Market-Leading Low Out-of-Band Noise
- Selectable Digital-Filter Latency and Performance
- No DC Blocking Capacitors Required
- Integrated Negative Charge Pump
- Intelligent Muting System; Soft Up or Down Ramp and Analog Mute for 120dB Mute SNR
- Integrated High-Performance Audio PLL With BCK Reference to Generate SCK Internally
- Accepts 16-, 24-, and 32-Bit Audio Data
- PCM Data Formats: I^2S, Left-Justified, Right-Justified, TDM
- SPI or I^2C Control
- Software or Hardware Configuration
- Automatic Power-Save Mode When LRCK And BCK Are Deactivated
- 1.8V or 3.3V Failsafe LVCMOS Digital Inputs
- Single Supply Operation:
  - 3.3V Analog, 1.8V or 3.3V Digital
- Integrated Power-On Reset
- Small 32-terminal QFN Package

2 Applications

- HiFi Smartphone
- A/V Receivers
- DVD, BD Players
- HDTV Receivers

3 Description

The PCM5242 is a monolithic CMOS integrated circuit that includes a stereo digital-to-analog converter and additional support circuitry in a small QFN package. The PCM5242 uses the latest generation of TI's advanced segment-DAC architecture to achieve excellent dynamic performance and improved tolerance to clock jitter.

The PCM5242 integrates a fully programmable miniDSP core, allowing developers to integrate filters, dynamic range controls, custom interpolators and other differentiating features to their products.

The PCM5242 provides 4.2V_{RMS} ground-centered differential outputs, allowing designers to eliminate DC blocking capacitors on the output, as well as external muting circuits traditionally associated with single supply line drivers.

The integrated PLL on the device removes the requirement for a system clock (commonly known as master clock), allowing a 3-wire I^2S connection and reducing system EMI.

Device Information(1)

PART NAME	PACKAGE	BODY SIZE (NOM)
PCM5242	VQFN (32)	5.00mm × 5.00mm

(1) For all available packages, see the orderable addendum at the end of the datasheet.

4 Simplified System Diagram

Figure 9.2.2.4.18*a*: PCM5242. Reprinted courtesy of Texas Instruments

The PCM5242 specification sheet is shown in Figure 9.2.2.4.18*b*. Note that this device is specified with sampling rates from 48 kHz to 192 kHz. Total harmonic distortion plus noise is typically 94 dB below a full-scale output when used with any of these sample rates. Due to its high resolution and 114 dB dynamic range, extra care must be taken during circuit layout to avoid hum pickup and RF interference.

7.5 Electrical Characteristics

All specifications at T_A = 25°C, AV_{DD} = CPV_{DD} = DV_{DD} = 3.3V, f_S = 48kHz, system clock = 512f_S and 24-bit data unless otherwise noted.

PARAMETER		TEST CONDITIONS	MIN	TYP	MAX	UNIT
	Resolution		16	24	32	Bits
Digital Input/Output						
	Logic Family: 3.3V LVCMOS compatible					
V_{IH}	Input logic level		0.7×DV_{DD}			V
V_{IL}					0.3×DV_{DD}	
I_{IH}	Input logic current	V_{IN} = V_{DD}			10	µA
I_{IL}		V_{IN} = 0V			–10	
V_{OH}	Output logic level	I_{OH} = –4mA	0.8×DV_{DD}			V
V_{OL}		I_{OL} = 4mA			0.22×DV_{DD}	
	Logic Family 1.8V LVCMOS compatible					
V_{IH}	Input logic level		0.7×DV_{DD}			V
V_{IL}					0.3×DV_{DD}	
I_{IH}	Input logic current	V_{IN} = V_{DD}			10	µA
I_{IL}		V_{IN} = 0V			–10	
V_{OH}	Output logic level	I_{OH} = –2mA	0.8×DV_{DD}			V
V_{OL}		I_{OL} = 2mA			0.22×DV_{DD}	
Dynamic Performance (PCM Mode)(1)(2)						
	THD+N at -1 dB(2)	f_S = 48kHz			–87	dB
		f_S = 96kHz		–94		
		f_S = 192kHz		–94		
	Dynamic range(2)	EIAJ, A-weighted, f_S = 48kHz	108	114		
		EIAJ, A-weighted, f_S = 96kHz		114		
		EIAJ, A-weighted, f_S = 192kHz		114		
	Signal-to-noise ratio(2)	EIAJ, A-weighted, f_S = 48kHz		114		
		EIAJ, A-weighted, f_S = 96kHz		114		
		EIAJ, A-weighted, f_S = 192kHz		114		
	Signal to noise ratio with analog mute(2)(3)	EIAJ, A-weighted, f_S = 48kHz	113	123		
		EIAJ, A-weighted, f_S = 96kHz	113	123		
		EIAJ, A-weighted, f_S = 192kHz	113	123		
	Channel Separation	f_S = 48kHz	100 / 95	109 / 103		dB
		f_S = 96kHz	100 / 95	109 / 103		
		f_S = 192kHz	100 / 95	109 / 103		

(1) Filter condition: THD+N: 20Hz HPF, 20kHz AES17 LPF Dynamic range: 20Hz HPF, 20kHz AES17 LPF, A-weighted Signal-to-noise ratio: 20Hz HPF, 20kHz AES17 LPF, A-weighted Channel separation: 20Hz HPF, 20kHz AES17 LPF Analog performance specifications are measured using the System Two Cascade™ audio measurement system by Audio Precision™ in the RMS mode.
(2) Output load is 10kΩ, with 470Ω output resistor and a 2.2nF shunt capacitor (see recommended output filter).
(3) Assert XSMT or both L-ch and R-ch PCM data are BPZ

Figure 9.2.2.4.18*b*: PCM5242 specifications. Reprinted courtesy of Texas Instruments

12.4.3: Applications of Digital-to-Analog Converter Integrated Circuits

Example 9.2.2.4.1

Perhaps the first thing many people think of when they hear the terms digital or digitized, is the audio compact disk, or CD for short. Home CD players are excellent examples of the use of precision DA circuitry in our everyday lives. Music data is stored on the CD with 16-bit resolution and a sampling rate of 44.1 kHz. This produces a Nyquist frequency of 22.05 kHz, which is high enough to encompass the hearing range of most humans. Error correction and auxiliary data is also stored on the disk. The data is stored on disk in the form of very tiny pits, which are read by a laser. The signal is then converted into the common electronic logic form where it is checked for error and adjusted as need be. The data stream is then fed to the DA converter for audio reconstruction. A single converter may be multiplexed between the two stereo channels, or two dedicated converters may be used. Oversampling in the range of 2X to 8X is often used for improved signal quality. A block diagram of the system is shown in Figure 9.2.2.4.19. The actual DAC portion seems almost trivial when compared to some of the more sophisticated elements.

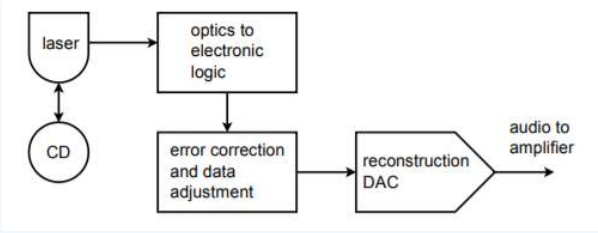


Figure 9.2.2.4.19: Audio compact disk playback system.

The storage density of the optical CD is quite remarkable. This small disk (less than 5 inches in diameter) can hold 70 minutes of music. Ignoring the auxiliary data, we can quickly calculate the total storage. We have two channels of 16-bit data, or 32 bits (4 bytes) per sample point. There are 44,100 samples per second for 70 minutes, yielding 185.22 megasamples. The total data storage is 5.927 gigabits, or 741 megabytes.

Example 9.2.2.4.2

As we have already mentioned, it is possible to connect DACs directly to microprocessor systems. Furthermore, the microprocessor may write to the DAC with no more effort than writing to a memory location. The microprocessor can write any series of data words we desire out to the DAC and can repeat a sequence virtually forever. Given this ability, we can make an arbitrary waveform generator. Instead of being locked into a set of predefined wave shapes as on ordinary function generators, this system allows for all manner of wave shapes. The accuracy and flexibility of the system will depend on its speed and the available DAC resolution.

The basic idea is one of table look-up. For example, let's say that we have a 16-bit system. We will create a table of data values for one cycle of the output waveform. For convenience sake, we might make the table size a handy power of 2, such as 256. In other words, a single output cycle will be chopped into 256 discrete time chunks. It is obvious, then, that the converter must be a few hundred times faster than the highest fundamental that we wish to produce. By increasing or decreasing the output data rate, we can change the frequency of the output fundamental. This is known as a variable sample rate technique. It is also possible to change the fundamental frequency with a fixed rate technique (this is somewhat more complex, but does offer certain advantages). An output flow chart is shown in Figure 9.2.2.4.20.

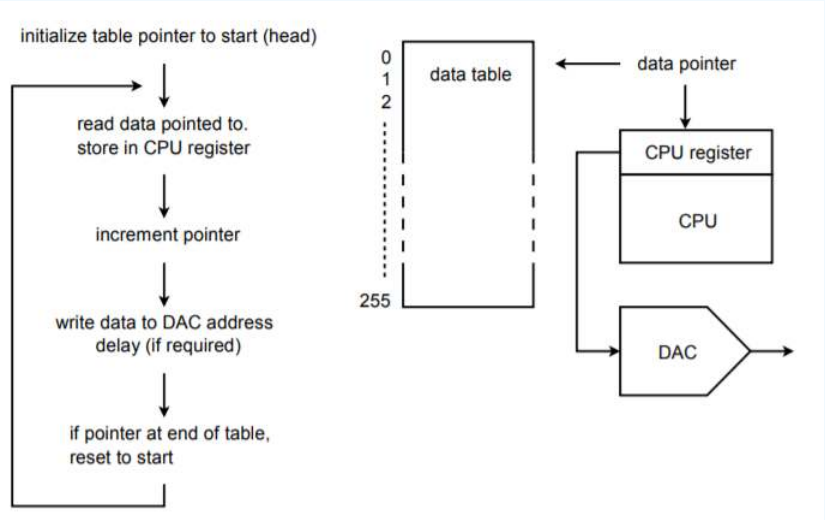


Figure 9.2.2.4.20: Arbitrary waveform generator.

Upon initialization, an address pointer is set to the starting address of the data table. The CPU reads the data from the table via the pointer. The pointer is incremented so that it now points to the next element in the table. (Some CPUs offer a post-increment addressing mode so that both steps may be performed in a single instruction.) Next, the CPU writes the data to the special DAC address. At this point, some form of software/hardware delay is invoked that sets the output data rate. After the delay, the CPU reads the next data element via the pointer and continues as in the first run. Once the 256th element is sent out,

the pointer is reset to the start of the table and the process continues on. In this way, the table can be thought of as circular, or never-ending. If the system software is written in a higher level language, the pointer/data table may be implemented as a simple array where the array index is set by a counter. This will not be as efficient as a direct assembly level approach, though.

The real beauty of this system is that the data table may contain virtually any sequence of data. The data could represent a sine, pulse, triangle, or other standard function. More importantly, the data could represent a sine wave with an embedded noise transient, or a signal containing a hum component just as easily. This data could come from three basic sources. First, the data table can be filled through direct computation if the time-domain equation of the desired function is known. Second, the data could be manufactured by the user through some form of interaction with a computer, perhaps with a mouse or drawing pad. Finally, the data may be derived from a real-world signal. That is, an analog-to-digital converter may be used to record the signal in digital form. The data may then be loaded into the table and played back repeatedly. The arbitrary waveform generator allows its user to test circuits and systems with a range of wave shapes that would be impossible or impractical to generate otherwise.

Example 9.2.2.4.3

Under computer control, DA converters can be used as part of an automated test equipment system. In order to fully characterize an electronic product, a number of individual tests need to be run. Setting up each individual test can be somewhat time consuming and is subject to operator error. Automating this procedure may improve repeatability and decrease testing time. There are many ways in which this process may be automated. We'll look at one approach.

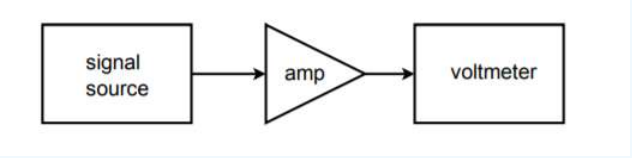


Figure 9.2.2.4.21: Simple test setup.

Let's assume that we would like to make frequency response measurements for an amplifier at 20 different frequencies. A circuit test system appropriate for this job is shown in Figure 9.2.2.4.21. To perform this test manually requires 20 distinct settings of the source signal, and 20 corresponding output readings. This can prove to be rather tedious if many units are to be tested. It would be very handy if there was some way in which the source frequency could be automatically changed to preset values. This is not particularly difficult. Most modern sources have control voltage inputs that may be used to set the frequency. The required control voltage can be created and accurately set through the use of a computer and DA converter. The computer can be programmed to send specific digital words to the DAC, which in turn feeds the signal source control input. In other words, the data word directly sets the frequency of the signal source. The computer can be programmed to send virtually any sequence of data words at almost any rate and do it all without operator intervention. All the operator needs to do is start the process. Test repeatability is very high with a system like this. A block diagram of this system is shown in Figure 9.2.2.4.22.

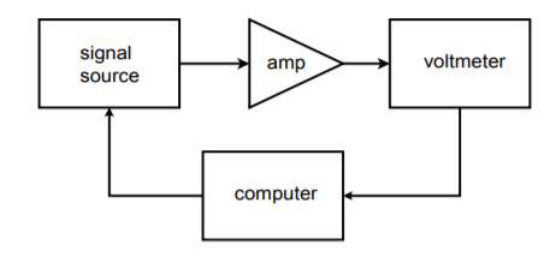


Figure 9.2.2.4.22: Automated test setup.

In order to record the data, the voltmeter may be connected to a strip chart recorder, or better yet, back to the computer. The data may be sent to the computer in digital form if the voltmeter is of fairly advanced design, or, with the inclusion of an analog-to-digital converter, the output signal may be directly sampled and manipulated by the computer. In either case, data files may be created for each unit tested and stored for later use. Also, convenient statistical analysis may be performed quickly

at the end of a test batch. Note that because the DA converter only generates a control signal, very high resolution and low distortion is normally not required. If a high resolution converter is used, it is possible to create the test signals in the computer (as in the arbitrary function generator), and dispense with the signal source.

The automated test system is only one possible application of instrument control. Another interesting example is in the generation of laser “light shows”. A block diagram of a simplified system is shown in Figure 9.2.2.4.23. In order to create the complex patterns seen by the audience, a laser beam is bounced off of tiny moving mirrors. The mirrors may be mounted on something as simple as a galvanometer. The galvanometer is fed by a DAC. The pattern that the laser beam makes is dependent on how the galvanometer moves the mirror, which is in turn controlled by the data words fed to the DAC. In practice, several mirrors may be used to deflect the beam along three axes.

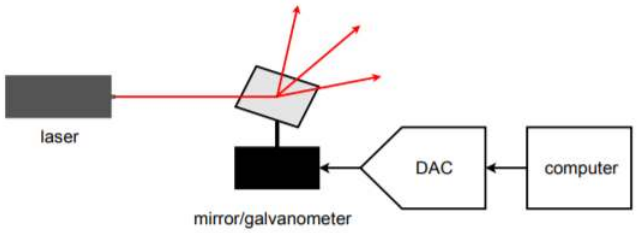


Figure 9.2.2.4.23: Computer-controlled lighting.

DA converters can be used to adjust any device with a control-voltage type input. Also, they may be used to control electromechanical devices that respond to an applied voltage. Their real advantage is the repeatability and flexibility they offer.

This page titled 9.2.2.4: Digital-to-Analog Conversion Techniques is shared under a CC BY-NC-SA 4.0 license and was authored, remixed, and/or curated by James M. Fiore via source content that was edited to the style and standards of the LibreTexts platform.

- **12.4: Digital-to-Analog Conversion Techniques** by James M. Fiore is licensed CC BY-NC-SA 4.0. Original source: http://www.dissidents.com/resources/OperationalAmplifiersAndLinearICs_3E.pdf.

9.2.2.5: Analog-to-Digital Conversion

Now that you know the basics behind digital-to-analog conversion, we may examine the converse system, analog-to-digital conversion. The analog-to-digital conversion process is sometimes referred to as quantization, implying the individual discrete steps that the output assumes. There are several techniques to produce the conversion. Some techniques are optimized for fastest possible conversion speed, and some for highest accuracy. We will investigate the more popular types.

The concept of AD conversion is simple enough: you wish to measure the voltage of the incoming waveform at specific instances in time. This measurement will be translated into a digital word. One practical limitation is the fact that the conversion circuitry may require a small amount of computation or translation time. For ultimate accuracy, then, it is important that the measured waveform not change during the conversion interval. To do this, specialized subcircuits called track-and-hold (or sample-and-hold) amplifiers are used. They are usually abbreviated as T/H or S/H. Their job is to capture an input potential and produce a steady output to feed to the AD converter. The T/H schematic symbol is shown in Figure 9.2.2.5.1. When the T/H logic is in track mode, the circuit acts as a simple buffer, so its output voltage equals its input. When the logic goes to the hold state, the output voltage locks at its present potential and stays there until the circuit is switched back to track mode. A representation of this operation is shown in Figure 9.2.2.5.2.

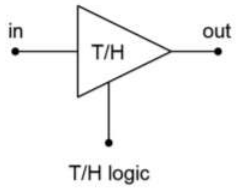


Figure 9.2.2.5.1: Symbol for the track-and-hold amplifier.

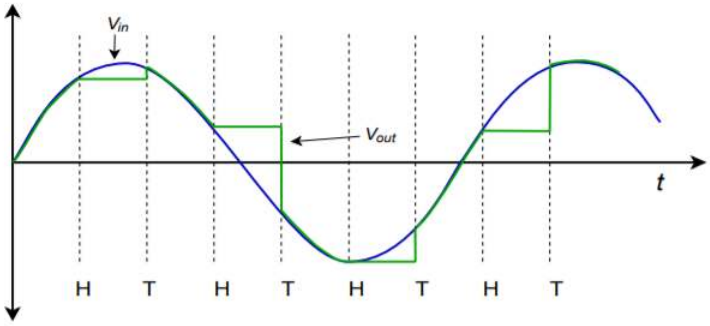


Figure 9.2.2.5.2: Track-and-hold operation.

In reality, the track-and-hold process is not perfect, and errors may arise. These errors are shown magnified in Figure 9.2.2.5.3. First of all, there is a small delay between the time the logic signal changes and the T/H starts to react. The exact amount of time is variable and is responsible for aperture error. This is equal to the voltage difference between the signals at the desired and actual times. Also, due to the dynamic nature of the process of switching from track to hold, some initial ringing may occur in the hold waveform.

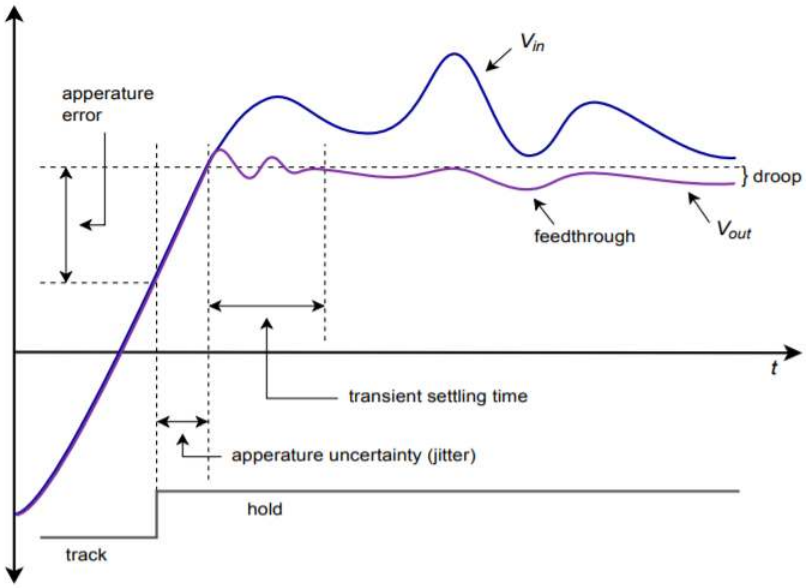


Figure 9.2.2.5.3: Track-and-hold errors.

You can see that as time progresses, the hold waveform tends to decay toward zero. This is because the held voltage is usually formed across a capacitor. Although the discharge time constant can be very long, it cannot be infinite, and thus, the charge eventually bleeds off. This parameter is measured by the droop rate (fundamental units of volts per second). Finally, the possibility of feedthrough error exists. If a large change occurs on the input waveform during the hold period, it is possible that a portion of the signal may "leak" through to the T/H output. These errors are particularly troublesome when working with high-resolution converters. Lower-resolution systems may not be adversely affected by these relatively small aberrations.

In order to create a T/H, a pair of high-impedance buffers are normally used, along with some form of switch element and a capacitor to hold the charge. Examples are shown in Figure 9.2.2.5.4. Figure 9.2.2.5.4a shows the general voltage type, open-loop T/H. The buffers exhibit high input impedance. When the switch is closed, op amp 1 directly feeds op amp 2, and therefore the output voltage equals the input voltage. Normally, the hold capacitor is relatively small and does not adversely affect the drive capability of op amp 1.

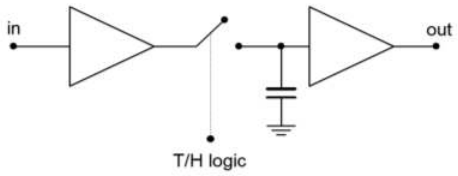


Figure 9.2.2.5.4a: Track-and-hold circuits. General open-loop voltage mode track-and-hold.

A more detailed version of this circuit is shown in Figure 9.2.2.5.4b. Each op amp is a FET input type for minimum input current draw. The switch is a simple JFET which is controlled by a comparator.

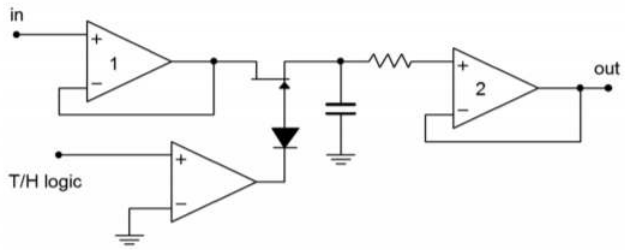


Figure 9.2.2.5.4b: Track-and-hold circuits. Op amp based track-and-hold.

When the T/H logic is high, the gate of the JFET is high, thus producing a low onresistance (i.e., a closed switch). When the comparator output goes low, the JFET is turned off creating a high impedance (i.e., open switch). In this state, op amp 2 is fed by the hold capacitor and buffers this potential to its output. For minimum droop, it is essential that the capacitor be a low leakage type and that op amp 2 have very low input bias current (e.g., FET input). The input resistor is used only to limit possible destructive discharge currents when the circuit is switched off. The diode positioned between the comparator and FET is used to prevent an excessively large positive comparator output potential from reaching the gate of the FET and possibly damaging it. (A comparator high will reverse-bias the series diode.) Figure 9.2.2.5.4*c* shows an alternate circuit using a closed-loop, current-mode approach. Note that the hold capacitor is now forming part of an integrator. While the open-loop form offers faster acquisition and settling times, the closed-loop system offers improved signal tracking.

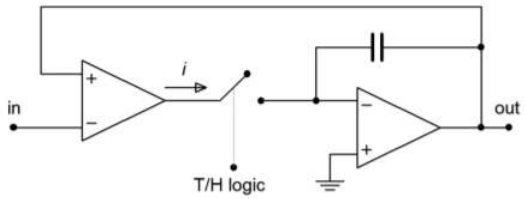


Figure 9.2.2.5.4*c*: Track-and-hold circuits. General closed-loop current mode track-and-hold.

A closed-loop voltage-mode is also possible, but the current form generally offers fewer problems with leakage and switching transients. For general-purpose work, a variety of track-and-hold amplifiers are available in IC form from several manufacturers.

Computer Simulation

A simulation of a track-and-hold circuit similar to the one shown in Figure 9.2.2.5.4*b* is shown in Figure 9.2.2.5.5. The 100 Hz input signal is being sampled at 2 kHz, or 20 times per cycle.

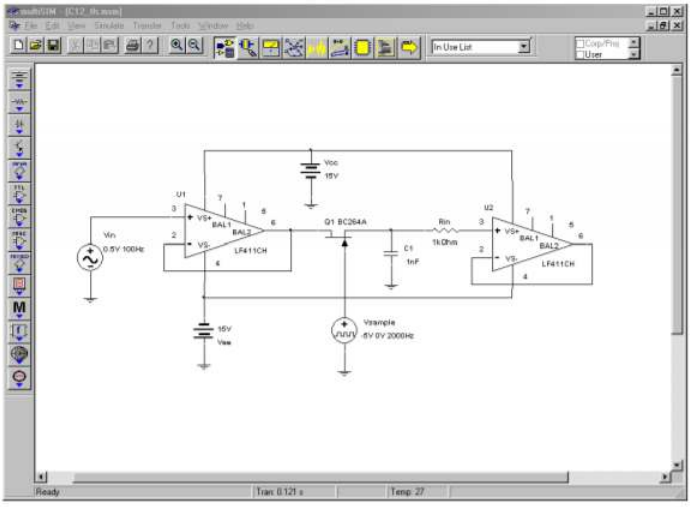


Figure 9.2.2.5.5*a*: Track and hold circuit in Multisim.

Three waveforms of interest are shown in the Transient Analysis graph. The input waveform is seen in the center as the smoothly varying sine wave. Along the bottom of the graph, narrow spikes can be seen. This is the T/H logic signal. The high portion of the pulse (at 0 volts) is the track logic, and the wider low signal (dropping below the −1 volt limit of the graph) represents the hold logic. The stair-stepped sine wave is the output voltage. Note how the circuit acquires or tracks the input signal during the track pulse and then remains at that level when the hold logic is applied. It is during these hold times that the computations will be performed by the analogto-digital converter.

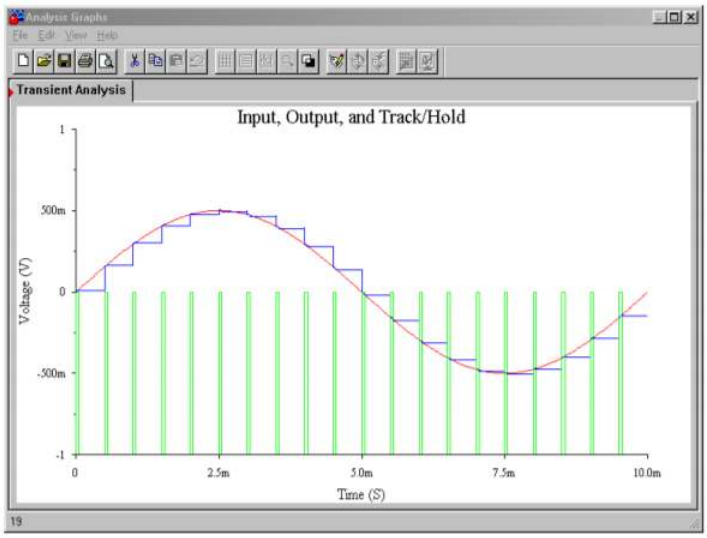


Figure 9.2.2.5.5b: Multisim waveforms for track-and-hold circuit.

12.5.1: Analog-to-Digital Conversion Techniques

Several different techniques have evolved for dealing with differing system requirements. These include flash, staircase, successive approximation, and delta-sigma forms.

Flash conversion is generally used for high-speed work, such as video applications. The circuits are usually low resolution. A flash converter is made up of a string of comparators as shown in Figure 9.2.2.5.6. The input signal is applied to all of the comparators simultaneously. Each comparator is also tied into a reference ladder. Effectively, there is one comparator for each quantization step. When a given signal is applied, a number of comparators towards the bottom of the string will produce a high level, as V_{in} will be greater than their references. Conversely, the comparators towards the top will indicate a low. The comparator at which the outputs shift from high to low indicates the step value closest to the input signal. The set of comparator outputs can be fed into a priority encoder that will turn this simple unweighted sequence into a normal binary word.

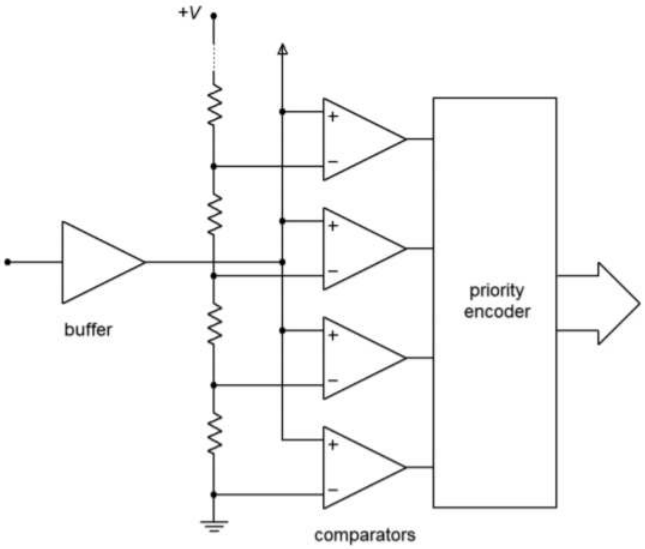


Figure 9.2.2.5.6: Flash converter.

The only time delay involved in the conversion is that of logic propagation delay. Therefore, this conversion technique is quite useful for rapidly changing signals. Its downfall lies in the fact that one comparator is needed for each possible output step change. An 8-bit flash converter requires 256 comparators, whereas a 16-bit version requires 65,536. Obviously, this is rather excessive,

and units in the 4- to 6-bit range are common. Although 6-bit resolution may appear at first to be too coarse for any application, it is actually quite useful for video displays.

For high-resolution work, some other technique must be used. One possibility is shown in Figure 9.2.2.5.7. This is called a staircase converter. Its operation is fairly simple. When a conversion is first started, the output of the counter will be all zeros. This produces a DAC output of zero, and thus, the comparator output will be high. The next clock cycle will increment the up counter, causing the DAC output to increase by one step-level. This signal is compared against the input, and if the input is greater, the output will remain high. The clock will continue to increment the counter in this fashion until the DAC output just exceeds the input level. At this point the comparator output drops low, indicating that the conversion is complete. This signal can then be used to latch the output of the counter. This circuit gets its name from the fact that the waveform produced by the DAC looks like a staircase. The staircase technique can be used for very high-resolution conversion, as long as an appropriate high resolution DAC is used. The major problem with this form is its very low conversion speed, due to the fact that there must be time to test every possible bit combination. Consequently, a 16-bit system requires 65,536 comparisons. Even if a fast 1 microsecond DAC is used, this would limit the sampling interval to nearly 66 milliseconds. This translates to a maximum input frequency before aliasing of only 7 Hz. Noting that the lowest frequency most humans can hear is about 20 Hz, this technique is hardly suitable for something like digital audio recording. In fact, for general-purpose work, the staircase system is avoided in favor of the successive approximation technique.

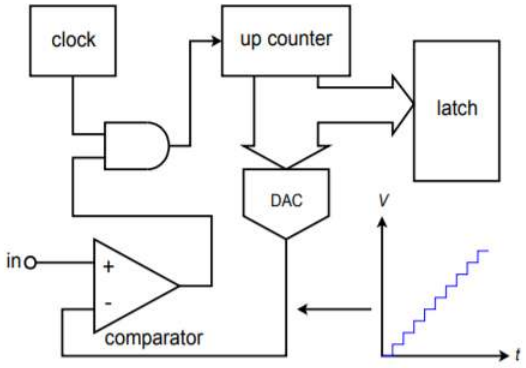


Figure 9.2.2.5.7: Staircase converter.

Successive approximation is a good general-purpose solution suitable for systems requiring resolutions in the 16-bit area. Instead of trying to convert the input signal at one instant like the flash converter, this technique creates the output bit by bit. Unlike the staircase converter, each individual level does not need to be tested. Instead, a binary search algorithm is used. You might think of it as making a series of guesses, each time getting a little closer to the result. Each guess results in a simple comparison. For an n-bit output, n comparisons need to be made.

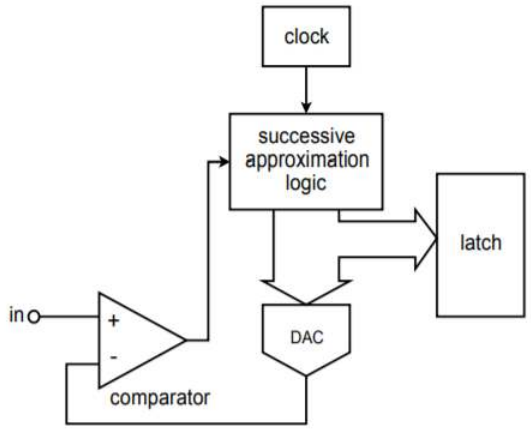


Figure 9.2.2.5.8: Successive-approximation converter.

A block diagram of a successive approximation converter is shown in Figure 9.2.2.5.8. Instead of a simple up counter, a circuit is used to implement the successive approximation algorithm. Here is how the circuit works. The first bit to be tested is the most significant bit. The DAC is fed a 1 with all of the remaining bits being set to 0 (i.e., 100000 ...). This word represents half of the maximum value capable by the system. The comparator output indicates whether V_{in} is greater or less than the resulting DAC output. If the comparator output is high, it means that the required digital output must be greater than the present word. If the comparator output is low, then the present digital word is too large, so the MSB is set to 0. At this point, the next most significant bit is tested by setting it to 1. The new word is fed to the DAC, and again, the comparator is used to determine whether or not the bit under test should remain at 1 or be reset. At this point the two most significant bits have been determined. The remaining bits are individually set to 1 and tested in a similar manner until the least significant bit is determined. In this way, a 16-bit system only requires 16 comparisons. With a 1 microsecond DAC, conversion takes only 16 microseconds. This translates to a sampling rate of 62.5 kHz; thus a maximum input frequency of 31.25 kHz is allowed. As you can see, this is far more efficient than the staircase technique.

For the highest resolutions combined with high sample rates, delta-sigma conversion techniques are popular. Basically, a low-resolution converter is run at a rate many times higher than the Nyquist frequency (perhaps 256 times higher). A special digital filter called a decimator converts the low-resolution high sample-rate data stream into a lower rate with higher resolution. This technique can achieve conversions in the 50 kHz range with 20-bit resolution. Design and analysis of delta-sigma modulators and digital filters is an advanced topic beyond the scope of this text. We will, however, look at a representative IC in the next section.

12.5.2: Analog-to-Digital Converter Integrated Circuits

In this section we will examine a few specific ADC ICs, along with selected applications. The ICs include the ADC0844, an 8-bit microprocessor-compatible unit; the ADC12181, a very fast 12-bit converter utilizing pipeling techniques; and the CS5396, a 24-bit converter designed primarily for audio applications.

ADC0844

A block diagram of the ADC0844 is shown in Figure 9.2.2.5.9. Along with the built-in clock and 8-bit successive approximation register (SAR), the IC also includes an input multiplexer, tri-state latches and read, write, and chip select logic pins. This means that the ADC0844 is easily interfaced to a microprocessor data bus and may be used as a memory-mapped I/O device. This IC is a moderate-speed device, showing a typical conversion speed of 40 microseconds. The timing diagram for the ADC0844 is shown in Figure 9.2.2.5.10. A conversion is initiated by bringing both the chip-select and write-logic lines low. The falling edge of write resets the converter and its rising edge starts the actual conversion. After the conversion period, which is set internally, the digital data may be transferred to the output latches with the readlogic line. Note that the pulse repetition rate of the write-logic line sets the sampling rate. Therefore, a small program running on the host microprocessor that reads and writes to the ADC may be used to control the sampling rate and store the data for later use.

National Semiconductor

ADC0844/ADC0848
8-Bit µP Compatible A/D Converters with Multiplexer Options

General Description

The ADC0844 and ADC0848 are CMOS 8-bit successive approximation A/D converters with versatile analog input multiplexers. The 4-channel or 8-channel multiplexers can be software configured for single-ended, differential or pseudo-differential modes of operation.

The differential mode provides low frequency input common mode rejection and allows offsetting the analog range of the converter. In addition, the A/D's reference can be adjusted enabling the conversion of reduced analog ranges with 8-bit resolution.

The A/Ds are designed to operate from the control bus of a wide variety of microprocessors. TRI-STATE® output latches that directly drive the data bus permit the A/Ds to be configured as memory locations or I/O devices to the microprocessor with no interface logic necessary.

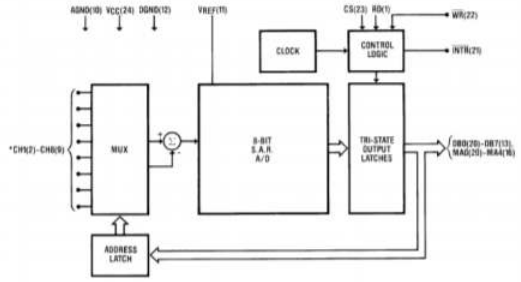
Features

- Easy interface to all microprocessors
- Operates ratiometrically or with 5 V_{DC} voltage reference
- No zero or full-scale adjust required
- 4-channel or 8-channel multiplexer with address logic
- Internal clock
- 0V to 5V input range with single 5V power supply
- 0.3" standard width 20-pin or 24-pin DIP
- 28 Pin Molded Chip Carrier Package

Key Specifications

■ Resolution	8 Bits
■ Total Unadjusted Error	±½ LSB and ± 1 LSB
■ Single Supply	5 V_{DC}
■ Low Power	15 mW
■ Conversion Time	40 µs

Block and Connection Diagrams



*ADC0848 shown in DIP Package CH5-CH8 not included on the ADC0844

Figure 9.2.2.5.9: The ADC0844. Reprinted courtesy of Texas Instrutments

https://eng.libretexts.org/@go/page/122598

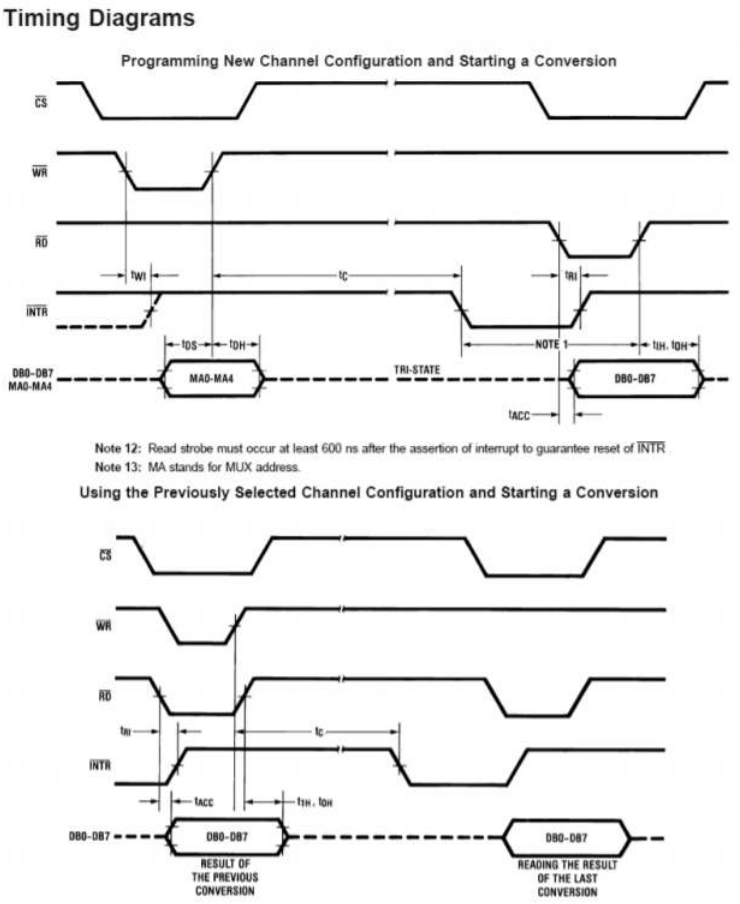
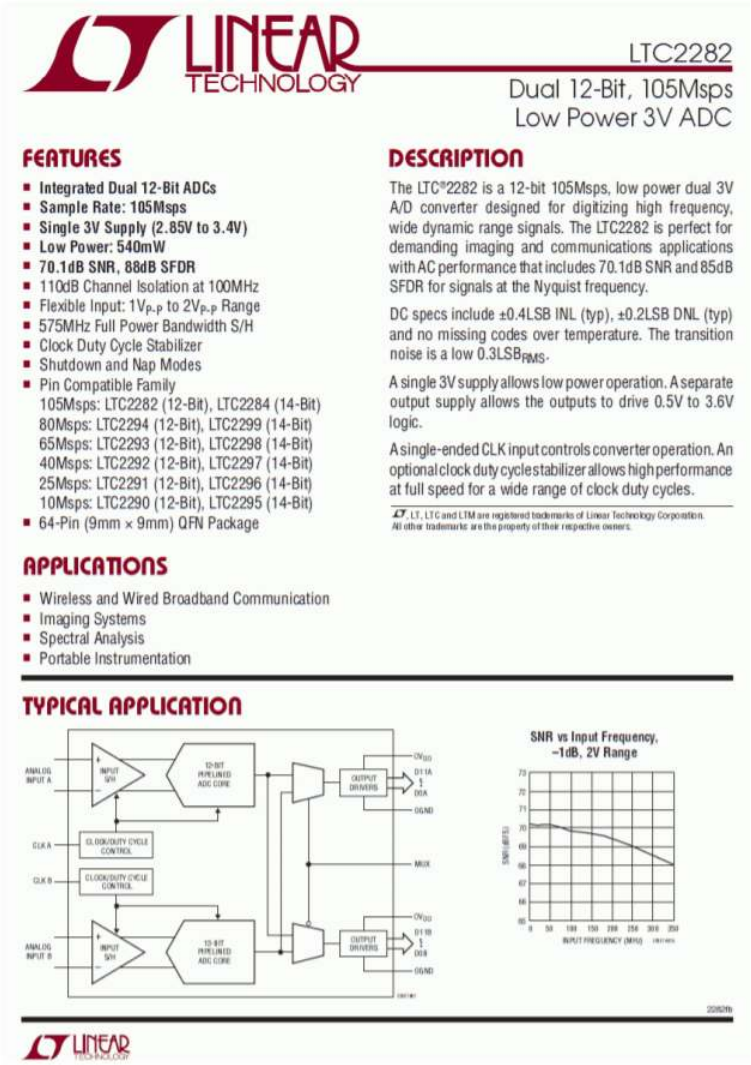


Figure 9.2.2.5.10: Timing diagram for the ADC0844. Reprinted courtesy of Texas Instrutments

LTC2282

The LTC2282 is a 12-bit, 105 mega-samples per second converter with an internal sample-and-hold. Its data sheet is shown in Figure 9.2.2.5.11. In order to achieve its combination of high sample rate with relatively high resolution, the LTC2282 relies on a technique known as pipelining. In this particular chip, the pipeline consists of six stages. Each stage produces a digital signal of just three bits and an error signal known as a residual. The residual is passed to the following stage where it is multiplied by a fixed gain, thus bringing it up to the former bit-weight. The newly resulting residue is passed to the next stage where the process is repeated. In essence, the signal propagates down the pipeline in a fashion conceptually similar to the successive approximation technique. Note that pipelining speeds the conversion process because once the residual is passed to the next stage, the prior stage(s) can begin work on the following sample(s).



LINEAR TECHNOLOGY

LTC2282

Dual 12-Bit, 105Msps
Low Power 3V ADC

FEATURES

- **Integrated Dual 12-Bit ADCs**
- **Sample Rate: 105Msps**
- **Single 3V Supply (2.85V to 3.4V)**
- **Low Power: 540mW**
- **70.1dB SNR, 88dB SFDR**
- 110dB Channel Isolation at 100MHz
- Flexible Input: 1V_{P-P} to 2V_{P-P} Range
- 575MHz Full Power Bandwidth S/H
- Clock Duty Cycle Stabilizer
- Shutdown and Nap Modes
- Pin Compatible Family
  105Msps: LTC2282 (12-Bit), LTC2284 (14-Bit)
  80Msps: LTC2294 (12-Bit), LTC2299 (14-Bit)
  65Msps: LTC2293 (12-Bit), LTC2298 (14-Bit)
  40Msps: LTC2292 (12-Bit), LTC2297 (14-Bit)
  25Msps: LTC2291 (12-Bit), LTC2296 (14-Bit)
  10Msps: LTC2290 (12-Bit), LTC2295 (14-Bit)
- 64-Pin (9mm × 9mm) QFN Package

DESCRIPTION

The LTC®2282 is a 12-bit 105Msps, low power dual 3V A/D converter designed for digitizing high frequency, wide dynamic range signals. The LTC2282 is perfect for demanding imaging and communications applications with AC performance that includes 70.1dB SNR and 85dB SFDR for signals at the Nyquist frequency.

DC specs include ±0.4LSB INL (typ), ±0.2LSB DNL (typ) and no missing codes over temperature. The transition noise is a low 0.3LSB_{RMS}.

A single 3V supply allows low power operation. A separate output supply allows the outputs to drive 0.5V to 3.6V logic.

A single-ended CLK input controls converter operation. An optional clock duty cycle stabilizer allows high performance at full speed for a wide range of clock duty cycles.

LT, LTC and LTM are registered trademarks of Linear Technology Corporation. All other trademarks are the property of their respective owners.

APPLICATIONS

- Wireless and Wired Broadband Communication
- Imaging Systems
- Spectral Analysis
- Portable Instrumentation

TYPICAL APPLICATION

Figure 9.2.2.5.11a: LTC2282. Reprinted courtesy of Linear Technology

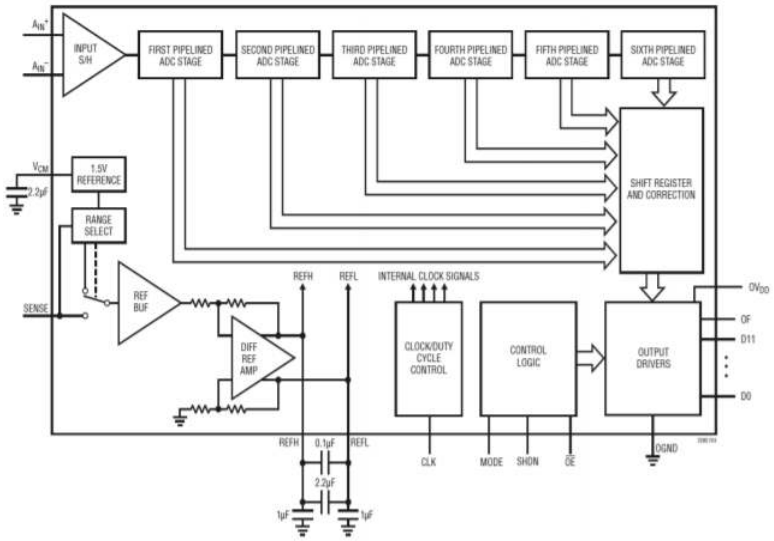


Figure 9.2.2.5.11b: LTC2282 block diagram. Reprinted courtesy of Linear Technology

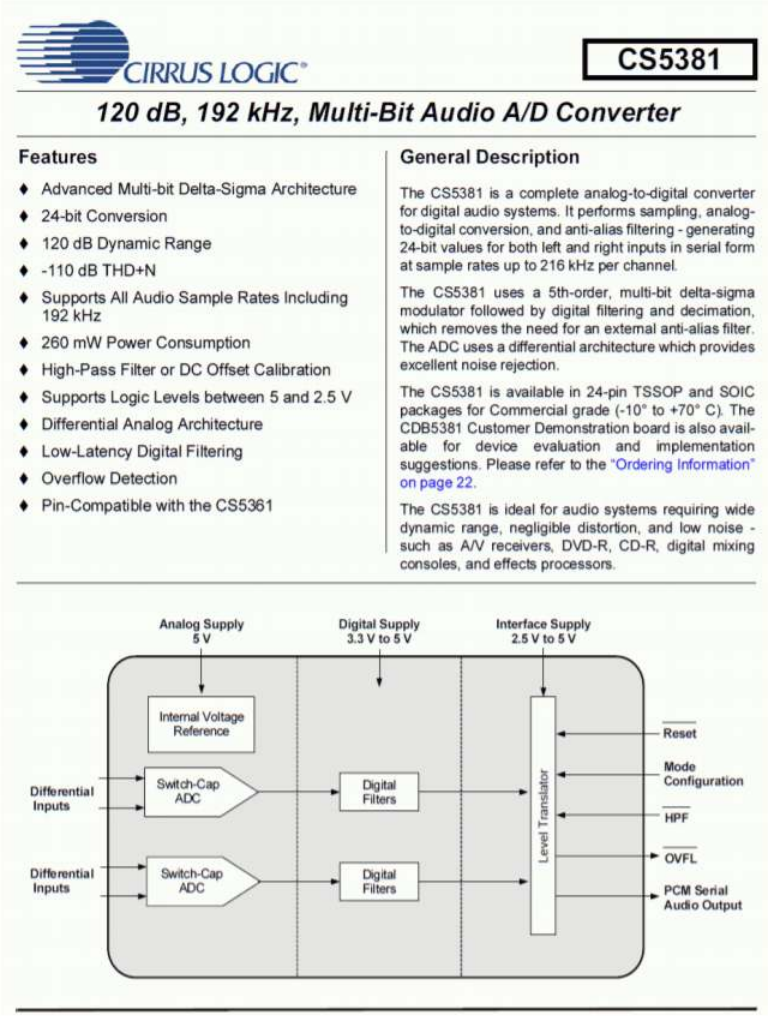
https://eng.libretexts.org/@go/page/122598

The LTC2282 is used for applications requiring both high speed and high resolution. It also offers self-calibration, single +5 V power supply operation, and low power consumption.

CS5381

The CS5381 offers stereo 24-bit resolution with a maximum sampling rate of 192 kHz. This is ideal for a wide range of high-quality digital audio systems. The CS5381 data sheets are shown in Figure 9.2.2.5.12. This IC uses the delta-sigma conversion technique and achieves a dynamic range of 120 dB. The signal-to-noise plus distortion rating (THD+N) is typically 110 dB. The device over-samples the input at 6.144 MHz and has logic settings for 1X, 2X and 4X effective conversion rates at its output. For example, the standard pro-audio sampling rate is 48 kHz with double rate at 96 kHz and quad rate at 192 kHz.

As with any high-resolution converter, you must be particularly careful with the circuit layout or excessive noise may result. Attention must be paid to proper power supply bypassing. The CS5381, like many ICs in this category, requires separate analog and digital power supplies. Although this IC only requires 5 V, note that there are separate physical connections for analog and digital power. Standard logic supplies are often too noisy for use on such precision devices and will degrade overall audio performance when applied to the analog portion of the IC.


CIRRUS LOGIC

CS5381

120 dB, 192 kHz, Multi-Bit Audio A/D Converter

Features

- Advanced Multi-bit Delta-Sigma Architecture
- 24-bit Conversion
- 120 dB Dynamic Range
- -110 dB THD+N
- Supports All Audio Sample Rates Including 192 kHz
- 260 mW Power Consumption
- High-Pass Filter or DC Offset Calibration
- Supports Logic Levels between 5 and 2.5 V
- Differential Analog Architecture
- Low-Latency Digital Filtering
- Overflow Detection
- Pin-Compatible with the CS5361

General Description

The CS5381 is a complete analog-to-digital converter for digital audio systems. It performs sampling, analog-to-digital conversion, and anti-alias filtering - generating 24-bit values for both left and right inputs in serial form at sample rates up to 216 kHz per channel.

The CS5381 uses a 5th-order, multi-bit delta-sigma modulator followed by digital filtering and decimation, which removes the need for an external anti-alias filter. The ADC uses a differential architecture which provides excellent noise rejection.

The CS5381 is available in 24-pin TSSOP and SOIC packages for Commercial grade (-10° to +70° C). The CDB5381 Customer Demonstration board is also available for device evaluation and implementation suggestions. Please refer to the "Ordering Information" on page 22.

The CS5381 is ideal for audio systems requiring wide dynamic range, negligible distortion, and low noise - such as A/V receivers, DVD-R, CD-R, digital mixing consoles, and effects processors.

Figure 9.2.2.5.12*a*: CS5381. Reprinted courtesy of Cirrus Logic, Inc.

https://eng.libretexts.org/@go/page/122598

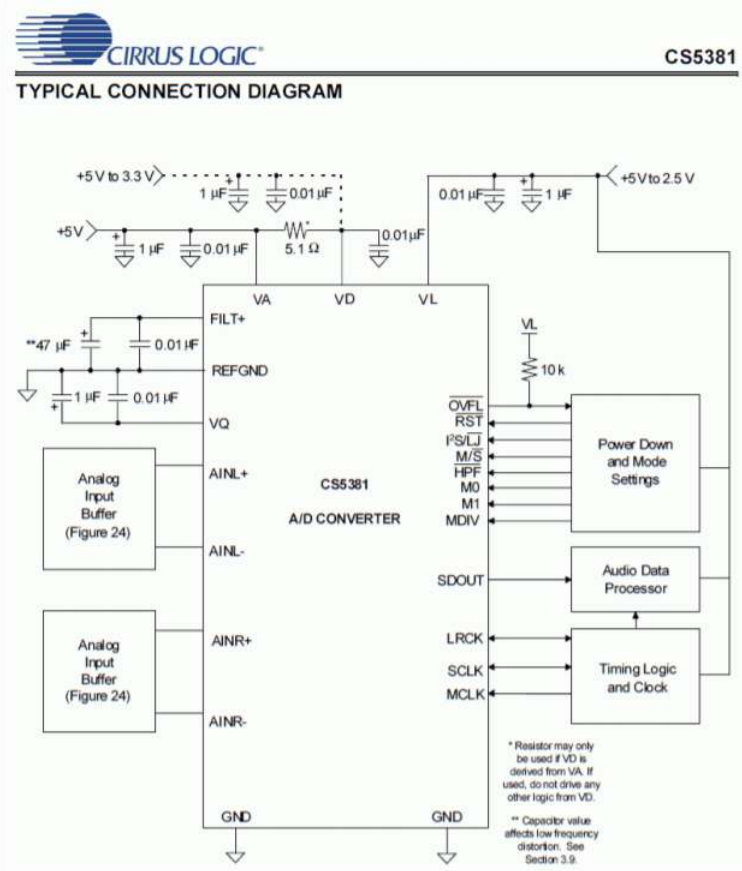


Figure 9.2.2.5.12*b*: CS5381 connection diagram. Reprinted courtesy of Cirrus Logic, Inc.

12.5.3: Applications of Analog-to-Digital Converter Integrated Circuits

Example 9.2.2.5.1

Perhaps the most straightforward application of analog-to-digital conversion is the acquisition of a signal. Once in the digital domain, the signal may be processed in a variety of ways. An example of this is the Digital Sampling Oscilloscope (DSO). A simplified block diagram of a DSO is shown in Figure 9.2.2.5.13. Since the final output of the system is a simple graph, it generally does not make sense to resolve the input beyond 8 bits (256 steps), and often even fewer bits may be used. Typically, high sampling rates are more important than fine resolution in this application. Consequently, 12- and 16-bit converters are not found here. Instead, lower-resolution converters capable of sampling at hundreds of MHz are used.

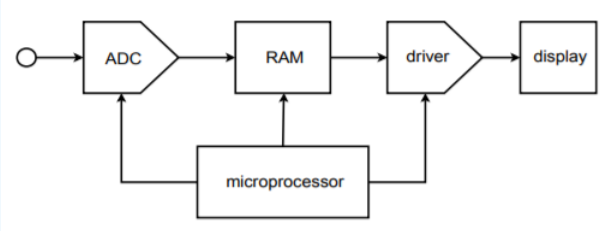


Figure 9.2.2.5.13: Basic digital sampling oscilloscope.

DSOs can usually be run in one of two modes: continuous, or single-shot. In single-shot mode, the ADC acquires the signal and stores it in memory. This data can then be routed to the display circuits continuously in order to create a trace on the display. The signal is captured and stored in computer memory, thus it can be replayed virtually forever without a loss of clarity. This is very useful for catching quick, non-repetitive transients. Once the signal is captured, it may be examined at

leisure. For that matter, if some form of level-sensing logic is included, sampling can be initiated by specific transient events. This is known as baby-sitting. For example, you might suspect that a circuit you have designed occasionally emits an undesirable voltage spike. It is not practical for you to hook up the circuit and stare at the face of an oscilloscope, perhaps for hours, waiting for the system to misbehave. Instead, the DSO can be programmed to wait for the transient event before recording. In this way, you can leave the system, and when you return some time later, the spike will have been recorded and will be waiting your inspection.



Figure 9.2.2.5.14: Commercial MDO. Copyright © Tektronix. Reprinted with permission. All rights reserved

An extension of this concept is pre-trigger recording. In this variation, the DSO is constantly recording and "throwing away" data. When the transient spike finally occurs, the DSO has a snapshot of the events leading up to the spike, as well as the spike itself. This can provide very useful information in some applications. In either case, since the data is in digital form, it may be off-loaded to a computer for further analysis. Indeed, many DSOs offer some interesting on-board analysis functions, including signal smoothing (noise reduction), trace cursors which allow for easy delta time/delta voltage computation, basic math functions, spectrum computation via FFT and more.

In continuous mode, a DSO appears to operate in much the same fashion as an ordinary analog oscilloscope. In this mode, the DSO samples the input signal and passes the data out to memory. From here the data is relayed to the display circuitry where a trace is produced. The trace is being constantly updated, so any change in the input will be quickly displayed.

The basic concept of the DSO has been expanded to the MSO and the MDO. The MSO, or Mixed Signal Oscilloscope, adds digital domain measurements while the MDO, or Mixed Domain Oscilloscope, extends the MSO into the frequency domain as well.

Example 9.2.2.5.2

Another interesting application of the analog-to-digital converter is in the digital sampling music keyboard or drum computer (generally referred to as a sampler). The usage of a sampler is quite straightforward, actually. The idea is to mimic the sound of a given instrument (such as a trumpet or flute) from the keyboard. Before the advent of the sampler, this was done by properly setting the filters, amplifiers and oscillators of a keyboard synthesizer. Although the resulting sounds were reasonably close to the desired instrument, they usually weren't close enough to fool the average listener. The sampler bypasses the problems of synthesizers by directly recording an instrument with an analog-to-digital converter. For example, a trumpet player might sound an A into a microphone that is connected to the sampler. This note is digitized and stored in RAM. Now, when the keyboard player hits the A key, the data is retrieved from RAM and fed to a DAC where it is reconstructed. The result is the exact same note that the trumpet player originally produced. By recording several different pitches from many different instruments, the keyboard player literally has the power of an entire orchestra at his fingertips. Of course, there is no limit to the sounds that might be sampled, and the keyboardist could just as easily use the sampler to "play" a collection of dog barks, door slams, and bird calls. A sampling drum computer is similar to a sampling keyboard, but replaces the standard musical keyboard with a series of buttons that allow the musician to create a programmed sequence of notes. This sequence can be

played back at any time, and at virtually any tempo. In this manner, an entire percussion section may be simulated using digital recordings of real drums. A block diagram of a typical musical sampling system is shown in Figure 9.2.2.5.15.

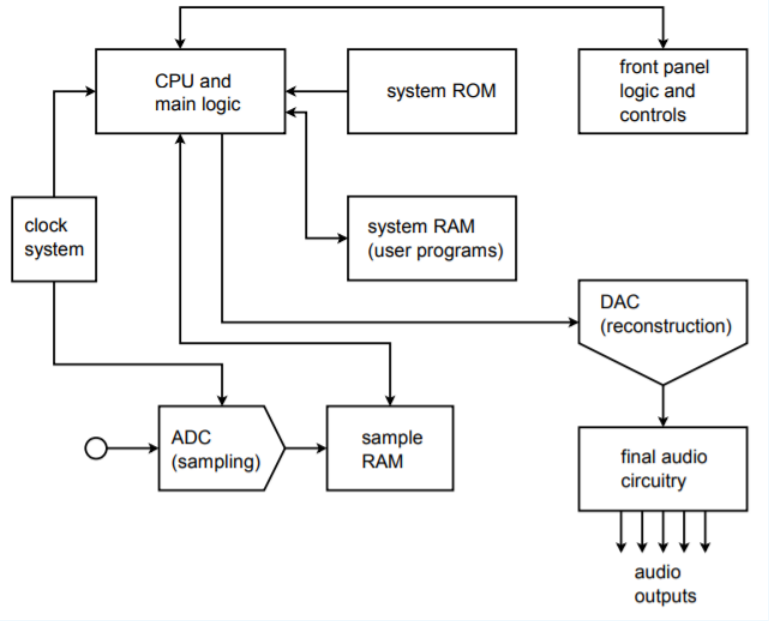


Figure 9.2.2.5.15: Block diagram of musical instrument sampling and playback device.

Further enhancements to the digital sampling musical instrument include direct connections to personal computers. In this way, the computer can be used to analyze and augment the sound samples in new ways. For example, the personal computer may be used to create a graphic display of the waveform, or compute and display the spectral components of the sound using a popular technique called the Fast Fourier Transform. An example is shown in Figure 9.2.2.5.16. Finally, the computer can be programmed to “play” the sampler. In this way, even non-keyboardists can take advantage of the inherent musical flexibility this system offers.

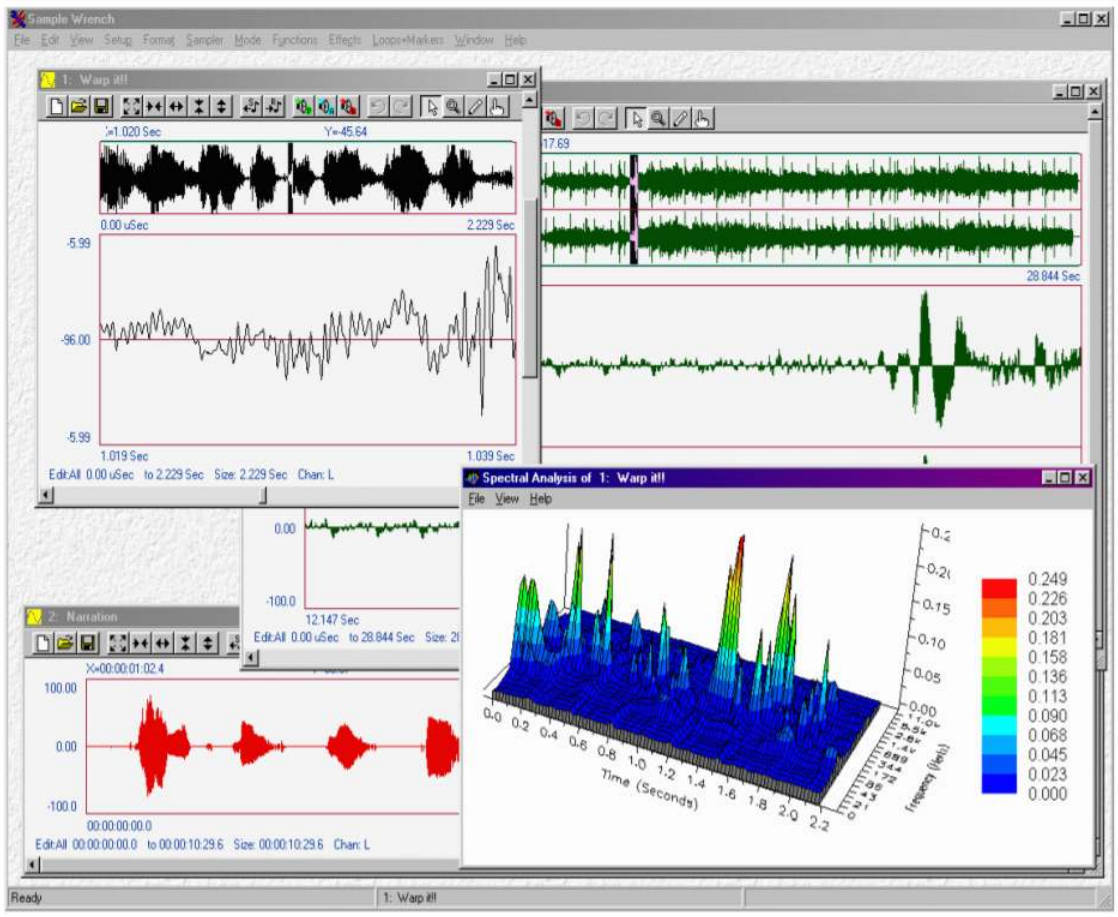


Figure 9.2.2.5.16: Commercial waveform display and editing software. Reprinted courtesy of dissidents

This page titled 9.2.2.5: Analog-to-Digital Conversion is shared under a CC BY-NC-SA 4.0 license and was authored, remixed, and/or curated by James M. Fiore via source content that was edited to the style and standards of the LibreTexts platform.

- **12.5: Analog-to-Digital Conversion** by James M. Fiore is licensed CC BY-NC-SA 4.0. Original source: http://www.dissidents.com/resources/OperationalAmplifiersAndLinearICs_3E.pdf.

9.2.2.6: Extended Topic - Digital Signal Processing

Much has been said in this chapter concerning the ability of personal computers to store and manipulate data in the digital domain. This process is called **digital signal processing (DSP)**. The sampled data is generally stored in RAM as a large array of values. Normally, the data is in integer form. This is because microprocessors are generally faster at performing integer calculations than floating point calculations. Also, AD converters produce integer values, so this is doubly convenient. As an example, if 8-bit resolution is used, each sample point will require one byte (8 bits). The byte is the fundamental unit of computer memory storage, and thus storage as a simple byte multiple (8 bits, 16 bits, 32 bits) is straightforward. The data may be stored in either unipolar (unsigned) or bipolar (signed) formats. For unsigned systems, the data value “0” means “most negative peak”. In signed systems, a two's compliment form is normally used. Here, a data value of “0” means “0 V”. All negative values have their most significant bit set (for an 8 bit representation, $-1 = 11111111, -2 = 11111110,$ etc.). Input signals are often AC, thus signed representations are quite common.

In BASIC, the data array might be called `data()`, and individual elements are accessed by properly setting the array index. The first element is `data(1)`, the second is `data(2)`, and the `N` th element is `data(N)`. Although BASIC is useful, it is not as powerful as languages such as C, which offer direct manipulation of memory addresses via pointers. In C, the starting, or base, address of the data might be given the name `data\_ptr`. To access different elements in the array, an offset is added to the base. The address of the second element is `data\_ptr + 1`, the third is `data\_ptr + 2`, and so on. Alternately, we could just increment the pointer continuously, as in `data\_ptr++`. Direct manipulation of pointers is generally faster for the microprocessor than array indexing. To find the value stored at a given address, C uses the indirection operator, `*`. To assign the second data value to the variable `x`, you would say `x = *(data\_ptr + 1)` or simply `x = *data\_ptr++` if the pointer is being continuously incremented. The BASIC equivalent is `x = data(2)`.

No matter how the data is accessed, any mathematical function may be applied to the data elements. In the examples below, C is used to illustrate the concepts. Only the processing portion of the code is shown. The source array is called `src()` and has `TOTAL` number of elements. The destination array is called `dest()`. Also, questions concerning integer versus floating point representation are bypassed in order to keep the examples as straightforward as possible. In the real world, problems such as execution speed, round-off error, and overflow cannot be ignored.

Let's start with something simple. Suppose we would like to simulate the operation of an inverting buffer. An inverting buffer simply multiplies the input signal by −1. Here is how we do this in the digital domain: each point is multiplied by −1.

```
for( i=0; i<TOTAL; i++ )
{
    *dest++ = -1 * (*src++);
}
```

We can extend this concept a bit further by replacing “−1” with a variable called “gain”. In this way, we can alter the signal amplitude digitally.

Another useful calculation is average value. For signed data, the average value is equal to the DC offset component.

```
sum = 0;
for( i=0; i<TOTAL; i++ )
{
     sum += *src++;
}
average = sum / TOTAL;
```

In order to add a DC offset, a constant is added to each data element in turn.

```
for( i=0; i<TOTAL; i++ )
{
    *dest++ = offset + (*src++);
}
```

We might wish to combine two different waveforms into a third waveform. Simple addition is all that is required. (The example assumes that both arrays are the same size).

```
for( i=0; i<TOTAL; i++ )
{
    *dest++ = (*src++) + (*src2++);
}
```

If the two source arrays are multiplied instead of added, signal modulation may be produced.

These examples are relatively simple. With proper programming, a wide variety of functions may be simulated including spectral analysis, translation to new sampling rates, and the realization of complex filter functions that are impractical to implement in the analog domain. The major drawback of DSP techniques is the inherent computing power required. Thus, real-time application of complex DSP principles is not possible without using powerful computers. Specialized ICs have been designed to implement DSP sub-functions very quickly. These chips make realtime DSP practical and ecomical. The DSP ICs are programmable, thus they offer the distinct advantage of being able to produce different functions at different times by loading new programs. In this way, they are far more flexible than dedicated analog circuits. For example, an all-DSP guitar effects unit encompassing a broad range of functions such as equalization, pitch shifting, chorusing, reverberation and other effects can be designed. The amplitude and spectral shaping is performed directly in the digital domain instead of using dedicated analog circuitry.

This page titled 9.2.2.6: Extended Topic - Digital Signal Processing is shared under a CC BY-NC-SA 4.0 license and was authored, remixed, and/or curated by James M. Fiore via source content that was edited to the style and standards of the LibreTexts platform.

- **12.6: Extended Topic - Digital Signal Processing** by James M. Fiore is licensed CC BY-NC-SA 4.0. Original source: http://www.dissidents.com/resources/OperationalAmplifiersAndLinearICs_3E.pdf.

9.2.2.7: Summary

In this chapter we have covered the basic concepts of analog-to-digital and digitalto-analog conversion. By placing a signal in the digital domain, a variety of new analysis, storage, and transmission techniques become available.

The process of transforming an analog signal into a digital representation is referred to as quantization, or more simply, as digitizing or sampling. One of the most popular methods used is PCM, pulse code modulation. In this scheme, the input signal is measured, or sampled, at a constant rate. Each sample point is represented as a digital word. The sequence of words describes the input signal and is used to recreate it, if necessary. The ultimate accuracy of the conversion is dependent on the resolution and sampling rate of the system. Resolution refers to the number of bits present in the digital word. An 8-bit word can represent 256 steps, whereas a 16-bit word is much finer, offering 65,536 discrete steps. With so many steps, round-off error is much less of a problem in 16-bit systems than in, say, 8- or 12-bit systems.

The minimum allowable sampling rate is twice the highest input frequency. In other words, at least two samples per cycle are required for the highest harmonic in the input signal. Another way of stating this is that no input frequency component can exceed the Nyquist frequency, which is defined as one-half of the sampling frequency. If the input exceeds this limit, alias distortion may occur. Aliases are nonharmonically-related frequencies that are effectively created by improper sampling. In order to remove all possibility of alias distortion, special high-order low-pass filters, called anti-alias filters, are normally placed before the sampling circuitry.

Two popular methods of analog-to-digital conversion are the flash and successive approximation techniques. Flash converters are very fast, but require one comparator per output step, so they are not normally used where high resolution is required. Successive approximation takes longer than flash conversion, but can produce resolutions in excess of 16 bits. In either case, the conversion process is not instantaneous, and any fluctuation of the input signal during the conversion can produce errors. In order to alleviate this difficulty, special track-and-hold amplifiers are used between the anti-alias filters and the AD converter to create a non-varying signal.

Once the signal has been digitized, it may be stored in RAM or some other media for future use, or directly analyzed. Often, a personal computer can prove to be very useful for waveform analysis.

To reconstruct the waveform, a digital-to-analog converter is used. In essence, this is usually little more than a weighted summing amplifier. Due to accuracy and construction constraints, an $R/2R$ ladder technique is often employed. In order to smooth out the resulting waveform and remove any remaining digital “glitches”, the signal is passed through a reconstruction filter. This is a low-pass filter and is often called a smoothing filter. Generally, the process of digital-to-analog conversion is much faster than analog-to-digital conversion. Indeed, the successive approximation analog-to-digital scheme requires an internal digital-to-analog converter.

Applications for AD and DA converters range from laboratory instruments such as digital sampling oscilloscopes and hand-held digital multimeters, to industrial instrument and device control. AD and DA systems have found a large application market in the commercial sector. Uses include the popular music CD player and musical keyboard equipment.

This page titled 9.2.2.7: Summary is shared under a CC BY-NC-SA 4.0 license and was authored, remixed, and/or curated by James M. Fiore via source content that was edited to the style and standards of the LibreTexts platform.

- **12.7: Summary** by James M. Fiore is licensed CC BY-NC-SA 4.0. Original source: http://www.dissidents.com/resources/OperationalAmplifiersAndLinearICs_3E.pdf.

9.2.2.8: Problems

12.8.1: Review Questions

1. What is PCM?

2. Define the term resolution.

3. What is quantization?

4. What is an alias, and how is it produced? How is an alias avoided?

5. Define Nyquist frequency, and discuss the importance of this parameter.

6. Explain how a summing amplifier may be used to create a digital-to-analog converter.

7. Explain the difference between integral nonlinearity and differential nonlinearity.

8. What is a smoothing (reconstruction) filter?

9. What is the purpose of a track-and-hold amplifier?

10. Detail the differences between flash conversion and the successive approximation technique. Where would each type be used? What are their limitations?

11. Give several examples of possible DSP functions.

12.8.2: Problems

Analysis Problems

1. Determine the number of quantization steps for a 10-bit system.

2. A 14-bit converter produces a maximum peak-to-peak output of 2.5 V. What is the step size?

3. Determine the dynamic range of the converters in Problems 1 and 2.

4. We wish to resolve a 1 V peak-to-peak signal to at least 1 mV. What is the minimum allowable number of bits in the converted data?

5. We wish to create analog signals using an arbitrary waveform generator. If we send out digital words at the rate of 50 kHz, what is the maximum allowable conversion speed for the DAC?

6. Assume that we are trying to digitize ultrasonic signals lying between 25 kHz and 45 kHz.

A) What is the Nyquist frequency?

B) What is the minimum acceptable sampling rate?

7. Determine the maximum allowable conversion time for the ADC of Problem 6.

8. Given a 14-bit ADC,

A) Determine the number of comparators required for the flash technique.

B) Determine the number of comparisons required if the successive approximation technique is used.

9. Assume that a single 16-bit ADC is connected to an embedded computer. The sampling rate is 10 kHz. Determine the data rate in bytes per second.

10. Referring to Problem 9, if the computing device has 350 k bytes of RAM available for data storage, how much time does this represent?

11. DAT (digital audio tape) recorders normally use a 16 bit representation with a sampling rate of 48 kHz. If the unit is used to record a performance of Stravinsky's “Rite of Spring” (35 minutes total), what is the required storage capacity in bytes?

12. If the data is transferred serially from the DAT of Problem 11 to a digital signal processing IC in real time, what is the width of each individual pulse?

13. A 12-bit 2-microsecond DAC is used as part of a discrete successive approximation analog-to-digital converter. Assuming that logic delays and signal settling times are negligible, determine:

A) The minimum time allowable between sample points.

B) The maximum input signal frequency without aliasing.

14. A 6-bit video DA converter produces a maximum output swing of approximately 1.25 V (unipolar). Determine the output voltage for the following digital input words.

A) 000001

B) 100000

C) 111111

D) 011101

15. A 10-bit instrumentation DAC produces an output of 16 mV with an input of 0000000100. Determine:

A) The step size

B) The maximum output signal.

16. An 8-bit ADC produces a full scale output of 11111111 with a 2 V input signal. Determine the output word given the following inputs. (Assume that this converter rounds to the nearest output value and is unipolar.)

A) 100 mV

B) 10 μV

C) 0 V

D) 1.259 V

17. Assume that comparator/logic delays, amplifier settling times, and other factors require 400 ns total in a particular IC fabrication technique. If this technology is used to create AD converters, determine the maximum conversion time for the following 8-bit converters:

A) Flash

B) Successive approximation

C) Staircase/Ramp type

Design Problems

18. We wish to digitize human voice signals. The maximum input frequency is to be limited to 3 kHz and resolution to better than 0.5% of the maximum input value is required.

A) Draw a block diagram of the complete system.

B) Determine the minimum bit requirement.

C) Determine the minimum sampling rate if the Nyquist rate is set to 25% greater than the theoretical minimum.

D) Determine the anti-alias filter tuning frequency.

E) Determine the preferred conversion technique.

F) Determine which of the ICs presented in the chapter is best suited for this system.

19. We wish to design a system capable of digitizing complex signals with a spectrum ranging from DC to 400 kHz. Accuracy must be at least 0.2% of full scale.

A) Draw a block diagram of the complete system.

B) Determine the minimum bit requirement.

C) Determine the minimum sampling rate if the Nyquist rate is set to 20% greater than the theoretical minimum.

D) Determine the anti-alias filter tuning frequency.

E) Determine the preferred conversion technique.

F) Determine which of the ICs presented in the chapter is best suited for this system.

20. Write a computer algorithm that can be used to “flip” digital data back to front. (i.e., play it backwards).

21. Write a computer algorithm that will determine the maximum peak value of the digital data.

22. Write a computer algorithm that will determine the RMS value of the digital data.

This page titled 9.2.2.8: Problems is shared under a CC BY-NC-SA 4.0 license and was authored, remixed, and/or curated by James M. Fiore via source content that was edited to the style and standards of the LibreTexts platform.

- **12.8: Problems** by James M. Fiore is licensed CC BY-NC-SA 4.0. Original source: http://www.dissidents.com/resources/OperationalAmplifiersAndLinearICs_3E.pdf.

SECTION OVERVIEW

9.2.3: Sampling and Reconstruction

9.2.3.1: Signal Sampling

9.2.3.2: Sampling Theorem

9.2.3.3: Signal Reconstruction

9.2.3.4: Perfect Reconstruction

9.2.3.5: Aliasing Phenomena

9.2.3.6: Anti-Aliasing Filters

9.2.3.7: Discrete Time Processing of Continuous Time Signals

This page titled 9.2.3: Sampling and Reconstruction is shared under a CC BY 4.0 license and was authored, remixed, and/or curated by Richard Baraniuk et al. via source content that was edited to the style and standards of the LibreTexts platform.

9.2.3.1: Signal Sampling

Introduction

Digital computers can process discrete time signals using extremely flexible and powerful algorithms. However, most signals of interest are continuous time signals, which is how data almost always appears in nature. This module introduces the concepts behind converting continuous time signals into discrete time signals through a process called sampling.

Sampling

Sampling a continuous time signal produces a discrete time signal by selecting the values of the continuous time signal at evenly spaced points in time. Thus, sampling a continuous time signal x with sampling period T_s gives the discrete time signal x_s defined by $x_s(n) = x(nT_s)$. The sampling angular frequency is then given by $\omega_s = 2\pi/T_s$.

It should be intuitively clear that multiple continuous time signals sampled at the same rate can produce the same discrete time signal since uncountably many continuous time functions could be constructed that connect the points on the graph of any discrete time function. Thus, sampling at a given rate does not result in an injective relationship. Hence, sampling is, in general, not invertible.

Example 9.2.3.1.1

For instance, consider the signals x,y defined by

$$x(t) = \frac{\sin(t)}{t}$$

$$y(t) = \frac{\sin(5t)}{t}$$

and their sampled versions x_S,y_s with sampling period $T_s = \pi/2$

$$x_s(n) = \frac{\sin(n\pi/2)}{n\pi/2}$$

$$y_s(n) = \frac{\sin(n5\pi/2)}{n\pi/2}.$$

Notice that since

$$\sin(n5\pi/2) = \sin(n2\pi + n\pi/2) = \sin(n\pi/2)$$

it follows that

$$y_s(n) = \frac{\sin(n\pi/2)}{n\pi/2} = x_s(n).$$

Hence, x and y provide an example of distinct functions with the same sampled versions at a specific sampling rate.

It is also useful to consider the relationship between the frequency domain representations of the continuous time function and its sampled versions. Consider a signal x sampled with sampling period T_s to produce the discrete time signal $x_s(n) = x(nT_s)$. The spectrum $X_s(\omega)$ for $\omega \in [-\pi, \pi)$ of x_s is given by

$$X_s(\omega) = \sum_{n=-\infty}^{\infty} x\left(nT_s\right) e^{-j\omega n}.$$

Using the continuous time Fourier transform, $x(tT_s)$ can be represented as

$$x\left(tT_s\right) = \frac{1}{2\pi T_s} \int_{-\infty}^{\infty} X\left(\frac{\omega_1}{T_s}\right) e^{j\omega_1 t} d\omega_1.$$

Thus, the unit sampling period version of $x(tT_s)$, which is $x(nT_s)$ can be represented as

$$x\left(nT_s\right) = \frac{1}{2\pi T_s}\int_{-\infty}^{\infty} X\left(\frac{\omega_1}{T_s}\right) e^{j\omega_1 n} d\omega_1.$$

This is algebraically equivalent to the representation

$$x\left(nT_s\right) = \frac{1}{T_s}\sum_{k=-\infty}^{\infty}\frac{1}{2\pi}\int_{-\pi}^{\pi} X\left(\frac{\omega_1 - 2\pi k}{T_s}\right) e^{j(\omega_1 - 2\pi k)n} d\omega_1,$$

which reduces by periodicity of complex exponentials to

$$x\left(nT_s\right) = \frac{1}{T_s}\sum_{k=-\infty}^{\infty}\frac{1}{2\pi}\int_{-\pi}^{\pi} X\left(\frac{\omega_1 - 2\pi k}{T_s}\right) e^{j\omega_1 n} d\omega_1.$$

Hence, it follows that

$$X_s(\omega) = \frac{1}{T_s}\sum_{k=-\infty}^{\infty}\sum_{n=-\infty}^{\infty}\left(\int_{-\pi}^{\pi} X\left(\frac{\omega_1 - 2\pi k}{T_s}\right) e^{j\omega_1 n} d\omega_1\right) e^{-j\omega n}.$$

Noting that the above expression contains a Fourier series and inverse Fourier series pair, it follows that

$$X_s(\omega) = \frac{1}{T_s}\sum_{k=-\infty}^{\infty} X\left(\frac{\omega - 2\pi k}{T_s}\right).$$

Hence, the spectrum of the sampled signal is, intuitively, the scaled sum of an infinite number of shifted and time scaled copies of original signal spectrum. Aliasing, which will be discussed in depth in later modules, occurs when these shifted spectrum copies overlap and sum together. Note that when the original signal x is bandlimited to $(-\pi/T_s, \pi/T_s)$ no overlap occurs, so each period of the sampled signal spectrum has the same form as the original signal spectrum. This suggest that if we sample a bandlimited signal at a sufficiently high sampling rate, we can recover it from its samples as will be further described in the modules on the Nyquist-Shannon sampling theorem and on perfect reconstruction.

Sampling Summary

Sampling a continuous time signal produces a discrete time signal by selecting the values of the continuous time signal at equally spaced points in time. However, we have shown that this relationship is not injective as multiple continuous time signals can be sampled at the same rate to produce the same discrete time signal. This is related to a phenomenon called aliasing which will be discussed in later modules. Consequently, the sampling process is not, in general, invertible. Nevertheless, as will be shown in the module concerning reconstruction, the continuous time signal can be recovered from its sampled version if some additional assumptions hold.

This page titled 9.2.3.1: Signal Sampling is shared under a CC BY 4.0 license and was authored, remixed, and/or curated by Richard Baraniuk et al. via source content that was edited to the style and standards of the LibreTexts platform.

- **10.1: Signal Sampling** by Richard Baraniuk et al. is licensed CC BY 3.0. Original source: https://cnx.org/contents/d2CEAGW515.4:l-e_wSqM24/Signal-Classifications-and-Properties.

9.2.3.2: Sampling Theorem

Introduction

With the introduction of the concept of signal sampling, which produces a discrete time signal by selecting the values of the continuous time signal at evenly spaced points in time, it is now possible to discuss one of the most important results in signal processing, the Nyquist-Shannon sampling theorem. Often simply called the sampling theorem, this theorem concerns signals, known as bandlimited signals, with spectra that are zero for all frequencies with absolute value greater than or equal to a certain level. The theorem implies that there is a sufficiently high sampling rate at which a bandlimited signal can be recovered exactly from its samples, which is an important step in the processing of continuous time signals using the tools of discrete time signal processing.

Nyquist-Shannon Sampling Theorem

Statement of the Sampling Theorem

The Nyquist-Shannon sampling theorem concerns signals with continuous time Fourier transforms that are only nonzero on the interval $(-B, B)$ for some constant B. Such a function is said to be bandlimited to $(-B, B)$. Essentially, the sampling theorem has already been implicitly introduced in the previous module concerning sampling. Given a continuous time signals x with continuous time Fourier transform X, recall that the spectrum X_s of sampled signal x_s with sampling period T_s is given by

$$X_s(\omega) = \frac{1}{T_s} \sum_{k=-\infty}^{\infty} X\left(\frac{\omega - 2\pi k}{T_s}\right).$$

It had previously been noted that if x is bandlimited to $(-\pi/T_s, \pi/T_s)$, the period of X_s centered about the origin has the same form as X scaled in frequency since no aliasing occurs. This is illustrated in Figure 9.2.3.2.1. Hence, if any two $(-\pi/T_s, \pi/T_s)$ bandlimited continuous time signals sampled to the same signal, they would have the same continuous time Fourier transform and thus be identical. Thus, for each discrete time signal there is a unique $(-\pi/T_s, \pi/T_s)$ bandlimited continuous time signal that samples to the discrete time signal with sampling period T_s. Therefore, this $(-\pi/T_s, \pi/T_s)$ bandlimited signal can be found from the samples by inverting this bijection.

This is the essence of the sampling theorem. More formally, the sampling theorem states the following. If a signal x is bandlimited to $(-B, B)$, it is completely determined by its samples with sampling rate $\omega_s = 2B$. That is to say, x can be reconstructed exactly from its samples x_s with sampling rate $\omega_s = 2B$. The angular frequency $2B$ is often called the angular Nyquist rate. Equivalently, this can be stated in terms of the sampling period $T_s = 2\pi/\omega_s$. If a signal x is bandlimited to $(-B, B)$, it is completely determined by its samples with sampling period $T_s = \pi/B$. That is to say, x can be reconstructed exactly from its samples x_s with sampling period T_s.

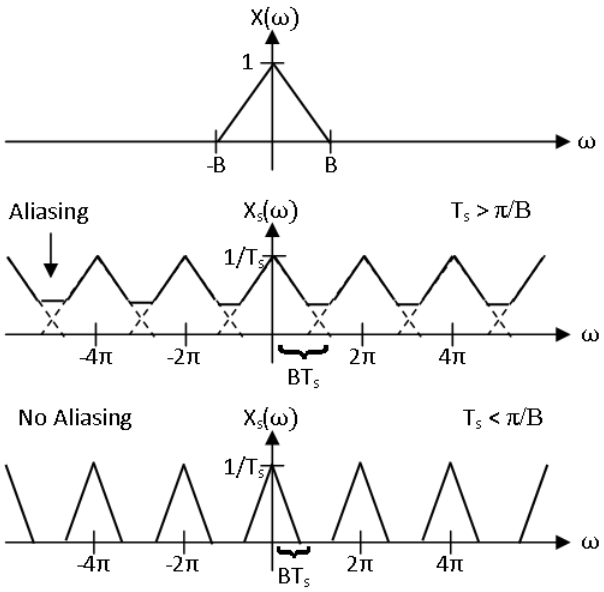


Figure 9.2.3.2.1: The spectrum of a bandlimited signals is shown as well as the spectra of its samples at rates above and below the Nyquist frequency. As is shown, no aliasing occurs above the Nyquist frequency, and the period of the samples spectrum centered about the origin has the same form as the spectrum of the original signal scaled in frequency. Below the Nyquist frequency, aliasing can occur and causes the spectrum to take a different than the original spectrum.

Proof of the Sampling Theorem

The above discussion has already shown the sampling theorem in an informal and intuitive way that could easily be refined into a formal proof. However, the original proof of the sampling theorem, which will be given here, provides the interesting observation that the samples of a signal with period T_s provide Fourier series coefficients for the original signal spectrum on $(-\pi/T_s, \pi/T_s)$.

Let x be a $(-\pi/T_s, \pi/T_s)$ bandlimited signal and x_s be its samples with sampling period T_s. We can represent x in terms of its spectrum X using the inverse continuous time Fourier transform and the fact that x is bandlimited. The result is

$$x(t) = \frac{1}{2\pi} \int_{-\pi/T_x}^{\pi/T_s} X(\omega) e^{j\omega t} d\omega$$

This representation of x may then be sampled with sampling period T_s to produce

$$x_s(n) = x_s(nT_s) = \frac{1}{2\pi} \int_{-\pi/T_s}^{\pi/T_s} X(\omega) e^{j\omega n T_s} d\omega$$

Noticing that this indicates that $x_s(n)$ is the nth continuous time Fourier series coefficient for $X(\omega)$ on the interval $(-\pi/T_s, \pi/T_s)$, it is shown that the samples determine the original spectrum $X(\omega)$ and, by extension, the original signal itself.

Perfect Reconstruction

Another way to show the sampling theorem is to derive the reconstruction formula that gives the original signal $\tilde{x} = x$ from its samples x_s with sampling period T_s, provided x is bandlimited to $(-\pi/T_s, \pi/T_s)$. This is done in the module on perfect reconstruction. However, the result, known as the Whittaker-Shannon reconstruction formula, will be stated here. If the requisite conditions hold, then the perfect reconstruction is given by

$$x(t) = \sum_{n=-\infty}^{\infty} x_s(n) \operatorname{sinc}(t/T_s - n)$$

where the sinc function is defined as

$$\operatorname{sinc}(t) = \frac{\sin(\pi t)}{\pi t}.$$

From this, it is clear that the set

$$\{\text{sinc}\,(t/T_s - n) \mid n \in \mathbb{Z}\}$$

forms an orthogonal basis for the set of $(-\pi/T_s, \pi/T_s)$ bandlimited signals, where the coefficients of a $(-\pi/T_s, \pi/T_s)$ signal in this basis are its samples with sampling period T_s.

Practical Implications

Discrete Time Processing of Continuous Time Signals

The Nyquist-Shannon Sampling Theorem and the Whittaker-Shannon Reconstruction formula enable discrete time processing of continuous time signals. Because any linear time invariant filter performs a multiplication in the frequency domain, the result of applying a linear time invariant filter to a bandlimited signal is an output signal with the same bandlimit. Since sampling a bandlimited continuous time signal above the Nyquist rate produces a discrete time signal with a spectrum of the same form as the original spectrum, a discrete time filter could modify the samples spectrum and perfectly reconstruct the output to produce the same result as a continuous time filter. This allows the use of digital computing power and flexibility to be leveraged in continuous time signal processing as well. This is more thoroughly described in the final module of this chapter.

Psychoacoustics

The properties of human physiology and psychology often inform design choices in technologies meant for interaction with people. For instance, digital devices dealing with sound use sampling rates related to the frequency range of human vocalizations and the frequency range of human auditory sensitivity. Because most of the sounds in human speech concentrate most of their signal energy between 5 Hz and 4 kHz, most telephone systems discard frequencies above 4 kHz and sample at a rate of 8 kHz. Discarding the frequencies greater than or equal to 4 kHz through use of an anti-aliasing filter is important to avoid aliasing, which would negatively impact the quality of the output sound as is described in a later module. Similarly, human hearing is sensitive to frequencies between 20 Hz and 20 kHz. Therefore, sampling rates for general audio waveforms placed on CDs were chosen to be greater than 40 kHz, and all frequency content greater than or equal to some level is discarded. The particular value that was chosen, 44.1 kHz, was selected for other reasons, but the sampling theorem and the range of human hearing provided a lower bound for the range of choices.

Sampling Theorem Summary

The Nyquist-Shannon Sampling Theorem states that a signal bandlimited to $(-\pi/T_s, \pi/T_s)$ can be reconstructed exactly from its samples with sampling period T_s. The Whittaker-Shannon interpolation formula, which will be further described in the section on perfect reconstruction, provides the reconstruction of the unique $(-\pi/T_s, \pi/T_s)$ bandlimited continuous time signal that samples to a given discrete time signal with sampling period T_s. This enables discrete time processing of continuous time signals, which has many powerful applications.

This page titled 9.2.3.2: Sampling Theorem is shared under a CC BY 4.0 license and was authored, remixed, and/or curated by Richard Baraniuk et al. via source content that was edited to the style and standards of the LibreTexts platform.

- **10.2: Sampling Theorem** by Richard Baraniuk et al. is licensed CC BY 3.0. Original source: https://cnx.org/contents/d2CEAGW515.4:l-e_wSqM24/Signal-Classifications-and-Properties.

9.2.3.3: Signal Reconstruction

Introduction

The sampling process produces a discrete time signal from a continuous time signal by examining the value of the continuous time signal at equally spaced points in time. Reconstruction, also known as interpolation, attempts to perform an opposite process that produces a continuous time signal coinciding with the points of the discrete time signal. Because the sampling process for general sets of signals is not invertible, there are numerous possible reconstructions from a given discrete time signal, each of which would sample to that signal at the appropriate sampling rate. This module will introduce some of these reconstruction schemes.

Reconstruction

Reconstruction Process

The process of reconstruction, also commonly known as interpolation, produces a continuous time signal that would sample to a given discrete time signal at a specific sampling rate. Reconstruction can be mathematically understood by first generating a continuous time impulse train

$$x_{imp}(t) = \sum_{n=-\infty}^{\infty} x_s(n)\delta\left(t - nT_s\right)$$

from the sampled signal x_s with sampling period T_s and then applying a lowpass filter G that satisfies certain conditions to produce an output signal $\tilde{x}$. If G has impulse response g, then the result of the reconstruction process, illustrated in Figure 9.2.3.3.1, is given by the following computation, the final equation of which is used to perform reconstruction in practice.

$$\begin{aligned}\tilde{x}(t) &= \left(x_{imp} * g\right)(t) \\ &= \int_{-\infty}^{\infty} x_{imp}(\tau)g(t-\tau)d\tau \\ &= \int_{-\infty}^{\infty} \sum_{n=-\infty}^{\infty} x_s(n)\delta\left(\tau - nT_s\right) g(t-\tau)d\tau \\ &= \sum_{n=-\infty}^{\infty} x_s(n) \int_{-\infty}^{\infty} \delta\left(\tau - nT_s\right) g(t-\tau)d\tau \\ &= \sum_{n=-\infty}^{\infty} x_s(n)g\left(t - nT_s\right) \end{aligned} \tag{9.2.3.3.1}$$

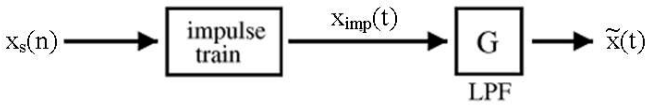


Figure 9.2.3.3.1: Block diagram of reconstruction process for a given lowpass filter G.

Reconstruction Filters

In order to guarantee that the reconstructed signal $\tilde{x}$ samples to the discrete time signal x_s from which it was reconstructed using the sampling period T_s, the lowpass filter G must satisfy certain conditions. These can be expressed well in the time domain in terms of a condition on the impulse response g of the lowpass filter G. The sufficient condition to be a reconstruction filters that we will require is that, for all $n \in \mathbb{Z}$,

$$g\left(nT_s\right) = \begin{cases} 1 & n = 0 \\ 0 & n \neq 0 \end{cases} = \delta(n)\ .$$

This means that gg sampled at a rate T_s produces a discrete time unit impulse signal. Therefore, it follows that sampling $\tilde{x}$ with sampling period T_s results in

$$\begin{aligned}\tilde{x}(nT_s) &= \sum_{m=-\infty}^{\infty} x_s(m)g(nT_s - mT_s) \\ &= \sum_{m=-\infty}^{\infty} x_s(m)g((n-m)T_s) \\ &= \sum_{m=-\infty}^{\infty} x_s(m)\delta(n-m) \\ &= x_s(n), \end{aligned} \tag{9.2.3.3.2}$$

which is the desired result for reconstruction filters.

Cardinal Basis Splines

Since there are many continuous time signals that sample to a given discrete time signal, additional constraints are required in order to identify a particular one of these. For instance, we might require our reconstruction to yield a spline of a certain degree, which is a signal described in piecewise parts by polynomials not exceeding that degree. Additionally, we might want to guarantee that the function and a certain number of its derivatives are continuous.

This may be accomplished by restricting the result to the span of sets of certain splines, called **basis splines** or **B-splines**. Specifically, if a nth degree spline with continuous derivatives up to at least order $n-1$ is required, then the desired function for a given T_s belongs to the span of $\{B_n(t/T_s - k) \mid k \in \mathbb{Z}\}$ where

$$B_n = B_0 * B_{n-1}$$

for $n \geq 1$ and

$$B_0(t) = \begin{cases} 1 & -1/2 < t < 1/2 \\ 0 & \text{otherwise} \end{cases}.$$

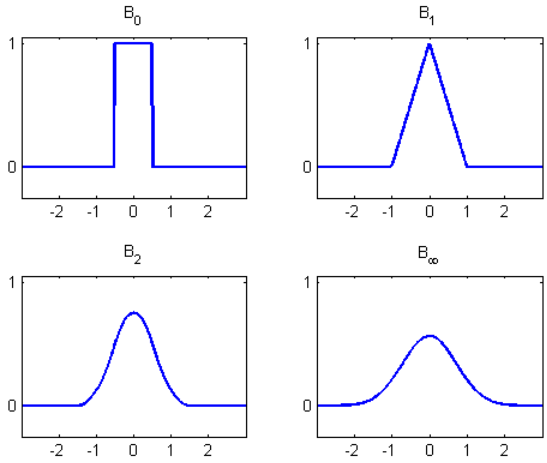


Figure 9.2.3.3.2: The basis splines B_n are shown in the above plots. Note that, except for the order 0 and order 1 functions, these functions do not satisfy the conditions to be reconstruction filters. Also notice that as the order increases, the functions approach the Gaussian function, which is exactly B_∞.

However, the basis splines B_n do not satisfy the conditions to be a reconstruction filter for $n \geq 2$ as is shown in Figure 9.2.3.3.2. Still, the B_n are useful in defining the cardinal basis splines, which do satisfy the conditions to be reconstruction filters. If we let b_n be the samples of B_n on the integers, it turns out that b_n has an inverse b_n^{-1} with respect to the operation of convolution for each n. This is to say that $b_n^{-1} * b_n = \delta$. The cardinal basis spline of order nn for reconstruction with sampling period T_s is defined as

$$\eta_n(t) = \sum_{k=-\infty}^{\infty} b_n^{-1}(k) B_n(t/T_s - k).$$

In order to confirm that this satisfies the condition to be a reconstruction filter, note that

$$\eta_n(mT_s) = \sum_{k=-\infty}^{\infty} b_n^{-1}(k) B_n(m-k) = \left(b_n^{-1} * b_n\right)(m) = \delta(m).$$

Thus, η_n is a valid reconstruction filter. Since η_n is an nth degree spline with continuous derivatives up to order $n-1$, the result of the reconstruction will be a nth degree spline with continuous derivatives up to order $n-1$.

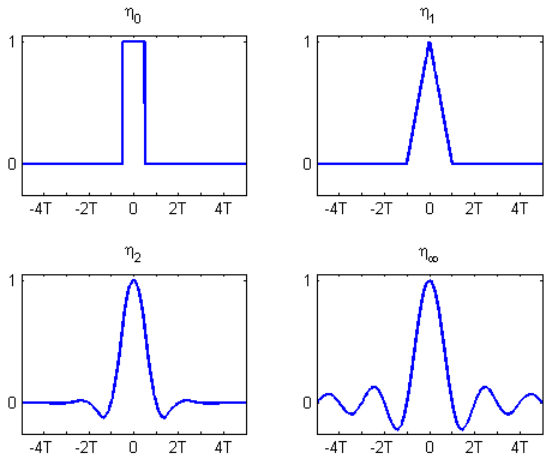


Figure 9.2.3.3.3: The above plots show cardinal basis spline functions η_0, η_1, η_2, and η_∞. Note that the functions satisfy the conditions to be reconstruction filters. Also, notice that as the order increases, the cardinal basis splines approximate the sinc function, which is exactly η_∞. Additionally, these filters are acausal.

The lowpass filter with impulse response equal to the cardinal basis spline η_0 of order 0 is one of the simplest examples of a reconstruction filter. It simply extends the value of the discrete time signal for half the sampling period to each side of every sample, producing a piecewise constant reconstruction. Thus, the result is discontinuous for all nonconstant discrete time signals.

Likewise, the lowpass filter with impulse response equal to the cardinal basis spline η_1 of order 1 is another of the simplest examples of a reconstruction filter. It simply joins the adjacent samples with a straight line, producing a piecewise linear reconstruction. In this way, the reconstruction is continuous for all possible discrete time signals. However, unless the samples are collinear, the result has discontinuous first derivatives.

In general, similar statements can be made for lowpass filters with impulse responses equal to cardinal basis splines of any order. Using the nth order cardinal basis spline η_n, the result is a piecewise degree nn polynomial. Furthermore, it has continuous derivatives up to at least order $n-1$. However, unless all samples are points on a polynomial of degree at most n, the derivative of order nn will be discontinuous.

Reconstructions of the discrete time signal given in Figure 9.2.3.3.4 using several of these filters are shown in Figure 9.2.3.3.5. As the order of the cardinal basis spline increases, notice that the reconstruction approaches that of the infinite order cardinal spline η_∞, the sinc function. As will be shown in the subsequent section on perfect reconstruction, the filters with impulse response equal to the sinc function play an especially important role in signal processing.

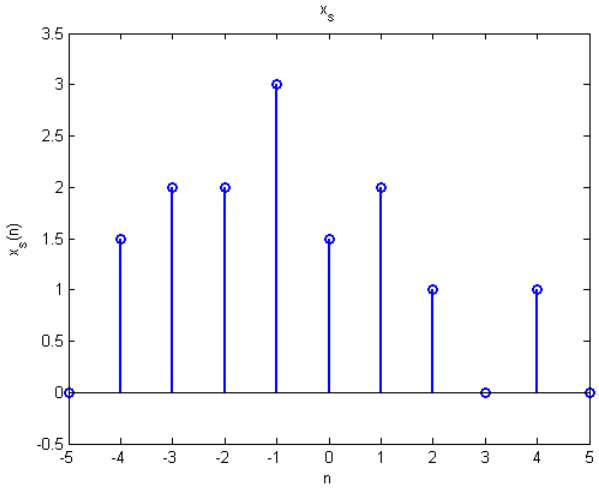


Figure 9.2.3.3.4: The above plot shows an example discrete time function. This discrete time function will be reconstructed using sampling period T_s using several cardinal basis splines in Figure 9.2.3.3.5.

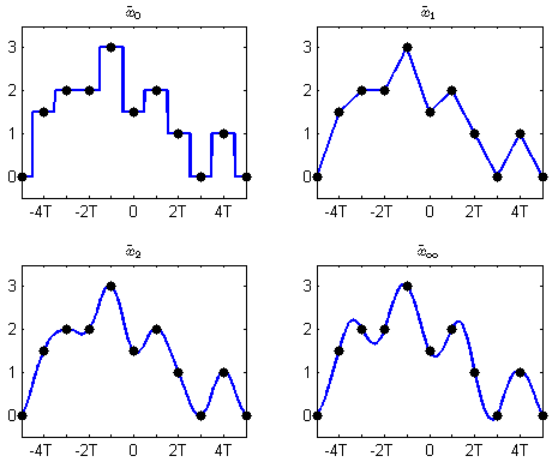


Figure 9.2.3.3.5: The above plots show interpolations of the discrete time signal given in Figure 9.2.3.3.4 using lowpass filters with impulse responses given by the cardinal basis splines shown in Figure 9.2.3.3.3. Notice that the interpolations become increasingly smooth and approach the sinc interpolation as the order increases.

Reconstruction Summary

Reconstruction of a continuous time signal from a discrete time signal can be accomplished through several schemes. However, it is important to note that reconstruction is not the inverse of sampling and only produces one possible continuous time signal that samples to a given discrete time signal. As is covered in the subsequent module, perfect reconstruction of a bandlimited continuous time signal from its sampled version is possible using the Whittaker-Shannon reconstruction formula, which makes use of the ideal lowpass filter and its sinc function impulse response, if the sampling rate is sufficiently high.

This page titled 9.2.3.3: Signal Reconstruction is shared under a CC BY 4.0 license and was authored, remixed, and/or curated by Richard Baraniuk et al. via source content that was edited to the style and standards of the LibreTexts platform.

- **10.3: Signal Reconstruction** by Richard Baraniuk et al. is licensed CC BY 3.0. Original source: https://cnx.org/contents/d2CEAGW515.4:l-e_wSqM24/Signal-Classifications-and-Properties.

9.2.3.4: Perfect Reconstruction

Introduction

If certain additional assumptions about the original signal and sampling rate hold, then the original signal can be recovered exactly from its samples using a particularly important type of filter. More specifically, it will be shown that if a bandlimited signal is sampled at a rate greater than twice its bandlimit, the Whittaker-Shannon reconstruction formula perfectly reconstructs the original signal. This formula makes use of the ideal lowpass filter, which is related to the sinc function. This is extremely useful, as sampled versions of continuous time signals can be filtered using discrete time signal processing, often in a computer. The results may then be reconstructed to produce the same continuous time output as some desired continuous time system.

Perfect Reconstruction

In order to understand the conditions for perfect reconstruction and the filter it employs, consider the following. As a beginning, a sufficient condition under which perfect reconstruction is possible will be discussed. Subsequently, the filter and process used for perfect reconstruction will be detailed.

Recall that the sampled version x_s of a continuous time signal x with sampling period T_s has a spectrum given by

$$X_s(\omega) = \frac{1}{T_s} \sum_{k=-\infty}^{\infty} X\left(\frac{\omega - 2\pi k}{T_s}\right).$$

As before, note that if x is bandlimited to $(-\pi/T_s, \pi/T_s)$, meaning that X is only nonzero on $(-\pi/T_s, \pi/T_s)$, then each period of X_s has the same form as X. Thus, we can identify the original spectrum X from the spectrum of the samples X_s and, by extension, the original signal x from its samples x_s at rate T_s if x is bandlimited to $(-\pi/T_s, \pi/T_s)$.

If a signal x is bandlimited to $(-B, B)$, then it is also bandlimited to $(-\pi/T_s, \pi/T_s)$ provided that $T_s < \pi/B$. Thus, if we ensure that x is sampled to x_s with sufficiently high sampling angular frequency $\omega_s = 2\pi/T_s > 2B$ and have a way of identifying the unique $(-\pi/T_s, \pi/T_s)$ bandlimited signal corresponding to a discrete time signal at sampling period T_s, then x_s can be used to reconstruct $\tilde{x} = x$ exactly. The frequency $2B$ is known as the angular Nyquist rate. Therefore, the condition that the sampling rate $\omega_s = 2\pi/T_s > 2B$ be greater than the Nyquist rate is a sufficient condition for perfect reconstruction to be possible.

The correct filter must also be known in order to perform perfect reconstruction. The ideal lowpass filter defined by $G(\omega) = T_s(u(\omega + \pi/T_s) - u(\omega - \pi/T_s))$, which is shown in Figure 9.2.3.4.1, removes all signal content not in the frequency range $(-\pi/T_s, \pi/T_s)$. Therefore, application of this filter to the impulse train $\sum_{n=-\infty}^{\infty} x_s(n)\delta(t - nT_s)$ results in an output bandlimited to $(-\pi/T_s, \pi/T_s)$.

We now only need to confirm that the impulse response g of the filter G satisfies our sufficient condition to be a reconstruction filter. The inverse Fourier transform of $G(\omega)$ is

$$g(t) = \operatorname{sinc}(t/T_s) = \begin{cases} 1 & t = 0 \\ \frac{\sin(\pi t/T_s)}{\pi t/T_s} & t \neq 0 \end{cases},$$

which is shown in Figure 9.2.3.4.6. Hence,

$$g(nT_s) = \operatorname{sinc}(n) = \begin{cases} 1 & n = 0 \\ \frac{\sin(\pi n)}{\pi n} & n \neq 0 \end{cases} = \begin{cases} 1 & n = 0 \\ 0 & n \neq 0 \end{cases} = \delta(n).$$

Therefore, the ideal lowpass filter G is a valid reconstruction filter. Since it is a valid reconstruction filter and always produces an output that is bandlimited to $(-\pi/T_s, \pi/T_s)$, this filter always produces the unique $(-\pi/T_s, \pi/T_s)$ bandlimited signal that samples to a given discrete time sequence at sampling period T_s when the impulse train $\sum_{n=-\infty}^{\infty} x_s(n)\delta(t - nT_s)$ is input.

Therefore, we can always reconstruct any $(-\pi/T_s, \pi/T_s)$ bandlimited signal from its samples at sampling period T_s by the formula

$$x(t) = \sum_{n=-\infty}^{\infty} x_s(n) \operatorname{sinc}(t/T_s - n).$$

This perfect reconstruction formula is known as the **Whittaker-Shannon interpolation formula** and is sometimes also called the **cardinal series**. In fact, the sinc function is the infinite order cardinal basis spline η_∞. Consequently, the set $\{\operatorname{sinc}(t/T_s - n) \mid n \in \mathbb{Z}\}$

forms a basis for the vector space of $(-\pi/T_s, \pi/T_s)$ bandlimited signals where the signal samples provide the corresponding coefficients. It is a simple exercise to show that this basis is, in fact, an orthogonal basis.

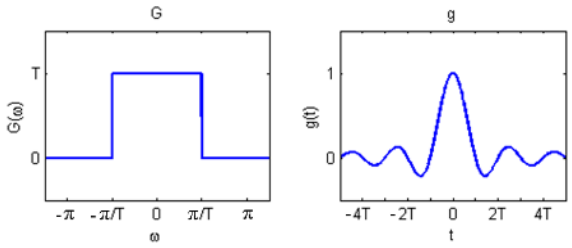


Figure 9.2.3.4.1: The above plots show the ideal lowpass filter and its inverse Fourier transform, the sinc function.

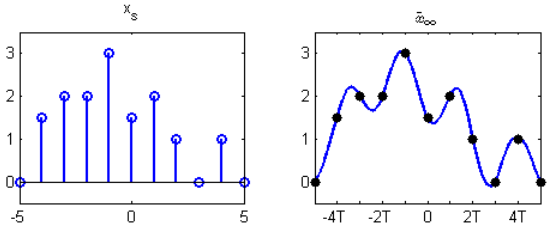


Figure 9.2.3.4.2: The plots show an example discrete time signal and its Whittaker-Shannon sinc reconstruction.

Perfect Reconstruction Summary

This module has shown that bandlimited continuous time signals can be reconstructed exactly from their samples provided that the sampling rate exceeds the Nyquist rate, which is twice the bandlimit. The Whittaker-Shannon reconstruction formula computes this perfect reconstruction using an ideal lowpass filter, with the resulting signal being a sum of shifted sinc functions that are scaled by the sample values. Sampling below the Nyquist rate can lead to aliasing which makes the original signal irrecoverable as is described in the subsequent module. The ability to perfectly reconstruct bandlimited signals has important practical implications for the processing of continuous time signals using the tools of discrete time signal processing.

This page titled 9.2.3.4: Perfect Reconstruction is shared under a CC BY 4.0 license and was authored, remixed, and/or curated by Richard Baraniuk et al. via source content that was edited to the style and standards of the LibreTexts platform.

- **10.4: Perfect Reconstruction** by Richard Baraniuk et al. is licensed CC BY 3.0. Original source: https://cnx.org/contents/d2CEAGW515.4:l-e_wSqM24/Signal-Classifications-and-Properties.

9.2.3.5: Aliasing Phenomena

Introduction

Through discussion of the Nyquist-Shannon sampling theorem and Whittaker-Shannon reconstruction formula, it has already been shown that a $(-B, B)$ continuous time signal can be reconstructed from its samples at rate $\omega_s = 2\pi/T_s$ via the sinc interpolation filter if $\omega_s > 2B$. Now, this module will investigate a problematic phenomenon, called aliasing, that can occur if this sufficient condition for perfect reconstruction does not hold. When aliasing occurs the spectrum of the samples has different form than the original signal spectrum, so the samples cannot be used to reconstruct the original signal through Whittaker-Shannon interpolation.

Aliasing

Aliasing occurs when each period of the spectrum of the samples does not have the same form as the spectrum of the original signal. Given a continuous time signals x with continuous time Fourier transform X, recall that the spectrum X_s of sampled signal x_s with sampling period T_s is given by

$$X_s(\omega) = \frac{1}{T_s} \sum_{k=-\infty}^{\infty} X\left(\frac{\omega - 2\pi k}{T_s}\right).$$

As has already been mentioned several times, if x is bandlimited to $(-\pi/T_s, \pi/T_s)$ then each period of X_s has the same form as X. However, if x is not bandlimited to $(-\pi/T_s, \pi/T_s)$, then the $X\left(\frac{\omega-2\pi k}{T_s}\right)$ can overlap and sum together. This is illustrated in Figure 9.2.3.5.1 in which sampling above the Nyquist frequency produces a samples spectrum of the same shape as the original signal, but sampling below the Nyquist frequency produces a samples spectrum with very different shape. Whittaker-Shannon interpolation of each of these sequences produces different results. The low frequencies not affected by the overlap are the same, but there is noise content in the higher frequencies caused by aliasing. Higher frequency energy masquerades as low energy content, a highly undesirable effect.

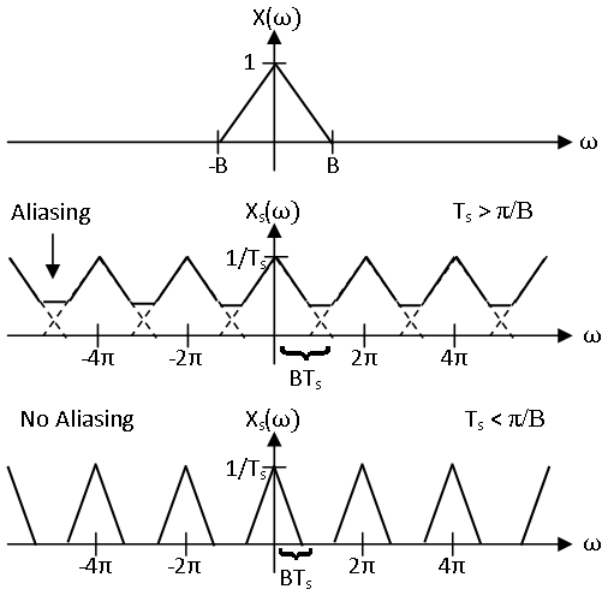


Figure 9.2.3.5.1: The spectrum of a bandlimited signals is shown as well as the spectra of its samples at rates above and below the Nyquist frequency. As is shown, no aliasing occurs above the Nyquist frequency, and the period of the samples spectrum centered about the origin has the same form as the spectrum of the original signal scaled in frequency. Below the Nyquist frequency, aliasing can occur and causes the spectrum to take a different than the original spectrum.

Unlike when sampling above the Nyquist frequency, sampling below the Nyquist frequency does not yield an injective (one-to-one) function from the $(-B, B)$ bandlimited continuous time signals to the discrete time signals. Any signal x with spectrum X which overlaps and sums to X_s samples to x_s. It should be intuitively clear that there are very many $(-B, B)$ bandlimited signals that sample to a given discrete time signal below the Nyquist frequency, as is demonstrated in Figure 9.2.3.5.2. It is quite easy to construct uncountably infinite families of such signals.

Aliasing obtains it name from the fact that multiple, in fact infinitely many, $(-B, B)$ bandlimited signals sample to the same discrete sequence if $\omega_s < 2B$. Thus, information about the original signal is lost in this noninvertible process, and these different signals effectively assume the same identity, an “alias”. Hence, under these conditions the Whittaker-Shannon interpolation formula will not produce a perfect reconstruction of the original signal but will instead give the unique $(-\omega_s/2, \omega_s/2)$ bandlimited signal that samples to the discrete sequence.

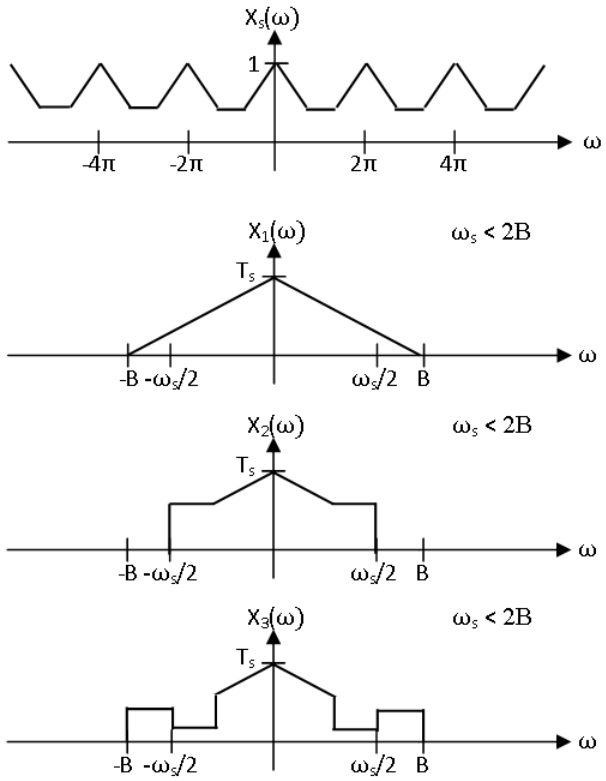


Figure 9.2.3.5.2: The spectrum of a discrete time signal x_s, taken from Figure 9.2.3.5.1, is shown along with the spectra of three $(-B, B)$ signals that sample to it at rate $\omega_s < 2B$. From the sampled signal alone, it is impossible to tell which, if any, of these was sampled at rate ω_s to produce x_s. In fact, there are infinitely many $(-B, B)$ bandlimited signals that sample to x_s at a sampling rate below the Nyquist rate.

Aliasing Demonstration

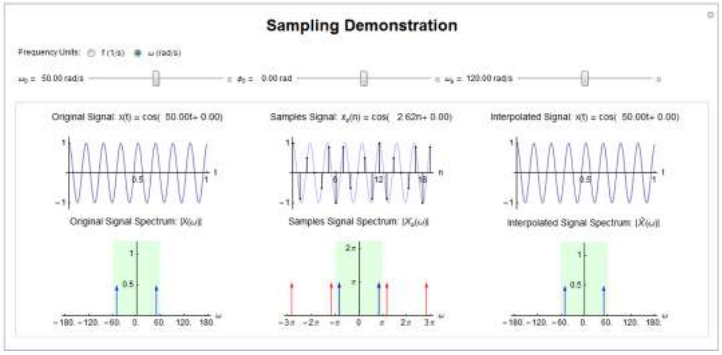


Figure 9.2.3.5.3: Interact (when online) with a Mathematica CDF demonstrating sampling and aliasing for a sinusoid. To Download, right-click and save target as .cdf.

Aliasing Summary

Aliasing, essentially the signal processing version of identity theft, occurs when each period of the spectrum of the samples does not have the same form as the spectrum of the original signal. As has been shown, there can be infinitely many $(-B, B)$ bandlimited signals that sample to a given discrete time signal x_s at a rate $\omega_s = 2\pi/T_s < 2B$ below the Nyquist frequency. However, there is a unique $(-B, B)$ bandlimited signal that samples to x_s, which is given by the Whittaker-Shannon interpolation of x_s, at rate $\omega_s \geq 2B$ as no aliasing occurs above the Nyquist frequency. Unfortunately, sufficiently high sampling rates cannot always be produced. Aliasing is detrimental to many signal processing applications, so in order to process continuous time signals using discrete time

tools, it is often necessary to find ways to avoid it other than increasing the sampling rate. Thus, anti-aliasing filters, are of practical importance.

This page titled 9.2.3.5: Aliasing Phenomena is shared under a CC BY 4.0 license and was authored, remixed, and/or curated by Richard Baraniuk et al. via source content that was edited to the style and standards of the LibreTexts platform.

- **10.5: Aliasing Phenomena** by Richard Baraniuk et al. is licensed CC BY 3.0. Original source: https://cnx.org/contents/d2CEAGW515.4:l-e_wSqM24/Signal-Classifications-and-Properties.

9.2.3.6: Anti-Aliasing Filters

Introduction

It has been shown that a $(-B, B)$ bandlimited signal can be perfectly reconstructed from its samples at a rate $\omega_s = 2\pi/T_s \geq B$. However, it is not always practically possible to produce sufficiently high sampling rates or to ensure that the input is bandlimited in real situations. Aliasing, which manifests itself as a difference in shape between the periods of the samples signal spectrum and the original spectrum, would occur without any further measures to correct this. Thus, it often becomes necessary to filter out signal energy at frequencies above $\omega_s/2$ in order to avoid the detrimental effects of aliasing. This is the role of the anti-aliasing filter, a lowpass filter applied before sampling to ensure that the signal is $(-\omega_s/2, \omega_s/2)$ bandlimited or at least nearly so.

Anti-Aliasing Filters

Aliasing can occur when a signal with energy at frequencies other that $(-B, B)$ is sampled at rate $\omega_s < 2B$. Thus, when sampling below the Nyquist frequency, it is desirable to remove as much signal energy outside the frequency range $(-B, B)$ as possible while keeping as much signal energy in the frequency range $(-B, B)$ as possible. This suggests that the ideal lowpass filter with cutoff frequency $\omega_s/2$ would be the optimal anti-aliasing filter to apply before sampling. While this is true, the ideal lowpass filter can only be approximated in real situations.

In order to demonstrate the importance of anti-aliasing filters, consider the calculation of the error energy between the original signal and its Whittaker-Shannon reconstruction from its samples taken with and without the use of an anti-aliasing filter. Let x be the original signal and $y = Gx$ be the anti-alias filtered signal where G is the ideal lowpass filter with cutoff frequency $\omega_s/2$. It is easy to show that the reconstructed spectrum using no anti-aliasing filter is given by

$$\widetilde{X}(\omega) = \begin{cases} T_s X_s\left(T_s \omega\right) & |\omega| < \omega_s/2 \\ 0 & \text{otherwise} \end{cases} = \begin{cases} \sum_{k=-\infty}^{\infty} X\left(\omega - k\omega_s\right) & |\omega| < \omega_s/2 \\ 0 & \text{otherwise} \end{cases} .$$

Thus, the reconstruction error spectrum for this case is

$$(X - \widetilde{X})(\omega) = \begin{cases} -\sum_{k=1}^{\infty} \left(X\left(\omega + k\omega_s\right) + X\left(\omega - k\omega_s\right)\right) & |\omega| < \omega_s/2 \\ X(\omega) & \text{otherwise} \end{cases} .$$

Similarly, the reconstructed spectrum using the ideal lowpass anti-aliasing filter is given by

$$\widetilde{Y}(\omega) = Y(\omega) = \begin{cases} X(\omega) & |\omega| < \omega_s/2 \\ 0 & \text{otherwise} \end{cases} .$$

Thus, the reconstruction error spectrum for this case is

$$(X - \widetilde{Y})(\omega) = \begin{cases} 0 & |\omega| < \omega_s/2 \\ X(\omega) & \text{otherwise} \end{cases} .$$

Hence, by Parseval's theorem, it follows that $\|x - \tilde{y}\| \leq \|x - \tilde{x}\|$. Also note that the spectrum of $\widetilde{Y}$ is identical to that of the original signal X at frequencies $\omega \in (-\omega_s/2, \omega_s/2)$. This is graphically shown in Figure 9.2.3.6.1.

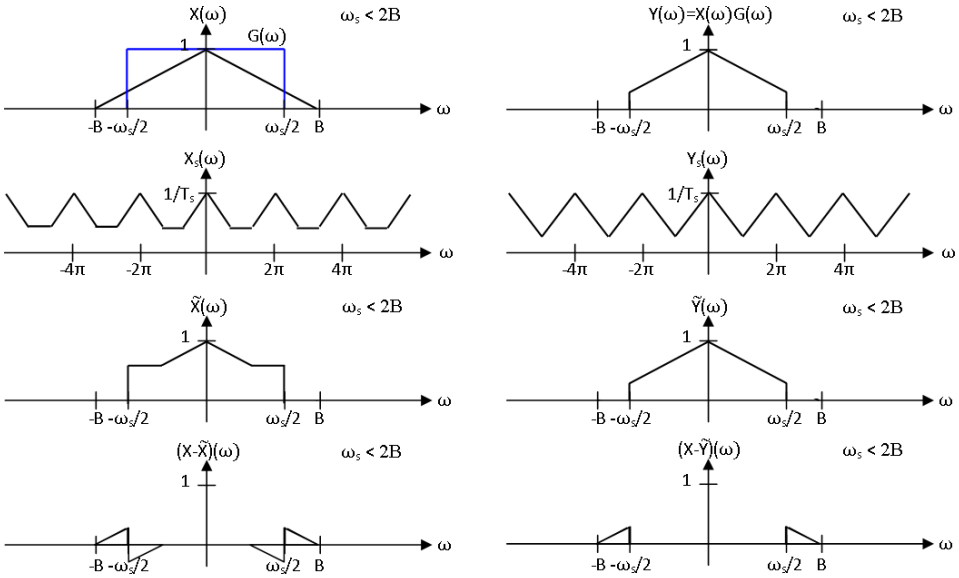


Figure 9.2.3.6.1: The figure above illustrates the use of an anti-aliasing filter to improve the process of sampling and reconstruction when using a sampling frequency below the Nyquist frequency. Notice that when using an ideal lowpass anti-aliasing filter, the reconstructed signal spectrum has the same shape as the original signal spectrum for all frequencies below half the sampling rate. This results in a lower error energy when using the anti-aliasing filter, as can be seen by comparing the error spectra shown.

Anti-Aliasing Filters Summary

As can be seen, anti-aliasing filters ensure that the signal is $(-\omega_s/2, \omega_s/2)$ bandlimited, or at least nearly so. The optimal anti-aliasing filter would be the ideal lowpass filter with cutoff frequency at $\omega_s/2$, which would ensure that the original signal spectrum and the reconstructed signal spectrum are equal on the interval $(-\omega_s/2, \omega_s/2)$. However, the ideal lowpass filter is not possible to implement in practice, and approximations must be accepted instead. Anti-aliasing filters are an important component of systems that implement discrete time processing of continuous time signals, as will be shown in the subsequent module.

This page titled 9.2.3.6: Anti-Aliasing Filters is shared under a CC BY 4.0 license and was authored, remixed, and/or curated by Richard Baraniuk et al. via source content that was edited to the style and standards of the LibreTexts platform.

- **10.6: Anti-Aliasing Filters** by Richard Baraniuk et al. is licensed CC BY 3.0. Original source: https://cnx.org/contents/d2CEAGW515.4:l-e_wSqM24/Signal-Classifications-and-Properties.

9.2.3.7: Discrete Time Processing of Continuous Time Signals

Introduction

Digital computers can process discrete time signals using extremely flexible and powerful algorithms. However, most signals of interest are continuous time signals, which is how data almost always appears in nature. Now that the theory supporting methods for generating a discrete time signal from a continuous time signal through sampling and then perfectly reconstructing the original signal from its samples without error has been discussed, it will be shown how this can be applied to implement continuous time, linear time invariant systems using discrete time, linear time invariant systems. This is of key importance to many modern technologies as it allows the power of digital computing to be leveraged for processing of analog signals.

Discrete Time Processing of Continuous Time Signals

Process Structure

With the aim of processing continuous time signals using a discrete time system, we will now examine one of the most common structures of digital signal processing technologies. As an overview of the approach taken, the original continuous time signal x is sampled to a discrete time signal x_s in such a way that the periods of the samples spectrum X_s is as close as possible in shape to the spectrum of X. Then a discrete time, linear time invariant filter H_2 is applied, which modifies the shape of the samples spectrum X_s but cannot increase the bandlimit of X_s, to produce another signal y_s. This is reconstructed with a suitable reconstruction filter to produce a continuous time output signal y, thus effectively implementing some continuous time system H_1. This process is illustrated in Figure 9.2.3.7.1, and the spectra are shown for a specific case in Figure 9.2.3.7.2.

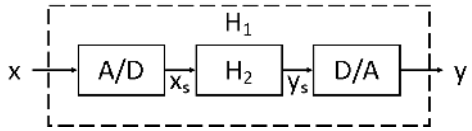


Figure 9.2.3.7.1: A block diagram for processing of continuous time signals using discrete time systems is shown.

Further discussion about each of these steps is necessary, and we will begin by discussing the analog to digital converter, often denoted by ADC or A/D. It is clear that in order to process a continuous time signal using discrete time techniques, we must sample the signal as an initial step. This is essentially the purpose of the ADC, although there are practical issues that which will be discussed later. An ADC takes a continuous time analog signal as input and produces a discrete time digital signal as output, with the ideal infinite precision case corresponding to sampling. As stated by the Nyquist-Shannon Sampling theorem, in order to retain all information about the original signal, we usually wish sample above the Nyquist frequency $\omega_s \geq 2B$ where the original signal is bandlimited to $(-B, B)$. When it is not possible to guarantee this condition, an anti-aliasing filter should be used.

The discrete time filter is where the intentional modifications to the signal information occur. This is commonly done in digital computer software after the signal has been sampled by a hardware ADC and before it is used by a hardware DAC to construct the output. This allows the above setup to be quite flexible in the filter that it implements. If sampling above the Nyquist frequency the. Any modifications that the discrete filter makes to this shape can be passed on to a continuous time signal assuming perfect reconstruction. Consequently, the process described will implement a continuous time, linear time invariant filter. This will be explained in more mathematical detail in the subsequent section. As usual, there are, of course, practical limitations that will be discussed later.

Finally, we will discuss the digital to analog converter, often denoted by DAC or D/A. Since continuous time filters have continuous time inputs and continuous time outputs, we must construct a continuous time signal from our filtered discrete time signal. Assuming that we have sampled a bandlimited at a sufficiently high rate, in the ideal case this would be done using perfect reconstruction through the Whittaker-Shannon interpolation formula. However, there are, once again, practical issues that prevent this from happening that will be discussed later.

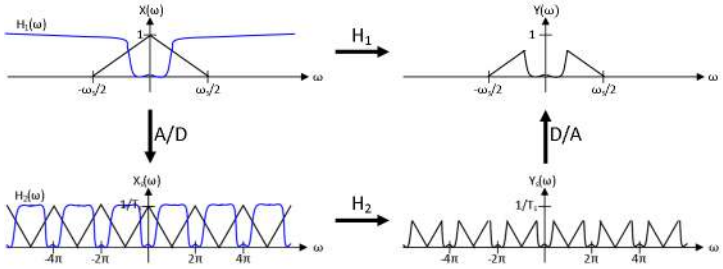


Figure 9.2.3.7.2: Spectra are shown in black for each step in implementing a continuous time filter using a discrete time filter for a specific signal. The filter frequency responses are shown in blue, and both are meant to have maximum value 1 in spite of the vertical scale that is meant only for the signal spectra. Ideal ADCs and DACs are assumed.

Discrete Time Filter

With some initial discussion of the process illustrated in Figure 9.2.3.7.1 complete, the relationship between the continuous time, linear time invariant filter H_1 and the discrete time, linear time invariant filter H_2 can be explored. We will assume the use of ideal, infinite precision ADCs and DACs that perform sampling and perfect reconstruction, respectively, using a sampling rate $\omega_s = 2\pi/T_s \geq 2B$ where the input signal x is bandlimited to $(-B, B)$. Note that these arguments fail if this condition is not met and aliasing occurs. In that case, pre-application of an anti-aliasing filter is necessary for these arguments to hold.

Recall that we have already calculated the spectrum X_s of the samples x_s given an input x with spectrum X as

$$X_s(\omega) = \frac{1}{T_s} \sum_{k=-\infty}^{\infty} X\left(\frac{\omega - 2\pi k}{T_s}\right).$$

Likewise, the spectrum Y_s of the samples y_s given an output y with spectrum Y is

$$Y_s(\omega) = \frac{1}{T_s} \sum_{k=-\infty}^{\infty} Y\left(\frac{\omega - 2\pi k}{T_s}\right).$$

From the knowledge that $y_s = (H_1 x)_s = H_2(x_s)$, it follows that

$$\sum_{k=-\infty}^{\infty} H_1\left(\frac{\omega - 2\pi k}{T_s}\right) X\left(\frac{\omega - 2\pi k}{T_s}\right) = H_2(\omega) \sum_{k=-\infty}^{\infty} X\left(\frac{\omega - 2\pi k}{T_s}\right).$$

Because X is bandlimited to $(-\pi/T_s, \pi/T_s)$, we may conclude that

$$H_2(\omega) = \sum_{k=-\infty}^{\infty} H_1\left(\frac{\omega - 2\pi k}{T_s}\right) (u(\omega - (2k-1)\pi) - u(\omega - (2k+1)\pi)).$$

More simply stated, H_2 is 2π periodic and $H_2(\omega) = H_1(\omega/T_s)$ for $\omega \in [-\pi, \pi)$.

Given a specific continuous time, linear time invariant filter H_1, the above equation solves the system design problem provided we know how to implement H_2. The filter H_2 must be chosen such that it has a frequency response where each period has the same shape as the frequency response of H_1 on $(-\pi/T_s, \pi/T_s)$. This is illustrated in the frequency responses shown in Figure 9.2.3.7.2.

We might also want to consider the system analysis problem in which a specific discrete time, linear time invariant filter H_2 is given, and we wish to describe the filter H_1. There are many such filters, but we can describe their frequency responses on $(-\pi/T_s, \pi/T_s)$ using the above equation. Isolating one period of $H_2(\omega)$ yields the conclusion that $H_1(\omega) = H_2(\omega T_s)$ for $\omega \in (-\pi/T_s, \pi/T_s)$. Because x was assumed to be bandlimited to $(-\pi/T_s, \pi/T_s)$, the value of the frequency response elsewhere is irrelevant.

Practical Considerations

As mentioned before, there are several practical considerations that need to be addressed at each stage of the process shown in Figure 9.2.3.7.1. Some of these will be briefly addressed here, and a more complete model of how discrete time processing of continuous time signals appears in Figure 9.2.3.7.3.

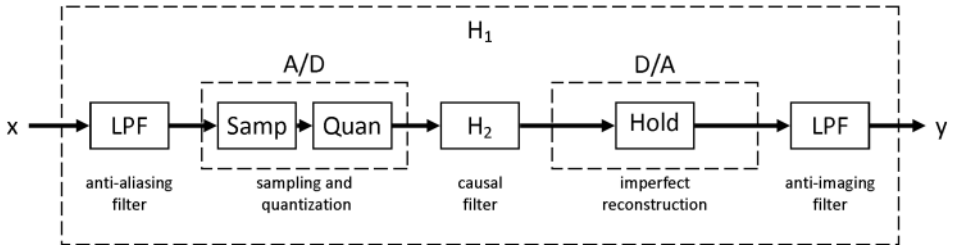


Figure 9.2.3.7.3: A more complete model of how discrete time processing of continuous time signals is implemented in practice. Notice the addition of anti-aliasing and anti-imaging filters to promote input and output bandlimitedness. The ADC is shown to perform sampling with quantization. The digital filter is further specified to be causal. The DAC is shown to perform imperfect reconstruction, a zero order hold in this case.

Anti-Aliasing Filter

In reality, we cannot typically guarantee that the input signal will have a specific bandlimit, and sufficiently high sampling rates cannot necessarily be produced. Since it is imperative that the higher frequency components not be allowed to masquerade as lower frequency components through aliasing, anti-aliasing filters with cutoff frequency less than or equal to $\omega_s/2$ must be used before the signal is fed into the ADC. The block diagram in Figure 9.2.3.7.3 reflects this addition.

As described in the previous section, an ideal lowpass filter removing all energy at frequencies above $\omega_s/2$ would be optimal. Of course, this is not achievable, so approximations of the ideal lowpass filter with low gain above $\omega_s/2$ must be accepted. This means that some aliasing is inevitable, but it can be reduced to a mostly insignificant level.

Signal Quantization

In our preceding discussion of discrete time processing of continuous time signals, we had assumed an ideal case in which the ADC performs sampling exactly. However, while an ADC does convert a continuous time signal to a discrete time signal, it also must convert analog values to digital values for use in a digital logic device, a phenomenon called quantization. The ADC subsystem of the block diagram in Figure 9.2.3.7.3 reflects this addition.

The data obtained by the ADC must be stored in finitely many bits inside a digital logic device. Thus, there are only finitely many values that a digital sample can take, specifically $2N$ where N is the number of bits, while there are uncountably many values an analog sample can take. Hence something must be lost in the quantization process. The result is that quantization limits both the range and precision of the output of the ADC. Both are finite, and improving one at constant number of bits requires sacrificing quality in the other.

Filter Implementability

In real world circumstances, if the input signal is a function of time, the future values of the signal cannot be used to calculate the output. Thus, the digital filter H_2 and the overall system H_1 must be causal. The filter annotation in Figure 9.2.3.7.3 reflects this addition. If the desired system is not causal but has impulse response equal to zero before some time t_0, a delay can be introduced to make it causal. However, if this delay is excessive or the impulse response has infinite length, a windowing scheme becomes necessary in order to practically solve the problem. Multiplying by a window to decrease the length of the impulse response can reduce the necessary delay and decrease computational requirements.

Take, for instance the case of the ideal lowpass filter. It is acausal and infinite in length in both directions. Thus, we must satisfy ourselves with an approximation. One might suggest that these approximations could be achieved by truncating the sinc impulse response of the lowpass filter at one of its zeros, effectively windowing it with a rectangular pulse. However, doing so would produce poor results in the frequency domain as the resulting convolution would significantly spread the signal energy. Other windowing functions, of which there are many, spread the signal less in the frequency domain and are thus much more useful for producing these approximations.

Anti-Imaging Filter

In our preceding discussion of discrete time processing of continuous time signals, we had assumed an ideal case in which the DAC performs perfect reconstruction. However, when considering practical matters, it is important to remember that the sinc function,

which is used for Whittaker-Shannon interpolation, is infinite in length and acausal. Hence, it would be impossible for an DAC to implement perfect reconstruction.

Instead, the DAC implements a causal zero order hold or other simple reconstruction scheme with respect to the sampling rate ω_s used by the ADC. However, doing so will result in a function that is not bandlimited to $(-\omega_s/2, \omega_s/2)$. Therefore, an additional lowpass filter, called an anti-imaging filter, must be applied to the output. The process illustrated in Figure 9.2.3.7.3 reflects these additions. The anti-imaging filter attempts to bandlimit the signal to $(-\omega_s/2, \omega_s/2)$, so an ideal lowpass filter would be optimal. However, as has already been stated, this is not possible. Therefore, approximations of the ideal lowpass filter with low gain above $\omega_s/2$ must be accepted. The anti-imaging filter typically has the same characteristics as the anti-aliasing filter.

Discrete Time Processing of Continuous Time Signals Summary

As has been show, the sampling and reconstruction can be used to implement continuous time systems using discrete time systems, which is very powerful due to the versatility, flexibility, and speed of digital computers. However, there are a large number of practical considerations that must be taken into account when attempting to accomplish this, including quantization noise and anti-aliasing in the analog to digital converter, filter implementability in the discrete time filter, and reconstruction windowing and associated issues in the digital to analog converter. Many modern technologies address these issues and make use of this process.

This page titled 9.2.3.7: Discrete Time Processing of Continuous Time Signals is shared under a CC BY 4.0 license and was authored, remixed, and/or curated by Richard Baraniuk et al. via source content that was edited to the style and standards of the LibreTexts platform.

- **10.7: Discrete Time Processing of Continuous Time Signals** by Richard Baraniuk et al. is licensed CC BY 3.0. Original source: https://cnx.org/contents/d2CEAGW515.4:l-e_wSqM24/Signal-Classifications-and-Properties.

SECTION OVERVIEW

9.3: Chapter 3 Supplementary Reading Materials

Resources developed by authors external to NIU with full prose and background materials:

Topic hierarchy

9.3.1: Importance of Statistics and Experimental Measurements

9.3.2: Total Uncertainty, Bias Error, and Propagation of Error

9.3: Chapter 3 Supplementary Reading Materials is shared under a not declared license and was authored, remixed, and/or curated by LibreTexts.

9.3.1: Importance of Statistics and Experimental Measurements

Here you can find pre-recorded lecture videos that cover this topic here: https://youtube.com/playlist?list=PL...EANc_k_3z5Y5n8

1.1 Performing Error and Data Analysis in the Lab:

Here are some common comments that I will typically overhear when students are in lab and talking about data analysis:

1. **How good is your data?**
2. **How well does your experimental data fit the theory?**
3. **What is the R^2 value?**
4. **What is the percent error?**

While well intentioned, these are **not the questions an experimentalist should ask**. And before any data set can be utilized for an application, the **quality** of the data must first be established.

For example in the first question is the student assessing the **accuracy? Precision? How well it fits with theory? We must be specific and careful with our language.** In terms of the second question, data is **not necessarily good** just because it agrees with theory. The quality of the data must be assessed before any conclusions can be drawn. **What is the reason the theory and experiments disagree? Is that theory applicable in this situation?** We must ask these questions!

You all probably have experience with R^2 as a **measure of the goodness of a fit,** but in this course we will do a **much more thorough error analysis.** We will go much deeper than R^2 so we will not stop there.

Finally, **never ever ever ever ever include percent error in any lab analysis** that we perform in this class. It is a meaningless error analysis. We never know the actual value or true value of a measurement, so we will not perform this error analysis.

When designing an experiment it is crucial to do the following:

• Set clearly defined goals and ideally create graphs/figures for study

• Experimental plan, apparatus, and DAQ for collecting data

• Perform appropriate statistical analysis to effectively extract critical information from raw data

• Interpret results and draw conclusions that are statistically significant

Additionally, when we speak of quality what we mean is to find the **actual or true value** of the physical quantity being measured. And often this is different than the **measured value** of the physical quantity. The difference between the **measured value** and the **actual or true value** of a measurement is the **error**.

Here we encounter a problem as we cannot calculate the error exactly unless we know the true value of the quantity being measured. Instead we **estimate the bounds or the likelihood that the error exceeds a specific value**. For example say that 95% (19 out of 20) of the yield strength measurements will have an error less than 1 MPa. **Thus the reading has an uncertainty of 1 MPa at confidence level of 95%**. We will calculate confidence levels or confidence intervals when quantifying error.

As experimentalists we often work with large data sets with a large number of measurements. The **error or uncertainty can be estimated with statistical tools when a large number of measurements are taken**. Later on in the class we will learn how to use some software tools like **Matlab, R, Python, and/or Mathematica** to perform error analysis. Much more on this later, but let's get to some basics of experimentation and measurements.

1.2 Measurements:

A **measurement** is a **quantitative** comparison between a predefined **standard** and a **measurand**.

What are some examples of different types of measurements we can make?

• Temperature

• Length

• Time

• Stress

• Strain

• Viscosity

• Force

• Torque

• ....

Measurand: the particular physical parameter begin observed and quantified, that is, the **input quantity** of the **measuring process.**

Standard: has a defined relationship to a unit of measurement for a given measurand. Typically a standard will be recognized by the NIST (National Institute of Standards and Technology).

What are some examples of standards?

• Tape measure

• Balance

• Speed of light

• Stopwatch

• Extensometer

• ...

2.1.1 What is the significance of measurements—why do we experimentally measure parameters?

• Obtain quantitative information on the actual state of physical variables and processes (T, P, V, etc.); this is critical especially for thermodynamics and finding thermodynamic equilibrium

• Vital for control processes which require measured discrepancy between actual and desired output (i.e. temperature control on sheet annealing line)

• Daily operations (temperature controls, heating and cooling, tire gauge, oil gauge, etc.)

• Establishing costs, Ti $140/lb precision is important when dealing with tons of Ti

• Temperature in bioreactor can influence cell growth

• Thermal conductivity, acceleration, material properties

2.2 Uncertainty in Measurements: Measurements Are Useless without Uncertainty Bounds or Error Bars

Measurements are only useful to us as engineers if we can quantify the **uncertainty** of a measurement. **No measurement is meaningful** or useful if they do not have an **associated error bar**. In this class and moving forward in your career as an engineer, be sure to place error bars on all your reported values. Error bars are critical because no measurement is perfect. If you measure the length of a tensile specimen with a micrometer, your measurement is likely to be different than that of your colleague. Thus, there will be some **uncertainty** in any measurement that one makes. In this section of this course we will develop a comprehensive framework for evaluating uncertainty and placing **tolerances and confidence intervals** on measured values. Much more on this later.

2.3 Using Correct Terminology When Assessing Measurement Quality/Apparatus Performance

• **Accuracy:** Difference between measured and true values. Manufacturer will specify maximum error but not confidence intervals typically.

• **Precision:** Difference between reported values during repeated measurements of same quantity.

• **Resolution:** Smallest increment of change in the measured value that can be determined by the readout scale. Often the same as precision.

• **Sensitivity:** The change of an instrument's output per unit change in the measured quantity. Higher sensitivity often indicates finer resolution, better precision, and higher accuracy.

2.4 Error and Uncertainty:

Error is the difference between the *measured* and *true* value of the quantity being measured. Unfortunately, the true value can never really be known so instead we quantify the **uncertainty** of a measurement at specific levels of confidence. Error can be categorized into two basic categories: **bias** or **precision** error/uncertainty. We will cover uncertainty and error analysis much more in-depth in the next several lectures, but never use percent error in this course.

We can begin with our discussion of error by defining the following equation:

$$Error = x_m - x_{true} \tag{9.3.1.1}$$

where x_m is the measured value and x_{true} is the true value.

When designing an experimental system or executing an experiment, the experimentalist's primary objective should be to minimize the error. After the experiment is completed we need to estimate a bound on the error with a degree of certainty or a confidence interval (i.e., 90%, 95%, 99%). Hopefully you have encountered confidence intervals before, but if not, no worries; we will discuss this concept in depth.

There are many different sources of inaccuracy or error, however most errors can be classified as:

(1) **Precision**

(2) **Bias**

(3) **Illegitimate**

(4) **Errors that are sometimes bias or sometimes precision errors**

In this course we will spend a considerable amount of time discussing **1. Precision** and **2. Bias** error in great detail. But we will discuss all these sources of error and how to account for them when designing experiments or presenting your results. Let's start with **Precision Error.**

2.4.1 Precision or Random Errors

I really do not like the term *random error*, but you will occasionally see that term in literature; in this course we will use the term *precision* primarily.

Precision errors are **different for each successive measurement**. Thus if you create a histogram, which shows the probability of encountering or measuring a specific measurand, the width of your distribution (hopefully it is Gaussian) will be due to precision error if that is the only source of error in your experiment. From this distribution, **statistical analysis can be utilized to estimate the size of the precision error.** Typically bias and precision errors occur simultaneously and contribute to the **total error.** More on this fun calculation later—hope you remember Taylor expansions....

Why Do We Have Precision Error?

(1) Disturbances to the equipment

(2) Fluctuating experimental conditions

(3) Insufficient measuring-system sensitivity

(4) Stochastic (Random) nature of measurements even in static conditions

Examples of Precision Error:

• Measuring temperature with thermocouple as temperature fluctuates

• Measure the length of a beam with a micrometer you will get slightly different measurements than classmates

• Changes in temperature conditions can affect swimming of bacteria

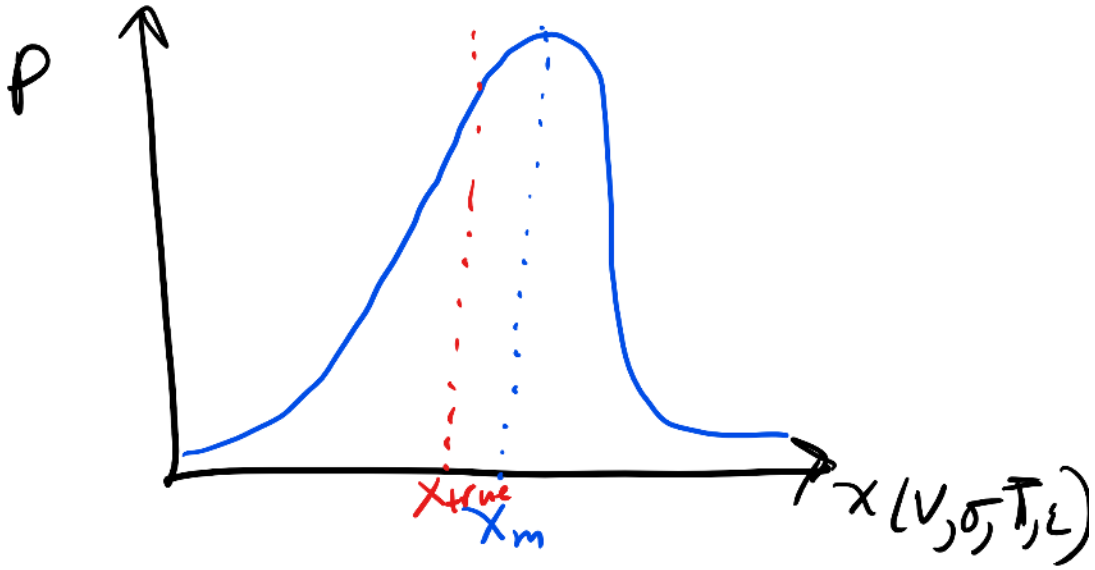


Figure 9.3.1.1: Precision Error Distribution.

2.4.2 Bias or Systematic Errors

I like the terms *bias* and *systematic* much better than *random*. However, in this course we will primarily use the term *bias error*.

Bias errors will occur the same way each time a measurement is made. For example a balance that consistently reads 0.5 mg heavier. In some literature you might see this termed as a fixed offset error. The key thing to note about bias errors is they are fixed and thus **they do not show a distribution**. This is in stark contrast to precision errors which show a distribution, typically Gaussian. Therefore if there is no distribution we cannot apply any **statistical analysis**.

There are multiple types of bias errors:

(1) Calibration errors

• This is the most common error.

• Typically calibration errors are **zero-offset errors**, which cause all reading to be offset by constant amount or *scale errors*.

• For example: a scale is constantly light by 1lb or a clock runs fast by 1 minute.

• Can be reduced via calibration procedures and proving the system via comparison with *known* standard.

(2) Consistently recurring human errors

• For example: experimenter who consistently jumps the gun when recording synchronized readings either time, voltage, etc.

(3) Errors caused by defective equipment

(4) **Loading errors**

• For example: not putting a DSC pan at the center of the DSC will skew temperature measurements.

(5) System resolution limits

• Bias error for a ruler with cm tick marks will be cm, compared to a micrometer with 1μm resolution, that is 4 order of magnitude difference.

3. Illegitimate Error

Illegitimate error can and most likely will occur to you at some point as you perform experiments as an engineer. Illegitimate errors include

• Blunders and mistakes

• Computational or calculation errors

• Analysis errors

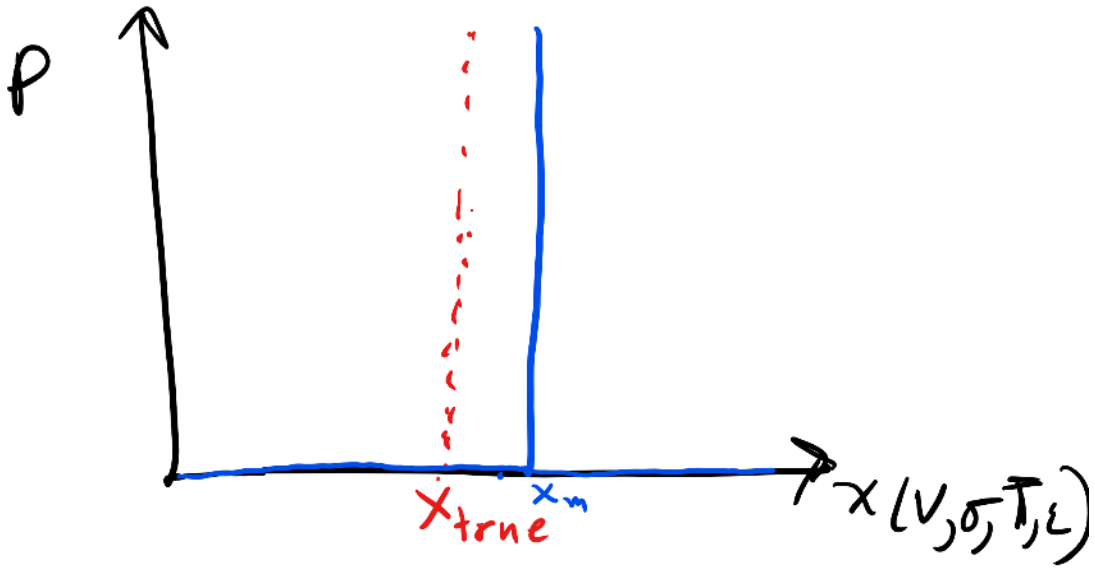


Figure 9.3.1.2: Bias Error Distribution

While these things can occur in the lab, **you must always perform the experiment again**. You cannot simply present your results in either a presentation or a form of technical writing and attribute error to illegitimate error. You must perform the experiment again.

4. Errors That Are Sometimes Bias and Sometimes Precision Error

These are the most difficult sources of error that we must analyze, as it is difficult to determine what type of error analysis must be done and which error has a distribution and which does not.

• Instrument Hysteresis

– Hysteresis error can be seen in stress strain measurements and is path dependent.

• Calibration drift and variation in test or environmental conditions

– Drift occurs when the response varies over time often due to sensitivity to temperature or humidity.

• Variations in procedure or definition among experimenters

Quantifying Total Uncertainty, Comprehensive Error Analysis

Again, typically we are concerned with **bias** and **precision** error when evaluating experimental data. And this data comes from two classes of experiments: i) *single- sample* and ii) *repeated-sample* experiments.

A **sample** refers to an individual measurement of a specific quantity. For example, measuring strain in a material several times under identical loading is a repeated-sample experiment. A single measurement of strain is a single-sample experiment, and we cannot extract the distribution of precision error in this case. In this course and in your career as an engineer you will most likely be taking multiple measurements, that is, repeating sample experiments. Single sample experiments are more uncommon and typically occur when the measurement you are making is extremely difficult and multiple experiments are not realistic to conduct, typically due to economic or practical reasons.

The total uncertainty U_x in a measurement of *x* is

$$U_x = \sqrt{B_x^2 + P_x^2} \tag{9.3.1.2}$$

where B_x if the bias uncertainty in a measurement of *x* and P_x is the precision uncertainty. Note that *x* is just a variable and we can talk about uncertainty in measuring volume (V), pressure (P), stress (σ), etc.

This is a largely empirically derived formula, however it does depend on the **assumption that the bias and precision uncertainty are independent sources of error.** Also the bias and precision uncertainty should have the **same confidence level** when calculating the **total uncertainty.** We will come back to this a bit later in excruciating detail for this calculation, so please be patient.

Minimizing Error in Designing Experiments

The best time to **minimize experimental error** is in the **design stage**. You should **perform a single-sample uncertainty analysis** of the proposed experimental arrangement **prior to building the apparatus**. This allows you to make a decision on whether the expected uncertainty is acceptable and to identify the sources of uncertainty. So when designing an experiment:

• **Avoid approaches** that require two large numbers to be measured to determine the small difference between them.

• **Design experiments or sensors that amplify the signal strength** to improve sensitivity.

• Build *null designs* in which the output is measured as a change from zero rather than as a change in nonzero value.

• Avoid experiments with large *correction factors* must be applied as part of the data-reduction procedure.

• Attempt to **minimize the influence of the measuring system on the measured variable.**

• **Calibrate** the entire system to minimize calibration-related bias errors.

Once we have established the error and uncertainty with a measurement, we also have to **report these results** and **explain our analysis**. **Technical writing** is a very distinct class of writing with many different formats and audiences. It can be difficult to strike the right **tone** and **level of technical information** that must be provided at times. In this class we will focus on **Laboratory Reports,** but as you can see below there are many different form of technical writing.

Forms of Technical Writing:

• **Executive summary**

– Audience is a supervisor. Summary should be concise, usually only a few paragraphs, and highlight key aspects of work in first paragraph.

– Consequences can be drastic as in the case of the Challenger Shuttle tragedy.

• **Laboratory note or technical memo**

– Limited audience, immediate supervisor

– Sufficient information should be included so the experimentalist can reconstruct the experiment

• **Progress report**

– Audience is supervisor of sponsoring agency

– Can vary in terms of length, structure, timelines, etc.

• **Full technical report**

– Wider audience, typically not people closely associated with the work

– Relates all facts pertinent to the project

– Contains enough detail so another professional can repeat experiments

• **Full technical report**

– Wider audience, typically not people closely associated with the work

– Relates all facts pertinent to project

– Contains enough detail so another professional can repeat experiments

Line Fitting: Identify Outliers and Physical Interpretation

When you are fitting a curve or a line to a data set **one must be careful to select a fit that is appropriate for the experiment being conducted**. Often researchers are too consumed with obtaining a **high R^2 value**. While the goodness of a fit is important, you must make sure that the equation that you are **fitting matches with the physical experiment that you are conducting**. For example, if you are measuring how an electrochemical cell voltage changes as a function of temperature, you must use the Nernst Equation. Looking at the data, perhaps a 12th degree polynomial fit will give you a better R^2 value but that equation has no physical meaning in terms of your analysis. So always keep that in mind.

When identifying outliers and a particular point is in doubt, temporarily exclude the data point and fit a new line through the remaining data. **If the fit is better, the point can possibly be dropped; however, it could indicate that the relationship is not linear. It may indicate another scaling regime or a transition in behavior.**

We can also apply this same error analysis procedure and to optimize your experimental apparatus and minimize error/total uncertainty. One can identify what component or part of the measuring process is generating the most error, and then one can redesign the apparatus to minimize that error.

9.3.1: Importance of Statistics and Experimental Measurements is shared under a CC BY-NC 4.0 license and was authored, remixed, and/or curated by LibreTexts.

9.3.2: Total Uncertainty, Bias Error, and Propagation of Error

Here you can find a series of pre-recorded lecture videos that cover this content: https://youtube.com/playlist?list=PLSKNWIzCsmjAmuzoj5TCkHKIAJGx8gWZu&si=wV67lD1o8Gp2ihNH

Bias and Single-Sample Uncertainty

Precision error can be calculated using **statistical methods** for repeat-sampled data; however, **bias error cannot** be uncovered using **statistical methods**. The only direct method is via comparison with measurements made using another more accurate apparatus. However, this is practically impossible. It is much too costly, time consuming, and I am going to show you another method to estimate the bias error right now. Instead we will utilize our **experience and knowledge of the device** and make an estimate on the **sizes of bias errors**. Then we will obtain an estimate of the precision and accuracy of our experimental apparatus and determine if we must redesign our apparatus to measure a particular quantity of interest.

Bias Error: Measuring Volume in a cylinder

What measurements do we need to measure the volume of a cylinder? Do you remember geometry?

The equation for the volume of a cylinder is

$$V = \pi r^2 l \tag{9.3.2.1}$$

where r is the radius of the cylinder and l is the length of the cylinder. When thinking about the uncertainty of these measurements, there are many things that may come to mind but looking at the equation above, we want to identify the number of independent variables as that is where uncertainty will originate from. So in the above equation we will have uncertain in r and l. Now of course when we calculate volume, we will have to measure r and l and there may be precision error in these measurements due to temperature fluctuation and thermal expansion, and we know how to quantify precision error. However, there will also be **bias error** present as well. Bias error will manifest itself in the type of device utilized to measure these quantities. For example, if you are using a yard stick with 1 yard intervals, that will have a much larger uncertainty than a micrometer, or some type of quantum laser technique...we unfortunately do not have that available in the lab.

Propagating Uncertainty

Usually we **measure several quantities** and use those measurements to **calculate a desired quantity** just like we saw in volume. Each measurement has some **associated uncertainty,** which will create an **uncertainty in the calculated result**. **How do we quantify that uncertainty?**

We **propagate the uncertainty**!

We will utilize a statistical theorem to propagate uncertainty in independent variables. Let's consider a linear function y of several **several independent variables** x_i with standard deviation σ_i.

The standard deviation of this equation would be

$$\sigma_y = \sqrt{\left(\frac{\partial y}{\partial x_1}\sigma_1\right)^2 + \left(\frac{\partial y}{\partial x_2}\sigma_2\right)^2 + ... + \left(\frac{\partial y}{\partial x_n}\sigma_n\right)^2} \tag{9.3.2.2}$$

Each variable x_i will have some associated uncertainty u_i which will then contribute to the total bias uncertainty in y which we will call u_y. In order to estimate u_y we will make a key assumption that each uncertainty is relatively small (reasonable) so that we can perform our favorite operation! A first order Taylor expansion of $y(x_i)$

$$y(x_1 + u_1, x_2 + u_2, ..., x_n + u_n) \approx y(x_1, x_2, ..., x_n) + \frac{\partial y}{\partial x_1}u_1 + \frac{\partial y}{\partial x_2}u_2 + ... + \frac{\partial y}{\partial x_n}u_n \tag{9.3.2.3}$$

Thus y is a linear function of independent variables and the **uncertainty will behave like standard deviations**:

$$u_y = \sqrt{\left(\frac{\partial y}{\partial x_1}u_1\right)^2 + \left(\frac{\partial y}{\partial x_2}u_2\right)^2 + ... + \left(\frac{\partial y}{\partial x_n}u_n\right)^2} \tag{9.3.2.4}$$

The key thing for this theorem to hold true and so that we can utilize this equation is that all uncertainties must be **independent of each other** and have the same odds.

Let's do some examples to make this clear.

Volume of a Cylinder

A tube of circular cross section has a nominal length of 50 m ± 0.5 m and a radius of 15m ± 0.1m. Determine the uncertainty in the volume.

So when we first approach this problem, we must identify if there is precision error, bias error, or both. When we read this problem, if we want to identify precision error do we have any information about the number of samples? A list of measurements? Or anything else?

Well, the answer in this problem is "no," so we have no precision error contribution, something to note when you are reporting your results in a lab report.

For identifying the bias error, do we know the tool we were using to measure the independent variables, manufacturer's information, or if we are really lucky, we are just given the uncertainties.

Well, in this problem, we are really lucky—we are just given the uncertainties so we just have to deal with bias error in calculating the total uncertainty!

The next step we should do is to identify what we need to measure, the equation required, and the number of independent variables.

In this problem

$$V = \pi r^2 l \tag{9.3.2.5}$$

So we have a function V (r,l) that is a function of two independent variables r and l both of which will have some uncertainty associated with the measurement as seen below.

Let's proceed with our solution to calculate the bias error in V, B_V we have to first take our differential

$$B_V = \sqrt{\left(\frac{\partial V}{\partial r} u_r\right)^2 + \left(\frac{\partial V}{\partial l} u_l\right)^2} \tag{9.3.2.6}$$

where we know from our problem statement that u_r = 0.1m and that u_l = 0.5m.

And we also know that

$$\frac{\partial V}{\partial r} = 2\pi r l \tag{9.3.2.7}$$

$$\frac{\partial V}{\partial l} = \pi r^2 \tag{9.3.2.8}$$

We can then calculate B_V as seen below by plugging in the **nominal or mean** values for r and l as well as the uncertainties associated with them.

So we find

$$B_V = 589.05 m^3 \tag{9.3.2.9}$$

and since in this problem we have no precision uncertainty, we find that the total uncertainty in volume, U_V is just B_V as seen below.

$$U_V = \sqrt{B_V^2 + P_r^2 + P_l^2} = B_V \tag{9.3.2.10}$$

where P_r = P_l = 0 since there is no precision error in this experiment.

It is critical when performing this analysis to keep your units consistent; this is where most people will make mistakes—always always always convert to SI units.

Now, often it is more useful to express your answers in terms of a fraction or a percent. We can do this simply by dividing our differentials and B_V equation above and rewrite it as

$$\frac{B_V}{V} = \sqrt{\left(\frac{\frac{\partial V}{\partial r}u_r}{V}\right)^2 + \left(\frac{\frac{\partial V}{\partial l}u_l}{V}\right)^2} \tag{9.3.2.11}$$

We can clearly see here that

$$\frac{\frac{\partial V}{\partial r}u_r}{V} = \frac{2u_r}{r} \tag{9.3.2.12}$$

and that

$$\frac{\frac{\partial V}{\partial l}u_l}{V} = \frac{u_l}{l} \tag{9.3.2.13}$$

Plugging this back into our equation we will find that now

$$B_V = 0.01667 \tag{9.3.2.14}$$

This is a unitless fraction and you should check that this is the case for yourself. If we want to convert this fraction to a percent, we must simply multiply by 100 in this problem. The nominal volume is 35343 m^3 so to double check our answer on the bias uncertainty, we can find that if we multiply our fraction by the nominal volume we indeed get that the uncertainty is 589.05 m^3. This is a nice way to double check your answer as the problems get more complex from here.

Let's do another example problem focusing on material mechanics.

Uniaxial Tension Test

I am trying to measure the stress in a rectangular block of aluminum under uniaxial tension. I repeated the experiment 14 times and find the force to exhibit a mean of 48.9N and a standard deviation of 3.24. I also measured the length of the block and width 14 times and found a mean width of 0.1m and a standard deviation of 0.01m, the length had a mean of 0.5m and a standard deviation of 0.02m. I measured the length and width with a micrometer and the force applied is ± 1N. What is the total uncertainty in stress?

Let's break down this problem: will we have precision error, bias error, or both?

We will have both precision error and bias error in this problem. Precision error comes from the multiple measurements made in this problem and we are also given standard deviations. We will also have bias error because we are given an uncertainty in force plus the tool used to measure length and width.

What is the equation we are working with?

Well, since we are dealing with uniaxial tension, we can use Hooke's law. We will talk about tensor notation much much more in depth soon.

$$\sigma = \frac{F}{A} = \frac{F}{l \cdot w} \tag{9.3.2.15}$$

We have a function $\sigma(F,l,w)$ that is a function of 3 independent variables, so we will have to take 3 partials.

At this point we have some options in terms of how to approach this problem. You can go directly to the partials, but when you have precision error, I would suggest calculating precision error first, expressing this uncertainty in percent, calculating bias error in percent, and then calculating total uncertainty in percent.

So to start with precision we have precision error in all 3 independent variables, so we can calculate the precision error as we have done previously

$$P_F = t_{0.025,13}\frac{S_F}{\sqrt{n}} \tag{9.3.2.16}$$

$$P_l = t_{0.025,13}\frac{S_l}{\sqrt{n}} \tag{9.3.2.17}$$

$$P_w = t_{0.025,13}\frac{S_w}{\sqrt{n}} \tag{9.3.2.18}$$

But as I mentioned we want to work with percentages, so we need to divide these values by the nominal value $\frac{P_F}{F}$, $\frac{P_l}{l}$, $\frac{P_w}{w}$.

So those values can be expressed as a dimensionless fraction as

$$\frac{P_F}{F} = 0.04 \tag{9.3.2.19}$$

$$\frac{P_l}{l} = 0.023 \tag{9.3.2.20}$$

$$\frac{P_w}{w} = 0.06 \tag{9.3.2.21}$$

Now we can move on to our partial equations. To calculate the bias error, B_σ we must use the following equation:

$$B_\sigma = \sqrt{\left(\frac{\partial\sigma}{\partial F}u_F\right)^2 + \left(\frac{\partial\sigma}{\partial l}u_l\right)^2 + \left(\frac{\partial\sigma}{\partial w}u_w\right)^2} \tag{9.3.2.22}$$

Now here you can plug and check and get an uncertainty in units of stress, *Pa*, but I would highly suggest that you convert to dimensionless units and a fraction and you will find your answer looks much nicer as seen below.

$$\frac{B_\sigma}{\sigma} = \sqrt{\left(\frac{u_F}{F}\right)^2 + \left(\frac{u_l}{l}\right)^2) + \left(\frac{u_w}{w}\right)^2} \tag{9.3.2.23}$$

Now you can plug and chug to solve this problem—but wait, I know that *F*,*l*,*w* are the nominal or mean values, but what is the uncertainty? Well, the uncertainty in *F* is given and we know the tool to measure length is a micrometer, so the bias uncertainty for that tool is 1μm—so we have our uncertainties.

Plugging in you will find that

$$\frac{B_\sigma}{\sigma} = \sqrt{\left(\frac{u_F}{F}\right)^2 + \left(\frac{u_l}{l}\right)^2) + \left(\frac{u_w}{w}\right)^2} = 0.02 \tag{9.3.2.24}$$

And now we can calculate the total uncertainty as a percent:

$$U_\sigma = \sqrt{\left(\frac{B_\sigma}{\sigma}\right)^2 + \left(\frac{P_F}{F}\right)^2 + \left(\frac{P_l}{l}\right)^2 + \left(\frac{P_w}{w}\right)^2} \cdot 100 = 7.93\% \tag{9.3.2.25}$$

You will get much more practice with these problems during class, on YouTube, and in Psets and exams, but this is generally how you should approach these problems.

Suggested Procedure for Calculating Total Uncertainty

(1) Determine if problem includes bias error, precision error, or both.

(2) Write out fundamental equation, that is, what you are trying to calculate and identify independent variables,

(3) Write out known quantities, that is, means, uncertainties, standard deviation, number of trials, etc.

(4) Start with precision error first and convert to percentages or fractions.

(5) Take partials and calculate bias error and convert to percentages or fractions.

(6) Combine precision and bias error to calculate total uncertainty.

Minimizing Error in Designing Experiments

The best time to **minimize experimental error** is in the **design stage**. You should **perform a single-sample uncertainty analysis** of the proposed experimental arrangement **prior to building the apparatus**. This allows one to make a decision on whether the

expected uncertainty is acceptable and to identify the sources of uncertainty. So when designing an experiment:

• **Avoid approaches** that require two large numbers to be measured to determine the small difference between them.

• **Design experiments or sensors that amplify the signal strength** to improve sensitivity.

• Build *null designs* in which the output is measured as a change from zero rather than as a change in nonzero value.

• Avoid experiments with large *correction factors* must be applied as part of the data-reduction procedure.

• Attempt to **minimize the influence of the measuring system on the measured variable.**

• **Calibrate** entire system to minimize calibration-related bias errors.

When identifying outliers and a particular point is in doubt, temporarily exclude the data point and fit a new line through the remaining data. **If the fit is better the point can possibly be dropped however it could indicate that the relationship is not linear it may indicate another scaling regime or a transition in behavior.**

9.3.2: Total Uncertainty, Bias Error, and Propagation of Error is shared under a CC BY-NC 4.0 license and was authored, remixed, and/or curated by LibreTexts.

9.4: Chapter 4 Supplementary Reading Materials

9.4: Chapter 4 Supplementary Reading Materials is shared under a not declared license and was authored, remixed, and/or curated by LibreTexts.

SECTION OVERVIEW

9.5: Chapter 5 Supplementary Reading Materials

Resources developed by authors external to NIU with full prose and background materials:

Topic hierarchy

9.5.1: Undamped Free Vibrations

9.5.2: Viscous Damped Free Vibrations

9.5.3: Friction (Coulomb) Damped Free Vibrations

9.5.4: Undamped Harmonic Forced Vibrations

9.5.5: Viscous Damped Harmonic Forced Vibrations

9.5: Chapter 5 Supplementary Reading Materials is shared under a not declared license and was authored, remixed, and/or curated by LibreTexts.

9.5.1: Undamped Free Vibrations

Vibrations occur in systems that attempt to return to their resting or equilibrium state when perturbed, or pushed away from their equilibrium state. The simplest vibrations to analyze are undamped, free vibrations with one degree of freedom.

"Undamped" means that there are no energy losses with movement (whether the losses are intentional, from adding dampers, or unintentional, through drag or friction). An undamped system will vibrate forever without any additional applied forces. A simple pendulum has very low damping, and will swing for a long time before stopping. "Damped" means that there are resistive forces and energy losses with movement that cause the system to stop moving eventually.

"Free" means that, after the initial perturbation, the only forces acting on the system are internal to the system (springs, dampers) and/or gravity. A tuning fork continues to vibrate after the one initial perturbation of being struck. In contrast, "forced" means there is an external, typically periodic, force acting on the system. A jackhammer vibrates due to having a supply of compressed air continually forcing the bit up and down, and it stops vibrating very quickly without that external periodic forcing.

"One degree of freedom" means that we will only consider systems with one mass vibrating along one direction (e.g. use variable x) or about one axis (e.g. use variable θ). Systems having more than one mass or vibrating along or about two or more axes have more than one degree of freedom.

We can derive the equation of the system by setting up a free body diagram. Consider a mass sitting on a frictionless surface, attached to a wall via a spring.

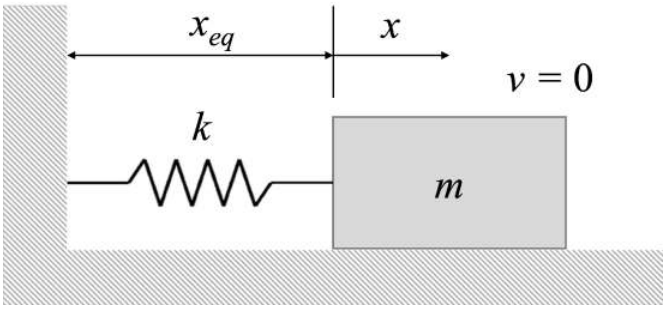


Figure 9.5.1.1: This is a system consisting of a mass attached to the wall via a spring, sitting on a frictionless surface. The system is currently in equilibrium, and the spring is not stretched or compressed.

The system above is in equilibrium. It is at rest, and will stay at rest unless some other force acts on it.

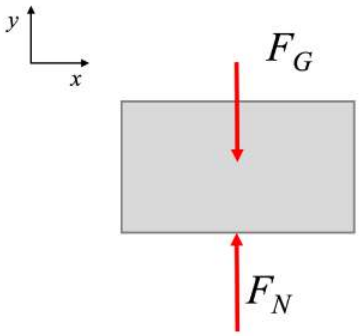


Figure 9.5.1.2: Free body diagram of the system in equilibrium position. Since the spring is at its unstretched length, it does not produce a force.

To start the system vibrating, we need to perturb it. Perturbation is moving the system away from equilibrium by a small amount.

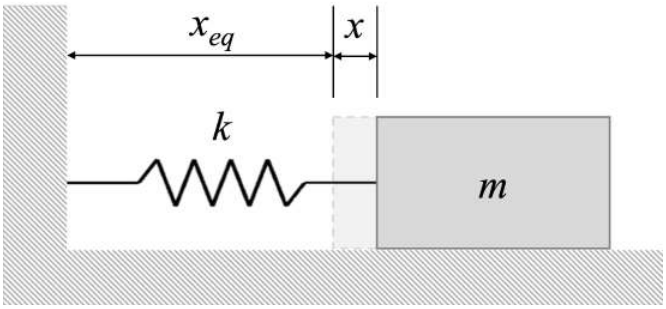


Figure 9.5.1.3: Mass-spring system perturbed by an amount $+x$ from equilibrium. This stretches the spring, causing a spring force which tends to pull the mass back toward equilibrium.

When we perturb this system, we either stretch or compress the spring. This generates a spring force, and the spring force is always in a direction that tends to pull the system back toward equilibrium.

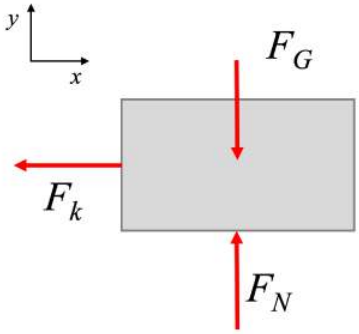


Figure 9.5.1.4: Free body diagram of the mass-spring system perturbed by an amount $+x$ from equilibrium. This stretches the spring, causing a spring force which tends to pull the mass back toward equilibrium.

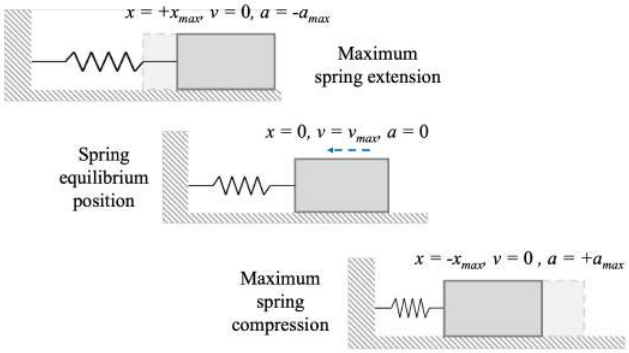


Figure 9.5.1.5: Mass-spring system vibration. After an initial perturbation, the system oscillates between two extreme positions ($-x_{max}$ and x_{max}). The extreme positions are turnaround points where the velocity is zero, but the spring force is at a maximum (maximum stretch/compression) and therefore the acceleration is maximum. As the system passes through the equilibrium position, the spring is no longer stretched (zero force, zero acceleration), but the velocity is at its maximum (this can be determined using conservation of energy) and the inertia of the system carries the mass past the equilibrium position. In the absence of damping or the application of another force, the system will oscillate forever.

We can generate the equation of motion of the system, and determine the specifics of how it will vibrate, by analyzing this perturbed state. Recall that the spring force or moment is:

$$\vec{F}_k = k\vec{x} \tag{9.5.1.1}$$

$$\vec{M}_k = k\vec{\omega} \tag{9.5.1.2}$$

Note that the spring constants in the above equations have different units, depending on whether the spring is linear (Newton/meter) or torsional (Newton-meter/radian), and that θ must be given in radians. The magnitude of the spring force depends on x, the distance perturbed **from the spring's unstretched length** (not necessarily the equilibrium position of the system), and the same is true for the moment of a torsional spring. The spring force or moment is in the direction/orientation opposite that of the displacement. That is, if you pull the mass to the right, the spring force points to the left.

The process for finding the equation of motion of the system is as follows:

1. Sketch the system with a small positive perturbation (x or θ).
2. Draw the free body diagram of the perturbed system. Ensure that the spring force has a direction opposing the perturbation.
3. Find the one equation of motion for the system in the perturbed coordinate using Newton's Second Law. Keep the same positive direction for position, and assign positive acceleration in the same direction.
4. Move all terms of the equation to one side, and check that all terms are positive. If all terms are not positive, there is an error in the direction of displacement, acceleration, and/or spring force.

For the example system above, with mass m and spring constant k, we derive the following:

$$\sum F_x = ma_x = m\ddot{x} \tag{9.5.1.3}$$

$$-F_k = m\ddot{x} \tag{9.5.1.4}$$

$$-kx = m\ddot{x} \tag{9.5.1.5}$$

$$m\ddot{x} + kx = 0 \tag{9.5.1.6}$$

This gives us a differential equation that describes the motion of the system. We can rewrite it in normal form:

$$m\ddot{x} + kx = 0 \tag{9.5.1.7}$$

$$\Rightarrow \ddot{x} + \frac{k}{m}x = 0 \tag{9.5.1.8}$$

$$\Rightarrow \ddot{x} + \omega_n^2 x = 0 \tag{9.5.1.9}$$

The term ω_n is called the **angular natural frequency of the system**, and has units of radians/second.

$$\omega_n^2 = \frac{k}{m} \tag{9.5.1.10}$$

$$\omega_n = \sqrt{\frac{k}{m}} \tag{9.5.1.11}$$

Assuming that the initial perturbation of the system can be described by the position and velocity of the mass at $t = 0$, then:

$$x(0) = x_0 \tag{9.5.1.12}$$

$$v(0) = \dot{x}(0) = v_0. \tag{9.5.1.13}$$

The solution to the differential equation, that provides the position $x(t)$ of the system at time t, is:

$$x(t) = C\sin\left(\omega_n t + \phi\right), \tag{9.5.1.14}$$

$$\text{where:} \quad \omega_n = \sqrt{\frac{k}{m}},\ C = \sqrt{\left(\frac{v_0}{\omega_n}\right)^2 + x_0^2},\ \phi = \tan^{-1}\left(\frac{x_0\omega_n}{v_0}\right). \tag{9.5.1.15}$$

The **amplitude** C describes the maximum displacement during the oscillations (i.e. x_{max}), and the **phase** ϕ describes how the sine function is shifted in time.

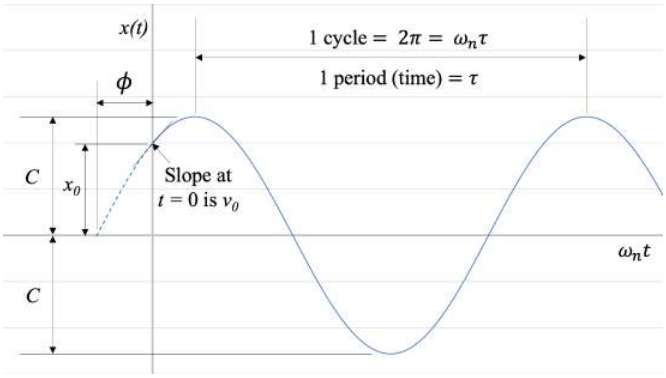


Figure 9.5.1.6: Displacement response of the mass spring system (solution to the differential equation).

For a system where there is torsional vibration (that is, the oscillation involves a rotation), the equations are similarly:

$$I\ddot{\theta} + k\theta = 0 \tag{9.5.1.16}$$

$$\Rightarrow \ddot{\theta} + \frac{k}{I}\theta = 0 \tag{9.5.1.17}$$

$$\Rightarrow \ddot{\theta} + \omega_n^2 \theta = 0, \tag{9.5.1.18}$$

$$\text{where } \omega_n = \sqrt{\frac{k}{I}}. \tag{9.5.1.19}$$

Assuming that the initial perturbation of the system can be described by the position and velocity of the mass at $t = 0$,

$$x(0) = x_0 \tag{9.5.1.20}$$

$$v(0) = \dot{x}(0) = v_0. \tag{9.5.1.21}$$

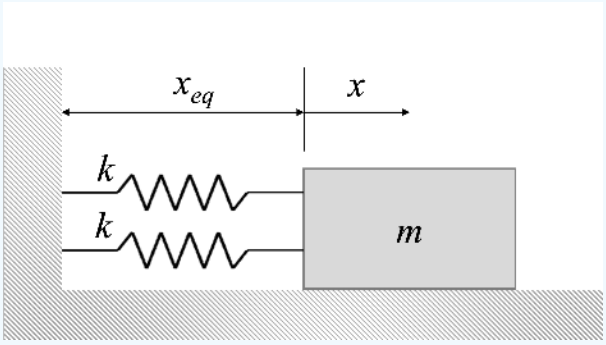
The solution to the differential equation, that provides the position $x(t)$ of the system at time t, is:

$$x(t) = C\sin(\omega_n t + \phi) \tag{9.5.1.22}$$

$$\text{where:} \quad \omega_n = \sqrt{\frac{k}{I}},\ C = \sqrt{\left(\frac{\omega_0}{\omega_n}\right)^2 + \theta_0^2},\ \phi = \tan^{-1}\left(\frac{\theta_0 \omega_n}{\omega_0}\right). \tag{9.5.1.23}$$

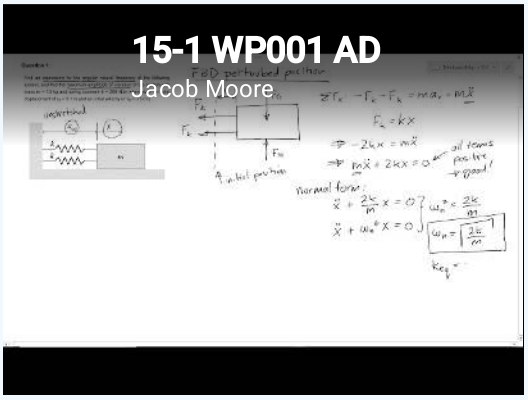
Question 1:

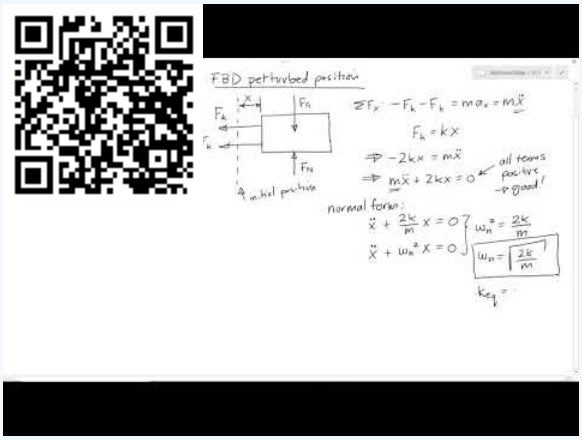
Find an expression for the angular natural frequency of the following system, and find the maximum amplitude of vibration of the system with mass m = 10 kg and spring constant k = 200 N/m when given an inital displacement of x_0 = 0.1 m and an initial velocity of v_0 = 0.3 m/s.



https://eng.libretexts.org/@go/page/122810

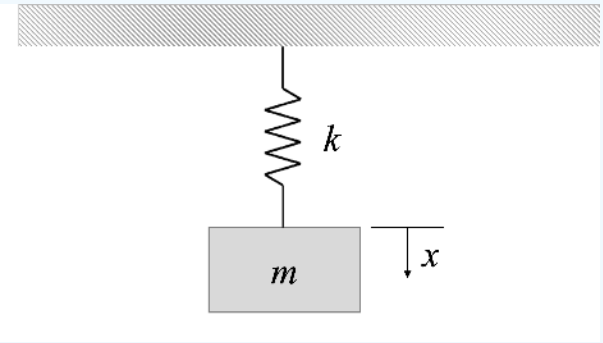
Solution:





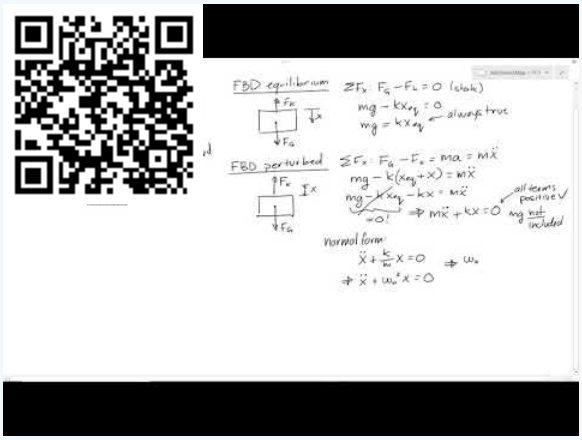
Question 2:

Determine the equation of motion of the system from Newton's Second Law. Assume mass m = 5 kg and spring constant k = 500 N/m. Find the initial displacement, x_0, such that the mass oscillates over a total range of 4 m. Assume the initial perturbation velocity, v_0, is 10 m/s.



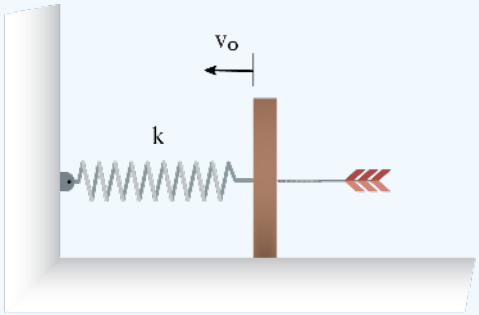
Solution:





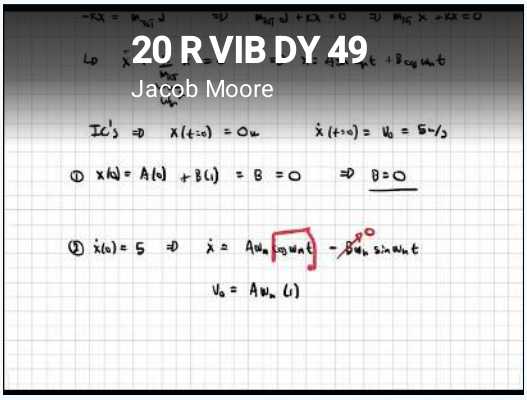
Question 3:

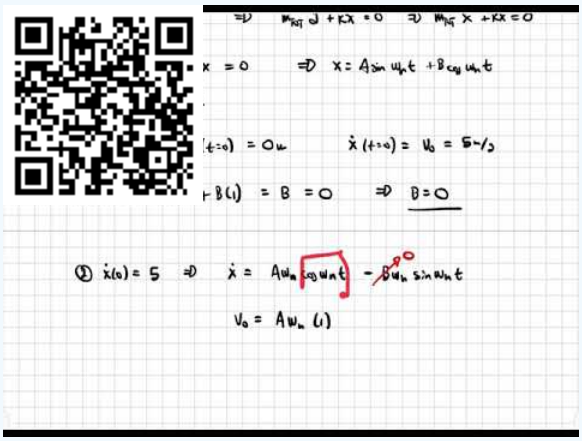
A target, with m=5kg, in an archery range is attached to a wall with a spring, k=50 N/m. An archer shoots a m=1kg arrow which sticks into the target. As a result of an impact with the arrow, the target has a horizontal initial velocity v_O=5m/s. Determine the maximum amplitude and the displacement at t=10s.



UBC Engineering

Solution:

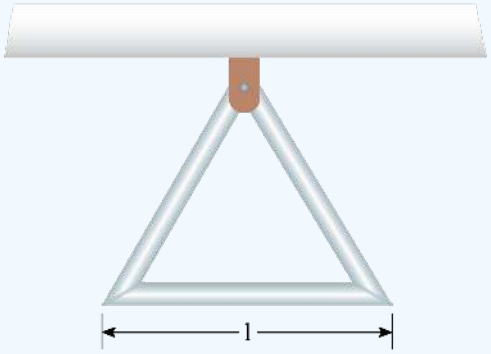




Transcript

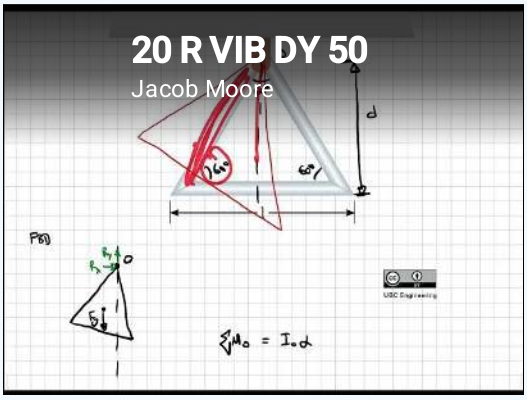
Question 4:

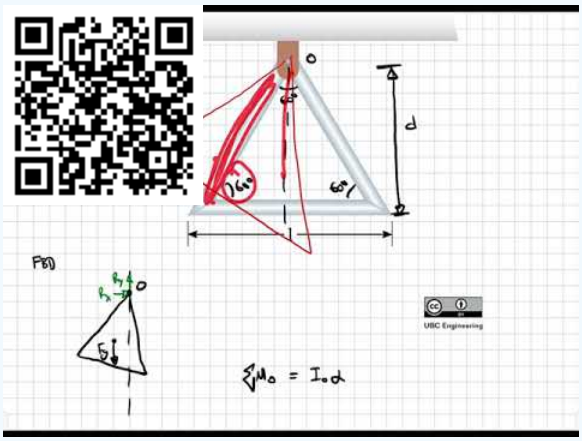
An equilateral triangle is made up of three individual bars. Each bar is 3m in length and 2kg in mass. Given that the frame is hung on the ceiling by one of its corners, determine the natural frequency.



UBC Engineering

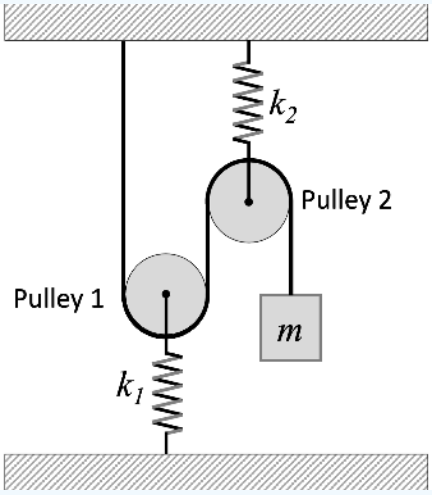
Solution:



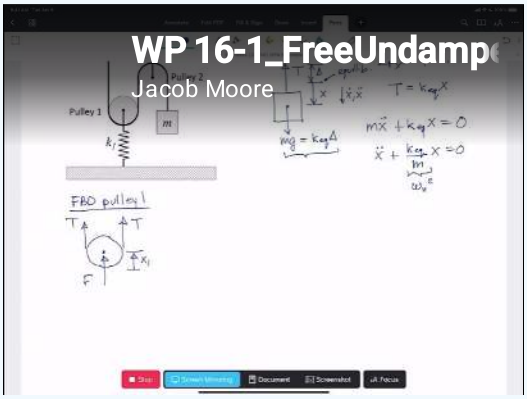


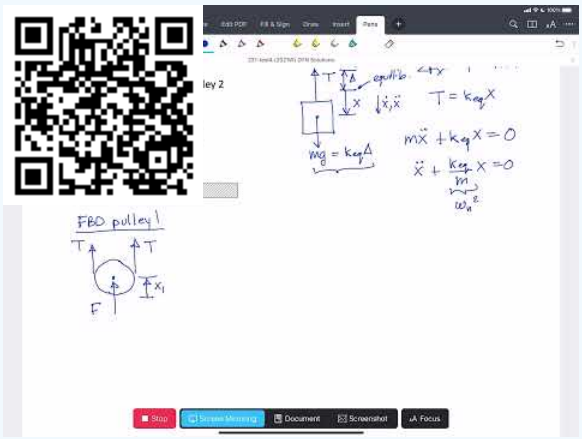
Question 5:

Find the natural angular frequency, ω_n, for the system shown. The pulleys are frictionless and have negligible mass. Values are: m = 8 kg, k_1= 300 N/m, k_2 = 400 N/m.



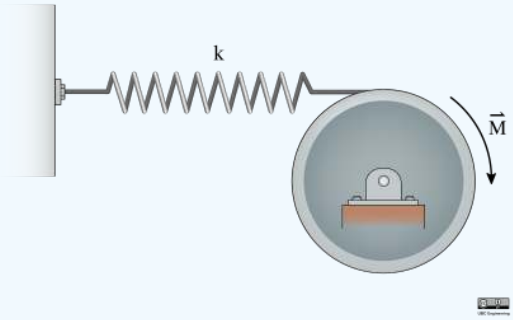
Solution:



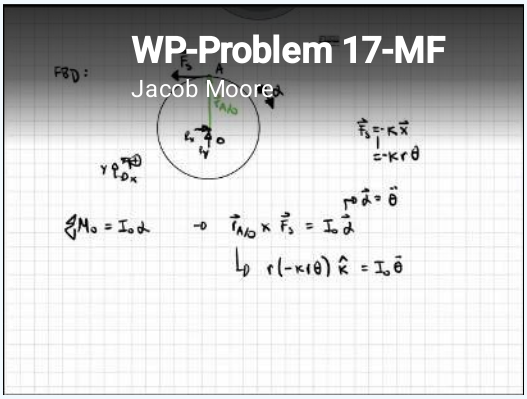


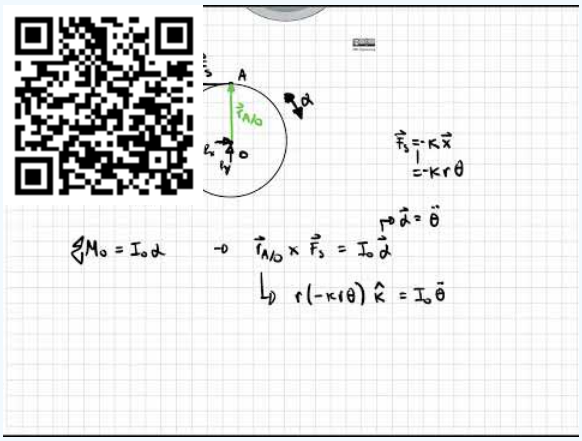
Question 6:

A spring with spring constant k = 90 N/m is connected to a disk with a mass 15 kg and radius r = 1 m. A moment of M= −40k^ N · m is required to keep the disk in static equilibrium. At time t = t_0 the moment is suddenly removed, and the disk begins to experience oscillatory motion due to the spring. With what period, τ, does the system oscillate? (You may assume sin θ = θ)



Solution:





This page titled 9.5.1: Undamped Free Vibrations is shared under a CC BY-SA 4.0 license and was authored, remixed, and/or curated by Jacob Moore & Contributors (Mechanics Map) via source content that was edited to the style and standards of the LibreTexts platform.

- **15.1: Undamped Free Vibrations** by Jacob Moore & Contributors is licensed CC BY-SA 4.0. Original source: http://mechanicsmap.psu.edu.

9.5.2: Viscous Damped Free Vibrations

Viscous damping is damping that is proportional to the velocity of the system. That is, the faster the mass is moving, the more damping force is resisting that motion. Fluids like air or water generate viscous drag forces.

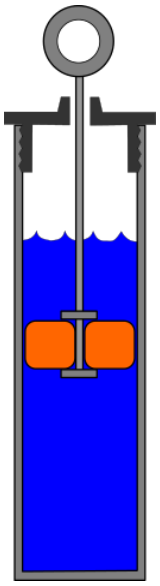


Figure 9.5.2.1: A diagram showing the basic mechanism in a viscous damper. As the system (mass) attached to the loop at the top vibrates up and down, the damper will resist motion in both directions due to the piston passing through the fluid. Image by Egmason, CC-BY SA.

We will only consider linear viscous dampers, that is where the damping force is linearly proportional to velocity. The equation for the force or moment produced by the damper, in either x or θ, is:

$$\vec{F}_c = c\dot{\vec{x}}, \tag{9.5.2.1}$$

$$\vec{M}_c = c\dot{\vec{\theta}}, \tag{9.5.2.2}$$

where c is the damping constant. This is a physical property of the damper based on the type of fluid, size of the piston, etc. Note that the units of c change depending on whether it is damping linear motion (N-s/m) or rotational motion (N-m s/rad).

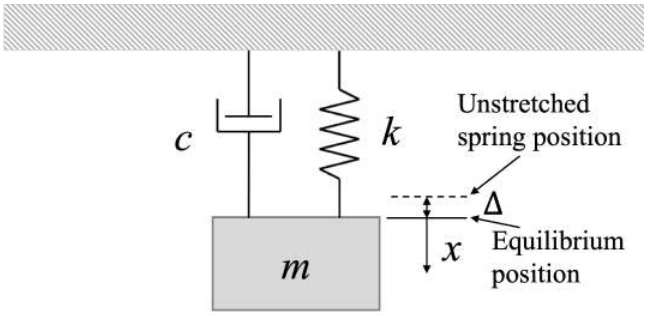


Figure 9.5.2.2: Diagram of a hanging mass-spring system, with a linear viscous damper, in equilibrium position. The spring is stretched from its natural length.

When the system is at rest in the equilibrium position, the damper produced no force on the system (no velocity), while the spring can produce force on the system, such as in the hanging mass shown above. Recall that this is the equilibrium position, but the spring is NOT at its unstretched length, as the static mass produces an extention of the spring.

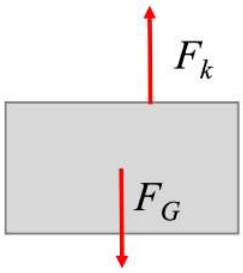


Figure 9.5.2.3: Free body diagram of the system in equilibrium position. The spring is at its equilibrium position, but it is stretched and does produce a force.

If we perturb the system (applying an initial displacement, an initial velocity, or both), the system will tend to move back to its equilibrium position. What that movement looks like will depend on the system parameters (m, c, and k).

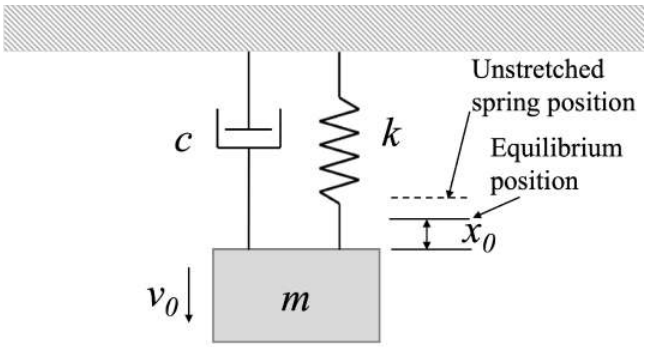


Figure 9.5.2.4: The system in a perturbed position. The spring is stretched further and the damper is extended, compared to their equilibrium positions.

To determine the equation of motion of the system, we draw a free body diagram of the system with perturbation and apply Newton's Second Law.

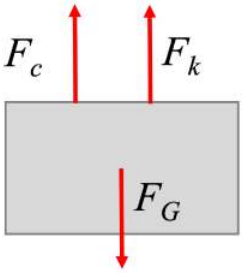


Figure 9.5.2.5: Free body diagram of the system with perturbation.

The process for finding the equation of motion of the system is again:

1. Sketch the system with a small positive perturbation (x or θ).
2. Draw the free body diagram of the perturbed system. Ensure that the spring force and the damper force have directions opposing the perturbation.
3. Find the one equation of motion for the system in the perturbed coordinate using Newton's Second Law. Keep the same positive direction for position, and assign positive acceleration in the same direction.
4. Move all terms of the equation to one side, and check that all terms are positive. If all terms are not positive, there is an error in the direction of displacement, acceleration, and/or spring or damper force.

For the example system above, with mass m, spring constant k and damping constant c, we derive the following:

$$\sum F_x = ma_x = m\ddot{x} \tag{9.5.2.3}$$

$$-F_k - F_c = m\ddot{x} \tag{9.5.2.4}$$

$$-kx - c(\dot{x}) = m\ddot{x} \tag{9.5.2.5}$$

$$m\ddot{x} + c(\dot{x}) + kx = 0 \tag{9.5.2.6}$$

This gives us a differential equation that describes the motion of the system. We can rewrite it in normal form:

$$m\ddot{x} + c\dot{x} + kx = 0 \tag{9.5.2.7}$$

$$\Rightarrow \ddot{x} + \frac{c}{m}\dot{x} + \frac{k}{m}x = 0 \tag{9.5.2.8}$$

$$\Rightarrow \ddot{x} + 2\zeta\omega_n\dot{x} + \omega_n^2 x = 0 \tag{9.5.2.9}$$

As before, the term ω_n is called the **angular natural frequency of the system**, and has units of rad/s.

$$\omega_n^2 = \frac{k}{m}\,; \quad \omega_n = \sqrt{\frac{k}{m}} \tag{9.5.2.10}$$

ζ (zeta) is called the damping ratio. It is a dimensionless term that indicates the level of damping, and therefore the type of motion of the damped system.

$$\zeta = \frac{c}{c_c} = \frac{\text{actual damping}}{\text{critical damping}} \tag{9.5.2.11}$$

The expression for critical damping comes from the solution of the differential equation. The solution to the system differential equation is of the form

$$x(t) = ae^{rt}, \tag{9.5.2.12}$$

where a is constant and the value(s) of r can be can be obtained by differentiating this general form of the solution and substituting into the equation of motion.

$$mr^2e^{rt} + cre^{rt} + ke^{rt} = 0 \tag{9.5.2.13}$$

$$\Rightarrow (mr^2 + cr + k)e^{rt} = 0 \tag{9.5.2.14}$$

Because the exponential term is never zero, we can divide both sides by that term and get:

$$mr^2 + cr + k = 0. \tag{9.5.2.15}$$

Using the quadratic formula, we can find the roots of the equation:

$$r_{1,2} = \frac{-c \pm \sqrt{c^2 - 4mk}}{2m} \tag{9.5.2.16}$$

Critical damping occurs when the term under the square root sign equals zero:

$$c_c^2 = 4mk \tag{9.5.2.17}$$

$$c_c = 2\sqrt{mk} = 2m\omega_n \tag{9.5.2.18}$$

Four Viscous Damping Cases:

There are four basic cases for the damping ratio. For the solutions that follow in each case, we will assume that the initial perturbation displacement of the system is x_0 and the initial perturbation velocity of the system is v_0.

1. $\zeta = 0$: Undamped

$$c = 0 \tag{9.5.2.19}$$

This is the case covered in the previous section. Undamped systems oscillate about the equilibrium position continuously, unless some other force is applied.

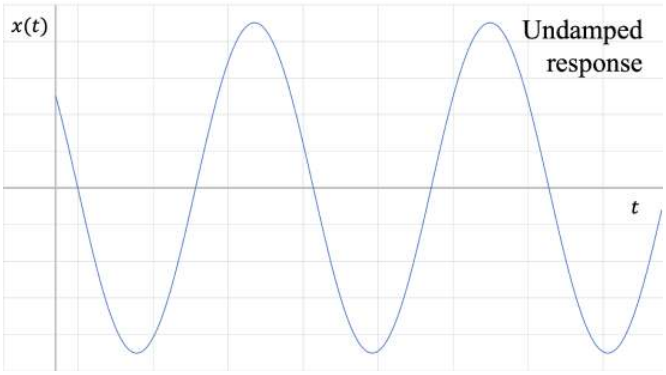


Figure 9.5.2.6: Response of an undamped system.

2. $\zeta > 1$: Overdamped

$$c^2 > 4mk \tag{9.5.2.20}$$

Roots are both real and negative, but not equal to each other. Overdamped systems move slowly toward equilibrium without oscillating.

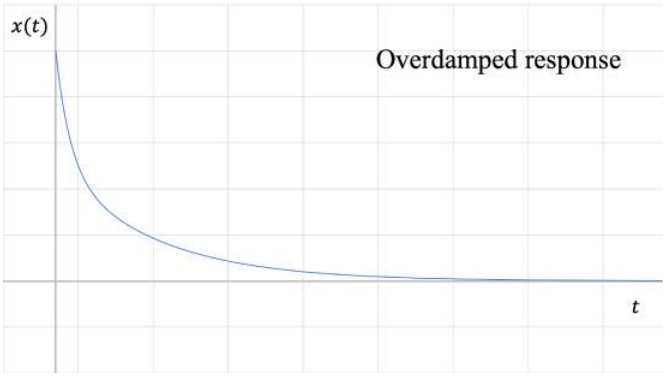


Figure 9.5.2.7: Response of an overdamped system.

The response for an overdamped system is:

$$x(t) = a_1 e^{\left(\frac{-c+\sqrt{c^2-4mk}}{2m}\right)t} + a_2 e^{\left(\frac{-c-\sqrt{c^2-4mk}}{2m}\right)t}, \tag{9.5.2.21}$$

$$\text{where } a_1 = \frac{-v_0 + r_2 x_0}{r_2 - r_1} \text{ and } a_2 = \frac{v_0 + r_1 x_0}{r_2 - r_1}. \tag{9.5.2.22}$$

3. $\zeta = 1$: Critically damped

$$c^2 = 4mk (= c_c^2) \tag{9.5.2.23}$$

Roots are real and both equal to $-\omega_n$. Critically-damped systems will allow the fastest return to equilibrium without oscillation.

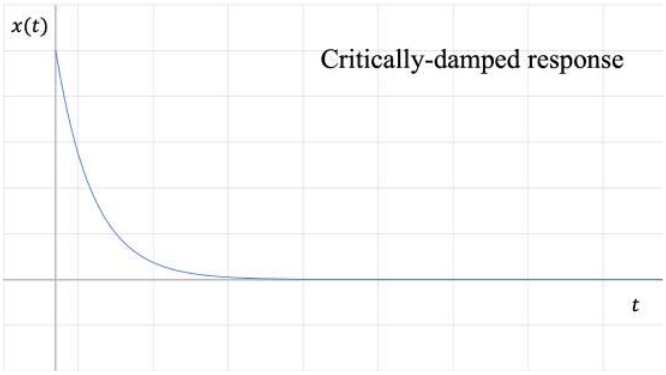


Figure 9.5.2.8: Response of an critically-damped system.

The solution for a critically-damped system is:

$$x(t) = (A + Bt)e^{-\omega_n t}, \tag{9.5.2.24}$$

$$\text{where } A = x_0 \text{ and } B = v_0 + x_0\omega_n. \tag{9.5.2.25}$$

4. $\zeta < 1$: Underdamped

$$c^2 < 4mk \tag{9.5.2.26}$$

The roots are complex numbers. Underdamped systems do oscillate around the equilibrium point; unlike undamped systems, the amplitude of the oscillations diminishes until the system eventually stops moving at the equilibrium position.

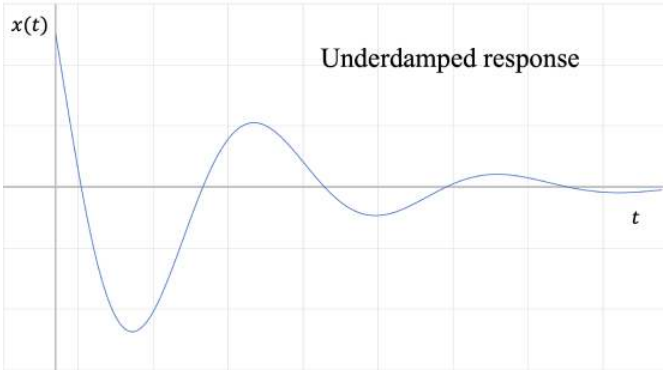


Figure 9.5.2.9: Response of an underdamped system.

The solution for an underdamped system is:

$$x(t) = [C_1 \sin(\omega_d t) + C_2 \cos(\omega_d t)]e^{-\omega_n \zeta t}, \tag{9.5.2.27}$$

$$\text{where } C_1 = \frac{v_0 + \omega_n \zeta x_0}{\omega_d},\ C_2 = x_0,\ \text{and } \zeta = \frac{c}{2m\omega_n}. \tag{9.5.2.28}$$

This can alternately be expressed as:

$$x(t) = A \sin(\omega_d t + \phi)\, e^{-\omega_n \zeta t} \tag{9.5.2.29}$$

Where:

$$\begin{aligned} A &= \sqrt{\frac{(v_0+\omega_n\zeta x_0)^2+(x_0\omega_d)^2}{\omega_d^2}} \\ \phi &= \tan^{-}1\left(\frac{x_0\omega_d}{v_0+\omega_n\zeta x_0}\right) \\ \zeta &= \frac{c}{2m\omega_n} \end{aligned} \tag{9.5.2.30}$$

ω_d is called the **damped natural frequency** of the system. It is always less than ω_n:

$$\omega_d = \omega_n \sqrt{1-\zeta^2}. \tag{9.5.2.31}$$

The period of the underdamped response differs from the undamped response as well.

$$\text{Undamped:} \quad \tau_n = \frac{2\pi}{\omega_n} \tag{9.5.2.32}$$

$$\text{Underdamped:} \quad \tau_d = \frac{2\pi}{\omega_d} \tag{9.5.2.33}$$

Comparison of Viscous Damping Cases:

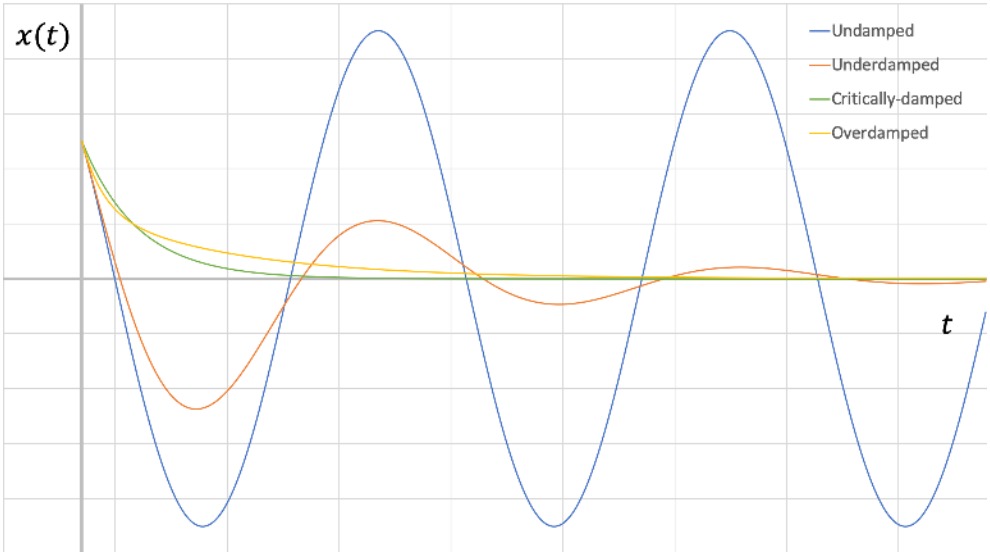
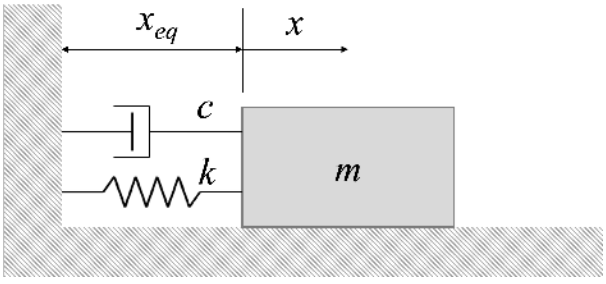


Figure 9.5.2.10: Responses for all four types of system (or values of damping ratio) in viscous damping. All four systems have the same mass and spring values, and have been given the same initial perturbations (initial position and initial velocity); this is apparent because they start at the same y-intercept and have the same slope at $x = 0$.

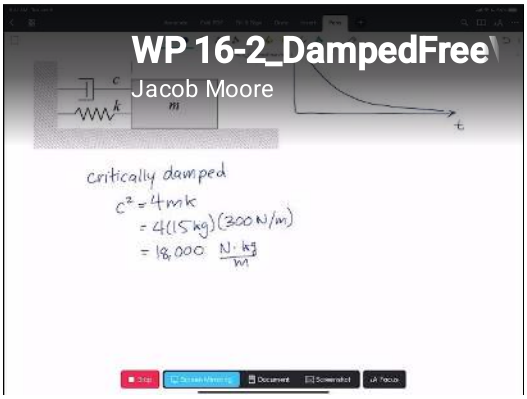
In the figure above, we can see that the critically-damped response results in the system returning to equilibrium the fastest. Also, we can see that the underdamped system amplitude is quite attenuated compared to the undamped case.

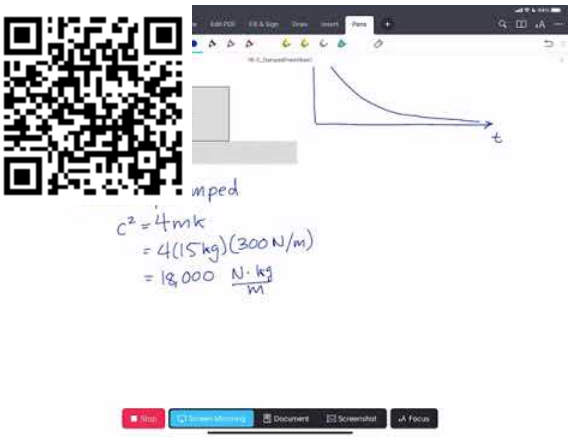
Question 1:

A 15 kg block on a frictionless surface is attached to a spring (k = 300 N/m). Find the damping constant, c, that will make the system critically damped.



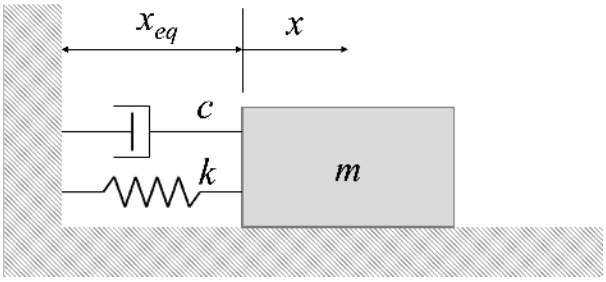
Solution:



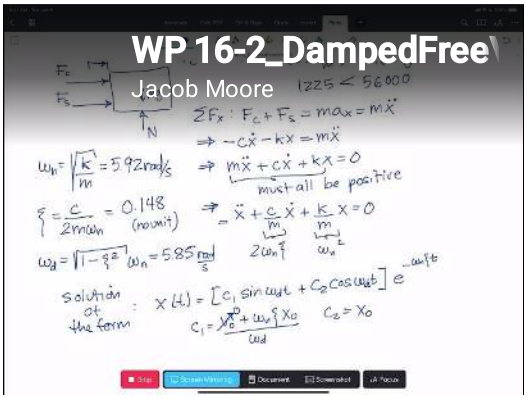


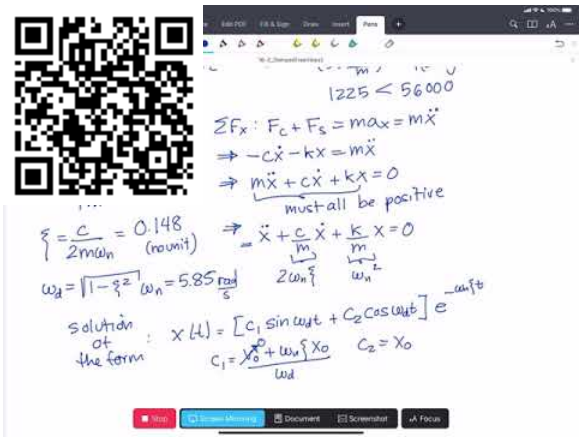
Question 2:

A 20 kg block on a frictionless surface is attached to a spring (k = 700 N/m) and a damper (c = 35 N-s/m). If the initial perturbation, x_0, is 0.2 m (v_0 = 0), how many half-cycles will it take for the amplitude of the oscillation to peak at half the original displacement or less (i.e. $|x_{peak}|<=|x_0/2|$)?



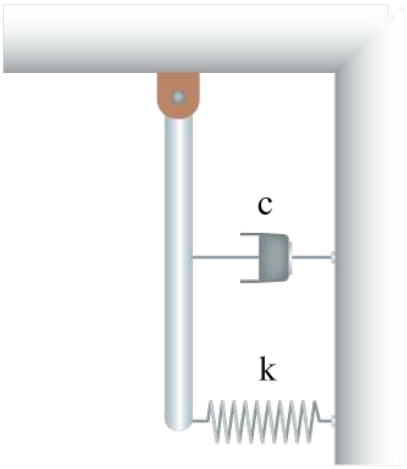
Solution:





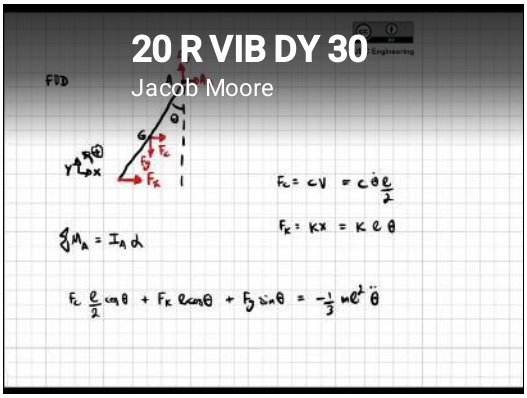
Question 3:

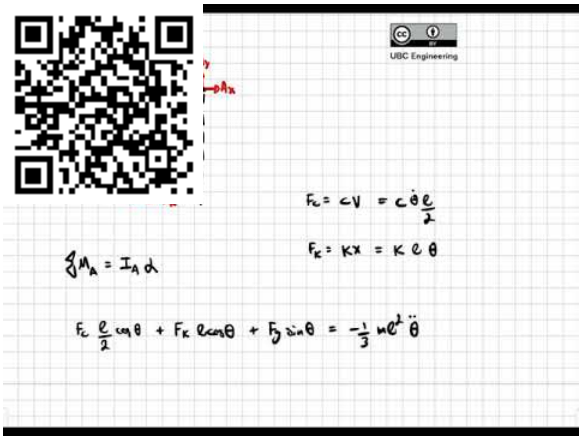
A bar of length 1.5m mass of 2kg is pinned to the ceiling. A spring, k=50N/m, is attached to the bottom of the bar and a damper, c=10Ns/m, is attached halfway down. Given a small angle displacement, find the damped frequency and the roots.



UBC Engineering

Solution:





This page titled 9.5.2: Viscous Damped Free Vibrations is shared under a CC BY-SA 4.0 license and was authored, remixed, and/or curated by Jacob Moore & Contributors (Mechanics Map) via source content that was edited to the style and standards of the LibreTexts platform.

- **15.2: Viscous Damped Free Vibrations** by Jacob Moore & Contributors is licensed CC BY-SA 4.0. Original source: http://mechanicsmap.psu.edu.

9.5.3: Friction (Coulomb) Damped Free Vibrations

Friction can also provide vibration damping. In this case, however, the damping is **not** proportional to the magnitude of velocity. It only depends on the direction of velocity.

We remember from the section on dry friction that the force of friction in sliding depends only on the coefficient of kinetic friction, μ_k, and the normal force, F_N.

$$F_f = \mu_k F_N \tag{9.5.3.1}$$

The above equation does not include velocity. We know that kinetic friction acts to oppose motion, however, so a more complete expression would be:

$$F_f = -\text{sgn}(\dot{x})\mu_k F_N, \tag{9.5.3.2}$$

where sgn is the "sign" function, a function that captures the sign (direction) of velocity. The above equation then indicates that the direction of friction is always opposite the direction of velocity, but the magnitude of velocity does not make a difference in the magnitude of friction.

The equation of motion of the system becomes:

$$m\ddot{x} + \mu mg\ \text{sgn}(\dot{x}) + kx = 0, \tag{9.5.3.3}$$

and the solution to this equation of motion is:

$$x(t) = \left(x_0 - \frac{(2n-1)\mu mg}{k}\right)\cos(\omega_n t) + \frac{\mu mg}{k}(-1)^{n+1}. \tag{9.5.3.4}$$

If we plot the response, we can see that there are several differences from a system with viscous damping.

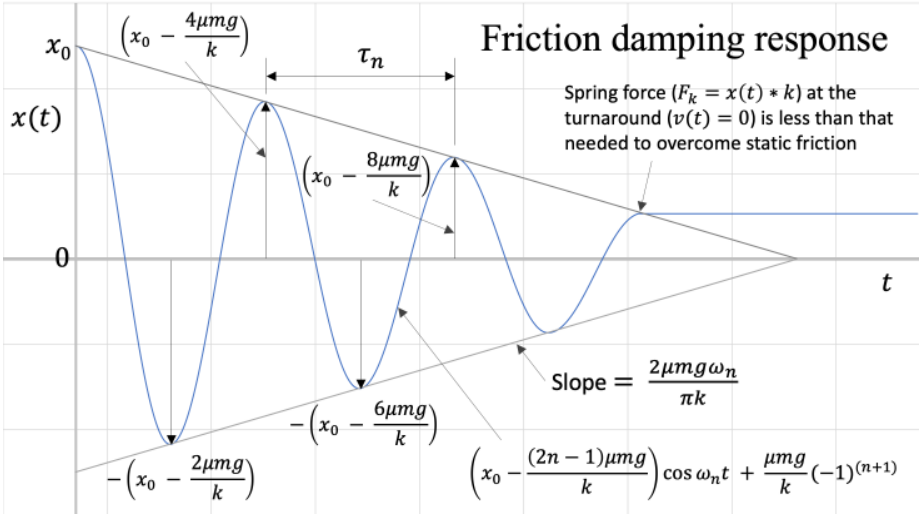


Figure 9.5.3.1: Response of the system in friction damping.

Some differences when compared to viscous damping include:

1. The system oscillates at the natural frequency of the system, not a damped natural frequency.
2. The bounding curves are linear, not exponential.
3. The system does not return to zero. This is because the magnitude of the friction force does not diminish as the system amplitude reduces, and at some point the spring force is no longer able to overcome the static friction that the system experiences when it changes direction ($v = 0$).

Comparison to Viscous Underdamped System

If we consider our simple linear mass-spring system, the magnitude of F_f does not change with velocity, unlike with viscous damping. If we plotted both types of damping for the same system, we would get the following:

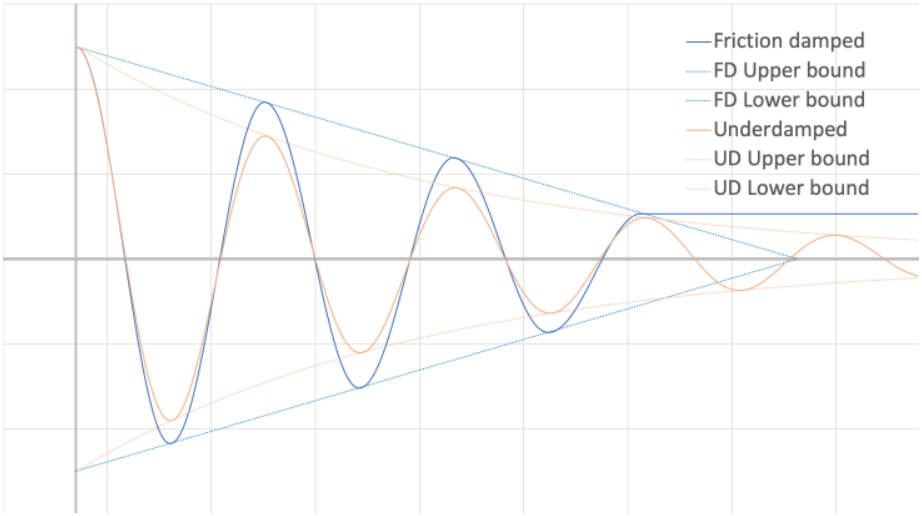
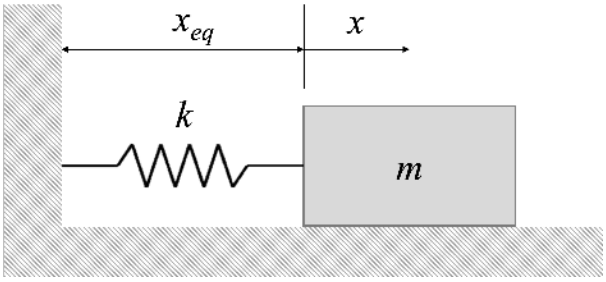


Figure 9.5.3.2: Response of the system in friction damping and in viscous damping, for the same initial conditions (x_0, v_0), spring constants and masses.

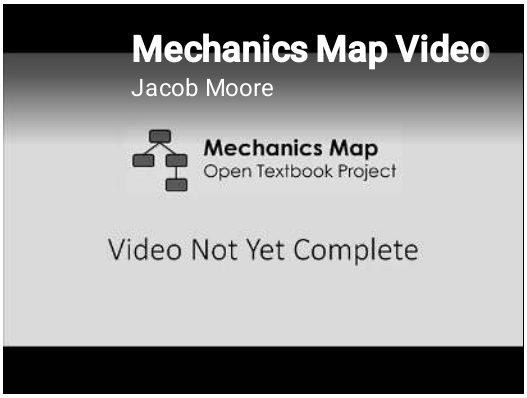
Note that the viscous damping has more reduction in amplitude earlier (despite relatively light damping), but continues oscillating past the point when the friction-damped system has stopped (specific relative values are dependent on the values of damping constant and coefficients of friction). Also note that since the viscous damping is relatively light, the difference in period between the two plots is quite small in this example.

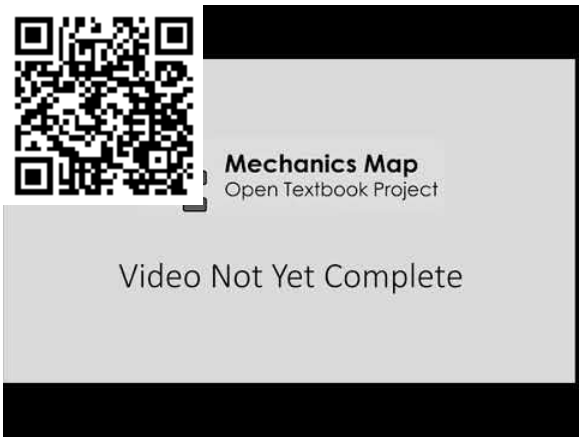
Question 1:

A 10 kg mass is attached to a spring with spring constant of 300 N/m. The system is initially perturbed by a distance of 0.25 m. Find the kinetic coefficient of friction, μ_k, such that the mass only moves 1/3 of the initial displacement distance past equilibrium.



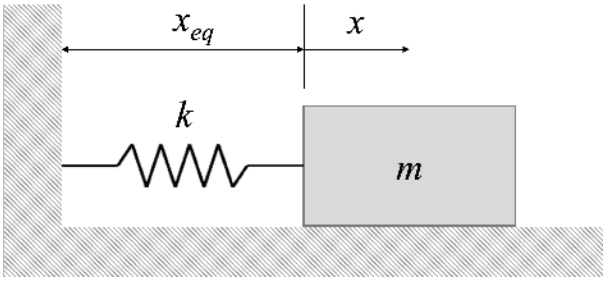
Solution:



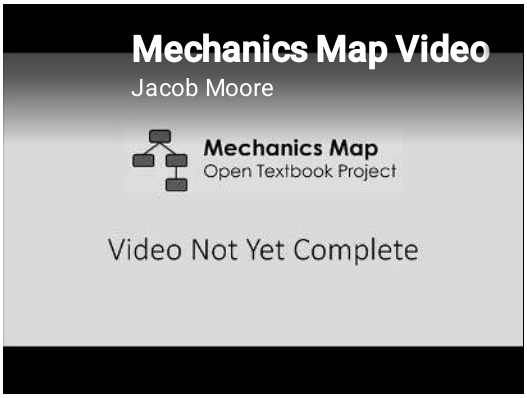


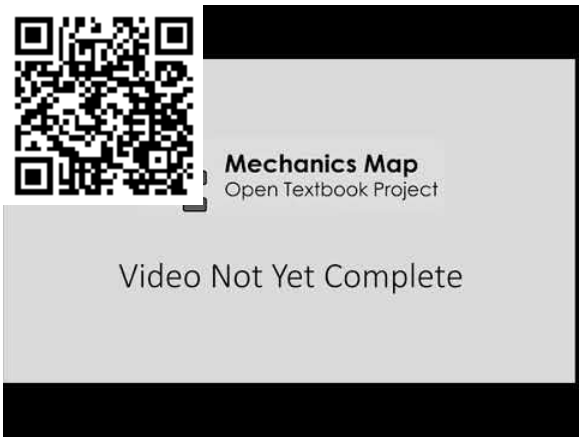
Question 2:

A 12 kg mass is attached to a spring with spring constant of 600 N/m. The system is initially perturbed by a distance of 1.2 m. Determine the displacement distance where the system will come to rest.



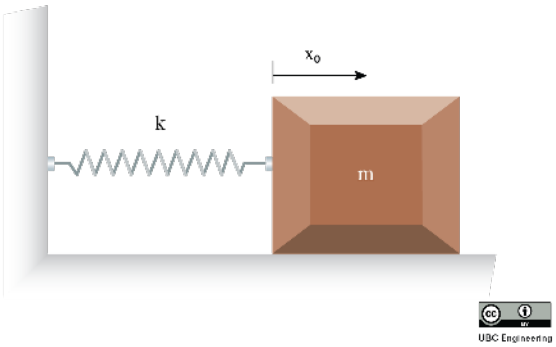
Solution:



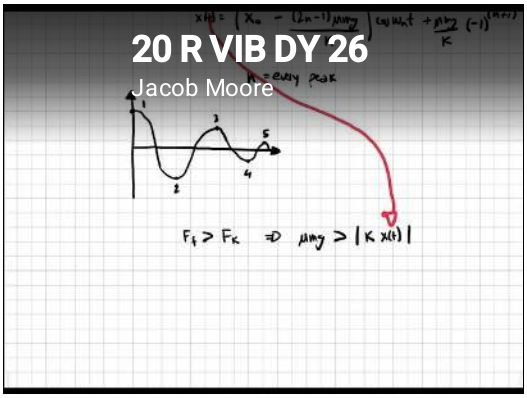


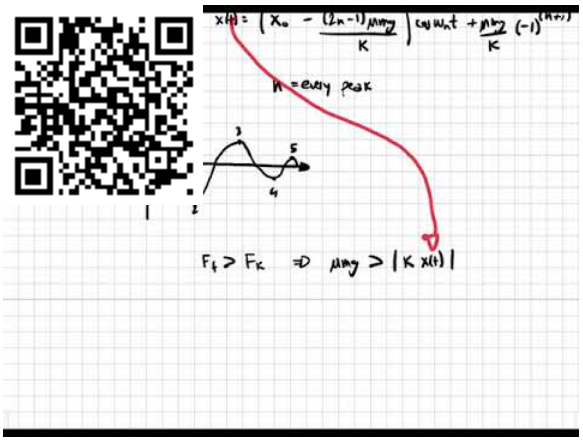
Question 3:

A box of mass m=5kg is connected to a spring, k=200N/m on the wall. The ground has a static and kinetic friction coefficient μ=0.2. Given an initial displacement of 1m, determine how long it takes to come to a stop.



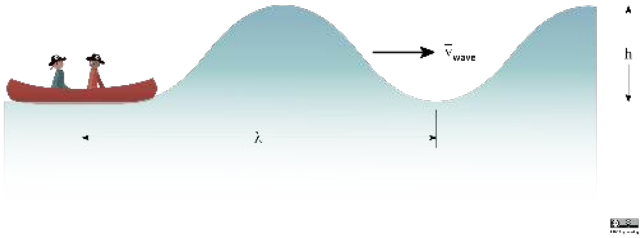
Solution:



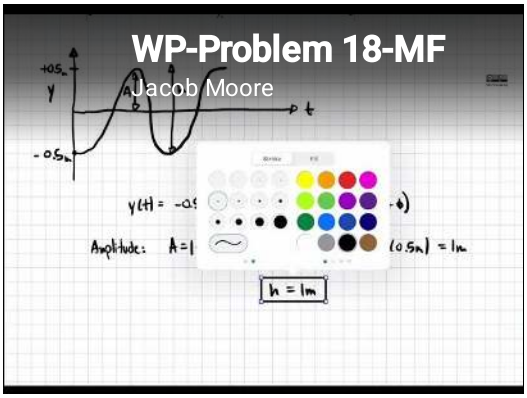


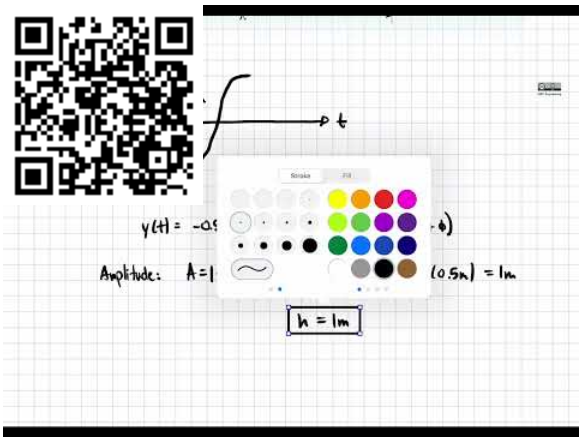
Question 4:

A small boat sits anchored in the ocean. The vertical displacement of the boat as a function of time is: y(t)=0.5 cos(0.419t)m. Waves are measured to have a wavelength λ = 15 m. What is the height and horizontal velocity of the waves with respect to the boat? Take the starting displacement of the boat to be at the bottom of the wave and y positive upward.



Solution:





This page titled 9.5.3: Friction (Coulomb) Damped Free Vibrations is shared under a CC BY-SA 4.0 license and was authored, remixed, and/or curated by Jacob Moore & Contributors (Mechanics Map) via source content that was edited to the style and standards of the LibreTexts platform.

- **15.3: Friction (Coulomb) Damped Free Vibrations** by Jacob Moore & Contributors is licensed CC BY-SA 4.0. Original source: http://mechanicsmap.psu.edu.

9.5.4: Undamped Harmonic Forced Vibrations

Often, mechanical systems are not undergoing free vibration, but are subject to some applied force that causes the system to vibrate. In this section, we will consider only harmonic (that is, sine and cosine) forces, but any changing force can produce vibration.

When we consider the free body diagram of the system, we now have an additional force to add: namely, the external harmonic excitation.

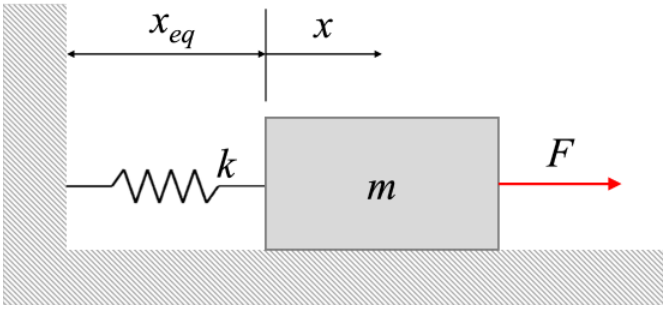


Figure 9.5.4.1: A mass-spring system with an external force, F, applying a harmonic excitation.

The equation of motion of the system above will be:

$$m\ddot{x} + kx = F, \tag{9.5.4.1}$$

where F is a force of the form:

$$F = F_0 \sin(\omega_0 t). \tag{9.5.4.2}$$

This equation of motion for the system can be re-written in standard form:

$$\ddot{x} + \frac{k}{m}x = \frac{F_0}{m}\sin(\omega_0 t). \tag{9.5.4.3}$$

The solution to this system consists of the superposition of two solutions: a particular solution, x_p (related to the forcing function), and a complementary solution, x_c (which is the solution to the system without forcing).

As we saw previously, the complementary solution is the solution to the undamped free system:

$$x_c = C\sin(\omega_n t + \phi). \tag{9.5.4.4}$$

We can obtain the particular solution by assuming a solution of the form:

$$x_p = D\sin(\omega_0 t), \tag{9.5.4.5}$$

where ω_0 is the frequency of the harmonic forcing function. We differentiate this form of the solution, and then sub into the above equation of motion:

$$\ddot{x}_p = -\omega_0^2 D\sin(\omega_0 t) \tag{9.5.4.6}$$

$$-m\omega_0^2 D\sin(\omega_0 t) + kD\sin(\omega_0 t) = F_0\sin(\omega_0 t) \tag{9.5.4.7}$$

After solving for D, we can then use it to find the particular solution, x_p:

$$D = \frac{\frac{F_0}{k}}{1 - \left(\frac{\omega_0}{\omega_n}\right)^2} \tag{9.5.4.8}$$

$$x_p = \frac{\frac{F_0}{k}}{1 - \left(\frac{\omega_0}{\omega_n}\right)^2} \sin(\omega_0 t). \tag{9.5.4.9}$$

Thus, the general solution for a forced, undamped system is:

$$x_G(t) = \frac{\frac{F_0}{k}}{1 - \left(\frac{\omega_0}{\omega_n}\right)^2} \sin(\omega_0 t) + C\sin(\omega_n t + \phi) \tag{9.5.4.10}$$

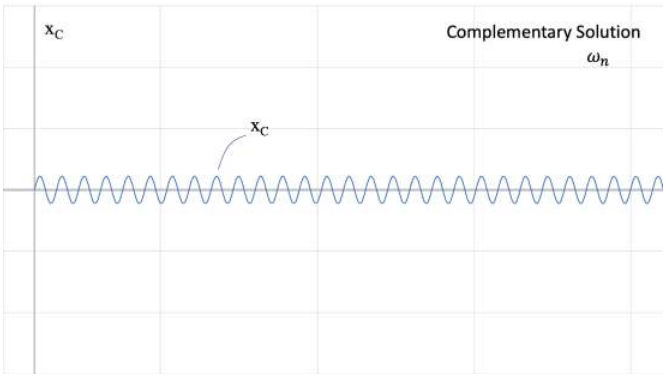


Figure 9.5.4.2: The complementary solution of the equation of motion. This represents the natural response of the system, and oscillates at the angular natural frequency. This is the transient response.

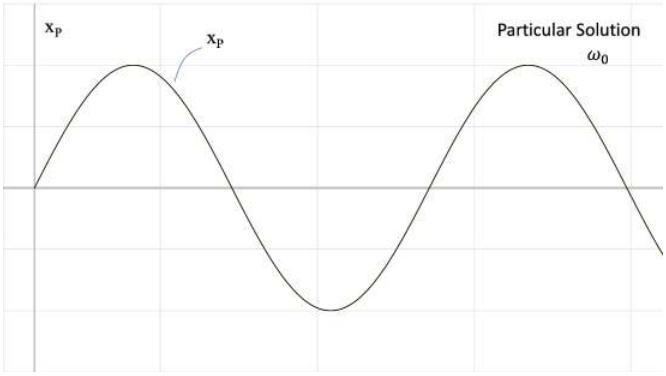


Figure 9.5.4.3: The particular solution of the equation of motion. This represents the forced response of the system, and oscillates at the angular forced frequency. This is the steady-state response.

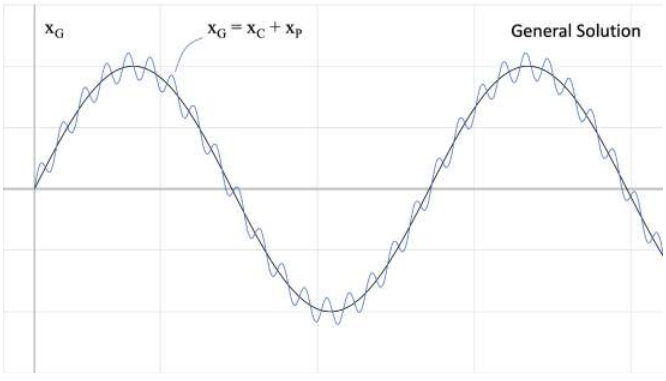


Figure 9.5.4.4: The general solution of the equation of motion. This represents the combined response of the system, and the sum of the complementary (or natural) and particular (or forced) responses.

The above figures show the two responses at different frequencies. Recall that the value of ω_n comes from the physical characteristics of the system (m, k) and ω_0 comes from the force being applied to the system. These responses are summed, to achieve the blue response (general solution) in Figure 9.5.4.4 above.

Steady-State Response:

In reality, this superimposed response does not last long. Every real system has some damping, and the natural response of the system will be damped out. As long as the external harmonic force is applied, however, the response to it will remain. When evaluating the response of the system to a harmonic forcing function, we will typically consider the steady-state response, when the natural response has been damped out and the response to the forcing function remains.

Amplitude of Forced Vibration

The amplitude of the steady-state forced vibration depends on the ratio of the forced frequency to the natural frequency. As ω_0 approaches ω_n (the ratio approaches 1), the magnitude D becomes very large. We can define a magnification factor:

$$MF = \frac{\frac{F_0}{k}}{\frac{F_0}{k}} = \frac{1}{1-\left(\frac{\omega_0}{\omega_n}\right)^2} \tag{9.5.4.11}$$

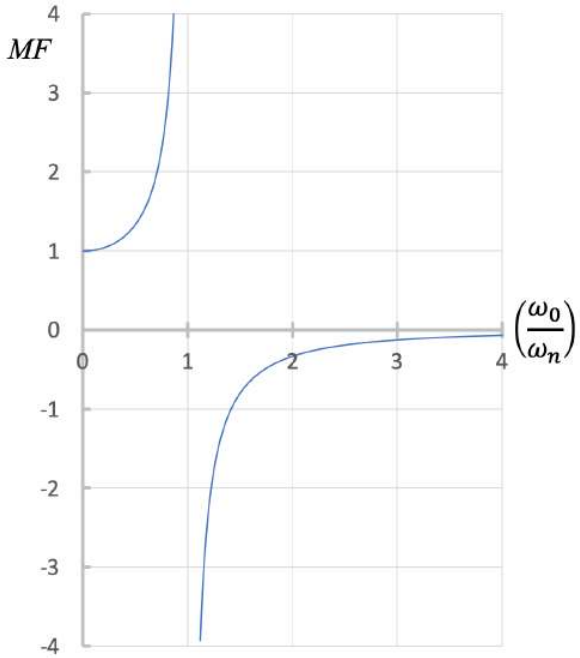


Figure 9.5.4.5: The magnification factor, MF, is defined as the ratio of the amplitude of the steady-state vibration to the displacement that would be achieved by static deflection.

From the figure above, we can discuss various cases:

- $\omega_0 = \omega_n$: Resonance occurs. This results in very large-amplitude vibrations, and is associated with high stress and failure to the system.
- $\omega_0 \sim 0, \mathbf{MF} \sim 1$: The forcing function is nearly static, leaving essentially the static deflection and limited natural vibration.
- $\omega_0 < \omega_n$: Magnification is positive and greater than 1, meaning the vibrations are in phase (when the force acts to the left, the system displaces to the left) and the amplitude of vibration is larger than the static deflection.
- $\omega_0 > \omega_n$: Magnification is negative and the absolute value is typically smaller than 1, meaning the vibration is out of phase with the motion of the forcing function (when the force acts to the left, the system displaces to the right) and the amplitude of vibration is smaller than the static deflection.
- $\omega_0 >> \omega_n$: The force is changing direction too fast for the block's motion to respond.

Rotating Unbalance:

One common cause of harmonic forced vibration in mechanical systems is rotating unbalance. This occurs when the axis of rotation does not pass through the center of mass, meaning that the center of mass experiences some acceleration instead of

remaining stationary. This causes a force on the axle that changes direction as the center of mass rotates. We can represent this as a small mass, m, rotating about the axis of rotation at some distance, called an eccentricity, e. The forced angular frequency, ω_0, in this case is the angular frequency of the rotating system.

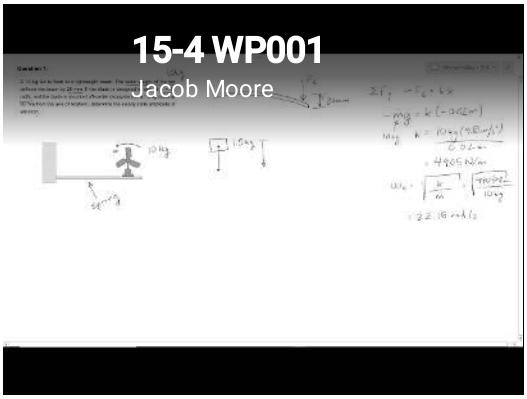
Worked Problems:

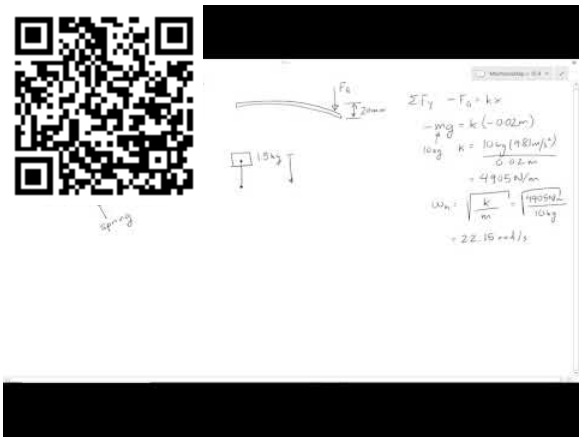
Question 1:

A 10 kg fan is fixed to a lightweight beam. The static weight of the fan deflects the beam by 20 mm. If the blade is designed to spin at ω = 15 rad/s, and the blade is mounted off-center (equivalent to a 1.5 kg mass at 50 mm from the axis of rotation), determine the steady-state amplitude of vibration.



Solution:





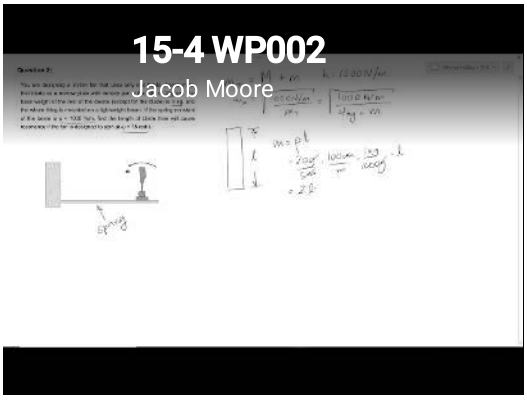
Question 2:

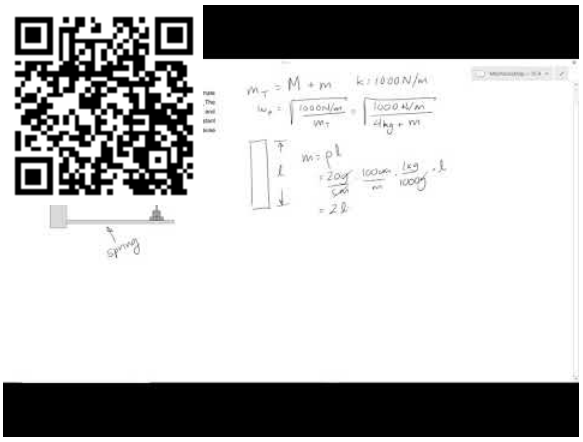
You are designing a stylish fan that uses only one blade. Approximate that blade as a narrow plate with density per unit length of 20g/cm. The base weight of the rest of the device (except for the blade) is 4 kg, and the whole thing is mounted on a lightweight

beam. If the spring constant of the beam is k = 1000 N/m, find the length of blade than will cause resonance if the fan is designed to spin at ω = 15 rad/s.



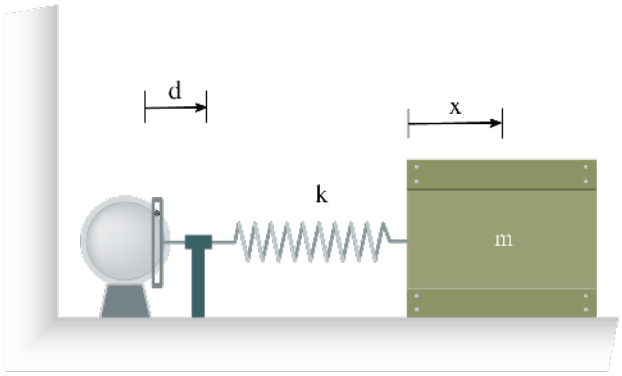
Solution:





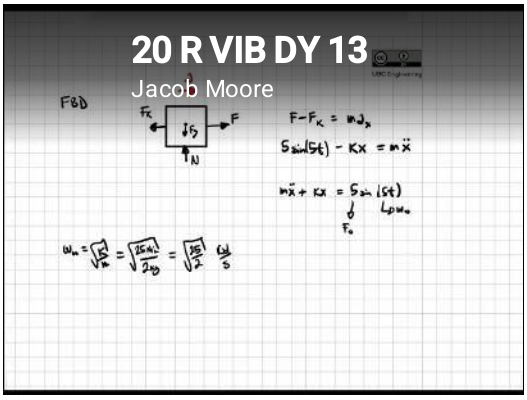
Question 3:

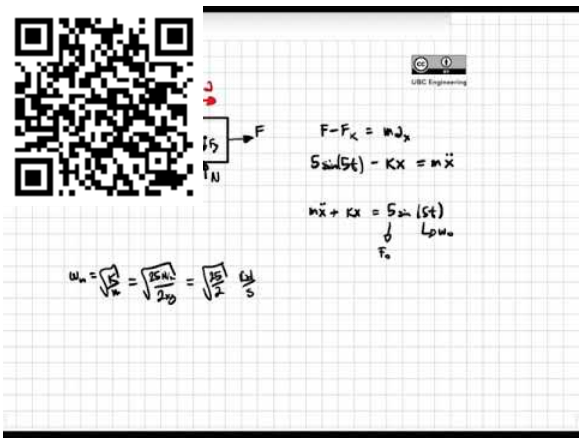
A 2kg box of orange juice contains the instructions “shake well”. An engineering student decides to get smart and connect the juice box with a spring, which has a spring constant k=25N/m, and applies a periodic force. The periodic force is described as F=5sin(5t). What is the maximum amplitude of the periodic motion and Magnification Factor at steady state?



UBC Engineering

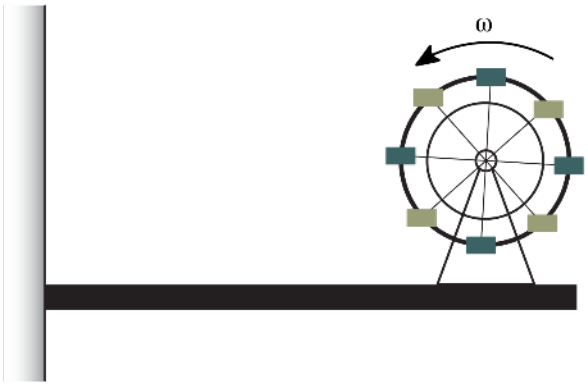
Solution:





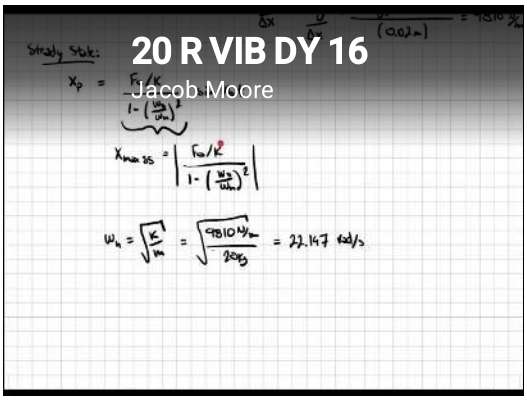
Question 4:

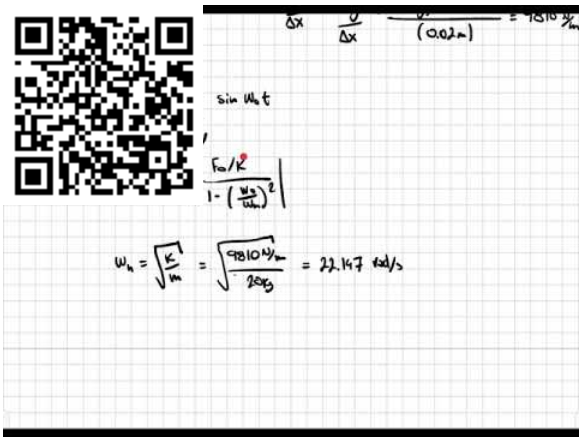
The world's smallest Ferris wheel (m=20kg) is mounted on the end of a horizontal beam for no apparent reason. The wheel is mounted eccentrically in such a way that the eccentricity is equivalent to a mass of 5kg located 0.15m away from the axis of rotation. The static weight of the Ferris wheel causes a deflection of 20mm in the beam. Given that the wheel spins at a rate of 15 rad/sec, find the steady-state amplitude of vibration.



UBC Engineering

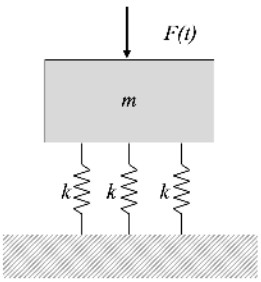
Solution:



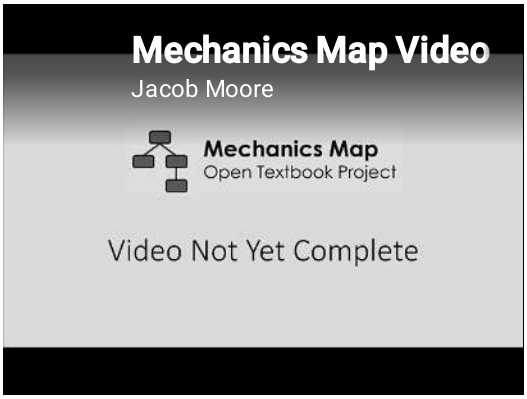


Question 5:

A 150 kg mass is supported on three springs. Find the spring constant, *k* [N/m], required for a magnification factor of 2 when *F(t) = 25 sin(4t)*.



Solution:





This page titled 9.5.4: Undamped Harmonic Forced Vibrations is shared under a CC BY-SA 4.0 license and was authored, remixed, and/or curated by Jacob Moore & Contributors (Mechanics Map) via source content that was edited to the style and standards of the LibreTexts platform.

- **15.4: Undamped Harmonic Forced Vibrations** by Jacob Moore & Contributors is licensed CC BY-SA 4.0. Original source: http://mechanicsmap.psu.edu.

9.5.5: Viscous Damped Harmonic Forced Vibrations

As described in the previous section, many vibrations are caused by an external harmonic forcing function (such as rotating unbalance). While we assumed that the natural vibrations of the system eventually damped out somehow, leaving the forced vibrations at steady-state, by explicitly including viscous damping in our model we can evaluate the system through the transient stage when the natural vibrations are present.

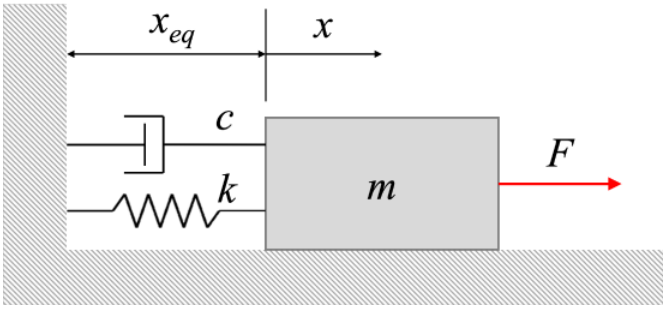


Figure 9.5.5.1: A mass-spring-damper system with an external force, F, applying a harmonic excitation.

Consider the system above. The equation of the system becomes:

$$m\ddot{x} + c\dot{x} + kx = F_0 \sin(\omega_0 t) \tag{9.5.5.1}$$

$$\Rightarrow \ddot{x} + \frac{c}{m}\dot{x} + \frac{k}{m}x = \frac{F_0}{m} \sin(\omega_0 t). \tag{9.5.5.2}$$

Because the natural vibrations will damp out with friction (as mentioned in undamped harmonic vibrations), we will only consider the particular solution. This particular solution will have the form:

$$x_p(t) = X' \sin(\omega_0 t - \phi'). \tag{9.5.5.3}$$

After solving, we determine that the expressions for X' and ϕ' are:

$$X' = \frac{\frac{F_0}{k}}{\sqrt{\left[1 - \left(\frac{\omega_0}{\omega_n}\right)^2\right]^2 + \left[2\frac{c}{c_c}\frac{\omega_0}{\omega_n}\right]^2}} \tag{9.5.5.4}$$

$$\phi' = \tan^{-1}\left[\frac{2\frac{c}{c_c}\frac{\omega_0}{\omega_n}}{1 - \left(\frac{\omega_0}{\omega_n}\right)^2}\right]. \tag{9.5.5.5}$$

The magnification factor now becomes:

$$MF = \frac{X'}{\frac{F_0}{k}} = \frac{1}{\sqrt{\left[1 - \left(\frac{\omega_0}{\omega_n}\right)^2\right]^2 + \left[2\frac{c}{c_c}\frac{\omega_0}{\omega_n}\right]^2}} \tag{9.5.5.6}$$

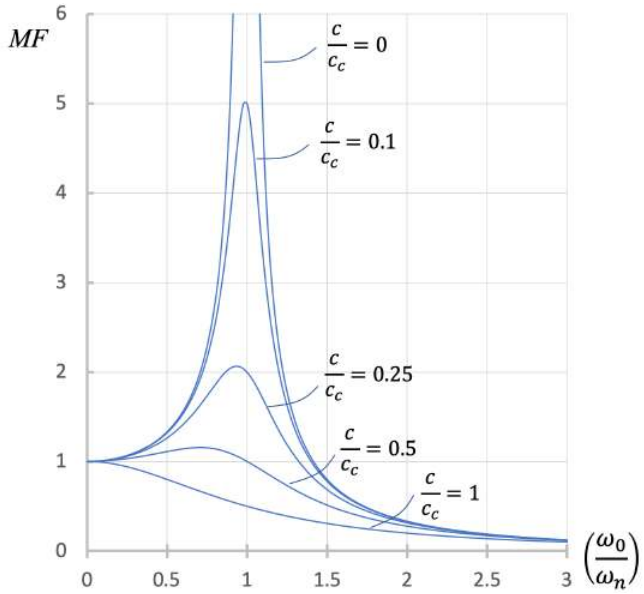


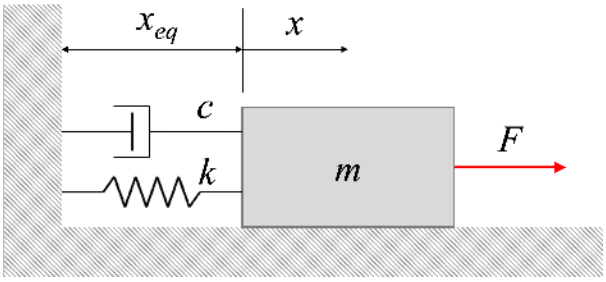
Figure 9.5.5.2: This figure shows the various magnification factors associated with different levels of (under)damping.

From the figure above, we can see that the extreme amplitudes at resonance only occur when the damping ratio = 0 and the ratio of frequencies is 1. Otherwise, damping inhibits the out-of-control vibrations that would otherwise be seen at resonance.

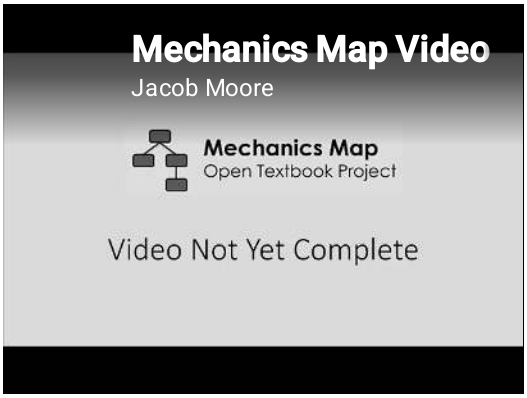
Worked Problems:

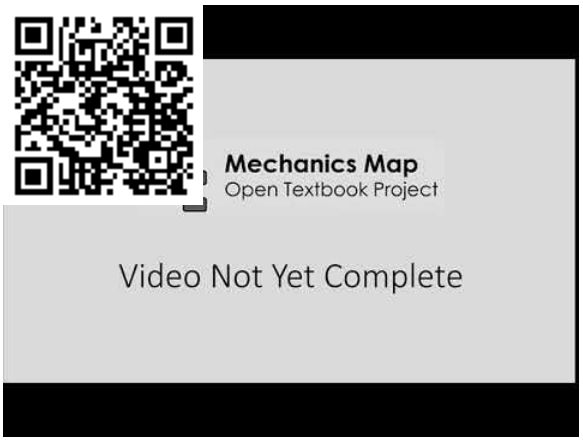
Question 1:

You are in a van that steadily accelerates from 20 m/s to 35 m/s over the course of 10 seconds. What is the rate of acceleration.



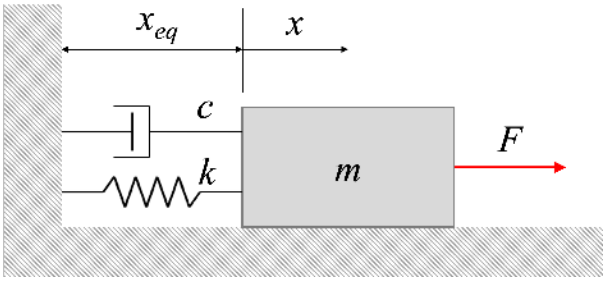
Solution:



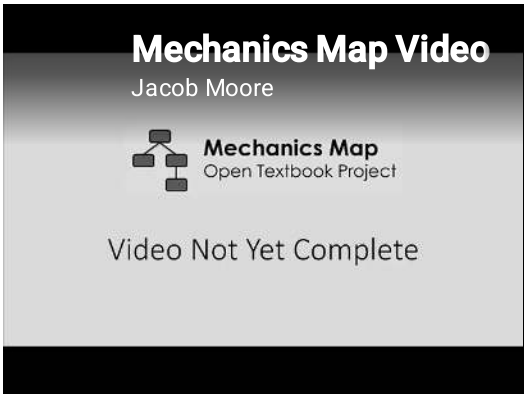


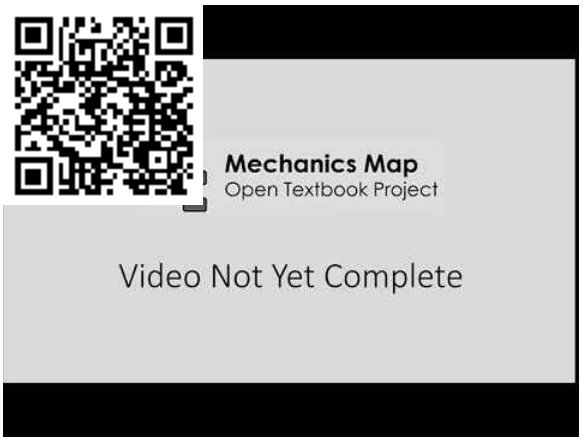
Question 2:

You are in a van that steadily accelerates from 20 m/s to 35 m/s over the course of 10 seconds. How far did you travel in meters over the course of those ten seconds.



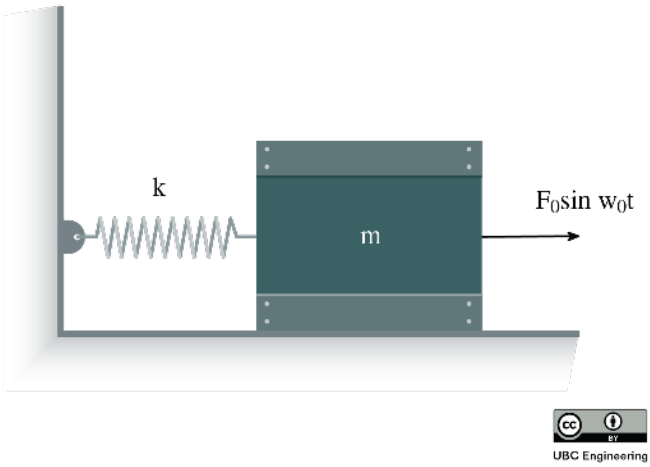
Solution:



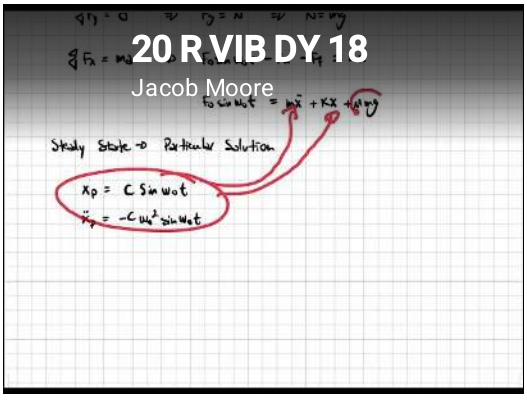


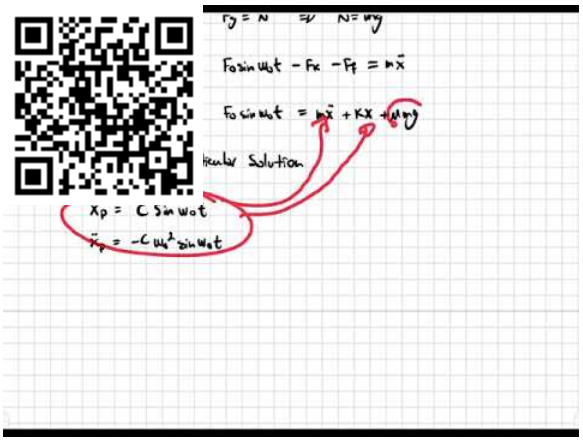
Question 3:

A periodic force F=5sin(3t) is applied to a 5kg load, which is connected to a 10 N/m spring. Given that the floor's coefficient of friction is μ=0.5, what is the maximum amplitude of the steady state function?



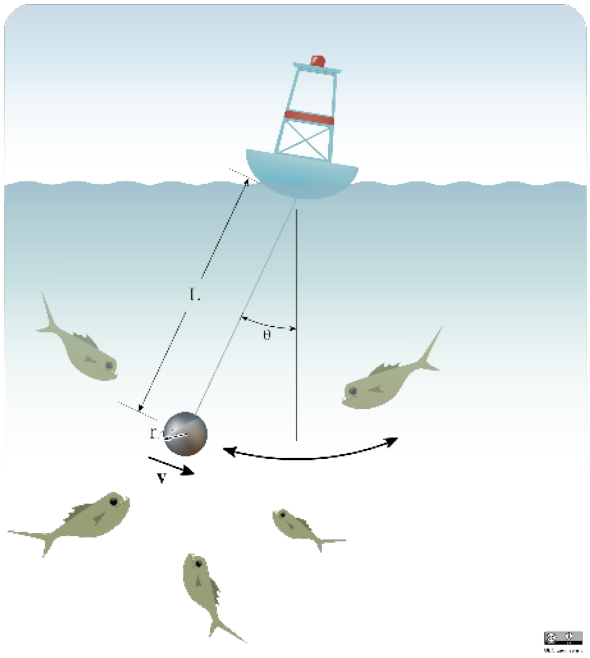
Solution:



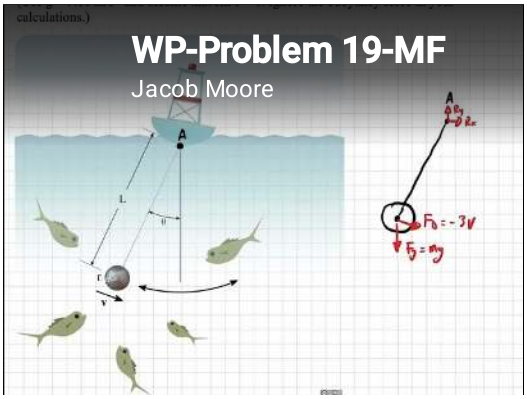


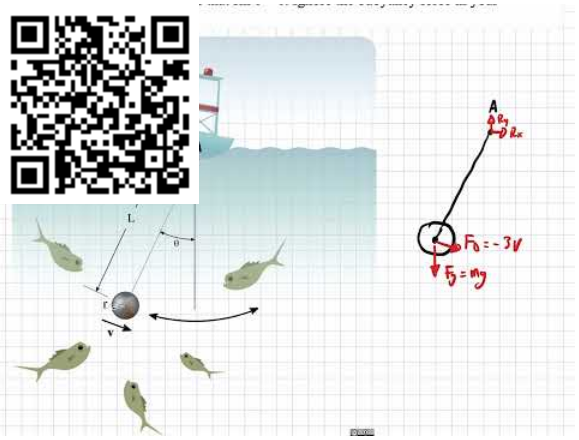
Question 4:

A new fish species discovered in the Fraser River has been observed to be attracted to swinging pendulums as depicted above. You want to observe this phenomenon, so you take a spherical metal ball of mass m = 0.585 kg attached to a rope of length L = 1.5 m and tie it to a buoy as shown. If the drag force of the ball is modelled by $F_d = -3v$, what radius must the ball be for oscillation to occur? (Use g = 9.81 m/s^2 and assume that sin θ = θ. Ignore the buoyancy force in your calculations.)



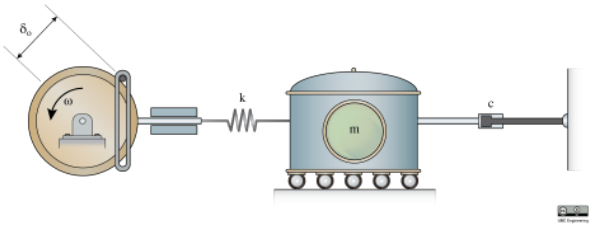
Solution:



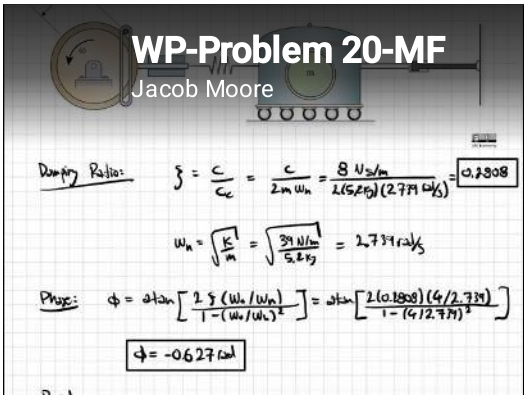


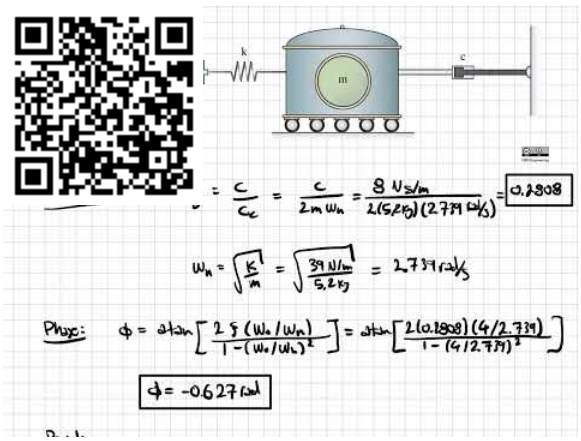
Question 5:

Your latest invention is a milkshake maker that uses vibrational movement to create the perfect milkshake. You start by adding all the frozen ingredients to the milkshake maker and you can approximate it as a uniform, solid container. The milk shaker and all the ingredients inside have combined mass of m = 5.2 kg. It is connected to a damper with damping constant c = 8 N·s/m on one side, and a spring of stiffness k = 39 N/m on the other. A rotating wheel causes periodic motion to keep the milkshake shaking where $\delta 0$ = 41 cm and the angular velocity is ω = 4 rad/s. Find the damping ratio ζ, the phase angle φ' of the steady state solution, the natural period of oscillation τn, the period of the steady state response $\tau 0$, and the period of the damped vibration τd.



Solution:





This page titled 9.5.5: Viscous Damped Harmonic Forced Vibrations is shared under a CC BY-SA 4.0 license and was authored, remixed, and/or curated by Jacob Moore & Contributors (Mechanics Map) via source content that was edited to the style and standards of the LibreTexts platform.

- **15.5: Viscous Damped Harmonic Forced Vibrations** by Jacob Moore & Contributors is licensed CC BY-SA 4.0. Original source: http://mechanicsmap.psu.edu.

SECTION OVERVIEW

9.6: Chapter 6 Supplementary Reading Materials

Resources developed by authors external to NIU with full prose and background materials:

Topic hierarchy

9.6.1: Introduction to the Frequency Domain

9.6.2: Complex Fourier Series

9.6.3: Classic Fourier Series

9.6.4: Fourier Series Approximation of Signals

9.6.5: Fourier Transform

9.6.6: Sampling and Reconstruction

- 9.6.6.1: Signal Sampling
- 9.6.6.2: Sampling Theorem
- 9.6.6.3: Signal Reconstruction
- 9.6.6.4: Perfect Reconstruction
- 9.6.6.5: Aliasing Phenomena
- 9.6.6.6: Anti-Aliasing Filters
- 9.6.6.7: Discrete Time Processing of Continuous Time Signals

9.6: Chapter 6 Supplementary Reading Materials is shared under a not declared license and was authored, remixed, and/or curated by LibreTexts.

9.6.1: Introduction to the Frequency Domain

Learning Objectives

- This module serves as an introduction to working in the frequency domain and thinking of signals in terms of their spectral components.
- The Fourier transform can be used to represent any signal in terms of frequency instead of time and facilitates the computation of the transfer function of a system.

In developing ways of analyzing linear circuits, we invented the impedance method because it made solving circuits easier. Along the way, we developed the notion of a circuit's frequency response or transfer function. This notion, which also applies to all linear, time-invariant systems, describes how the circuit responds to a sinusoidal input when we express it in terms of a complex exponential. We also learned the Superposition Principle for linear systems: The system's output to an input consisting of a sum of two signals is the sum of the system's outputs to each individual component.

The study of the frequency domain combines these two notions--a system's sinusoidal response is easy to find and a linear system's output to a sum of inputs is the sum of the individual outputs--to develop the crucial idea of a signal's **spectrum**. We begin by finding that those signals that can be represented as a sum of sinusoids is very large. In fact, **all signals can be expressed as a superposition of sinusoids**.

As this story unfolds, we'll see that information systems rely heavily on spectral ideas. For example, radio, television, and cellular telephones transmit over different portions of the spectrum. In fact, spectrum is so important that communications systems are regulated as to which portions of the spectrum they can use by the Federal Communications Commission in the United States and by International Treaty for the world (see Frequency Allocations). Calculating the spectrum is easy: The **Fourier transform** defines how we can find a signal's spectrum.

This page titled 9.6.1: Introduction to the Frequency Domain is shared under a CC BY 4.0 license and was authored, remixed, and/or curated by Don H. Johnson (Connections) via source content that was edited to the style and standards of the LibreTexts platform.

- **4.1: Introduction to the Frequency Domain** by Don H. Johnson is licensed CC BY 1.0. Original source: https://cnx.org/contents/d442r0wh9.72:g9deOnx519.

9.6.2: Complex Fourier Series

Learning Objectives

- Definition of the complex Fourier series.

In an earlier module, we showed that a square wave could be expressed as a superposition of pulses. As useful as this decomposition was in this example, it does not generalize well to other periodic signals: How can a superposition of pulses equal a smooth signal like a sinusoid? Because of the importance of sinusoids to linear systems, you might wonder whether they could be added together to represent a large number of periodic signals. You would be right and in good company as well. Euler and Gauss in particular worried about this problem, and Jean Baptiste Fourier got the credit even though tough mathematical issues were not settled until later. They worked on what is now known as the **Fourier series**: representing **any** periodic signal as a superposition of sinusoids.

But the Fourier series goes well beyond being another signal decomposition method. Rather, the Fourier series begins our journey to appreciate how a signal can be described in either the time-domain or the frequency-domain with **no** compromise.

Let **s(t)** be a **periodic** signal with period **T**. We want to show that periodic signals, even those that have constant-valued segments like a square wave, can be expressed as sum of **harmonically** related sine waves: sinusoids having frequencies that are integer multiples of the **fundamental frequency**. Because the signal has period **T**, the fundamental frequency is **1/T**. The complex Fourier series expresses the signal as a superposition of complex exponentials having frequencies:

$$\frac{k}{T}, k = \{...., -1, 0, 1, ...\}$$

$$s(t) = \sum_{k=-\infty}^{\infty} c_k e^{i\frac{2\pi kt}{T}}$$

with

$$c_k = \frac{1}{2}\left(a_k - ib_k\right)$$

The real and imaginary parts of the **Fourier coefficients c_k** are written in this unusual way for convenience in defining the classic Fourier series. The zeroth coefficient equals the signal's average value and is real- valued for real-valued signals: $c_0 = a_0$. The family of functions

$$\left\{e^{i\frac{2\pi kt}{T}}\right\}$$

are called **basis functions** and form the foundation of the Fourier series. No matter what the periodic signal might be, these functions are always present and form the representation's building blocks. They depend on the signal period **T**, and are indexed by **k**.

Note

Assuming we know the period, knowing the Fourier coefficients is equivalent to knowing the signal. Thus, it makes no difference if we have a time-domain or a frequency- domain characterization of the signal.

Exercise 9.6.2.1

What is the complex Fourier series for a sinusoid?

Solution

Because of Euler's relation,

$$\sin(2\pi ft) = \frac{1}{2i}e^{i2\pi ft} - \frac{1}{2i}e^{-(i2\pi ft)}$$

Thus,

$$c_1 = \frac{1}{2i},\ c_{-1} = -\frac{1}{2i}$$

and the other coefficients are zero.

To find the Fourier coefficients, we note the orthogonality property

$$\int_0^T e^{i\frac{2\pi kt}{T}} e^{(-i)\frac{2\pi lt}{T}} dt = \begin{cases} T & \text{if } k = l \\ 0 & \text{if } k \neq l \end{cases}$$

Assuming for the moment that the complex Fourier series "works," we can find a signal's complex Fourier coefficients, its **spectrum**, by exploiting the orthogonality properties of harmonically related complex exponentials.

Simply multiply each side of the Fourier Series equation by

$$e^{(-i2\pi lt)}$$

and integrate over the interval **[0,T]**.

$$c_k = \frac{1}{T}\int_0^T s(t) e^{-(i\frac{2\pi kt}{T})} dt$$

$$c_0 = \frac{1}{T}\int_0^T s(t) dt$$

Example 9.6.2.1:

Finding the Fourier series coefficients for the square wave $\mathbf{sq_T(t)}$ is very simple. Mathematically, this signal can be expressed as

$$sq_T(t) = \begin{cases} 1 & \text{if } 0 < t < \frac{T}{2} \\ -1 & \text{if } \frac{T}{2} < t < T \end{cases}$$

The expression for the Fourier coefficients has the form

$$c_k = \frac{1}{T}\int_0^{\frac{T}{2}} e^{-(i\frac{2\pi kt}{T})} dt - \frac{1}{T}\int_{\frac{T}{2}}^T e^{-(i\frac{2\pi kt}{T})} dt$$

When integrating an expression containing **i**, treat it just like any other constant.

The two integrals are very similar, one equaling the negative of the other. The final expression becomes

$$c_k = \frac{-2}{i2\pi k}\left((-1)^k - 1\right)$$

$$c_k = \begin{cases} \frac{2}{i\pi k} & \text{if } k = odd \\ 0 & \text{if } k = even \end{cases}$$

$$sq(t) = \sum_{k\in\{\ldots,-3,-1,1,3,\ldots\}} \frac{2}{i2\pi k} e^{(i)\frac{2\pi kt}{T}}$$

Consequently, the square wave equals a sum of complex exponentials, but only those having frequencies equal to odd multiples of the fundamental frequency **1/T**. The coefficients decay slowly as the frequency index **k** increases. This index corresponds to the **k-th** harmonic of the signal's period.

Properties of Fourier Series Spectrum

A signal's Fourier series spectrum $\mathbf{c_k}$ has interesting properties.

Property 1

If **s(t)** is real,

$$c_k = \overline{c_{-k}}$$

Real-valued periodic signals have conjugate-symmetric spectra.

This result follows from the integral that calculates the $\mathbf{c_k}$ from the signal. Furthermore, this result means that:

$$\Re(c_k) = \Re(c_{-k})$$

The real part of the Fourier coefficients for real-valued signals is even. Similarly,

$$\Im(c_k) = \Im(c_{-k})$$

The imaginary parts of the Fourier coefficients have odd symmetry. Consequently, if you are given the Fourier coefficients for positive indices and zero and are told the signal is real-valued, you can find the negative-indexed coefficients, hence the entire spectrum. This kind of symmetry,

$$c_k = \overline{c_{-k}}$$

is known as **conjugate symmetry**.

Property 2

If **s(-t) = s(t)**, which says the signal has even symmetry about the origin,

$$c_{-k} = c_k$$

Given the previous property for real-valued signals, the Fourier coefficients of even signals are real-valued. A real-valued Fourier expansion amounts to an expansion in terms of only cosines, which is the simplest example of an even signal.

Property 3

If **s(-t) = - s(t)**, which says the signal has odd symmetry,

$$c_{-k} = -c_k$$

Therefore, the Fourier coefficients are purely imaginary. The square wave is a great example of an odd-symmetric signal.

Property 4

The spectral coefficients for a periodic signal delayed by **τ**, **s(t-τ)**, are:

$$c_k e^{-\frac{i2\pi k\tau}{T}}$$

where $\mathbf{c_k}$ denotes the spectrum of **s(t)**.

Delaying a signal by **τ** seconds results in a spectrum having a **linear phase shift** of

$$-\frac{2\pi k\tau}{T}$$

in comparison to the spectrum of the undelayed signal. Note that the spectral magnitude is unaffected. Showing this property is easy.

$$\frac{1}{T}\int_0^T s(t-\tau)e^{(-i)\frac{2\pi kt}{T}}dt = \frac{1}{T}\int_{-\tau}^{T-\tau} s(t)e^{(-i)\frac{2\pi k(t+\tau)}{T}}dt$$

$$\therefore \quad \frac{1}{T}\int_0^T s(t-\tau)e^{(-i)\frac{2\pi kt}{T}}dt = \frac{1}{T}e^{(-i)\frac{2\pi kt}{T}}\int_{-\tau}^{T-\tau} s(t)e^{(-i)\frac{2\pi kt}{T}}dt$$

Note that the range of integration extends over a period of the integrand. Consequently, it should not matter how we integrate over a period, which means that:

$$\int_{-\tau}^{T-\tau}(\cdot)dt = \int_0^T(\cdot)dt$$

and we have our result.

The complex Fourier series obeys **Parseval's Theorem**, one of the most important results in signal analysis. This general mathematical result says you can calculate a signal's power in either the time domain or the frequency domain.

Parseval's Theorem

Average power calculated in the time domain equals the power calculated in the frequency domain.

$$\frac{1}{T}\int_0^T s^2(t)dt = \sum_{-\infty}^{\infty}(|c_k|)^2$$

This result is a (simpler) re-expression of how to calculate a signal's power than with the real-valued Fourier series expression for power.

Let's calculate the Fourier coefficients of the periodic pulse signal shown in Figure 4.2.1 below.

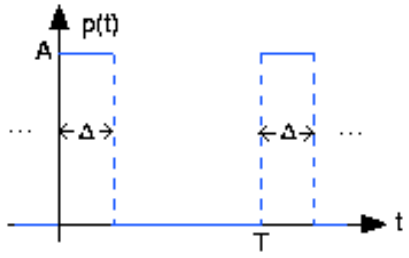


Figure 4.2.1 Periodic Pulse Signal

The pulse width is **Δ**, the period **T**, and the amplitude **A**. The complex Fourier spectrum of this signal is given by:

$$c_k = \frac{1}{T}\int_0^{\Delta} Ae^{-\frac{i2\pi kt}{T}}dt = -\left(\frac{A}{i2\pi k}\left(e^{-\frac{i2\pi\Delta}{T}}-1\right)\right)$$

At this point, simplifying this expression requires knowing an interesting property.

$$1-e^{-(i\theta)} = e^{-\frac{i\theta}{2}}\left(e^{\frac{i\theta}{2}}-e^{-\frac{i\theta}{2}}\right) = e^{-\frac{i\theta}{2}}2i\sin\left(\frac{\theta}{2}\right)$$

Armed with this result, we can simply express the Fourier series coefficients for our pulse sequence.

$$c_k = Ae^{-\frac{i\pi k\Delta}{T}}\frac{\sin\left(\frac{\pi k\Delta}{T}\right)}{\pi k}$$

Because this signal is real-valued, we find that the coefficients do indeed have conjugate symmetry:

$$c_k = \overline{c_{-k}}$$

The periodic pulse signal has neither even nor odd symmetry; consequently, no additional symmetry exists in the spectrum. Because the spectrum is complex valued, to plot it we need to calculate its magnitude and phase.

$$|c_k| = A\left|\frac{\sin\left(\frac{\pi k\Delta}{T}\right)}{\pi k}\right|$$

$$\angle c_k = -\frac{\pi k\Delta}{T} + \pi\, neg\left|\frac{\sin\left(\frac{\pi k\Delta}{T}\right)}{\pi k}\right| sign(k)$$

The function **neg(•)** equals -1 if its argument is negative and zero otherwise. The somewhat complicated expression for the phase results because the sine term can be negative; magnitudes must be positive, leaving the occasional negative values to be accounted for as a phase shift of Math input error

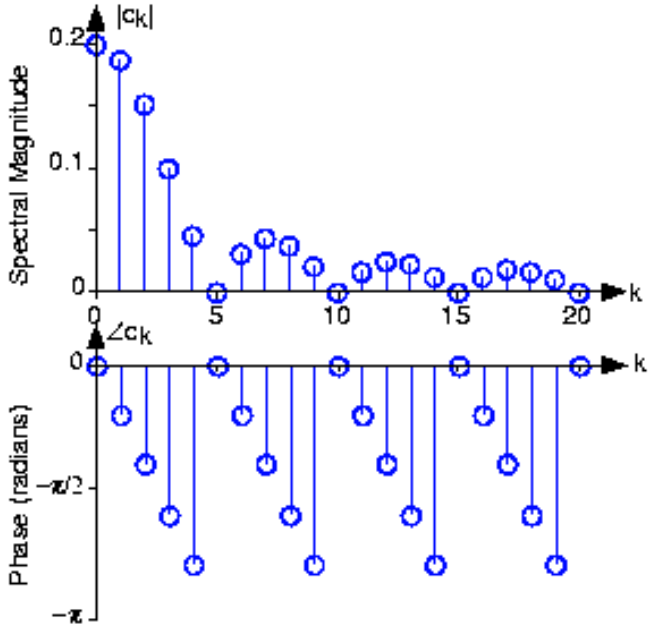


Figure 4.2.2 Periodic Pulse Sequence

The magnitude and phase of the periodic pulse sequence's spectrum is shown for positive-frequency indices. Here ***Δ/T = 0.2*** *and* ***A = 1***

Also note the presence of a linear phase term (the first term in $\angle c_k$ is proportional to frequency **k/T**). Comparing this term with that predicted from delaying a signal, a delay of **Δ/2** is present in our signal. Advancing the signal by this amount centers the pulse about the origin, leaving an even signal, which in turn means that its spectrum is real-valued. Thus, our calculated spectrum is consistent with the properties of the Fourier spectrum.

? Exercise 9.6.2.1

What is the value of c_0? Recalling that this spectral coefficient corresponds to the signal's average value, does your answer make sense?

Solution

$$c_0 = \frac{A\Delta}{T}$$

This quantity clearly corresponds to the periodic pulse signal's average value.

The phase plot shown in Figure 4.2.2 requires some explanation as it does not seem to agree with what the equation for $\angle c_k$ suggests. There, the phase has a linear component, with a jump of Math input error every time the sinusoidal term changes sign. We must realize that any integer multiple of **2**Math input error can be added to a phase at each frequency **without** affecting the value of the complex spectrum. We see that at frequency index **4** the phase is nearly -Math input error. The phase at index **5** is undefined because the magnitude is zero in this example. At index **6**, the formula suggests that the phase of the linear term should be less than -Math input error(more negative). In addition, we expect a shift of -Math input error in the phase between indices **4 and 6**. Thus, the phase value predicted by the formula is a little less than -**(2**Math input error**).** Because we can add **2**Math input error without affecting the value of the spectrum at index **6**, the result is a slightly negative number as shown. Thus, the formula and the plot do agree. In phase calculations like those made in MATLAB, values are usually confined to the range **[-**Math input error**,**Math input error**]** by adding some (possibly negative) multiple of **2**Math input error to each phase value.

This page titled 9.6.2: Complex Fourier Series is shared under a CC BY 4.0 license and was authored, remixed, and/or curated by Don H. Johnson (Connections) via source content that was edited to the style and standards of the LibreTexts platform.

- **4.2: Complex Fourier Series** by Don H. Johnson is licensed CC BY 1.0. Original source: https://cnx.org/contents/d442r0wh9.72:g9deOnx519.

9.6.3: Classic Fourier Series

Learning Objectives

- Signals can be composed by a superposition of an infinite number of sine and cosine functions.
- The coefficients of the superposition depend on the signal being represented and are equivalent to knowing the function itself.

The classic Fourier series as derived originally expressed a periodic signal (period **T**) in terms of harmonically related sines and cosines.

$$s(t) = a_0 + \sum_{k=1}^{\infty} a_k \cos\left(\frac{2\pi kt}{T}\right) + \sum_{k=1}^{\infty} b_k \sin\left(\frac{2\pi kt}{T}\right)$$

The complex Fourier series and the sine-cosine series are identical, each representing a signal's spectrum. The **Fourier coefficients**, $\mathbf{a_k}$ and $\mathbf{b_k}$, express the real and imaginary parts respectively of the spectrum while the coefficients $\mathbf{c_k}$ of the complex Fourier series express the spectrum as a magnitude and phase. Equating the classic Fourier series to the complex Fourier series, an extra factor of two and complex conjugate become necessary to relate the Fourier coefficients in each.

$$c_k = \frac{1}{2}(a_k - ib_k)$$

Exercise 9.6.3.1

Derive this relationship between the coefficients of the two Fourier series.

Solution

Write the coefficients of the complex Fourier series in Cartesian form as:

$$c_k = A_k + iB_k$$

Now substitute into the expression for the complex Fourier series.

$$\sum_{k=-\infty}^{\infty} c_k e^{i\frac{2\pi kt}{T}} = \sum_{k=-\infty}^{\infty} (A_k + iB_k)\, e^{i\frac{2\pi kt}{T}}$$

Simplifying each term in the sum using Euler's formula,

$$(A_k + iB_k)\, e^{i\frac{2\pi kt}{T}} = (A_k + iB_k)\left(\cos\left(\frac{2\pi kt}{T}\right) + i\sin\left(\frac{2\pi kt}{T}\right)\right)$$

$$(A_k + iB_k)\, e^{i\frac{2\pi kt}{T}} = A_k \cos\left(\frac{2\pi kt}{T}\right) - B_k \sin\left(\frac{2\pi kt}{T}\right) + i\left(A_k \sin\left(\frac{2\pi kt}{T}\right) + B_k \cos\left(\frac{2\pi kt}{T}\right)\right)$$

We now combine terms that have the same frequency index **in magnitude**. Because the signal is real-valued, the coefficients of the complex Fourier series have conjugate symmetry:

$$c_{-k} = \overline{c_k} \text{ or } A_{-k} = \overline{A_k} \text{ and } B_{-k} = \overline{B_k}$$

After we add the positive-indexed and negative-indexed terms, each term in the Fourier series becomes:

$$2A_k \cos\left(\frac{2\pi kt}{T}\right) - 2B_k \sin\left(\frac{2\pi kt}{T}\right)$$

To obtain the classic Fourier series, we must have:

$$2A_k = a_k \text{ and } 2B_k = -b_k$$

Just as with the complex Fourier series, we can find the Fourier coefficients using the **orthogonality** properties of sinusoids. Note that the cosine and sine of harmonically related frequencies, even the **same** frequency, are orthogonal.

$$\forall k,l,k \in \mathbb{Z} l \in \mathbb{Z} : \left(\int_0^T \sin\left(\frac{2\pi kt}{T}\right) \cos\left(\frac{2\pi lt}{T}\right) dt = 0 \right)$$

$$\int_0^T \sin\left(\frac{2\pi kt}{T}\right) \sin\left(\frac{2\pi lt}{T}\right) dt = \begin{cases} \frac{T}{2} & \text{if } (k=l) \wedge (k \neq 0) \wedge (l \neq 0) \\ 0 & \text{if } (k \neq l) \vee (k = 0 = l) \end{cases}$$

$$\int_0^T \cos\left(\frac{2\pi kt}{T}\right) \cos\left(\frac{2\pi lt}{T}\right) dt = \begin{cases} \frac{T}{2} & \text{if } (k=l) \wedge (k \neq 0) \wedge (l \neq 0) \\ T & \text{if } k = 0 = l \\ 0 & \text{if } k \neq l \end{cases}$$

These orthogonality relations follow from the following important trigonometric identities.

$$\sin(\alpha)\sin(\beta) = \frac{1}{2}\left(\cos(\alpha-\beta) - \cos(\alpha+\beta)\right)$$

$$\cos(\alpha)\cos(\beta) = \frac{1}{2}\left(\cos(\alpha+\beta) + \cos(\alpha-\beta)\right)$$

$$\sin(\alpha)\cos(\beta) = \frac{1}{2}\left(\sin(\alpha+\beta) + \sin(\alpha-\beta)\right)$$

These identities allow you to substitute a sum of sines and/or cosines for a product of them. Each term in the sum can be integrated by noticing one of two important properties of sinusoids.

- The integral of a sinusoid over an **integer** number of periods equals zero.
- The integral of the **square** of a unit-amplitude sinusoid over a period **T** equals **T/2**.

To use these, let's, for example, multiply the Fourier series for a signal by the cosine of the **l**$^{\textbf{th}}$ harmonic:

$$\cos\left(\frac{2\pi lt}{T}\right)$$

and integrate.

The idea is that, because integration is linear, the integration will sift out all but the term involving $\mathbf{a_l}$.

$$\int_0^T s(t)\cos\left(\frac{2\pi lt}{T}\right) dt = \int_0^T a_0 \cos\left(\frac{2\pi lt}{T}\right) dt + \sum_{k=1}^{\infty} a_k \int_0^T \cos\left(\frac{2\pi kt}{T}\right)\cos\left(\frac{2\pi lt}{T}\right) dt + \sum_{k=1}^{\infty} b_k \int_0^T \sin\left(\frac{2\pi kt}{T}\right)\cos\left(\frac{2\pi lt}{T}\right) dt$$

The first and third terms are zero; in the second, the only non-zero term in the sum results when the indices **k** and **l** are equal (but not zero), in which case we obtain:

$$\frac{a_1 T}{2}$$

$$If \;\; k = 0 = l \;\; a_0 T \; is \; obtained$$

Consequently,

$$\forall l, l \neq 0 : \left(a_l = \frac{2}{T} \int_0^T s(t)\cos\left(\frac{2\pi lt}{T}\right) dt \right)$$

All of the Fourier coefficients can be found similarly.

$$a_0 = \frac{2}{T}\int_0^T s(t)dt$$

$$\forall k, k \neq 0 : \left(a_k = \frac{2}{T} \int_0^T s(t)\cos\left(\frac{2\pi kt}{T}\right) dt \right)$$

$$b_k = \frac{2}{T}\int_0^T s(t)\sin\left(\frac{2\pi kt}{T}\right) dt$$

? Exercise 9.6.3.1

The expression for $\mathbf{a_0}$ is referred to as the **average value** of **s(t)**. Why?

Solution

The average of a set of numbers is the sum divided by the number of terms. Viewing signal integration as the limit of a Riemann sum, the integral corresponds to the average.

? Exercise 9.6.3.1

What is the Fourier series for a unit-amplitude square wave?

Solution

We found that the complex Fourier series coefficients are given by

$$c_k = \frac{2}{i\pi k}$$

The coefficients are pure imaginary, which means $\mathbf{a_k}$**=0**. The coefficients of the sine terms are given by:

$$b_k = -\left(2\Im(c_k)\right)$$

so that:

$$b_k = \begin{cases} \frac{4}{\pi k} & \text{if } k \; odd \\ 0 & \text{if } k \; even \end{cases}$$

Thus, the Fourier series for the square wave is

$$sq(t) = \sum_{k \in \{1,3,\ldots\}} \frac{4}{\pi k} \sin\left(\frac{2\pi k t}{T}\right)$$

✓ Example 9.6.3.1:

Let's find the Fourier series representation for the half-wave rectified sinusoid.

$$s(t) = \begin{cases} \sin\frac{2\pi t}{T} & \text{if } 0 \le t \le \frac{T}{2} \\ 0 & \text{if } \frac{T}{2} \le t < T \end{cases}$$

Begin with the sine terms in the series; to find $\mathbf{b_k}$ we must calculate the integral

$$b_k = \int_0^{\frac{T}{2}} \sin\left(\frac{2\pi t}{T}\right) \sin\left(\frac{2\pi k t}{T}\right) dt$$

Using our trigonometric identities turns our integral of a product of sinusoids into a sum of integrals of individual sinusoids, which are much easier to evaluate.

$$\int_0^{\frac{T}{2}} \sin\left(\frac{2\pi t}{T}\right) \sin\left(\frac{2\pi k t}{T}\right) dt = \frac{1}{2} \int_0^{\frac{T}{2}} \cos\left(\frac{2\pi(k-1)t}{T}\right) - \cos\left(\frac{2\pi(k+1)t}{T}\right) dt$$

$$\int_0^{\frac{T}{2}} \sin\left(\frac{2\pi t}{T}\right) \sin\left(\frac{2\pi k t}{T}\right) dt = \begin{cases} \frac{1}{2} & \text{if } k = 1 \\ 0 & \text{if } otherwise \end{cases}$$

Thus,

$$b_1 = \frac{1}{2}$$

$$b_2 = b_3 = \ldots = 0$$

On to the cosine terms. The average value, which corresponds to $\mathbf{a_0}$, equals $\mathbf{1/\pi}$. The remainder of the cosine coefficients are easy to find, but yield the complicated result:

$$a_k = \begin{cases} -\left(\frac{2}{\pi}\frac{1}{k^2-1}\right) & \text{if } k \in \{2,4,\ldots\} \\ 0 & \text{if } k \; odd \end{cases}$$

Thus, the Fourier series for the half-wave rectified sinusoid has non-zero terms for the average, the fundamental, and the even harmonics.

This page titled 9.6.3: Classic Fourier Series is shared under a CC BY 4.0 license and was authored, remixed, and/or curated by Don H. Johnson (Connections) via source content that was edited to the style and standards of the LibreTexts platform.

- **4.3: Classic Fourier Series** by Don H. Johnson is licensed CC BY 1.0. Original source: https://cnx.org/contents/d442r0wh9.72:g9deOnx519.

9.6.4: Fourier Series Approximation of Signals

Learning Objectives

- Fourier series approximations.

It is interesting to consider the sequence of signals that we obtain as we incorporate more terms into the Fourier series approximation of the half-wave rectified sine wave. Define $\mathbf{s_K(t)}$ to be the signal containing **K+1** Fourier terms.

$$s_K(t) = a_0 + \sum_{k=1}^{K} a_k \cos\left(\frac{2\pi kt}{T}\right) + \sum_{k=1}^{K} b_k \sin\left(\frac{2\pi kt}{T}\right)$$

Figure 4.5.1 below shows how this sequence of signals portrays the signal more accurately as more terms are added.

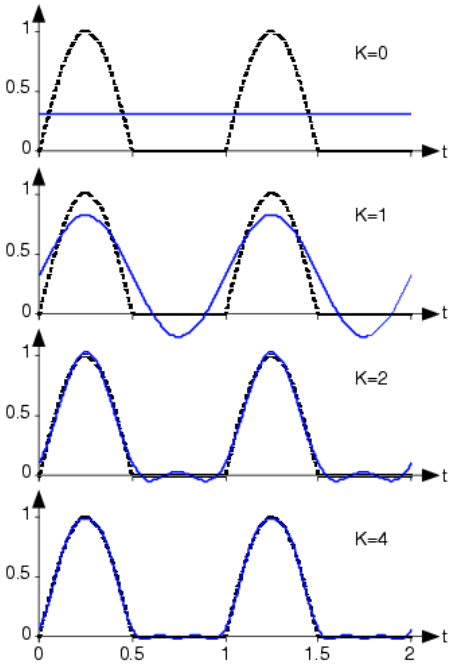


Figure 4.5.1 The Fourier series spectrum of a half-wave rectified sinusoid is shown in the upper portion.

*The index indicates the multiple of the fundamental frequency at which the signal has energy. The cumulative effect of adding terms to the Fourier series for the half-wave rectified sine wave is shown in the bottom portion. The dashed line is the actual signal, with the solid line showing the finite series approximation to the indicated number of terms, **K+1**.*

We need to assess quantitatively the accuracy of the Fourier series approximation so that we can judge how rapidly the series approaches the signal. When we use a

$$\varepsilon_K(t) = \sum_{k=K+1}^{\infty} a_k \cos\left(\frac{2\pi kt}{T}\right) + \sum_{k=K+1}^{\infty} b_k \sin\left(\frac{2\pi kt}{T}\right)$$

To find the rms error, we must square this expression and integrate it over a period. Again, the integral of most cross-terms is zero, leaving

$$rms(\varepsilon_K) = \sqrt{\frac{1}{2}\sum_{k=K+1}^{\infty} a_k^2 + b_k^2}$$

Figure 4.5.2 shows how the error in the Fourier series for the half-wave rectified sinusoid decreases as more terms are incorporated. In particular, the use of four terms, as shown in the bottom plot of Figure 4.5.1 has a rms error (relative to the rms value of the signal) of about 3%. The Fourier series in this case converges quickly to the signal.

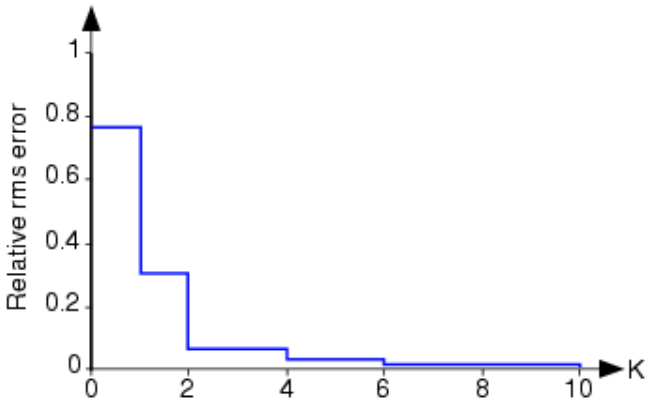


Figure4.5.2 Approximation error for a half-wave rectified sinusoid

The rms error calculated according to the equation above is shown as a function of the number of terms in the series for the half-wave rectified sinusoid. The error has been normalized by the rms value of the signal.

We can look at Figure 4.5.3 to see the power spectrum and the rms approximation error for the square wave.

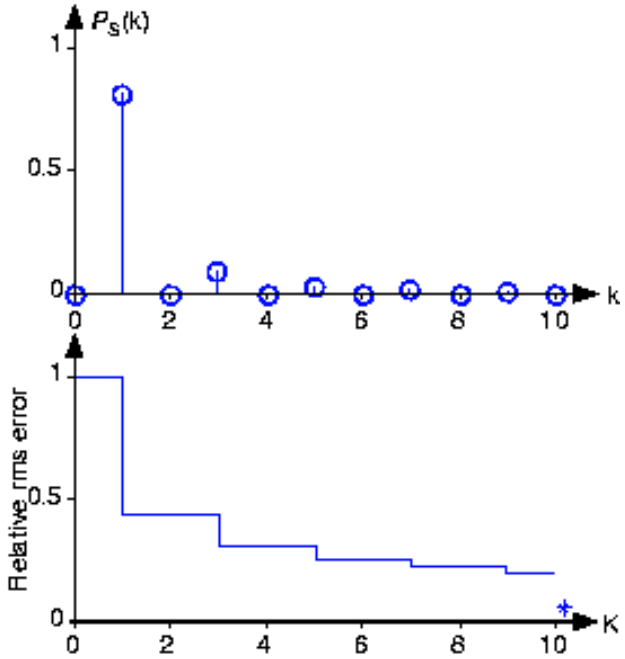


Figure 4.5.3 The upper plot shows the power spectrum of the square wave, and the lower plot the rms error of the finite-length Fourier series approximation to the square wave. The asterisk denotes the rms error when the number of terms **K** in the Fourier series equals 99.

Because the Fourier coefficients decay more slowly here than for the half-wave rectified sinusoid, the rms error is not decreasing quickly. Said another way, the square-wave's spectrum contains more power at higher frequencies than does the half-wave-rectified sinusoid. This difference between the two Fourier series results because the half-wave rectified sinusoid's Fourier coefficients are proportional to $\mathbf{1/k^2}$ while those of the square wave are proportional to **1/k**. If fact, after 99 terms of the square wave's approximation, the error is bigger than 10 terms of the approximation for the half-wave rectified sinusoid. Mathematicians have shown that no signal has an rms approximation error that decays more slowly than it does for the square wave.

? Exercise 9.6.4.1

Calculate the harmonic distortion for the square wave.

Solution

Total harmonic distortion in the square wave is:

$$1 - \frac{1}{2}(\frac{4}{\pi})^2 = 20\%$$

More than just decaying slowly, Fourier series approximation shown in Figure 4.5.4 exhibits interesting behavior.

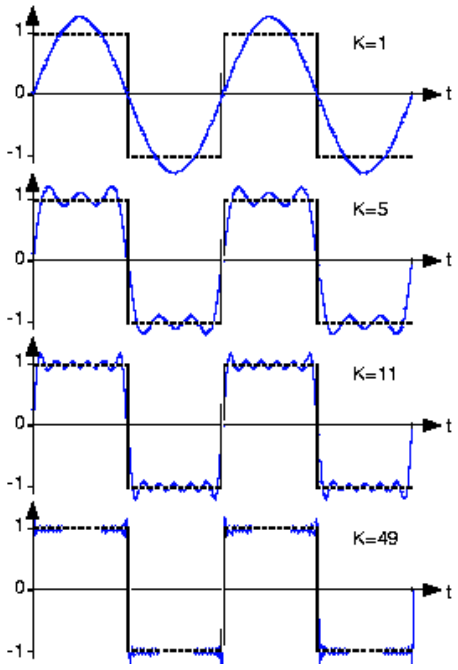


Figure 4.5.4 Fourier series approximation to **sq(t).** The number of terms in the Fourier sum is indicated in each plot, and the square wave is shown as a dashed line over two periods.

Although the square wave's Fourier series requires more terms for a given representation accuracy, when comparing plots it is not clear that the two are equal. Does the Fourier series really equal the square wave at **all** values of

Consider this mathematical question intuitively: Can a discontinuous function, like the square wave, be expressed as a sum, even an infinite one, of continuous signals? One should at least be suspicious, and in fact, it can't be thus expressed. This issue brought Fourier much criticism from the French Academy of Science (Laplace, Lagrange, Monge and LaCroix comprised the review committee) for several years after its presentation on 1807. It was not resolved for almost a century, and its resolution is interesting and important to understand from a practical viewpoint.

The extraneous peaks in the square wave's Fourier series **never** disappear; they are termed **Gibb's phenomenon** after the American physicist Josiah Willard Gibbs. They occur whenever the signal is discontinuous, and will always be present whenever the signal has jumps.

Let's return to the question of equality; how can the equal sign in the definition of the Fourier series be justified? The partial answer is that **pointwise**—each and every value of **t**—equality is **not** guaranteed. However, mathematicians later in the nineteenth century showed that the rms error of the Fourier series was always zero.

$$\lim_{k \to \infty} rms(\varepsilon_K) = 0$$

What this means is that the error between a signal and its Fourier series approximation may not be zero, but that its rms value will be zero! It is through the eyes of the rms value that we redefine equality: The usual definition of equality is called **pointwise equality**: Two signals $\mathbf{s_1(t)}$, $\mathbf{s_2(t)}$ are said to be equal pointwise if:

$$s_1(t) = s_2(t)$$

for all values of **t**. A new definition of equality is **mean-square equality**: Two signals are said to be equal in the mean square if:

$$rms(s_1 - s_2) = 0$$

For Fourier series, Gibb's phenomenon peaks have finite height and zero width. The error differs from zero only at isolated points —whenever the periodic signal contains discontinuities—and equals about 9% of the size of the discontinuity. The value of a

function at a finite set of points does not affect its integral. This effect underlies the reason why defining the value of a discontinuous function, like we refrained from doing in defining the step function, at its discontinuity is meaningless. Whatever you pick for a value has no practical relevance for either the signal's spectrum or for how a system responds to the signal. The Fourier series value "at" the discontinuity is the average of the values on either side of the jump.

This page titled 9.6.4: Fourier Series Approximation of Signals is shared under a CC BY 4.0 license and was authored, remixed, and/or curated by Don H. Johnson (Connections) via source content that was edited to the style and standards of the LibreTexts platform.

- **4.5: Fourier Series Approximation of Signals** by Don H. Johnson is licensed CC BY 1.0. Original source: https://cnx.org/contents/d442r0wh9.72:g9deOnx519.

9.6.5: Fourier Transform

The Fourier transform is the underlying principle for frequency-domain description of signals. We begin with the Fourier series. Consider a signal $f(t)$ continuous on the time interval $[0, T]$, which then repeats with period T off to negative and positive infinity. It can be shown that

$$f(t) = A_o + \sum_{k=1}^{\infty} [A_n \cos(n\omega_o t) + B_n \sin(n\omega_o t)] \qquad \text{in which:} \tag{9.6.5.1}$$

$$\omega_o = 2\pi/T,$$

$$A_0 = \frac{1}{T} \int_0^T f(t)\, dt,$$

$$A_n = \frac{2}{T} \int_0^T f(t) \cos(n\omega_o t)\, dt,$$

$$B_n = \frac{2}{T} \int_0^T f(t) \sin(n\omega_o t)\, dt.$$

This says that the time-domain signal $f(t)$ has an exact (if you carry all the infinity of terms) representation of a constant plus scaled cosines and sines. As we will see later, the impact of this second, frequency-domain representation is profound, as it allows an entirely new set of tools for manipulation and analysis of signals and systems. A compact form of these expressions for the Fourier series can be written using complex exponentials:

$$f(t) = \sum_{n=-\infty}^{\infty} C_n e^{in\omega_o t}, \tag{9.6.5.2}$$

$$\text{where } C_n = \frac{1}{T} \int_0^T f(t) e^{-in\omega_o t}\, dt.$$

Of course, C_n can be a complex number. In making these inner product calculations, orthogonality of the harmonic functions is useful:

$$\int_0^{2\pi} sin(nt)\, sin(mt)\, dt = 0, \text{ for } n \geq 1,\ m \geq 1,\ n \neq m$$

$$\int_0^{2\pi} cos(nt)\, cos(mt)\, dt = 0, \text{ for } n \geq 1,\ m \geq 1,\ n \neq m$$

$$\int_0^{2\pi} sin(nt)\, cos(mt)\, dt = 0, \text{ for } n \geq 1,\ m \geq 1.$$

Now let's go to a different class of signal, one that is not periodic, but has a finite integral of absolute value. Obviously, such a signal has to approach zero at distances far from the origin. We can write a more elegant transformation:

$$F(\omega) = \int_{-\infty}^{\infty} f(t) e^{-i\omega t}\, dt, \tag{9.6.5.3}$$

$$f(t) = \frac{1}{2\pi} \int_{-\infty}^{\infty} F(\omega) e^{i\omega\tau}\, d\tau. \tag{9.6.5.4}$$

This is the real Fourier transform: a time-domain signal is transformed into a (complex) frequency-domain version, and it can be transformed back. On working it through, we see that derivatives and integrals look this way through the transform:

$$f(t) \longleftrightarrow F(\omega) \tag{9.6.5.5}$$

$$\frac{d^n f(t)}{dt^n} \longleftrightarrow (i\omega)^n F(\omega) \tag{9.6.5.6}$$

$$\int_{-\infty}^{t} f(\tau)\, d\tau \longleftrightarrow \frac{1}{i\omega} F(\omega). \tag{9.6.5.7}$$

Another very important property of the Fourier transform is Parseval's Relation:

$$\int_{-\infty}^{\infty} f^2(t)\, dt = \frac{1}{2\pi} \int_{-\infty}^{\infty} F(\omega) F^*(\omega)\, d\omega = \int_{-\infty}^{\infty} |F(\omega)|^2\, d\omega, \tag{9.6.5.8}$$

where the $*$-superscript indicates the complex conjugate. We will give more properties for the related Laplace transform in a later section. But as is, the Fourier transform is immediately useful for solving linear differential equations with constant coefficients (LTI systems):

$$mx'' + bx' + cx = u(t) \longleftrightarrow [-m\omega^2 + i\omega b + k]X(\omega) = U(\omega), \tag{9.6.5.9}$$

$$\text{so that } X(\omega) = \frac{1}{-m\omega^2 + i\omega b + k} U(\omega).$$

Hence, the action of the differential equation to relate $f(t)$ with $x(t)$ is, in the frequency domain, captured by the function

$$H(\omega) = \frac{1}{-m\omega^2 + i\omega b + k} \tag{9.6.5.10}$$

Putting two and two together, we then assert that $X(\omega) = H(\omega)U(\omega)$; in the Fourier space, the system response is the product of impulse response function, and the input! To back this up, we show now that convolution in the time-domain is equivalent to multiplication in the frequency domain:

$$\begin{aligned}
X(\omega) &= \mathcal{F}\left[\int_{-\infty}^{\infty} u(\tau)h(t-\tau)\, d\tau\right] \\
&= \int_{-\infty}^{\infty}\int_{-\infty}^{\infty} u(\tau)h(t-\tau)e^{-i\tau t}\, dt d\tau \\
&= \int_{-\infty}^{\infty}\int_{-\infty}^{\infty} u(\tau)h(\xi)e^{-i\omega(\xi+\tau)}\, d\xi d\tau \qquad \text{because } e^{-i\omega t} = e^{-i\omega(t-\tau+\tau)} \\
&= \int_{-\infty}^{\infty} e^{-i\omega\xi}h(\xi)\, d\xi \int_{-\infty}^{\infty} e^{-i\omega\tau}u(\tau)\, d\tau \\
&= H(\omega)U(\omega).
\end{aligned}$$

The central role of the impulse response should be reiterated here. It is a complete definition of the system, and for systems of differential equations, it is a specific function of the parameters and of the frequency ω. The Fourier Transform of the impulse response is called the system transfer function, and we often refer to the transfer function as "the system," even though it is actually a (transformed) signal.

By way of summary, we can write

$$y(t) = h(t) * u(t), \tag{9.6.5.11}$$

and

$$Y(\omega) = H(\omega)U(\omega). \tag{9.6.5.12}$$

This page titled 9.6.5: Fourier Transform is shared under a CC BY-NC-SA 4.0 license and was authored, remixed, and/or curated by Franz S. Hover & Michael S. Triantafyllou (MIT OpenCourseWare) via source content that was edited to the style and standards of the LibreTexts platform.

- **2.8: Fourier Transform** by Franz S. Hover & Michael S. Triantafyllou is licensed CC BY-NC-SA 4.0. Original source: https://ocw.mit.edu/courses/mechanical-engineering/2-017j-design-of-electromechanical-robotic-systems-fall-2009.

SECTION OVERVIEW

9.6.6: Sampling and Reconstruction

9.6.6.1: Signal Sampling

9.6.6.2: Sampling Theorem

9.6.6.3: Signal Reconstruction

9.6.6.4: Perfect Reconstruction

9.6.6.5: Aliasing Phenomena

9.6.6.6: Anti-Aliasing Filters

9.6.6.7: Discrete Time Processing of Continuous Time Signals

This page titled 9.6.6: Sampling and Reconstruction is shared under a CC BY 4.0 license and was authored, remixed, and/or curated by Richard Baraniuk et al. via source content that was edited to the style and standards of the LibreTexts platform.

9.6.6.1: Signal Sampling

Introduction

Digital computers can process discrete time signals using extremely flexible and powerful algorithms. However, most signals of interest are continuous time signals, which is how data almost always appears in nature. This module introduces the concepts behind converting continuous time signals into discrete time signals through a process called sampling.

Sampling

Sampling a continuous time signal produces a discrete time signal by selecting the values of the continuous time signal at evenly spaced points in time. Thus, sampling a continuous time signal x with sampling period T_s gives the discrete time signal x_s defined by $x_s(n) = x(nT_s)$. The sampling angular frequency is then given by $\omega_s = 2\pi/T_s$.

It should be intuitively clear that multiple continuous time signals sampled at the same rate can produce the same discrete time signal since uncountably many continuous time functions could be constructed that connect the points on the graph of any discrete time function. Thus, sampling at a given rate does not result in an injective relationship. Hence, sampling is, in general, not invertible.

Example 9.6.6.1.1

For instance, consider the signals x,y defined by

$$x(t) = \frac{\sin(t)}{t}$$

$$y(t) = \frac{\sin(5t)}{t}$$

and their sampled versions x_S,y_s with sampling period $T_s = \pi/2$

$$x_s(n) = \frac{\sin(n\pi/2)}{n\pi/2}$$

$$y_s(n) = \frac{\sin(n5\pi/2)}{n\pi/2}.$$

Notice that since

$$\sin(n5\pi/2) = \sin(n2\pi + n\pi/2) = \sin(n\pi/2)$$

it follows that

$$y_s(n) = \frac{\sin(n\pi/2)}{n\pi/2} = x_s(n).$$

Hence, x and y provide an example of distinct functions with the same sampled versions at a specific sampling rate.

It is also useful to consider the relationship between the frequency domain representations of the continuous time function and its sampled versions. Consider a signal x sampled with sampling period T_s to produce the discrete time signal $x_s(n) = x(nT_s)$. The spectrum $X_s(\omega)$ for $\omega \in [-\pi, \pi)$ of x_s is given by

$$X_s(\omega) = \sum_{n=-\infty}^{\infty} x\left(nT_s\right) e^{-j\omega n}.$$

Using the continuous time Fourier transform, $x(tT_s)$ can be represented as

$$x\left(tT_s\right) = \frac{1}{2\pi T_s} \int_{-\infty}^{\infty} X\left(\frac{\omega_1}{T_s}\right) e^{j\omega_1 t} d\omega_1.$$

Thus, the unit sampling period version of $x(tT_s)$, which is $x(nT_s)$ can be represented as

$$x\left(nT_s\right) = \frac{1}{2\pi T_s}\int_{-\infty}^{\infty} X\left(\frac{\omega_1}{T_s}\right) e^{j\omega_1 n} d\omega_1.$$

This is algebraically equivalent to the representation

$$x\left(nT_s\right) = \frac{1}{T_s}\sum_{k=-\infty}^{\infty}\frac{1}{2\pi}\int_{-\pi}^{\pi} X\left(\frac{\omega_1 - 2\pi k}{T_s}\right) e^{j(\omega_1 - 2\pi k)n} d\omega_1,$$

which reduces by periodicity of complex exponentials to

$$x\left(nT_s\right) = \frac{1}{T_s}\sum_{k=-\infty}^{\infty}\frac{1}{2\pi}\int_{-\pi}^{\pi} X\left(\frac{\omega_1 - 2\pi k}{T_s}\right) e^{j\omega_1 n} d\omega_1.$$

Hence, it follows that

$$X_s(\omega) = \frac{1}{T_s}\sum_{k=-\infty}^{\infty}\sum_{n=-\infty}^{\infty}\left(\int_{-\pi}^{\pi} X\left(\frac{\omega_1 - 2\pi k}{T_s}\right) e^{j\omega_1 n} d\omega_1\right) e^{-j\omega n}.$$

Noting that the above expression contains a Fourier series and inverse Fourier series pair, it follows that

$$X_s(\omega) = \frac{1}{T_s}\sum_{k=-\infty}^{\infty} X\left(\frac{\omega - 2\pi k}{T_s}\right).$$

Hence, the spectrum of the sampled signal is, intuitively, the scaled sum of an infinite number of shifted and time scaled copies of original signal spectrum. Aliasing, which will be discussed in depth in later modules, occurs when these shifted spectrum copies overlap and sum together. Note that when the original signal x is bandlimited to $(-\pi/T_s, \pi/T_s)$ no overlap occurs, so each period of the sampled signal spectrum has the same form as the original signal spectrum. This suggest that if we sample a bandlimited signal at a sufficiently high sampling rate, we can recover it from its samples as will be further described in the modules on the Nyquist-Shannon sampling theorem and on perfect reconstruction.

Sampling Summary

Sampling a continuous time signal produces a discrete time signal by selecting the values of the continuous time signal at equally spaced points in time. However, we have shown that this relationship is not injective as multiple continuous time signals can be sampled at the same rate to produce the same discrete time signal. This is related to a phenomenon called aliasing which will be discussed in later modules. Consequently, the sampling process is not, in general, invertible. Nevertheless, as will be shown in the module concerning reconstruction, the continuous time signal can be recovered from its sampled version if some additional assumptions hold.

This page titled 9.6.6.1: Signal Sampling is shared under a CC BY 4.0 license and was authored, remixed, and/or curated by Richard Baraniuk et al. via source content that was edited to the style and standards of the LibreTexts platform.

- **10.1: Signal Sampling** by Richard Baraniuk et al. is licensed CC BY 3.0. Original source: https://cnx.org/contents/d2CEAGW515.4:l-e_wSqM24/Signal-Classifications-and-Properties.

9.6.6.2: Sampling Theorem

Introduction

With the introduction of the concept of signal sampling, which produces a discrete time signal by selecting the values of the continuous time signal at evenly spaced points in time, it is now possible to discuss one of the most important results in signal processing, the Nyquist-Shannon sampling theorem. Often simply called the sampling theorem, this theorem concerns signals, known as bandlimited signals, with spectra that are zero for all frequencies with absolute value greater than or equal to a certain level. The theorem implies that there is a sufficiently high sampling rate at which a bandlimited signal can be recovered exactly from its samples, which is an important step in the processing of continuous time signals using the tools of discrete time signal processing.

Nyquist-Shannon Sampling Theorem

Statement of the Sampling Theorem

The Nyquist-Shannon sampling theorem concerns signals with continuous time Fourier transforms that are only nonzero on the interval $(-B, B)$ for some constant B. Such a function is said to be bandlimited to $(-B, B)$. Essentially, the sampling theorem has already been implicitly introduced in the previous module concerning sampling. Given a continuous time signals x with continuous time Fourier transform X, recall that the spectrum X_s of sampled signal x_s with sampling period T_s is given by

$$X_s(\omega) = \frac{1}{T_s} \sum_{k=-\infty}^{\infty} X\left(\frac{\omega - 2\pi k}{T_s}\right).$$

It had previously been noted that if x is bandlimited to $(-\pi/T_s, \pi/T_s)$, the period of X_s centered about the origin has the same form as X scaled in frequency since no aliasing occurs. This is illustrated in Figure 9.6.6.2.1. Hence, if any two $(-\pi/T_s, \pi/T_s)$ bandlimited continuous time signals sampled to the same signal, they would have the same continuous time Fourier transform and thus be identical. Thus, for each discrete time signal there is a unique $(-\pi/T_s, \pi/T_s)$ bandlimited continuous time signal that samples to the discrete time signal with sampling period T_s. Therefore, this $(-\pi/T_s, \pi/T_s)$ bandlimited signal can be found from the samples by inverting this bijection.

This is the essence of the sampling theorem. More formally, the sampling theorem states the following. If a signal x is bandlimited to $(-B, B)$, it is completely determined by its samples with sampling rate $\omega_s = 2B$. That is to say, x can be reconstructed exactly from its samples x_s with sampling rate $\omega_s = 2B$. The angular frequency $2B$ is often called the angular Nyquist rate. Equivalently, this can be stated in terms of the sampling period $T_s = 2\pi/\omega_s$. If a signal x is bandlimited to $(-B, B)$, it is completely determined by its samples with sampling period $T_s = \pi/B$. That is to say, x can be reconstructed exactly from its samples x_s with sampling period T_s.

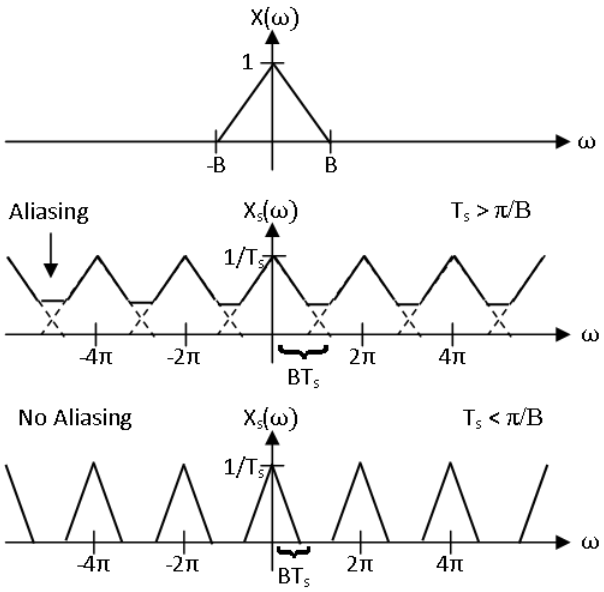


Figure 9.6.6.2.1: The spectrum of a bandlimited signals is shown as well as the spectra of its samples at rates above and below the Nyquist frequency. As is shown, no aliasing occurs above the Nyquist frequency, and the period of the samples spectrum centered about the origin has the same form as the spectrum of the original signal scaled in frequency. Below the Nyquist frequency, aliasing can occur and causes the spectrum to take a different than the original spectrum.

Proof of the Sampling Theorem

The above discussion has already shown the sampling theorem in an informal and intuitive way that could easily be refined into a formal proof. However, the original proof of the sampling theorem, which will be given here, provides the interesting observation that the samples of a signal with period T_s provide Fourier series coefficients for the original signal spectrum on $(-\pi/T_s, \pi/T_s)$.

Let x be a $(-\pi/T_s, \pi/T_s)$ bandlimited signal and x_s be its samples with sampling period T_s. We can represent x in terms of its spectrum X using the inverse continuous time Fourier transform and the fact that x is bandlimited. The result is

$$x(t) = \frac{1}{2\pi} \int_{-\pi/T_x}^{\pi/T_s} X(\omega) e^{j\omega t} d\omega$$

This representation of x may then be sampled with sampling period T_s to produce

$$x_s(n) = x_s(nT_s) = \frac{1}{2\pi} \int_{-\pi/T_s}^{\pi/T_s} X(\omega) e^{j\omega n T_s} d\omega$$

Noticing that this indicates that $x_s(n)$ is the nth continuous time Fourier series coefficient for $X(\omega)$ on the interval $(-\pi/T_s, \pi/T_s)$, it is shown that the samples determine the original spectrum $X(\omega)$ and, by extension, the original signal itself.

Perfect Reconstruction

Another way to show the sampling theorem is to derive the reconstruction formula that gives the original signal $\tilde{x} = x$ from its samples x_s with sampling period T_s, provided x is bandlimited to $(-\pi/T_s, \pi/T_s)$. This is done in the module on perfect reconstruction. However, the result, known as the Whittaker-Shannon reconstruction formula, will be stated here. If the requisite conditions hold, then the perfect reconstruction is given by

$$x(t) = \sum_{n=-\infty}^{\infty} x_s(n) \operatorname{sinc}(t/T_s - n)$$

where the sinc function is defined as

$$\operatorname{sinc}(t) = \frac{\sin(\pi t)}{\pi t}.$$

From this, it is clear that the set

$$\{\text{sinc}\,(t/T_s - n) \mid n \in \mathbb{Z}\}$$

forms an orthogonal basis for the set of $(-\pi/T_s, \pi/T_s)$ bandlimited signals, where the coefficients of a $(-\pi/T_s, \pi/T_s)$ signal in this basis are its samples with sampling period T_s.

Practical Implications

Discrete Time Processing of Continuous Time Signals

The Nyquist-Shannon Sampling Theorem and the Whittaker-Shannon Reconstruction formula enable discrete time processing of continuous time signals. Because any linear time invariant filter performs a multiplication in the frequency domain, the result of applying a linear time invariant filter to a bandlimited signal is an output signal with the same bandlimit. Since sampling a bandlimited continuous time signal above the Nyquist rate produces a discrete time signal with a spectrum of the same form as the original spectrum, a discrete time filter could modify the samples spectrum and perfectly reconstruct the output to produce the same result as a continuous time filter. This allows the use of digital computing power and flexibility to be leveraged in continuous time signal processing as well. This is more thoroughly described in the final module of this chapter.

Psychoacoustics

The properties of human physiology and psychology often inform design choices in technologies meant for interaction with people. For instance, digital devices dealing with sound use sampling rates related to the frequency range of human vocalizations and the frequency range of human auditory sensitivity. Because most of the sounds in human speech concentrate most of their signal energy between 5 Hz and 4 kHz, most telephone systems discard frequencies above 4 kHz and sample at a rate of 8 kHz. Discarding the frequencies greater than or equal to 4 kHz through use of an anti-aliasing filter is important to avoid aliasing, which would negatively impact the quality of the output sound as is described in a later module. Similarly, human hearing is sensitive to frequencies between 20 Hz and 20 kHz. Therefore, sampling rates for general audio waveforms placed on CDs were chosen to be greater than 40 kHz, and all frequency content greater than or equal to some level is discarded. The particular value that was chosen, 44.1 kHz, was selected for other reasons, but the sampling theorem and the range of human hearing provided a lower bound for the range of choices.

Sampling Theorem Summary

The Nyquist-Shannon Sampling Theorem states that a signal bandlimited to $(-\pi/T_s, \pi/T_s)$ can be reconstructed exactly from its samples with sampling period T_s. The Whittaker-Shannon interpolation formula, which will be further described in the section on perfect reconstruction, provides the reconstruction of the unique $(-\pi/T_s, \pi/T_s)$ bandlimited continuous time signal that samples to a given discrete time signal with sampling period T_s. This enables discrete time processing of continuous time signals, which has many powerful applications.

This page titled 9.6.6.2: Sampling Theorem is shared under a CC BY 4.0 license and was authored, remixed, and/or curated by Richard Baraniuk et al. via source content that was edited to the style and standards of the LibreTexts platform.

- **10.2: Sampling Theorem** by Richard Baraniuk et al. is licensed CC BY 3.0. Original source: https://cnx.org/contents/d2CEAGW515.4:l-e_wSqM24/Signal-Classifications-and-Properties.

9.6.6.3: Signal Reconstruction

Introduction

The sampling process produces a discrete time signal from a continuous time signal by examining the value of the continuous time signal at equally spaced points in time. Reconstruction, also known as interpolation, attempts to perform an opposite process that produces a continuous time signal coinciding with the points of the discrete time signal. Because the sampling process for general sets of signals is not invertible, there are numerous possible reconstructions from a given discrete time signal, each of which would sample to that signal at the appropriate sampling rate. This module will introduce some of these reconstruction schemes.

Reconstruction

Reconstruction Process

The process of reconstruction, also commonly known as interpolation, produces a continuous time signal that would sample to a given discrete time signal at a specific sampling rate. Reconstruction can be mathematically understood by first generating a continuous time impulse train

$$x_{imp}(t) = \sum_{n=-\infty}^{\infty} x_s(n)\delta\left(t - nT_s\right)$$

from the sampled signal x_s with sampling period T_s and then applying a lowpass filter G that satisfies certain conditions to produce an output signal $\tilde{x}$. If G has impulse response g, then the result of the reconstruction process, illustrated in Figure 9.6.6.3.1, is given by the following computation, the final equation of which is used to perform reconstruction in practice.

$$\begin{aligned}\tilde{x}(t) &= \left(x_{imp} * g\right)(t) \\ &= \int_{-\infty}^{\infty} x_{imp}(\tau)g(t-\tau)d\tau \\ &= \int_{-\infty}^{\infty} \sum_{n=-\infty}^{\infty} x_s(n)\delta\left(\tau - nT_s\right) g(t-\tau)d\tau \\ &= \sum_{n=-\infty}^{\infty} x_s(n) \int_{-\infty}^{\infty} \delta\left(\tau - nT_s\right) g(t-\tau)d\tau \\ &= \sum_{n=-\infty}^{\infty} x_s(n)g\left(t - nT_s\right) \end{aligned} \tag{9.6.6.3.1}$$

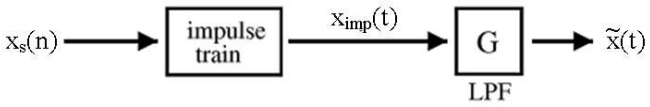


Figure 9.6.6.3.1: Block diagram of reconstruction process for a given lowpass filter G.

Reconstruction Filters

In order to guarantee that the reconstructed signal $\tilde{x}$ samples to the discrete time signal x_s from which it was reconstructed using the sampling period T_s, the lowpass filter G must satisfy certain conditions. These can be expressed well in the time domain in terms of a condition on the impulse response g of the lowpass filter G. The sufficient condition to be a reconstruction filters that we will require is that, for all $n \in \mathbb{Z}$,

$$g\left(nT_s\right) = \begin{cases} 1 & n = 0 \\ 0 & n \neq 0 \end{cases} = \delta(n)\ .$$

This means that gg sampled at a rate T_s produces a discrete time unit impulse signal. Therefore, it follows that sampling $\tilde{x}$ with sampling period T_s results in

$$\begin{aligned}\tilde{x}(nT_s) &= \sum_{m=-\infty}^{\infty} x_s(m)g(nT_s - mT_s) \\ &= \sum_{m=-\infty}^{\infty} x_s(m)g((n-m)T_s) \\ &= \sum_{m=-\infty}^{\infty} x_s(m)\delta(n-m) \\ &= x_s(n), \end{aligned} \tag{9.6.6.3.2}$$

which is the desired result for reconstruction filters.

Cardinal Basis Splines

Since there are many continuous time signals that sample to a given discrete time signal, additional constraints are required in order to identify a particular one of these. For instance, we might require our reconstruction to yield a spline of a certain degree, which is a signal described in piecewise parts by polynomials not exceeding that degree. Additionally, we might want to guarantee that the function and a certain number of its derivatives are continuous.

This may be accomplished by restricting the result to the span of sets of certain splines, called **basis splines** or **B-splines**. Specifically, if a nth degree spline with continuous derivatives up to at least order $n-1$ is required, then the desired function for a given T_s belongs to the span of $\{B_n(t/T_s - k) \mid k \in \mathbb{Z}\}$ where

$$B_n = B_0 * B_{n-1}$$

for $n \geq 1$ and

$$B_0(t) = \begin{cases} 1 & -1/2 < t < 1/2 \\ 0 & \text{otherwise} \end{cases}.$$

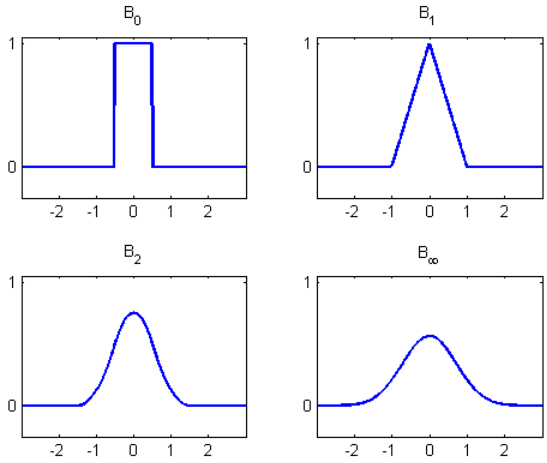


Figure 9.6.6.3.2: The basis splines B_n are shown in the above plots. Note that, except for the order 0 and order 1 functions, these functions do not satisfy the conditions to be reconstruction filters. Also notice that as the order increases, the functions approach the Gaussian function, which is exactly B_∞.

However, the basis splines B_n do not satisfy the conditions to be a reconstruction filter for $n \geq 2$ as is shown in Figure 9.6.6.3.2. Still, the B_n are useful in defining the cardinal basis splines, which do satisfy the conditions to be reconstruction filters. If we let b_n be the samples of B_n on the integers, it turns out that b_n has an inverse b_n^{-1} with respect to the operation of convolution for each n. This is to say that $b_n^{-1} * b_n = \delta$. The cardinal basis spline of order nn for reconstruction with sampling period T_s is defined as

$$\eta_n(t) = \sum_{k=-\infty}^{\infty} b_n^{-1}(k) B_n(t/T_s - k).$$

In order to confirm that this satisfies the condition to be a reconstruction filter, note that

$$\eta_n(mT_s) = \sum_{k=-\infty}^{\infty} b_n^{-1}(k) B_n(m-k) = \left(b_n^{-1} * b_n\right)(m) = \delta(m).$$

Thus, η_n is a valid reconstruction filter. Since η_n is an nth degree spline with continuous derivatives up to order $n-1$, the result of the reconstruction will be a nth degree spline with continuous derivatives up to order $n-1$.

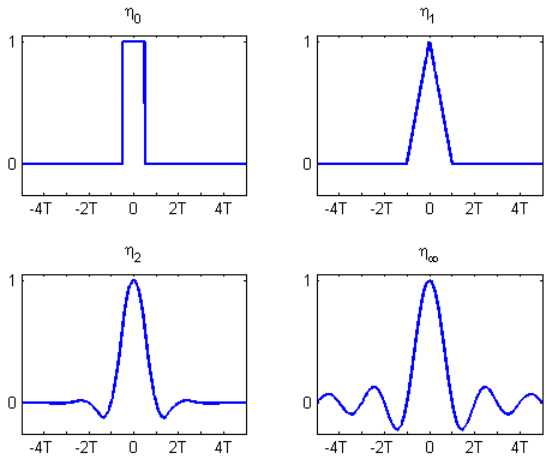


Figure 9.6.6.3.3: The above plots show cardinal basis spline functions η_0, η_1, η_2, and η_∞. Note that the functions satisfy the conditions to be reconstruction filters. Also, notice that as the order increases, the cardinal basis splines approximate the sinc function, which is exactly η_∞. Additionally, these filters are acausal.

The lowpass filter with impulse response equal to the cardinal basis spline η_0 of order 0 is one of the simplest examples of a reconstruction filter. It simply extends the value of the discrete time signal for half the sampling period to each side of every sample, producing a piecewise constant reconstruction. Thus, the result is discontinuous for all nonconstant discrete time signals.

Likewise, the lowpass filter with impulse response equal to the cardinal basis spline η_1 of order 1 is another of the simplest examples of a reconstruction filter. It simply joins the adjacent samples with a straight line, producing a piecewise linear reconstruction. In this way, the reconstruction is continuous for all possible discrete time signals. However, unless the samples are collinear, the result has discontinuous first derivatives.

In general, similar statements can be made for lowpass filters with impulse responses equal to cardinal basis splines of any order. Using the nth order cardinal basis spline η_n, the result is a piecewise degree nn polynomial. Furthermore, it has continuous derivatives up to at least order $n-1$. However, unless all samples are points on a polynomial of degree at most n, the derivative of order nn will be discontinuous.

Reconstructions of the discrete time signal given in Figure 9.6.6.3.4 using several of these filters are shown in Figure 9.6.6.3.5. As the order of the cardinal basis spline increases, notice that the reconstruction approaches that of the infinite order cardinal spline η_∞, the sinc function. As will be shown in the subsequent section on perfect reconstruction, the filters with impulse response equal to the sinc function play an especially important role in signal processing.

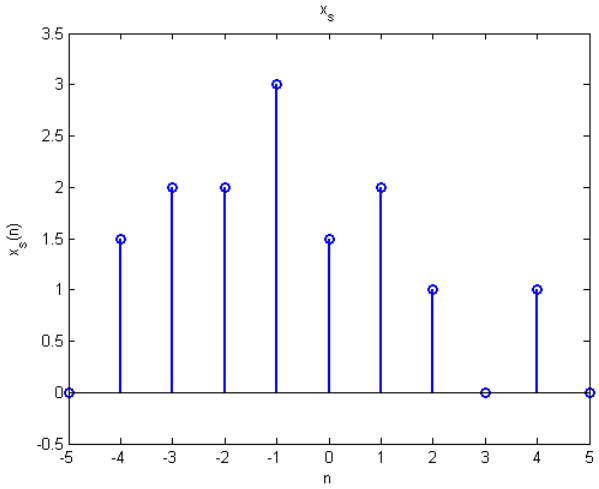


Figure 9.6.6.3.4: The above plot shows an example discrete time function. This discrete time function will be reconstructed using sampling period T_s using several cardinal basis splines in Figure 9.6.6.3.5.

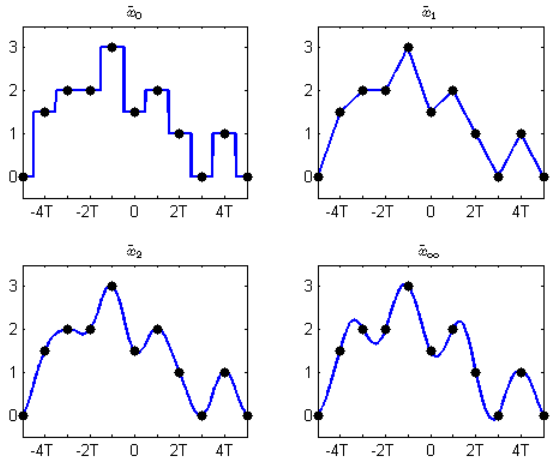


Figure 9.6.6.3.5: The above plots show interpolations of the discrete time signal given in Figure 9.6.6.3.4 using lowpass filters with impulse responses given by the cardinal basis splines shown in Figure 9.6.6.3.3. Notice that the interpolations become increasingly smooth and approach the sinc interpolation as the order increases.

Reconstruction Summary

Reconstruction of a continuous time signal from a discrete time signal can be accomplished through several schemes. However, it is important to note that reconstruction is not the inverse of sampling and only produces one possible continuous time signal that samples to a given discrete time signal. As is covered in the subsequent module, perfect reconstruction of a bandlimited continuous time signal from its sampled version is possible using the Whittaker-Shannon reconstruction formula, which makes use of the ideal lowpass filter and its sinc function impulse response, if the sampling rate is sufficiently high.

This page titled 9.6.6.3: Signal Reconstruction is shared under a CC BY 4.0 license and was authored, remixed, and/or curated by Richard Baraniuk et al. via source content that was edited to the style and standards of the LibreTexts platform.

- **10.3: Signal Reconstruction** by Richard Baraniuk et al. is licensed CC BY 3.0. Original source: https://cnx.org/contents/d2CEAGW515.4:l-e_wSqM24/Signal-Classifications-and-Properties.

9.6.6.4: Perfect Reconstruction

Introduction

If certain additional assumptions about the original signal and sampling rate hold, then the original signal can be recovered exactly from its samples using a particularly important type of filter. More specifically, it will be shown that if a bandlimited signal is sampled at a rate greater than twice its bandlimit, the Whittaker-Shannon reconstruction formula perfectly reconstructs the original signal. This formula makes use of the ideal lowpass filter, which is related to the sinc function. This is extremely useful, as sampled versions of continuous time signals can be filtered using discrete time signal processing, often in a computer. The results may then be reconstructed to produce the same continuous time output as some desired continuous time system.

Perfect Reconstruction

In order to understand the conditions for perfect reconstruction and the filter it employs, consider the following. As a beginning, a sufficient condition under which perfect reconstruction is possible will be discussed. Subsequently, the filter and process used for perfect reconstruction will be detailed.

Recall that the sampled version x_s of a continuous time signal x with sampling period T_s has a spectrum given by

$$X_s(\omega) = \frac{1}{T_s} \sum_{k=-\infty}^{\infty} X\left(\frac{\omega - 2\pi k}{T_s}\right).$$

As before, note that if x is bandlimited to $(-\pi/T_s, \pi/T_s)$, meaning that X is only nonzero on $(-\pi/T_s, \pi/T_s)$, then each period of X_s has the same form as X. Thus, we can identify the original spectrum X from the spectrum of the samples X_s and, by extension, the original signal x from its samples x_s at rate T_s if x is bandlimited to $(-\pi/T_s, \pi/T_s)$.

If a signal x is bandlimited to $(-B, B)$, then it is also bandlimited to $(-\pi/T_s, \pi/T_s)$ provided that $T_s < \pi/B$. Thus, if we ensure that x is sampled to x_s with sufficiently high sampling angular frequency $\omega_s = 2\pi/T_s > 2B$ and have a way of identifying the unique $(-\pi/T_s, \pi/T_s)$ bandlimited signal corresponding to a discrete time signal at sampling period T_s, then x_s can be used to reconstruct $\tilde{x} = x$ exactly. The frequency $2B$ is known as the angular Nyquist rate. Therefore, the condition that the sampling rate $\omega_s = 2\pi/T_s > 2B$ be greater than the Nyquist rate is a sufficient condition for perfect reconstruction to be possible.

The correct filter must also be known in order to perform perfect reconstruction. The ideal lowpass filter defined by $G(\omega) = T_s(u(\omega + \pi/T_s) - u(\omega - \pi/T_s))$, which is shown in Figure 9.6.6.4.1, removes all signal content not in the frequency range $(-\pi/T_s, \pi/T_s)$. Therefore, application of this filter to the impulse train $\sum_{n=-\infty}^{\infty} x_s(n)\delta(t - nT_s)$ results in an output bandlimited to $(-\pi/T_s, \pi/T_s)$.

We now only need to confirm that the impulse response g of the filter G satisfies our sufficient condition to be a reconstruction filter. The inverse Fourier transform of $G(\omega)$ is

$$g(t) = \operatorname{sinc}(t/T_s) = \begin{cases} 1 & t = 0 \\ \frac{\sin(\pi t/T_s)}{\pi t/T_s} & t \neq 0 \end{cases},$$

which is shown in Figure 9.6.6.4.6. Hence,

$$g(nT_s) = \operatorname{sinc}(n) = \begin{cases} 1 & n = 0 \\ \frac{\sin(\pi n)}{\pi n} & n \neq 0 \end{cases} = \begin{cases} 1 & n = 0 \\ 0 & n \neq 0 \end{cases} = \delta(n).$$

Therefore, the ideal lowpass filter G is a valid reconstruction filter. Since it is a valid reconstruction filter and always produces an output that is bandlimited to $(-\pi/T_s, \pi/T_s)$, this filter always produces the unique $(-\pi/T_s, \pi/T_s)$ bandlimited signal that samples to a given discrete time sequence at sampling period T_s when the impulse train $\sum_{n=-\infty}^{\infty} x_s(n)\delta(t - nT_s)$ is input.

Therefore, we can always reconstruct any $(-\pi/T_s, \pi/T_s)$ bandlimited signal from its samples at sampling period T_s by the formula

$$x(t) = \sum_{n=-\infty}^{\infty} x_s(n) \operatorname{sinc}(t/T_s - n).$$

This perfect reconstruction formula is known as the **Whittaker-Shannon interpolation formula** and is sometimes also called the **cardinal series**. In fact, the sinc function is the infinite order cardinal basis spline η_∞. Consequently, the set $\{\operatorname{sinc}(t/T_s - n) \mid n \in \mathbb{Z}\}$

forms a basis for the vector space of $(-\pi/T_s, \pi/T_s)$ bandlimited signals where the signal samples provide the corresponding coefficients. It is a simple exercise to show that this basis is, in fact, an orthogonal basis.

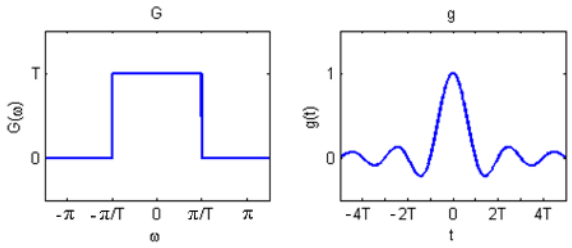


Figure 9.6.6.4.1: The above plots show the ideal lowpass filter and its inverse Fourier transform, the sinc function.

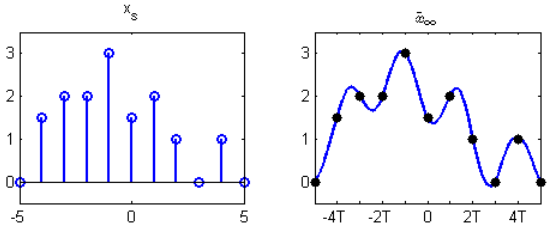


Figure 9.6.6.4.2: The plots show an example discrete time signal and its Whittaker-Shannon sinc reconstruction.

Perfect Reconstruction Summary

This module has shown that bandlimited continuous time signals can be reconstructed exactly from their samples provided that the sampling rate exceeds the Nyquist rate, which is twice the bandlimit. The Whittaker-Shannon reconstruction formula computes this perfect reconstruction using an ideal lowpass filter, with the resulting signal being a sum of shifted sinc functions that are scaled by the sample values. Sampling below the Nyquist rate can lead to aliasing which makes the original signal irrecoverable as is described in the subsequent module. The ability to perfectly reconstruct bandlimited signals has important practical implications for the processing of continuous time signals using the tools of discrete time signal processing.

This page titled 9.6.6.4: Perfect Reconstruction is shared under a CC BY 4.0 license and was authored, remixed, and/or curated by Richard Baraniuk et al. via source content that was edited to the style and standards of the LibreTexts platform.

- **10.4: Perfect Reconstruction** by Richard Baraniuk et al. is licensed CC BY 3.0. Original source: https://cnx.org/contents/d2CEAGW515.4:l-e_wSqM24/Signal-Classifications-and-Properties.

9.6.6.5: Aliasing Phenomena

Introduction

Through discussion of the Nyquist-Shannon sampling theorem and Whittaker-Shannon reconstruction formula, it has already been shown that a $(-B, B)$ continuous time signal can be reconstructed from its samples at rate $\omega_s = 2\pi/T_s$ via the sinc interpolation filter if $\omega_s > 2B$. Now, this module will investigate a problematic phenomenon, called aliasing, that can occur if this sufficient condition for perfect reconstruction does not hold. When aliasing occurs the spectrum of the samples has different form than the original signal spectrum, so the samples cannot be used to reconstruct the original signal through Whittaker-Shannon interpolation.

Aliasing

Aliasing occurs when each period of the spectrum of the samples does not have the same form as the spectrum of the original signal. Given a continuous time signals x with continuous time Fourier transform X, recall that the spectrum X_s of sampled signal x_s with sampling period T_s is given by

$$X_s(\omega) = \frac{1}{T_s} \sum_{k=-\infty}^{\infty} X\left(\frac{\omega - 2\pi k}{T_s}\right).$$

As has already been mentioned several times, if x is bandlimited to $(-\pi/T_s, \pi/T_s)$ then each period of X_s has the same form as X. However, if x is not bandlimited to $(-\pi/T_s, \pi/T_s)$, then the $X\left(\frac{\omega-2\pi k}{T_s}\right)$ can overlap and sum together. This is illustrated in Figure 9.6.6.5.1 in which sampling above the Nyquist frequency produces a samples spectrum of the same shape as the original signal, but sampling below the Nyquist frequency produces a samples spectrum with very different shape. Whittaker-Shannon interpolation of each of these sequences produces different results. The low frequencies not affected by the overlap are the same, but there is noise content in the higher frequencies caused by aliasing. Higher frequency energy masquerades as low energy content, a highly undesirable effect.

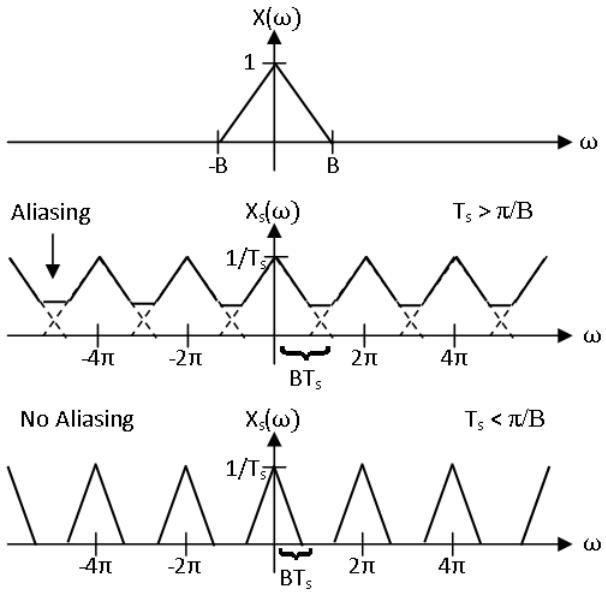


Figure 9.6.6.5.1: The spectrum of a bandlimited signals is shown as well as the spectra of its samples at rates above and below the Nyquist frequency. As is shown, no aliasing occurs above the Nyquist frequency, and the period of the samples spectrum centered about the origin has the same form as the spectrum of the original signal scaled in frequency. Below the Nyquist frequency, aliasing can occur and causes the spectrum to take a different than the original spectrum.

Unlike when sampling above the Nyquist frequency, sampling below the Nyquist frequency does not yield an injective (one-to-one) function from the $(-B, B)$ bandlimited continuous time signals to the discrete time signals. Any signal x with spectrum X which overlaps and sums to X_s samples to x_s. It should be intuitively clear that there are very many $(-B, B)$ bandlimited signals that sample to a given discrete time signal below the Nyquist frequency, as is demonstrated in Figure 9.6.6.5.2. It is quite easy to construct uncountably infinite families of such signals.

Aliasing obtains it name from the fact that multiple, in fact infinitely many, $(-B, B)$ bandlimited signals sample to the same discrete sequence if $\omega_s < 2B$. Thus, information about the original signal is lost in this noninvertible process, and these different signals effectively assume the same identity, an “alias”. Hence, under these conditions the Whittaker-Shannon interpolation formula will not produce a perfect reconstruction of the original signal but will instead give the unique $(-\omega_s/2, \omega_s/2)$ bandlimited signal that samples to the discrete sequence.

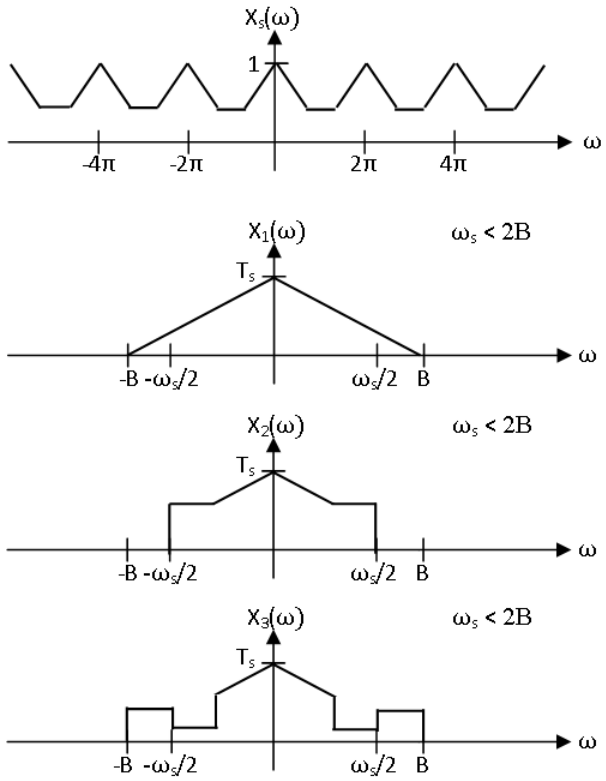


Figure 9.6.6.5.2: The spectrum of a discrete time signal x_s, taken from Figure 9.6.6.5.1, is shown along with the spectra of three $(-B, B)$ signals that sample to it at rate $\omega_s < 2B$. From the sampled signal alone, it is impossible to tell which, if any, of these was sampled at rate ω_s to produce x_s. In fact, there are infinitely many $(-B, B)$ bandlimited signals that sample to x_s at a sampling rate below the Nyquist rate.

Aliasing Demonstration

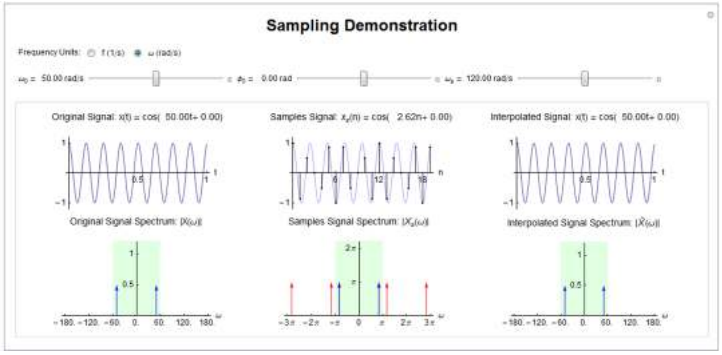


Figure 9.6.6.5.3: Interact (when online) with a Mathematica CDF demonstrating sampling and aliasing for a sinusoid. To Download, right-click and save target as .cdf.

Aliasing Summary

Aliasing, essentially the signal processing version of identity theft, occurs when each period of the spectrum of the samples does not have the same form as the spectrum of the original signal. As has been shown, there can be infinitely many $(-B, B)$ bandlimited signals that sample to a given discrete time signal x_s at a rate $\omega_s = 2\pi/T_s < 2B$ below the Nyquist frequency. However, there is a unique $(-B, B)$ bandlimited signal that samples to x_s, which is given by the Whittaker-Shannon interpolation of x_s, at rate $\omega_s \geq 2B$ as no aliasing occurs above the Nyquist frequency. Unfortunately, sufficiently high sampling rates cannot always be produced. Aliasing is detrimental to many signal processing applications, so in order to process continuous time signals using discrete time

tools, it is often necessary to find ways to avoid it other than increasing the sampling rate. Thus, anti-aliasing filters, are of practical importance.

This page titled 9.6.6.5: Aliasing Phenomena is shared under a CC BY 4.0 license and was authored, remixed, and/or curated by Richard Baraniuk et al. via source content that was edited to the style and standards of the LibreTexts platform.

- **10.5: Aliasing Phenomena** by Richard Baraniuk et al. is licensed CC BY 3.0. Original source: https://cnx.org/contents/d2CEAGW515.4:l-e_wSqM24/Signal-Classifications-and-Properties.

9.6.6.6: Anti-Aliasing Filters

Introduction

It has been shown that a $(-B, B)$ bandlimited signal can be perfectly reconstructed from its samples at a rate $\omega_s = 2\pi/T_s \geq B$. However, it is not always practically possible to produce sufficiently high sampling rates or to ensure that the input is bandlimited in real situations. Aliasing, which manifests itself as a difference in shape between the periods of the samples signal spectrum and the original spectrum, would occur without any further measures to correct this. Thus, it often becomes necessary to filter out signal energy at frequencies above $\omega_s/2$ in order to avoid the detrimental effects of aliasing. This is the role of the anti-aliasing filter, a lowpass filter applied before sampling to ensure that the signal is $(-\omega_s/2, \omega_s/2)$ bandlimited or at least nearly so.

Anti-Aliasing Filters

Aliasing can occur when a signal with energy at frequencies other that $(-B, B)$ is sampled at rate $\omega_s < 2B$. Thus, when sampling below the Nyquist frequency, it is desirable to remove as much signal energy outside the frequency range $(-B, B)$ as possible while keeping as much signal energy in the frequency range $(-B, B)$ as possible. This suggests that the ideal lowpass filter with cutoff frequency $\omega_s/2$ would be the optimal anti-aliasing filter to apply before sampling. While this is true, the ideal lowpass filter can only be approximated in real situations.

In order to demonstrate the importance of anti-aliasing filters, consider the calculation of the error energy between the original signal and its Whittaker-Shannon reconstruction from its samples taken with and without the use of an anti-aliasing filter. Let x be the original signal and $y = Gx$ be the anti-alias filtered signal where G is the ideal lowpass filter with cutoff frequency $\omega_s/2$. It is easy to show that the reconstructed spectrum using no anti-aliasing filter is given by

$$\widetilde{X}(\omega) = \begin{cases} T_s X_s\left(T_s \omega\right) & |\omega| < \omega_s/2 \\ 0 & \text{otherwise} \end{cases} = \begin{cases} \sum_{k=-\infty}^{\infty} X\left(\omega - k\omega_s\right) & |\omega| < \omega_s/2 \\ 0 & \text{otherwise} \end{cases} .$$

Thus, the reconstruction error spectrum for this case is

$$(X - \widetilde{X})(\omega) = \begin{cases} -\sum_{k=1}^{\infty} \left(X\left(\omega + k\omega_s\right) + X\left(\omega - k\omega_s\right)\right) & |\omega| < \omega_s/2 \\ X(\omega) & \text{otherwise} \end{cases} .$$

Similarly, the reconstructed spectrum using the ideal lowpass anti-aliasing filter is given by

$$\widetilde{Y}(\omega) = Y(\omega) = \begin{cases} X(\omega) & |\omega| < \omega_s/2 \\ 0 & \text{otherwise} \end{cases} .$$

Thus, the reconstruction error spectrum for this case is

$$(X - \widetilde{Y})(\omega) = \begin{cases} 0 & |\omega| < \omega_s/2 \\ X(\omega) & \text{otherwise} \end{cases} .$$

Hence, by Parseval's theorem, it follows that $\|x - \tilde{y}\| \leq \|x - \tilde{x}\|$. Also note that the spectrum of $\widetilde{Y}$ is identical to that of the original signal X at frequencies $\omega \in (-\omega_s/2, \omega_s/2)$. This is graphically shown in Figure 9.6.6.6.1.

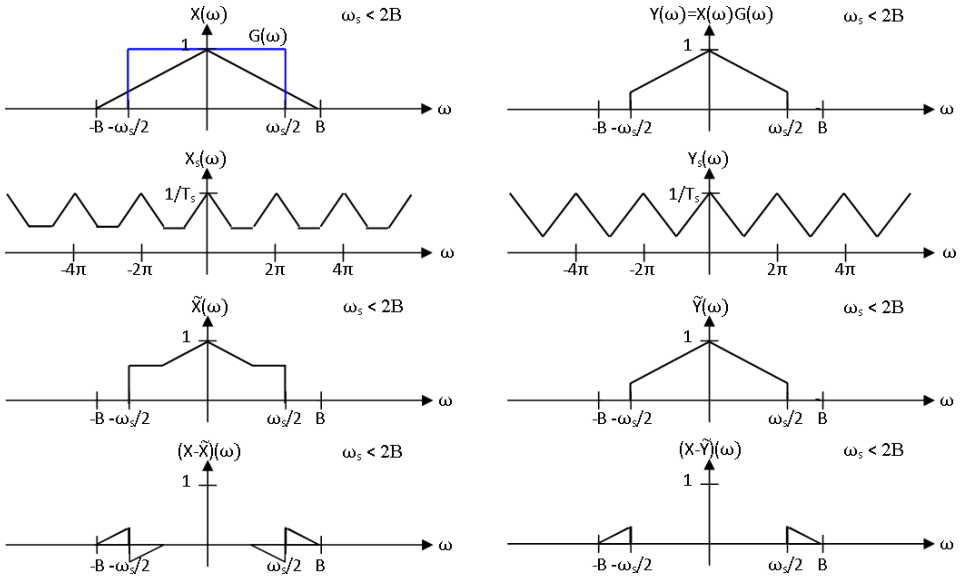


Figure 9.6.6.6.1: The figure above illustrates the use of an anti-aliasing filter to improve the process of sampling and reconstruction when using a sampling frequency below the Nyquist frequency. Notice that when using an ideal lowpass anti-aliasing filter, the reconstructed signal spectrum has the same shape as the original signal spectrum for all frequencies below half the sampling rate. This results in a lower error energy when using the anti-aliasing filter, as can be seen by comparing the error spectra shown.

Anti-Aliasing Filters Summary

As can be seen, anti-aliasing filters ensure that the signal is $(-\omega_s/2, \omega_s/2)$ bandlimited, or at least nearly so. The optimal anti-aliasing filter would be the ideal lowpass filter with cutoff frequency at $\omega_s/2$, which would ensure that the original signal spectrum and the reconstructed signal spectrum are equal on the interval $(-\omega_s/2, \omega_s/2)$. However, the ideal lowpass filter is not possible to implement in practice, and approximations must be accepted instead. Anti-aliasing filters are an important component of systems that implement discrete time processing of continuous time signals, as will be shown in the subsequent module.

This page titled 9.6.6.6: Anti-Aliasing Filters is shared under a CC BY 4.0 license and was authored, remixed, and/or curated by Richard Baraniuk et al. via source content that was edited to the style and standards of the LibreTexts platform.

- **10.6: Anti-Aliasing Filters** by Richard Baraniuk et al. is licensed CC BY 3.0. Original source: https://cnx.org/contents/d2CEAGW515.4:l-e_wSqM24/Signal-Classifications-and-Properties.

9.6.6.7: Discrete Time Processing of Continuous Time Signals

Introduction

Digital computers can process discrete time signals using extremely flexible and powerful algorithms. However, most signals of interest are continuous time signals, which is how data almost always appears in nature. Now that the theory supporting methods for generating a discrete time signal from a continuous time signal through sampling and then perfectly reconstructing the original signal from its samples without error has been discussed, it will be shown how this can be applied to implement continuous time, linear time invariant systems using discrete time, linear time invariant systems. This is of key importance to many modern technologies as it allows the power of digital computing to be leveraged for processing of analog signals.

Discrete Time Processing of Continuous Time Signals

Process Structure

With the aim of processing continuous time signals using a discrete time system, we will now examine one of the most common structures of digital signal processing technologies. As an overview of the approach taken, the original continuous time signal x is sampled to a discrete time signal x_s in such a way that the periods of the samples spectrum X_s is as close as possible in shape to the spectrum of X. Then a discrete time, linear time invariant filter H_2 is applied, which modifies the shape of the samples spectrum X_s but cannot increase the bandlimit of X_s, to produce another signal y_s. This is reconstructed with a suitable reconstruction filter to produce a continuous time output signal y, thus effectively implementing some continuous time system H_1. This process is illustrated in Figure 9.6.6.7.1, and the spectra are shown for a specific case in Figure 9.6.6.7.2.

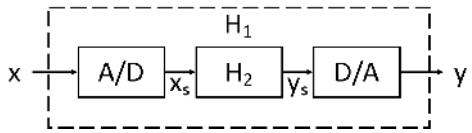


Figure 9.6.6.7.1: A block diagram for processing of continuous time signals using discrete time systems is shown.

Further discussion about each of these steps is necessary, and we will begin by discussing the analog to digital converter, often denoted by ADC or A/D. It is clear that in order to process a continuous time signal using discrete time techniques, we must sample the signal as an initial step. This is essentially the purpose of the ADC, although there are practical issues that which will be discussed later. An ADC takes a continuous time analog signal as input and produces a discrete time digital signal as output, with the ideal infinite precision case corresponding to sampling. As stated by the Nyquist-Shannon Sampling theorem, in order to retain all information about the original signal, we usually wish sample above the Nyquist frequency $\omega_s \geq 2B$ where the original signal is bandlimited to $(-B, B)$. When it is not possible to guarantee this condition, an anti-aliasing filter should be used.

The discrete time filter is where the intentional modifications to the signal information occur. This is commonly done in digital computer software after the signal has been sampled by a hardware ADC and before it is used by a hardware DAC to construct the output. This allows the above setup to be quite flexible in the filter that it implements. If sampling above the Nyquist frequency the. Any modifications that the discrete filter makes to this shape can be passed on to a continuous time signal assuming perfect reconstruction. Consequently, the process described will implement a continuous time, linear time invariant filter. This will be explained in more mathematical detail in the subsequent section. As usual, there are, of course, practical limitations that will be discussed later.

Finally, we will discuss the digital to analog converter, often denoted by DAC or D/A. Since continuous time filters have continuous time inputs and continuous time outputs, we must construct a continuous time signal from our filtered discrete time signal. Assuming that we have sampled a bandlimited at a sufficiently high rate, in the ideal case this would be done using perfect reconstruction through the Whittaker-Shannon interpolation formula. However, there are, once again, practical issues that prevent this from happening that will be discussed later.

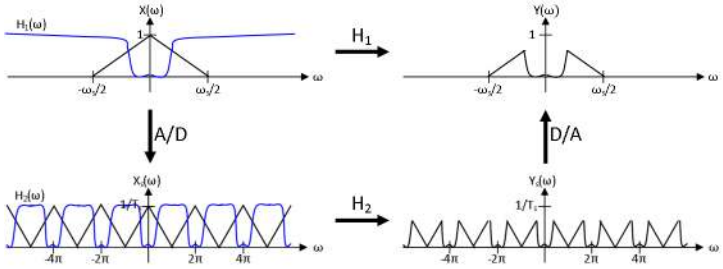


Figure 9.6.6.7.2: Spectra are shown in black for each step in implementing a continuous time filter using a discrete time filter for a specific signal. The filter frequency responses are shown in blue, and both are meant to have maximum value 1 in spite of the vertical scale that is meant only for the signal spectra. Ideal ADCs and DACs are assumed.

Discrete Time Filter

With some initial discussion of the process illustrated in Figure 9.6.6.7.1 complete, the relationship between the continuous time, linear time invariant filter H_1 and the discrete time, linear time invariant filter H_2 can be explored. We will assume the use of ideal, infinite precision ADCs and DACs that perform sampling and perfect reconstruction, respectively, using a sampling rate $\omega_s = 2\pi/T_s \geq 2B$ where the input signal x is bandlimited to $(-B, B)$. Note that these arguments fail if this condition is not met and aliasing occurs. In that case, pre-application of an anti-aliasing filter is necessary for these arguments to hold.

Recall that we have already calculated the spectrum X_s of the samples x_s given an input x with spectrum X as

$$X_s(\omega) = \frac{1}{T_s} \sum_{k=-\infty}^{\infty} X\left(\frac{\omega - 2\pi k}{T_s}\right).$$

Likewise, the spectrum Y_s of the samples y_s given an output y with spectrum Y is

$$Y_s(\omega) = \frac{1}{T_s} \sum_{k=-\infty}^{\infty} Y\left(\frac{\omega - 2\pi k}{T_s}\right).$$

From the knowledge that $y_s = (H_1 x)_s = H_2(x_s)$, it follows that

$$\sum_{k=-\infty}^{\infty} H_1\left(\frac{\omega - 2\pi k}{T_s}\right) X\left(\frac{\omega - 2\pi k}{T_s}\right) = H_2(\omega) \sum_{k=-\infty}^{\infty} X\left(\frac{\omega - 2\pi k}{T_s}\right).$$

Because X is bandlimited to $(-\pi/T_s, \pi/T_s)$, we may conclude that

$$H_2(\omega) = \sum_{k=-\infty}^{\infty} H_1\left(\frac{\omega - 2\pi k}{T_s}\right) (u(\omega - (2k-1)\pi) - u(\omega - (2k+1)\pi)).$$

More simply stated, H_2 is 2π periodic and $H_2(\omega) = H_1(\omega/T_s)$ for $\omega \in [-\pi, \pi)$.

Given a specific continuous time, linear time invariant filter H_1, the above equation solves the system design problem provided we know how to implement H_2. The filter H_2 must be chosen such that it has a frequency response where each period has the same shape as the frequency response of H_1 on $(-\pi/T_s, \pi/T_s)$. This is illustrated in the frequency responses shown in Figure 9.6.6.7.2.

We might also want to consider the system analysis problem in which a specific discrete time, linear time invariant filter H_2 is given, and we wish to describe the filter H_1. There are many such filters, but we can describe their frequency responses on $(-\pi/T_s, \pi/T_s)$ using the above equation. Isolating one period of $H_2(\omega)$ yields the conclusion that $H_1(\omega) = H_2(\omega T_s)$ for $\omega \in (-\pi/T_s, \pi/T_s)$. Because x was assumed to be bandlimited to $(-\pi/T_s, \pi/T_s)$, the value of the frequency response elsewhere is irrelevant.

Practical Considerations

As mentioned before, there are several practical considerations that need to be addressed at each stage of the process shown in Figure 9.6.6.7.1. Some of these will be briefly addressed here, and a more complete model of how discrete time processing of continuous time signals appears in Figure 9.6.6.7.3.

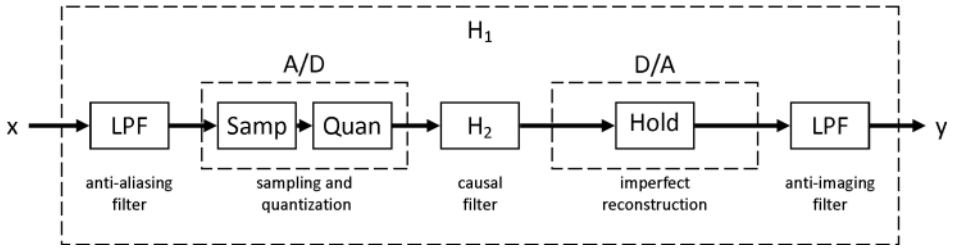


Figure 9.6.6.7.3: A more complete model of how discrete time processing of continuous time signals is implemented in practice. Notice the addition of anti-aliasing and anti-imaging filters to promote input and output bandlimitedness. The ADC is shown to perform sampling with quantization. The digital filter is further specified to be causal. The DAC is shown to perform imperfect reconstruction, a zero order hold in this case.

Anti-Aliasing Filter

In reality, we cannot typically guarantee that the input signal will have a specific bandlimit, and sufficiently high sampling rates cannot necessarily be produced. Since it is imperative that the higher frequency components not be allowed to masquerade as lower frequency components through aliasing, anti-aliasing filters with cutoff frequency less than or equal to $\omega_s/2$ must be used before the signal is fed into the ADC. The block diagram in Figure 9.6.6.7.3 reflects this addition.

As described in the previous section, an ideal lowpass filter removing all energy at frequencies above $\omega_s/2$ would be optimal. Of course, this is not achievable, so approximations of the ideal lowpass filter with low gain above $\omega_s/2$ must be accepted. This means that some aliasing is inevitable, but it can be reduced to a mostly insignificant level.

Signal Quantization

In our preceding discussion of discrete time processing of continuous time signals, we had assumed an ideal case in which the ADC performs sampling exactly. However, while an ADC does convert a continuous time signal to a discrete time signal, it also must convert analog values to digital values for use in a digital logic device, a phenomenon called quantization. The ADC subsystem of the block diagram in Figure 9.6.6.7.3 reflects this addition.

The data obtained by the ADC must be stored in finitely many bits inside a digital logic device. Thus, there are only finitely many values that a digital sample can take, specifically $2N$ where N is the number of bits, while there are uncountably many values an analog sample can take. Hence something must be lost in the quantization process. The result is that quantization limits both the range and precision of the output of the ADC. Both are finite, and improving one at constant number of bits requires sacrificing quality in the other.

Filter Implementability

In real world circumstances, if the input signal is a function of time, the future values of the signal cannot be used to calculate the output. Thus, the digital filter H_2 and the overall system H_1 must be causal. The filter annotation in Figure 9.6.6.7.3 reflects this addition. If the desired system is not causal but has impulse response equal to zero before some time t_0, a delay can be introduced to make it causal. However, if this delay is excessive or the impulse response has infinite length, a windowing scheme becomes necessary in order to practically solve the problem. Multiplying by a window to decrease the length of the impulse response can reduce the necessary delay and decrease computational requirements.

Take, for instance the case of the ideal lowpass filter. It is acausal and infinite in length in both directions. Thus, we must satisfy ourselves with an approximation. One might suggest that these approximations could be achieved by truncating the sinc impulse response of the lowpass filter at one of its zeros, effectively windowing it with a rectangular pulse. However, doing so would produce poor results in the frequency domain as the resulting convolution would significantly spread the signal energy. Other windowing functions, of which there are many, spread the signal less in the frequency domain and are thus much more useful for producing these approximations.

Anti-Imaging Filter

In our preceding discussion of discrete time processing of continuous time signals, we had assumed an ideal case in which the DAC performs perfect reconstruction. However, when considering practical matters, it is important to remember that the sinc function,

which is used for Whittaker-Shannon interpolation, is infinite in length and acausal. Hence, it would be impossible for an DAC to implement perfect reconstruction.

Instead, the DAC implements a causal zero order hold or other simple reconstruction scheme with respect to the sampling rate ω_s used by the ADC. However, doing so will result in a function that is not bandlimited to $(-\omega_s/2, \omega_s/2)$. Therefore, an additional lowpass filter, called an anti-imaging filter, must be applied to the output. The process illustrated in Figure 9.6.6.7.3 reflects these additions. The anti-imaging filter attempts to bandlimit the signal to $(-\omega_s/2, \omega_s/2)$, so an ideal lowpass filter would be optimal. However, as has already been stated, this is not possible. Therefore, approximations of the ideal lowpass filter with low gain above $\omega_s/2$ must be accepted. The anti-imaging filter typically has the same characteristics as the anti-aliasing filter.

Discrete Time Processing of Continuous Time Signals Summary

As has been show, the sampling and reconstruction can be used to implement continuous time systems using discrete time systems, which is very powerful due to the versatility, flexibility, and speed of digital computers. However, there are a large number of practical considerations that must be taken into account when attempting to accomplish this, including quantization noise and anti-aliasing in the analog to digital converter, filter implementability in the discrete time filter, and reconstruction windowing and associated issues in the digital to analog converter. Many modern technologies address these issues and make use of this process.

This page titled 9.6.6.7: Discrete Time Processing of Continuous Time Signals is shared under a CC BY 4.0 license and was authored, remixed, and/or curated by Richard Baraniuk et al. via source content that was edited to the style and standards of the LibreTexts platform.

- **10.7: Discrete Time Processing of Continuous Time Signals** by Richard Baraniuk et al. is licensed CC BY 3.0. Original source: https://cnx.org/contents/d2CEAGW515.4:l-e_wSqM24/Signal-Classifications-and-Properties.

SECTION OVERVIEW

9.7: Chapter 7 Supplementary Reading Materials

Resources developed by authors external to NIU with full prose and background materials:

Topic hierarchy

9.7.1: Design of Experiments

9.7.1.1: Design of Experiments via Taguchi Methods - Orthogonal Arrays

9.7.1.2: Design of experiments via factorial designs

9.7.1.3: Design of Experiments via Random Design

9.7.1.4: Summary- Summary on Control Architectures' philosophies, advantages, and disadvantages.

9.7: Chapter 7 Supplementary Reading Materials is shared under a not declared license and was authored, remixed, and/or curated by LibreTexts.

SECTION OVERVIEW

9.7.1: DESIGN OF EXPERIMENTS

9.7.1.1: DESIGN OF EXPERIMENTS VIA TAGUCHI METHODS - ORTHOGONAL ARRAYS

9.7.1.2: DESIGN OF EXPERIMENTS VIA FACTORIAL DESIGNS

9.7.1.3: DESIGN OF EXPERIMENTS VIA RANDOM DESIGN

9.7.1.4: SUMMARY- SUMMARY ON CONTROL ARCHITECTURES' PHILOSOPHIES, ADVANTAGES, AND DISADVANTAGES.

This page titled 9.7.1: Design of Experiments is shared under a CC BY-NC 4.0 license and was authored, remixed, and/or curated by Peter Woolf et al. via source content that was edited to the style and standards of the LibreTexts platform.

9.7.1.1: DESIGN OF EXPERIMENTS VIA TAGUCHI METHODS - ORTHOGONAL ARRAYS

The **Taguchi method** involves reducing the variation in a process through robust design of experiments. The overall objective of the method is to produce high quality product at low cost to the manufacturer. The Taguchi method was developed by Dr. Genichi Taguchi of Japan who maintained that variation. Taguchi developed a method for designing experiments to investigate how different parameters affect the mean and variance of a process performance characteristic that defines how well the process is functioning. The experimental design proposed by Taguchi involves using orthogonal arrays to organize the parameters affecting the process and the levels at which they should be varies. Instead of having to test all possible combinations like the factorial design, the Taguchi method tests pairs of combinations. This allows for the collection of the necessary data to determine which factors most affect product quality with a minimum amount of experimentation, thus saving time and resources. The Taguchi method is best used when there is an intermediate number of variables (3 to 50), few interactions between variables, and when only a few variables contribute significantly.

The Taguchi arrays can be derived or looked up. Small arrays can be drawn out manually; large arrays can be derived from deterministic algorithms. Generally, arrays can be found online. The arrays are selected by the number of parameters (variables) and the number of levels (states). This is further explained later in this article. Analysis of variance on the collected data from the Taguchi design of experiments can be used to select new parameter values to optimize the performance characteristic. The data from the arrays can be analyzed by plotting the data and performing a visual analysis, ANOVA, bin yield and Fisher's exact test, or Chi-squared test to test significance.

In this article, the specific steps involved in the application of the Taguchi method will be described and examples of using the Taguchi method to design experiments will be given.

9.7.1.1.1: SUMMARY OF TAGUCHI METHOD

9.7.1.1.1.1: PHILOSOPHY OF THE TAGUCHI METHOD

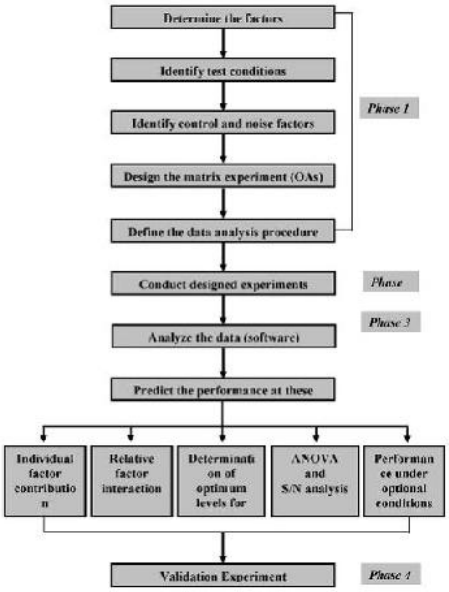
1. **Quality should be designed into a product, not inspected into it.** Quality is designed into a process through system design, parameter design, and tolerance design. Parameter design, which will be the focus of this article, is performed by determining what process parameters most affect the product and then designing them to give a specified target quality of product. Quality "inspected into" a product means that the product is produced at random quality levels and those too far from the mean are simply thrown out.
2. **Quality is best achieved by minimizing the deviation from a target. The product should be designed so that it is immune to uncontrollable environmental factors.** In other words, the signal (product quality) to noise (uncontrollable factors) ratio should be high.
3. **The cost of quality should be measured as a function of deviation from the standard and the losses should be measured system wide.** This is the concept of the loss function, or the overall loss incurred upon the customer and society from a product of poor quality. Because the producer is also a member of society and because customer dissatisfaction will discourage future patronage, this cost to customer and society will come back to the producer.

9.7.1.1.1.2: TAGUCHI METHOD DESIGN OF EXPERIMENTS

The general steps involved in the Taguchi Method are as follows:

1. Define the process objective, or more specifically, a target value for a performance measure of the process. This may be a flow rate, temperature, etc. The target of a process may also be a minimum or maximum; for example, the goal may be to maximize the output flow rate. The deviation in the performance characteristic from the target value is used to define the loss function for the process.
2. Determine the design parameters affecting the process. Parameters are variables within the process that affect the performance measure such as temperatures, pressures, etc. that can be easily controlled. The number of levels that the parameters should be varied at must be specified. For example, a temperature might be varied to a low and high value of 40 C and 80 C. Increasing the number of levels to vary a parameter at increases the number of experiments to be conducted.
3. Create orthogonal arrays for the parameter design indicating the number of and conditions for each experiment. The selection of orthogonal arrays is based on the number of parameters and the levels of variation for each parameter, and will be expounded below.
4. Conduct the experiments indicated in the completed array to collect data on the effect on the performance measure.
5. Complete data analysis to determine the effect of the different parameters on the performance measure.

See below for a pictorial depiction of these and additional possible steps, depending on the complexity of the analysis.



A detailed description of the execution of these steps will be discussed next.

9.7.1.1.2: TAGUCHI LOSS FUNCTION

The goal of the Taguchi method is to reduce costs to the manufacturer and to society from variability in manufacturing processes. Taguchi defines the difference between the target value of the performance characteristic of a process, τ, and the measured value, y, as a loss function as shown below.

$$l(y) = k_c(y-\tau)^2$$

The constant, k_c, in the loss function can be determined by considering the specification limits or the acceptable interval, delta.

$$k_c = \frac{C}{\Delta^2}$$

The difficulty in determining k_c is that τ and C are sometimes difficult to define.

If the goal is for the performance characteristic value to be minimized, the loss function is defined as follows:

$$l(y) = k_c y^2$$

where $\tau = 0$.

If the goal is for the performance characteristic value to maximized, the loss function is defined as follows:

$$l(y) = \frac{k_c}{y^2}$$

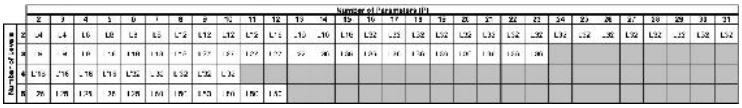
The loss functions described here are the loss to a customer from one product. By computing these loss functions, the overall loss to society can also be calculated.

9.7.1.1.3: DETERMINING PARAMETER DESIGN ORTHOGONAL ARRAY

The effect of many different parameters on the performance characteristic in a condensed set of experiments can be examined by using the orthogonal array experimental design proposed by Taguchi. Once the parameters affecting a process that can be controlled have been determined, the levels at which these parameters should be varied must be determined. Determining what levels of a variable to test requires an in-depth understanding of the process, including the minimum, maximum, and current value of the parameter. If the difference between the minimum and maximum value of a parameter is large, the values being tested can be further apart or more values can be tested. If the range of a parameter is small, then less values can be tested or the values tested can be closer together. For example, if the temperature of a reactor jacket can be varied between 20 and 80 degrees C and it is known that the current operating jacket temperature is 50 degrees C, three levels might be chosen at 20, 50, and 80 degrees C. Also, the cost of conducting experiments must be considered when determining the number of levels of a parameter to include in the experimental design. In the previous example of jacket temperature, it would be cost prohibitive to do 60 levels at 1 degree intervals. Typically, the number of levels for all parameters in the experimental design is chosen to be the same to aid in the selection of the proper orthogonal array.

Knowing the number of parameters and the number of levels, the proper orthogonal array can be selected. Using the array selector table shown below, the name of the appropriate array can be found by looking at the column and row corresponding to the number of parameters and number of levels. Once the name has been determined (the subscript represents the number of experiments that must be completed), the predefined array can be looked up. Links are provided to many of the predefined arrays given in the array selector table. These arrays were created using an algorithm Taguchi developed, and allows for each variable and setting to be tested equally. For example, if we have three parameters (voltage, temperature, pressure) and two levels (high, low), it can be seen the proper array is L4. Clicking on the link L4 to view the L4 array, it can be seen four different experiments are given in the array. The levels designated as 1, 2, 3 etc. should be replaced in the array with the actual level values to be varied and P1, P2, P3 should be replaced with the actual parameters (i.e. voltage, temperature, etc.)

Array Selector



Links to Orthogonal Arrays

The following links are connected to images of the orthogonal array named in the link title:

L4 Array

L8 Array

L9 Array

L12 Array

L16 Array

L'16 Array

L18 Array

L25 Array

L27 Array

L32 Array

L'32 Array

L36 Array

L50 Array

9.7.1.1.3.1: IMPORTANT NOTES REGARDING SELECTION + USE OF ORTHOGONAL ARRAYS

Note 1

The array selector assumes that each parameter has the same number of levels. Sometimes this is not the case. Generally, the highest value will be taken or the difference will be split.

The following examples offer insight on choosing and properly using an orthogonal array. Examples 1 and 2 focus on array choice, while Example 3 will demonstrate how to use an orthogonal array in one of these situations.

Example 1:

parameter: A, B, C, D = 4 # levels: 3, 3, 3, 2 = ~3 array: L9

Example 2:

parameter: A, B, C, D, E, F = 6 # levels: 4, 5, 3, 2, 2, 2 = ~3 array: modified L16

Example 3:

A reactor's behavior is dependent upon impeller model, mixer speed, the control algorithm employed, and the cooling water valve type. The possible values for each are as follows:

Impeller model: A, B, or C

Mixer speed: 300, 350, or 400 RPM

Control algorithm: PID, PI, or P

Valve type: butterfly or globe

There are 4 parameters, and each one has 3 levels with the exception of valve type. The highest number of levels is 3, so we will use a value of 3 when choosing our orthogonal array.

Using the array selector above, we find that the appropriate orthogonal array is L9:

Experiment	P1	P2	P3	P4
1	1	1	1	1
2	1	2	2	2
3	1	3	3	3
4	2	1	2	3
5	2	2	3	1
6	2	3	1	2
7	3	1	3	2
8	3	2	1	3
9	3	3	2	1

When we replace P1, P2, P3, and P4 with our parameters and begin filling in the parameter values, we find that the L9 array includes 3 levels for valve type, while our system only has 2. The appropriate strategy is to fill in the entries for P4=3 with 1 or 2 in a random, balanced way. For example:

Experiment	Impeller	Motor Speed	Control	Valve
1	A	300	PID	BF
2	A	350	PI	G
3	A	400	P	G
4	B	300	PI	BF
5	B	350	P	BF
6	B	400	PID	G
7	C	300	P	G
8	C	350	PID	BF
9	C	400	PI	BF

Here, the third value was chosen twice as butterfly and once as global.

Note 2

If the array selected based on the number of parameters and levels includes more parameters than are used in the experimental design, ignore the additional parameter columns. For example, if a process has 8 parameters with 2 levels each, the L12 array should be selected according to the array selector. As can be seen below, the L12 Array has columns for 11 parameters (P1-P11). The right 3 columns should be ignored.

Experiment	P1	P2	P3	P4	P5	P6	P7	P8	P9	P10	P11
1	1	1	1	1	1	1	1	1	1	1	1
2	1	1	1	1	1	2	2	2	2	2	2
3	1	1	2	2	2	1	1	1	2	2	2
4	1	2	1	2	2	1	2	2	1	1	2
5	1	2	2	1	2	2	1	2	1	2	1
6	1	2	2	1	2	2	1	2	1	1	1
7	1	2	2	2	1	2	2	1	2	1	1
8	2	1	2	1	2	2	2	1	1	1	2
9	2	1	1	2	2	2	1	2	2	1	1
10	2	2	2	1	1	1	1	2	2	1	2
11	2	2	1	2	1	2	1	1	1	2	2
12	2	2	1	1	2	1	2	1	2	2	1

9.7.1.1.4: ANALYZING EXPERIMENTAL DATA

Once the experimental design has been determined and the trials have been carried out, the measured performance characteristic from each trial can be used to analyze the relative effect of the different parameters. To demonstrate the data analysis procedure, the following L9 array will be used, but the principles can be transferred to any type of array.

In this array, it can be seen that any number of repeated observations (trials) may be used. Ti,j represents the different trials with i = experiment number and j = trial number. It should be noted that the Taguchi method allows for the use of a noise matrix including external factors affecting the process outcome rather than repeated trials, but this is outside of the scope of this article.

Experiment Number	P1	P2	P3	P4	T1	T2	...	TN
1	1	1	1	1	$T_{1,1}$	$T_{1,2}$	...	$T_{1,N}$
2	1	2	2	2	$T_{2,1}$	$T_{2,2}$	...	$T_{2,N}$
3	1	3	3	3	$T_{3,1}$	$T_{3,2}$	...	$T_{3,N}$
4	2	1	2	3	$T_{4,1}$	$T_{4,2}$	...	$T_{4,N}$
5	2	2	3	1	$T_{5,1}$	$T_{5,2}$	...	$T_{5,N}$
6	2	3	1	2	$T_{6,1}$	$T_{6,2}$	...	$T_{6,N}$
7	3	1	3	2	$T_{7,1}$	$T_{7,2}$	...	$T_{7,N}$
8	3	2	1	3	$T_{8,1}$	$T_{8,2}$	...	$T_{8,N}$
9	3	3	2	1	$T_{9,1}$	$T_{9,2}$	...	$T_{9,N}$

To determine the effect each variable has on the output, the signal-to-noise ratio, or the SN number, needs to be calculated for each experiment conducted. The calculation of the SN for the first experiment in the array above is shown below for the case of a specific target value of the performance characteristic. In the equations below, yi is the mean value and si is the variance. yi is the value of the performance characteristic for a given experiment.

$$SN_i = 10 \log \frac{\bar{y_i}^2}{{s_i}^2}$$

$$SN_i = 10 \log \frac{\bar{y}_i^2}{s_i^2}$$

where

$$\bar{y}_i = \frac{1}{N_i} \sum_{u=1}^{N_i} y_{i,u}$$

$$s_i^2 = \frac{1}{N_i - 1} \sum_{u=1}^{N_i} (y_{i,u} - \bar{y}_i)$$

and

- i is the experimental number,
- u is the trial number, and
- N_i is the numb er of trials for experiment i

For the case of minimizing the performance characteristic, the following definition of the SN ratio should be calculated:

$$SN_i = -10 \log \left(\sum_{u=1}^{N_i} \frac{y_u^2}{N_i} \right)$$

For the case of maximizing the performance characteristic, the following definition of the SN ratio should be calculated:

$$SN_i = -10 \log \left[\frac{1}{N_i} \sum_{u=1}^{N_i} \frac{1}{y_u^2} \right]$$

After calculating the SN ratio for each experiment, the average SN value is calculated for each factor and level. This is done as shown below for Parameter 3 (P3) in the array:

Experiment Number	P1	P2	P3	P4	SN
1	1	1	1	1	SN1
2	1	2	2	2	SN2
3	1	3	3	3	SN3
4	2	1	2	3	SN4
5	2	2	3	1	SN5
6	2	3	1	2	SN6
7	3	1	3	2	SN7
8	3	2	1	3	SN8
9	3	3	2	1	SN9

$$SN_{P3,1} = \frac{(S_{N1} + S_{N6} + S_{N8})}{3}$$

$$SN_{P3,2} = \frac{(S_{N2} + S_{N4} + S_{N9})}{3}$$

$$SN_{P3,3} = \frac{(S_{N3} + S_{N5} + S_{N7})}{3}$$

Once these SN ratio values are calculated for each factor and level, they are tabulated as shown below and the range R (R = high SN - low SN)of the SN for each parameter is calculated and entered into the table. The larger the R value for a parameter, the larger the effect the variable has on the process. This is because the same change in signal causes a larger effect on the output variable being measured.

Level	P1	P2	P3	P4
1	SNP1,1	SNP2,1	SNP3,1	SNP4,1
2	SNP1,2	SNP2,2	SNP2,2	SNP4,2
3	SNP1,3	SNP2,3	SNP3,3	SNP4,3
Δ	RP1	RP2	RP3	RP4
Rank	...	...	...	...

Please refer to the Worked Out Example for a numeric example of how the data analysis procedure described here is applied.

9.7.1.1.5: ADVANTAGES AND DISADVANTAGES

An advantage of the Taguchi method is that it emphasizes a mean performance characteristic value close to the target value rather than a value within certain specification limits, thus improving the product quality. Additionally, Taguchi's method for experimental design is straightforward and easy to apply to many engineering situations, making it a powerful yet simple tool. It can be used to quickly narrow down the scope of a research project or to identify problems in a manufacturing process from data already in existence. Also, the Taguchi method allows for the analysis of many different parameters without a prohibitively high amount of experimentation. For example, a process with 8 variables, each with 3 states, would require 6561 (3^8) experiments to test all variables. However using Taguchi's orthogonal arrays, only 18 experiments are necessary, or less than .3% of the original number of experiments. In this way, it allows for the identification of key parameters that have the most effect on the performance characteristic value so that further experimentation on these parameters can be performed and the parameters that have little effect can be ignored.

The main disadvantage of the Taguchi method is that the results obtained are only relative and do not exactly indicate what parameter has the highest effect on the performance characteristic value. Also, since orthogonal arrays do not test all variable combinations, this method should not be used with all relationships between all variables are needed. The Taguchi method has been criticized in the literature for difficulty in accounting for interactions between parameters. Another limitation is that the Taguchi methods are offline, and therefore inappropriate for a dynamically changing process such as a simulation study. Furthermore, since Taguchi methods deal with designing quality in rather than correcting for poor quality, they are applied most effectively at early stages of process development. After design variables are specified, use of experimental design may be less cost effective.

9.7.1.1.6: OTHER METHODS OF EXPERIMENTAL DESIGN

Two other methods for determining experimental design are factorial design and random design. For scenarios with a small number of parameters and levels (1-3) and where each variable contributes significantly, factorial design can work well to determine the specific interactions between variables. However, factorial design gets increasingly complex with an increase in the number of variables. For large systems with many variables (50+) where there are few interactions between variables, random design can be used. Random design assigns each variable a state based on a uniform sample (ex: 3 states = 0.33 probability) for the selected number of experiments. When used properly (in a large system), random design usually produces an experimental design that is desired. However, random design works poorly for systems with a small number of variables.

To obtain a even better understanding of these three different methods, it's good to get a visual of these three methods. It will illustrate the degree of efficiency for each experimental design depending on the number of variables and the number of states for each variable. The following will have the three experimental designs for the same scenario.

Scenario. You have a CSTR that has four(4) variables and each variable has three or two states. You are to design an experiment to systematically test the effect of each of the variables in the current CSTR.

These variables can take the following values:
Impellers: model A, B, or C
Motor speed for mixer: 300, 350, or 400 RPM
Control algorithm: PID, PI, or P only
Cooling water valve type: butterfly or globe

Experimental Design #1: Factorial Design By looking at the # variables and # states, there should be a total of 54 experiments because (3impellers)(3speeds)(3controllers)(2valves)=54. Here's a list of these 54 experiments:

A, 300,PID, B	A, 300,PI, B	A, 300,P, B	A, 300,PI, G	A, 300,PID, G	A, 300,P, G
B, 300,PID, B	B, 300,PI, B	B, 300,P, B	B, 300,PI, G	B, 300,PID, G	B, 300,P, G
C, 300,PID, B	C, 300,PI, B	C, 300,P, B	C, 300,PI, G	C, 300,PID, G	C, 300,P, G
A, 350,PID, B	A, 350,PI, B	A, 350,P, B	A, 350,PI, G	A, 350,PID, G	A, 350,P, G
B, 350,PID, B	B, 350,PI, B	B, 350,P, B	B, 350,PI, G	B, 350,PID, G	B, 350,P, G
C, 350,PID, B	C, 350,PI, B	C, 350,P, B	C, 350,PI, G	C, 350,PID, G	C, 350,P, G
A, 400,PID, B	A, 400,PI, B	A, 400,P, B	A, 400,PI, G	A, 400,PID, G	A, 400,P, G
B, 400,PID, B	B, 400,PI, B	B, 400,P, B	B, 400,PI, G	B, 400,PID, G	B, 400,P, G
C, 400,PID, B	C, 400,PI, B	C, 400,P, B	C, 400,PI, G	C, 400,PID, G	C, 400,P, G

Experimental Design #2: Taguchi Method Since you know the # of states and variables, you can refer to the table above in this wiki and obtain the correct Taguchi array. It turns out to be a L9 array.

L9

Experiment	P1	P2	P3	P4
1	1	1	1	1
2	1	2	2	2
3	1	3	3	3
4	2	1	2	3
5	2	2	3	1
6	2	3	1	2
7	3	1	3	2
8	3	2	1	3
9	3	3	2	1

With the actual variables and states, the L9 array should look like the following:

Experiment	Impeller	Moter Speed	Control	Valve
1	A	300	PID	BF
2	A	350	PI	G
3	A	400	P	G
4	B	300	PI	BF
5	B	350	P	BF
6	B	400	PID	G
7	C	300	P	G
8	C	350	PID	BF
9	C	400	PI	BF

Experimental Design #3: Random Design

Since we do not know the number of signal recoveries we want and we don't know the probabilities of each state to happen, it will be difficult to construct a random design table. It will mostly be used for extreme large experiments. Refer to the link below to help you obtain a better grasp on the random design concept.

http://groups.csail.mit.edu/drl/journal_club/papers/CS2-Candes-Romberg-05.pdf

EXAMPLE 9.7.1.1.1

A microprocessor company is having difficulty with its current yields. Silicon processors are made on a large die, cut into pieces, and each one is tested to match specifications. The company has requested that you run experiments to increase processor yield. The factors that affect processor yields are temperature, pressure, doping amount, and deposition rate.

Determine the Taguchi experimental design orthogonal array. The operating conditions for each parameter and level are listed below:

- **A: Temperature**
- A1 = 100ºC
- A2 = 150ºC (current)
- A3 = 200ºC
- **B: Pressure**
- B1 = 2 psi
- B2 = 5 psi (current)
- B3 = 8 psi
- **C: Doping Amount**
- C1 = 4%
- C2 = 6% (current)
- C3 = 8%
- **D: Deposition Rate**
- D1 = 0.1 mg/s
- D2 = 0.2 mg/s (current)
- D3 = 0.3 mg/s

Conducting three trials for each experiment, the data below was collected. Compute the SN ratio for each experiment for the target value case, create a response chart, and determine the parameters that have the highest and lowest effect on the processor yield.

Experiment Number	Temperature	Pressure	Doping Amount	Deposition Rate	Trial 1	Trial 2	Trial 3	Mean
1	100	2	4	0.1	87.3	82.3	70.7	80.1
2	100	5	6	0.2	74.8	70.7	63.2	69.6
3	100	8	8	0.3	56.5	54.9	45.7	52.1
4	150	2	6	0.3	79.8	78.2	62.3	73.4
5	150	5	8	0.1	77.3	76.5	54.9	69.3
6	150	8	4	0.2	89	87.3	83.2	86.5
7	200	2	8	0.2	64.8	62.3	55.7	60.9
8	200	5	4	0.3	99	93.2	87.3	93.2
9	200	8	6	0.1	75.7	74	63.2	71

Solution

a

The L9 orthogonal array should be used. The filled in orthogonal array should look like this:

Experiment Number	Temperature	Pressure	Doping Amount	Deposition Rate
1	100	2	4	0.1
2	100	5	6	0.2
3	100	8	8	0.3
4	150	2	6	0.3
5	150	5	8	0.1
6	150	8	4	0.2
7	200	2	8	0.2
8	200	5	4	0.3
9	200	8	6	0.1

This setup allows the testing of all four variables without having to run 81 [=3^4=(3 Temperatures)(3 Pressures)(3 Doping Amounts)(3 Deposition rates)] separate trials.b) Question:

b) Shown below is the calculation and tabulation of the SN ratio.

$$S_{m1} = \frac{(87.3 + 82.3 + 70.7)^2}{3} = 19248.0$$

$$S_{T1} = 87.3^2 + 82.3^2 + 70.7^2 = 19393.1$$

$$S_{e1} = S_{T1} - S_{m1} = 19393.1 - 19248.0 = 145.0$$

$$V_{e1} = \frac{S_{e1}}{N-1} = \frac{145.1}{2} = 72.5$$



Experiment Number	A (temp)	B (pres)	C (dop)	D (dep)	T1	T2	T3	SN
1	1	1	1	1	87.3	82.3	70.7	19.5
2	1	2	2	2	74.8	70.7	63.2	21.4
3	1	3	3	3	56.5	54.0	45.7	19.3
4	2	1	2	3	79.8	78.2	62.3	17.6
5	2	2	3	1	77.3	76.5	54.0	14.3
6	2	3	1	2	89.0	87.3	83.2	29.2
7	3	1	3	2	64.8	62.3	55.7	22.2
8	3	2	1	3	99.0	93.2	87.3	24.0
9	3	3	2	1	75.7	74.0	63.2	20.4

Shown below is the response table. This table was created by calculating an average SN value for each factor. A sample calculation is shown for Factor B (pressure):

Experiment Number	A (temp)	B (pres)	C (dop)	D (dep)	SN
1	1	1	1	1	19.5
2	1	2	2	2	21.4
3	1	3	3	3	19.3
4	2	1	2	3	17.6
5	2	2	3	1	14.3
6	2	3	1	2	29.2
7	3	1	3	2	22.2
8	3	2	1	3	24.0
9	3	3	2	1	20.4

$$SN_{B1} = \frac{(19.5 + 17.6 + 22.2)}{3} = 19.8$$

$$SN_{B2} = \frac{(21.4 + 14.3 + 24.0)}{3} = 19.9$$

$$SN_{B3} = \frac{(19.3 + 29.2 + 20.4)}{3} = 23.0$$

The effect of this factor is then calculated by determining the range:

$$\Delta = Max - Min = 23.0 - 19.8 = 3.2$$

Level	A (temp)	B (pres)	C (dop)	D (dep)
1	20.1	19.8	24.2	18.1
2	20.4	19.9	19.8	24.3
3	22.2	23.0	18.6	20.3
Δ	2.2	3.2	5.6	6.3
Rank	4	3	2	1

It can be seen that deposition rate has the largest effect on the processor yield and that temperature has the smallest effect on the processor yield.

NOTE: *The data for the following example is not real, and details pertaining to microprocessor fabrication may not be completely accurate.*

a) Solution:

✔ EXAMPLE 9.7.1.1.2: SESAME SEED SUFFERING

You have just produced one thousand 55 gallon drums of sesame oil for sale to your distributors. However, just before you are to ship oil, one of your employees remembers that one of the oil barrels was temporarily used to store insecticide and is almost surely contaminated. Unfortunately, all of the barrels look the same.

One barrel of sesame oil sells for $1000, while each assay for insecticide in food oil costs $1200 and takes 3 days. Tests for insecticide are extremely expensive. What do you do?

Solution

Extreme multiplexing. This is similar to using a Taguchi method but optimized for very sparse systems and specific cases. For example, instead of 1000 barrels, let us consider 8 barrels for now, one of which is contaminated. We could test each one, but that would be highly expensive. Another solution is to mix samples from each barrel and test the mixtures.

- Mix barrels 1,2,3,4 ---> Sample A
- Mix barrels 1,2,5,6 ---> Sample B
- Mix barrels 1,3,5,7 ---> Sample C

We claim that from testing only these three mixtures, we can determine which of the 8 barrels was contaminated. Let us consider some possible results of these tests. We will use the following label scheme: +/-,+/-,+/- in order of A, B, C. Thus, +,-,+ indicates A and C showed contamination but not B.

- Possible Result 1: -,-,- The only barrel not mixed in was #8, so it is contaminated.
- Possible Result 2: +,-,- Barrel #4 appears in A, but not in B and C. Since only A returned positive, barrel #4 was contaiminated.
- Possible Result 3: -,+,- Barrel #6 appears in B, but not in A and C. Since only B returned positive, barrel #6 was contaminated.

We can see that we have $2^3 = 8$ possible results, each of which corresponds to a particular barrel being contaminated. We leave the rest of the cases for the reader to figure out.

Solution with 1,000 barrels: Mix samples from each barrel and test mixtures. Each mixture will consist of samples from a unique combination of 500 barrels. Experiments required = $log_2(1000)$=~10.

Solution with 1,000,000 barrels: Experiments required = $log_2(1000000)$=~20.

Thus, by using extreme multiplexing, we can greatly reduce the # of experiments needed, since the # of experiments scales with log_2(# of barrels) instead of # of barrels.

? EXERCISE 9.7.1.1.1

You are producing decaffeinated coffee using supercritical carbon dioxide as a solvent. To optimize the effectiveness of caffeine removal, you would like to test 2 different possible pressures of CO2, 3 possible temperatures, 3 ratios of CO2 to coffee beans, 3 residence times of supercritical CO2 with beans, and 2 different procedures for preroasting the beans prior to caffeine extraction. What is the most appropriate method to design your optimization experiments?

a. Factorial design
b. Taguchi methods
c. Random design

d. None of the above. Coffee should never be decaffeinated

Answer

To perform a full factorial experiment, you'd need to account for: (2 pressures)(3 temperatures)(3 ratios)(3 times)(2 roasts) = 108 experiments! Thus, factorial design is not a practical choice: a good rule of thumb is 1-3 variables with few states for a manageable factorial analysis. However, selecting 3 for the number of levels and consulting the array selector, we see that an L18 array will suffice for a Taguchi analysis. 18 is a much more feasible number of experiments than 108. Finally, random design is only practical when >50 variables are involved and few will have significant impact on the system, which is not the case here. Thus, although the author is of the opinion that choice 'D' has significant merit, the correct answer is 'B'.

? EXERCISE 9.7.1.1.2

Which of the following is inconsistent with the Taguchi philosophy of quality control?

a. Variation is the opposite of quality
b. Interactions between parameters affecting product quality are unimportant and should never be considered
c. Customer dissatisfaction is the most important measure for process variation, and should be minimized
d. A high signal to noise ratio (SN) should be present in a process to make it robust against uncontrollable factors that would cause variation

Answer

Choice 'A' is another way of expressing point 3 of the Taguchi philosophy above. Similarly, choice 'C' is another way of expressing the cost function, and choice 'D' is consistent with philosophy 2. Although it is true that the Taguchi methods do not account well for interactions between variables, choice 'B' is incorrect for two reasons. 1) Pair-wise interactions between variables are accounted for in Taguchi orthogonal arrays. 2) Negligibility of interactions between parameters is not so much a central tenet of Taguchi philosophy as a necessary simplifying assumption. An analog is decoupling of MIMO systems. Since applying this assumption yields good results, it is consistent with Taguchi thinking to say that makes it valid.

9.7.1.1.7: REFERENCES

- Vuchkov, I.N. and Boyadjieva, L.N. *Quality Improvement with Design of Experiments: A Response Surface Approach.* Kluwer Academic Publishers. Dordrecht, 2001.
- http://www.ee.iitb.ac.in/~apte/CV_PRA_TAGUCHI_INTRO.htm
- http://www.york.ac.uk/depts/maths/tables/orthogonal.htm
- www.freequality.org/sites/www_freequality_org/documents/tools/Tagarray_files/tamatrix.htm
- Roy, R. K., A Primer on the Taguchi Method, Van Nostrand Reinhold, New York 1990.
- Ravella Sreenivas Rao, C. Ganesh Kumar, R. Shetty Prakasham, Phil J. Hobbs (2008) The Taguchi methodology as a statistical tool for biotechnological applications: A critical appraisal Biotechnology Journal 3:510–523.

This page titled 9.7.1.1: Design of Experiments via Taguchi Methods - Orthogonal Arrays is shared under a CC BY 4.0 license and was authored, remixed, and/or curated by Stephanie Fraley, John Zalewski, Mike Oom, Ben Terrien, & Ben Terrien via source content that was edited to the style and standards of the LibreTexts platform.

- **14.1: Design of Experiments via Taguchi Methods - Orthogonal Arrays** by Ben Terrien, John Zalewski, Mike Oom, Stephanie Fraley is licensed CC BY 3.0. Original source: https://open.umn.edu/opentextbooks/textbooks/chemical-process-dynamics-and-controls.

9.7.1.2: DESIGN OF EXPERIMENTS VIA FACTORIAL DESIGNS

Factorial design is an important method to determine the effects of multiple variables on a response. Traditionally, experiments are designed to determine the effect of ONE variable upon ONE response. R.A. Fisher showed that there are advantages by combining the study of multiple variables in the same factorial experiment. Factorial design can reduce the number of experiments one has to perform by studying multiple factors simultaneously. Additionally, it can be used to find both main effects (from each independent factor) and interaction effects (when both factors must be used to explain the outcome). However, factorial design can only give relative values, and to achieve actual numerical values the math becomes difficult, as regressions (which require minimizing a sum of values) need to be performed. Regardless, factorial design is a useful method to design experiments in both laboratory and industrial settings.

Factorial design tests all possible conditions. Because factorial design can lead to a large number of trials, which can become expensive and time-consuming, factorial design is best used for a small number of variables with few states (1 to 3). Factorial design works well when interactions between variables are strong and important and where every variable contributes significantly.

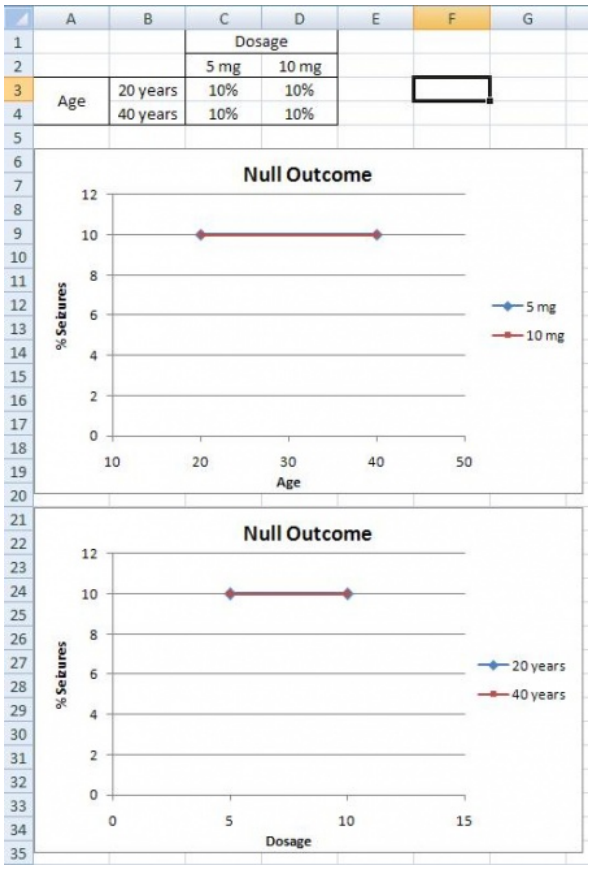
9.7.1.2.1: WHAT IS FACTORIAL DESIGN?

9.7.1.2.1.1: FACTORIAL DESIGN EXAMPLE

The easiest way to understand how factorial design works is to read an example. Suppose that you, a scientist working for the FDA, would like to study and measure the probability of patients suffering from seizures after taking a new pharmaceutical drug called CureAll. CureAll is a novel drug on the market and can cure nearly any ailment of the body. You along with your co-workers at the FDA have decided to test two dosage levels: 5 mg and 10 mg. You are also interested in determining whether the drug side-effects differ between younger adults (age 20) and older adults (age 40). Based on the given information, you see that there are two factors: dosage and age. Factors are the main categories to explore when determining the cause of seizures in patients. Under each of these factors, there are different levels: 5 and 10 mg for the dosage; 20 and 40 years for age. A level is basically one of the subdivisions that make up a factor. From this information, we can see that we have a 2 x 2 factorial design, which means that we will have 2 * 2 = 4 groups. A group is set of conditions that will make up that particular experiment.

9.7.1.2.1.2: NULL OUTCOME

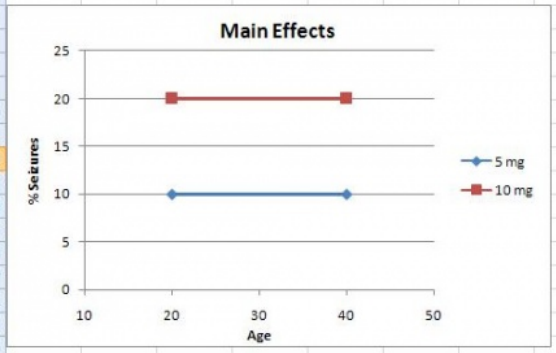
A null outcome situation is when the outcome of your experiment is the same regardless of how the levels within your experiment were combined. From the example above, a null outcome would exist if you received the same percentage of seizures occurring in patients with varying dose and age. The graphs below illustrate no change in the percentage of seizures for all factors, so you can conclude that the chance of suffering from a seizure is not affected by the dosage of the drug or the age of the patient.

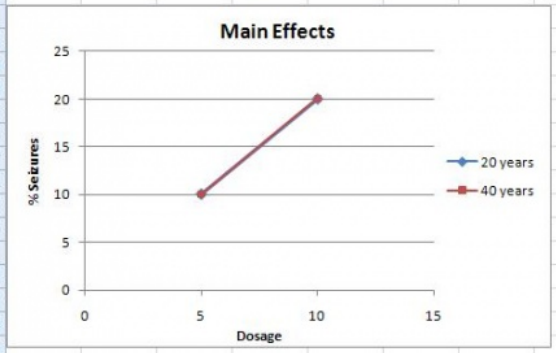


9.7.1.2.1.3: MAIN EFFECTS

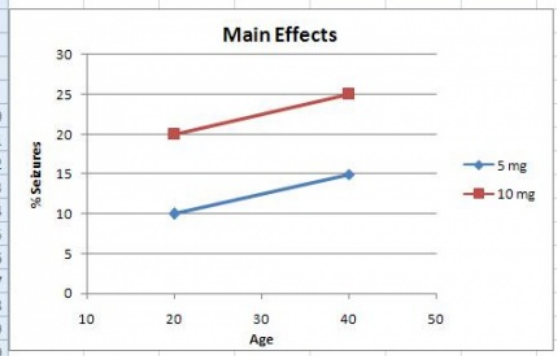
A main effects situation is when there exists a consistent trend among the different levels of a factor. From the example above, suppose you find that as dosage increases, the percentage of people who suffer from seizures increases as well. You also notice that age does not play a role; both 20 and 40 year olds suffer the same percentage of seizures for a given amount of CureAll. From this information, you can conclude that the chance of a patient suffering a seizure is minimized at lower dosages of the drug (5 mg). The second graph illustrates that with increased drug dosage there is an increased percentage of seizures, while the first graph illustrates that with increased age there is no change in the percentage of seizures. Both of these graphs only contain one main effect, since only dose has an effect the percentage of seizures. Whereas, graphs three and four have two main effects, since dose and age both have an effect on the percentage of seizures.

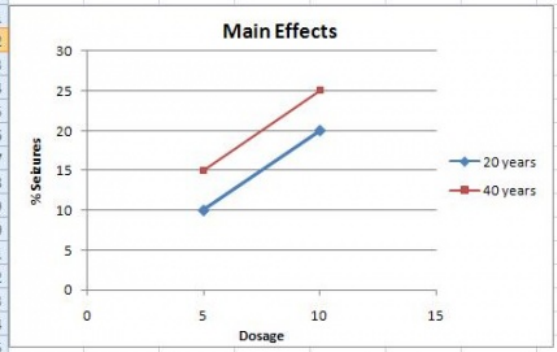
		Dosage	
		5 mg	10 mg
Age	20 years	10%	20%
	40 years	10%	20%





		Dosage	
		5 mg	10 mg
Age	20 years	10%	20%
	40 years	15%	25%



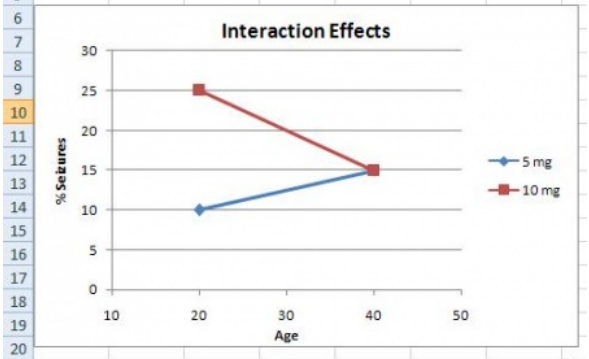


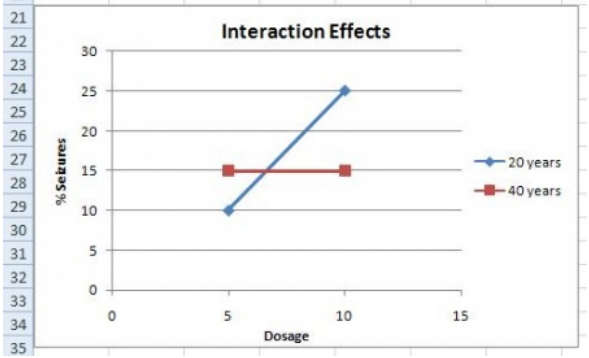
https://eng.libretexts.org/@go/page/122627

9.7.1.2.1.4: INTERACTION EFFECTS

The interaction effects situation is the last outcome that can be detected using factorial design. From the example above, suppose you find that 20 year olds will suffer from seizures 10% of the time when given a 5 mg CureAll pill, while 20 year olds will suffer 25% of the time when given a 10 mg CureAll pill. When 40 year olds, however, are given a 5 mg pill or a 10 mg pill, 15% suffer from seizures at both of these dosages. This correlation can be seen in the graphs below. There is an increasing chance of suffering from a seizure at higher doses for 20 year olds, but no difference in suffering from seizures for 40 year olds. Thus, there must be an interaction effect between the dosage of CureAll, and the age of the patient taking the drug. When you have an interaction effect it is impossible to describe your results accurately without mentioning both factors. You can always spot an interaction in the graphs because when there are lines that are not parallel an interaction is present. If you observe the main effect graphs above, you will notice that all of the lines within a graph are parallel. In contrast, for interaction effect graphs, you will see that the lines are not parallel.

		Dosage	
		5 mg	10 mg
Age	20 years	10%	25%
	40 years	15%	15%





9.7.1.2.2: MATHEMATICAL ANALYSIS APPROACH

In the previous section, we looked at a qualitative approach to determining the effects of different factors using factorial design. Now we are going to shift gears and look at factorial design in a quantitative approach in order to determine how much influence the factors in an experiment have on the outcome.

9.7.1.2.2.1: HOW TO DEAL WITH A 2^N FACTORIAL DESIGN

Suppose you have two variables $\boldsymbol{A}$ and $\boldsymbol{B}$ and each have two levels a_1, a_2 and b_1, b_2. You would measure combination effects of $\boldsymbol{A}$ and $\boldsymbol{B}$ (a_1b_1, a_1b_2, a_2b_1, a_2b_2). Since we have two factors, each of which has two levels, we say that we have a 2 x 2 or a 2^2 factorial design. Typically, when performing factorial design, there will be two levels, and n different factors. Thus, the general form of factorial design is 2^n.

In order to find the main effect of $\boldsymbol{A}$, we use the following equation:

$$A = (a_2b_1 - a_1b_1) + (a_2b_2 - a_1b_2)$$

Similarly, the main effect of B is given by:

$$B = (b_2a_1 - b_1a_1) + (b_2a_2 - b_1a_2)$$

By the traditional experimentation, each experiment would have to be isolated separately to fully find the effect on B. This would have resulted in 8 different experiments being performed. Note that only four experiments were required in factorial designs to solve for the eight

values in A and B. This shows how factorial design is a timesaver.

By taking the coefficients in A and B, the table below was created.

	Coefficient Effect			
	a_1b_1	a_2b_1	a_1b_2	a_2b_2
A	-1	+1	-1	+1
B	-1	-1	+1	+1
AB	+1	-1	-1	+1

AB is found by multiplying the coefficients of a_xb_x to get the new coefficient effect.

The additional complication is the fact that more than one trial/replication is required for accuracy, so this requires adding up each sub-effect (e.g adding up the three trials of a_1b_1). By adding up the coefficient effects with the sub-effects (multiply coefficient with sub-effect), a **total factorial effect** can be found. This value will determine if the factor has a significant effect on the outcome. For larger numbers, the factor can be considered extremely important and for smaller numbers, the factor can be considered less important. The sign of the number also has a direct correlation to the effect being positive or negative.

To get a mean factorial effect, the totals needs to be divided by 2 times the number of **replicates**, where a replicate is a repeated experiment.

$$\text{mean factorial effect} = \frac{\text{total factorial effect}}{2r}$$

By adding a third variable (C), the process of obtaining the coefficients becomes significantly complicated. The main factorial effect for A:

$$A = (a_2b_1c_1 - a_1b_1c_1) + (a_2b_2c_1 - a_1b_2c_1) + (a_2b_1c_2 - a_1b_1c_2) + (a_2b_2c_2 - a_1b_2c_2)$$

The table of coefficients is listed below

	Coefficient Effect							
	$a_1b_1c_1$	$a_2b_1c_1$	$a_1b_2c_1$	$a_2b_2c_1$	$a_1b_1c_2$	$a_2b_1c_2$	$a_1b_2c_2$	$a_2b_2c_2$
A	-1	+1	-1	+1	-1	+1	-1	+1
B	-1	-1	+1	+1	-1	-1	+1	+1
AB	+1	-1	-1	+1	+1	-1	-1	+1
C	-1	-1	-1	-1	+1	+1	+1	+1
AC	+1	-1	+1	-1	-1	+1	-1	+1
BC	+1	+1	-1	-1	-1	-1	+1	+1
ABC	-1	+1	+1	-1	+1	-1	-1	+1

It is clear that in order to find the total factorial effects, you would have to find the main effects of the variable and then the coefficients. Yates Algorithm can be used to simplify the process.

9.7.1.2.2.2: YATES ALGORITHM

Frank Yates created an algorithm to easily find the total factorial effects in a 2^n factorial that is easily programmable in Excel. While this algorithm is fairly straightforward, it is also quite tedious and is limited to 2^n factorial designs. Thus, modern technology has allowed for this analysis to be done using statistical software programs through regression.

Steps:

1. In the first column, list all the individual experimental combinations

According to the yates order, such as follows for a 2^3 factorial design

- - -

+ - -

- + -

+ + -

- - +

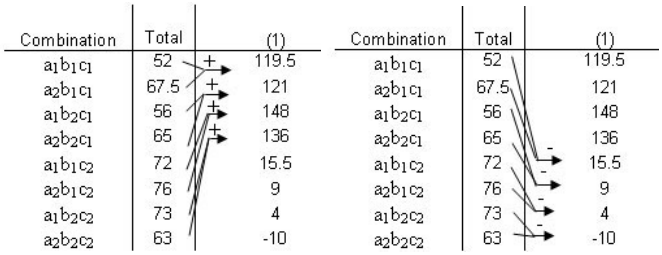
+ - +

- + +

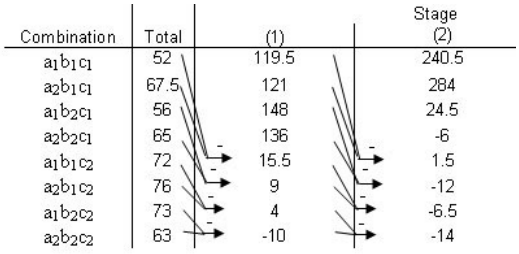
+ + +

2. In the second column, list all the totals for each combination

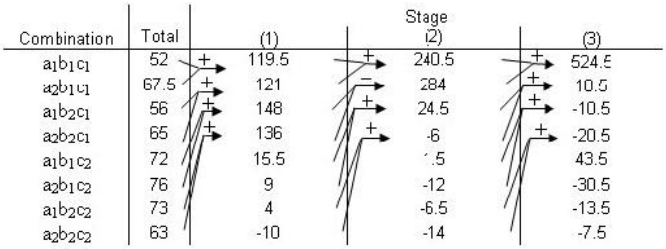
3. The 1st four entries in the third column (Stage 1) are obtained by adding pairs together from the "totals list" (previous column). The next four numbers are obtained by subtracting the top number from the bottom number of each pair.

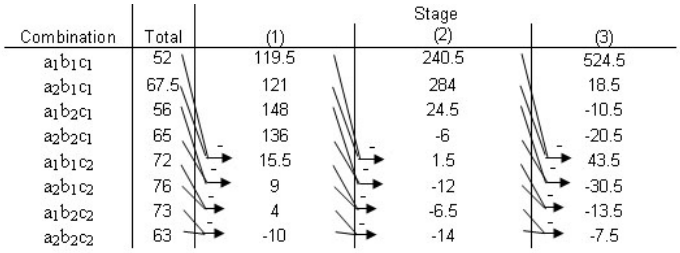


4. The fourth column (Stage 2) is obtained in the same fashion, but this time adding and subtracting pairs from Stage 1.



5. The fifth column (Stage 3) is obtained in the same fashion, but this time adding and subtracting pairs from Stage 2.





6. Continue with Stages until reaching *n*, or the number of factors. This final column is the Effect Total. A positive value means a positive correlation, and a negative values means a negative correlation. These values are all relative, however, so there is no way to compare different effect totals from different experiments.

Ignoring the first row, look in the last stage and find the variable that has the largest relative number, then that row indicates the **MAIN TOTAL EFFECT**. The Main Total Effect can be related to input variables by moving along the row and looking at the first column. If the row in the first column is $a_2b_1c_1$ then the main total effect is A. The row for $a_1b_2c_1$ would be for B. The row for $a_2b_1c_2$ would be for AC.

This main total effect value for each variable or variable combination will be some value that signifies the relationship between the output and the variable. For instance, if your value is positive, then there is a positive relationship between the variable and the output (i.e. as you increase the variable, the output increases as well). A negative value would signify a negative relationship. Notice, however, that the values are all relative to one another. So while the largest main total effect value in one set of experiments may have a value of 128, another experiment may have its largest main total effect value be 43. There is no way to determine if a value of 128 in one experiment has more control over its output than a value of 43 does, but for the purposes of comparing variables within an experiment, the main total effect does allow you to see the relative control the variables have on the output.

9.7.1.2.2.3: FACTORIAL DESIGN EXAMPLE REVISITED

Recall the example given in the previous section What is Factorial Design? In the example, there were two factors and two levels, which gave a 2^2 factorial design. The Yates Algorithm can be used in order to quantitatively determine which factor affects the percentage of seizures the most. For the use of the Yates algorithm, we will call age factor A with a_1 = 20 years, and a_2 = 40 years. Likewise, we will call dosage factor B, with b_1 = 5 mg, and b_2 = 10 mg. The data for the three outcomes is taken from the figures given in the example, assuming that the data given resulted from multiple trials.

Null Outcome

The following Yates algorithm table using the data for the null outcome was constructed. As seen in the table, the values of the main total factorial effect are 0 for A, B, and AB. This proves that neither dosage or age have any effect on percentage of seizures.

Combination	Total	Stage 1	Stage 2	Main Total Factorial Effect
a_1b_1	10	20	40	Doesn't Matter
a_2b_1	10	20	0	A
a_1b_2	10	0	0	B
a_2b_2	10	0	0	AB

Main Effect

The following Yates algorithm table using the data from the first two graphs of the main effects section was constructed. Besides the first row in the table, the row with the largest main total factorial effect is the B row, while the main total effect for A is 0. This means that dosage (factor B) affects the percentage of seizures, while age (factor A) has no effect, which is also what was seen graphically.

Combination	Total	Stage 1	Stage 2	Main Total Factorial Effect
a_1b_1	10	20	60	Doesn't Matter
a_2b_1	10	40	0	A
a_1b_2	20	0	20	B
a_2b_2	20	0	0	AB

The following Yates algorithm table using the data from second two graphs of the main effects section was constructed. Besides the first row in the table, the main total effect value was 10 for factor A and 20 for factor B. This means that both age and dosage affect percentage seizures. However, since the value for B is larger, dosage has a larger effect on percentage of seizures than age. This is what was seen graphically, since the graph with dosage on the horizontal axis has a slope with larger magnitude than the graph with age on the horizontal axis.

Combination	Total	Stage 1	Stage 2	Main Total Factorial Effect
a_1b_1	10	25	70	Doesn't Matter
a_2b_1	15	45	10	A
a_1b_2	20	5	20	B
a_2b_2	25	5	0	AB

Interaction Effect

The following Yates algorithm table was constructed using the data from the interaction effects section. Since the main total factorial effect for AB is non-zero, there are interaction effects. This means that it is impossible to correlate the results with either one factor or another; both factors must be taken into account.

Combination	Total	Stage 1	Stage 2	Main Total Factorial Effect
a_1b_1	10	25	65	Doesn't Matter
a_2b_1	15	40	-5	A
a_1b_2	25	5	15	B
a_2b_2	15	-10	-15	AB

9.7.1.2.3: CHEMICAL ENGINEERING APPLICATIONS

It should be quite clear that factorial design can be easily integrated into a chemical engineering application. Many chemical engineers face problems at their jobs when dealing with how to determine the effects of various factors on their outputs. For example, suppose that you have a reactor and want to study the effect of temperature, concentration and pressure on multiple outputs. In order to minimize the number

of experiments that you would have to perform, you can utilize factorial design. This will allow you to determine the effects of temperature and pressure while saving money on performing unnecessary experiments.

A 2007 study on converting wheat straw to fuel utilized factorial design to study the effect of four factors on the composition and susceptibility to enzyme hydrolysis of the final product. (Perez, et.al.). The table below shows the full factorial design for the study. The four factors that were studied all had only two levels and dealt with pretreatment parameters. They were: Water temperature, residence time, solid fraction and overpressure in the reactor. It is not necessary to understand what each of these are to understand the experimental design. As seen in the table below, there were sixteen trials, or 2^4 experiments.

Table 1. Full factorial design conditions for Liquid Hot Water pretreatment of wheat straw

Full factorial design				
Run	A: Temp (?C)	B: Time (min)	C: Overpressure (bar)	D: Subs.loading (% w/v)
1	170	0	0	5
2	200	0	0	5
3	170	40	0	5
4	200	40	0	5
5	170	0	30	5
6	200	0	30	5
7	170	40	30	5
8	200	40	30	5
9	170	0	0	10
10	200	0	0	10
11	170	40	0	10
12	200	40	0	10
13	170	0	30	10
14	200	0	30	10
15	170	40	30	10
16	200	40	30	10

Source: Perez, et. al.

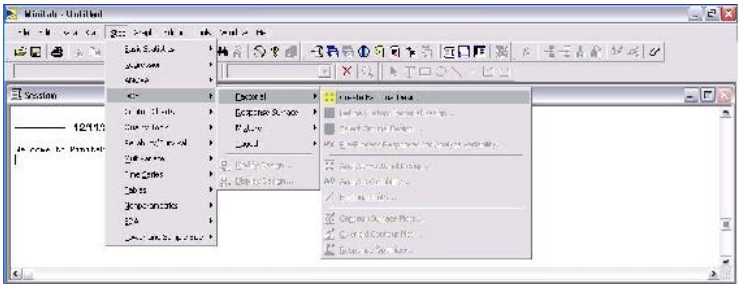
9.7.1.2.4: MINITAB DOE EXAMPLE

Minitab 15 Statistical Software is a powerful statistics program capable of performing regressions, ANOVA, control charts, DOE, and much more. Minitab is especially useful for creating and analyzing the results for DOE studies. It is possible to create factorial, response surface, mixture, and taguchi method DOEs in Minitab. The general method for creating factorial DOEs is discussed below.

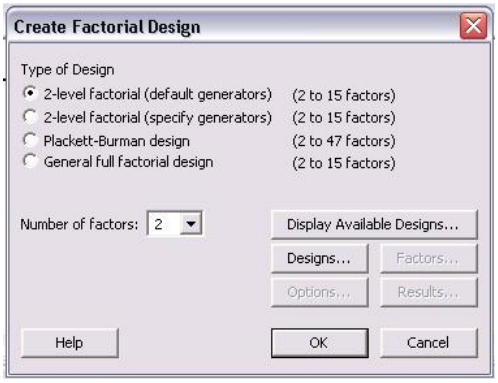
9.7.1.2.4.1: CREATING FACTORIAL DOE

Minitab provides a simple and user-friendly method to design a table of experiments. Additionally, analysis of multiple responses (results obtained from experimentation) to determine which parameters significantly affect the responses is easy to do with Minitab. Minitab 15 Statistical Software can be used via Virtual CAEN Labs by going to Start>All Programs>Math and Numerical Methods>Minitab Solutions>Minitab 15 Statistical Software.

The first step is creating the DOE by specifying the number of levels (typically 2) and number of responses. To do this, go to Stat>DOE>Factorial>Create Factorial Design as shown in the image below.

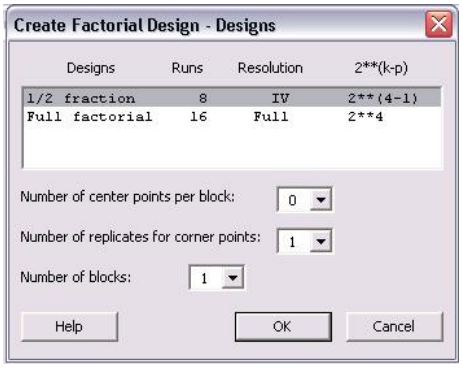


The next image is the "Create Factorial Design" options menu.



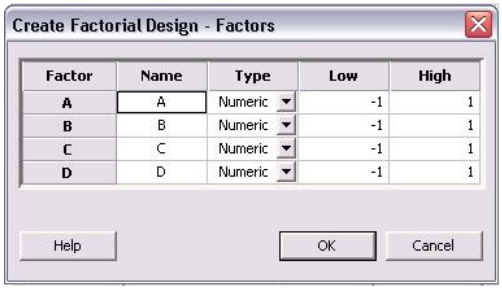
For a 2 level design, click the "2-level factorial (default generators)" radio button. Then specify the number of factors between 2 and 15. Other designs such as Plackett-Burman or a General full factorial design can be chosen. For information about these designs, please refer to the "Help" menu.

After the number of factors is chosen, click on the "Designs..." option to see the following menu.



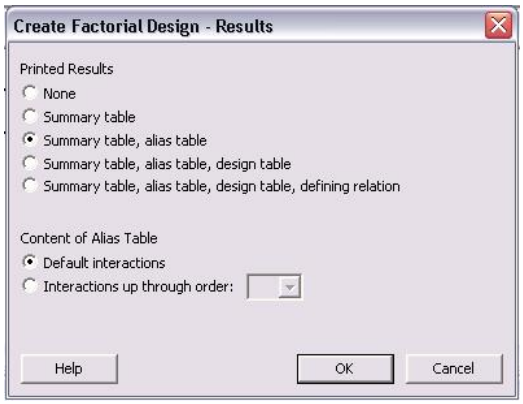
In this menu, a 1/2 fraction or full factorial design can be chosen. Although the full factorial provides better resolution and is a more complete analysis, the 1/2 fraction requires half the number of runs as the full factorial design. In lack of time or to get a general idea of the relationships, the 1/2 fraction design is a good choice. Additionally, the number of center points per block, number of replicates for corner points, and number of blocks can be chosen in this menu. Consult the "Help" menu for details about these options. Click "Ok" once the type of design has been chosen.

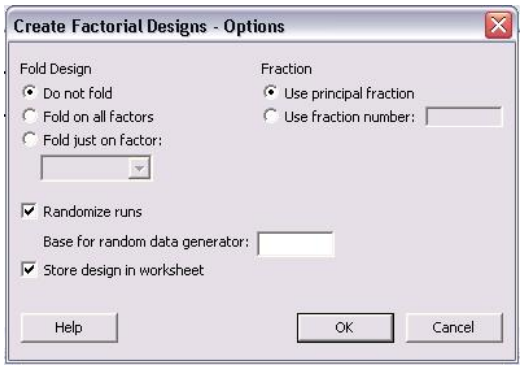
Once the design has been chosen, the "Factors...", "Options..." and "Results..." buttons become active in the "Create Factorial Designs" option menu. Click on "Factors..." button to see the following menu.



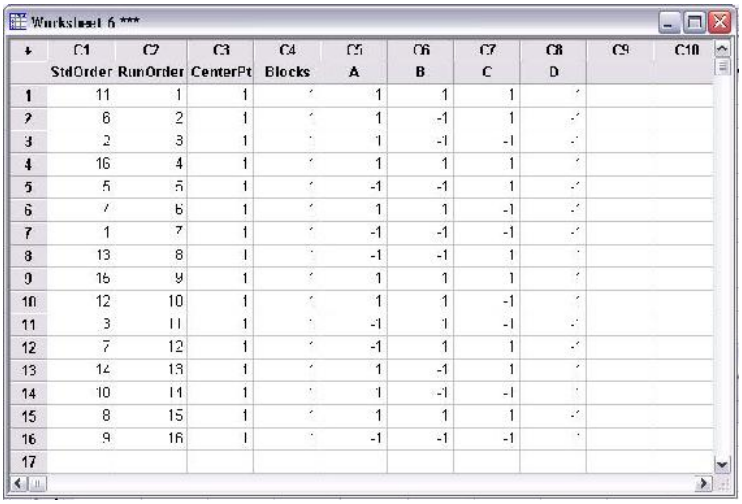
The above image is for a 4 factor design. Factors A - D can be renamed to represent the actual factors of the system. The factors can be numeric or text. Additionally, a low and high value are initially listed as -1 and 1, where -1 is the low and 1 is the high value. The low and high levels for each factor can be changed to their actual values in this menu. Click "OK" once this is completed.

The necessary steps for creating the DOE are complete, but other options for "Results..." and "Options..." can be specified. The menus for "Results..." and "Options..." are shown below.





In the main "Create Factorial Design" menu, click "OK" once all specifications are complete. The following table is obtained for a 2-level, 4 factor, full factorial design. None of the levels were specified as they appear as -1 and 1 for low and high levels, respectively.



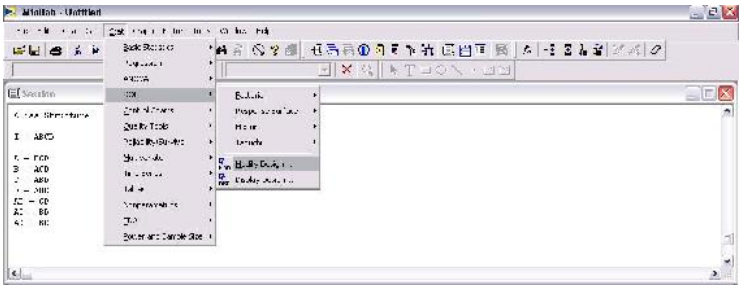
Worksheet 6 ***

	C1	C2	C3	C4	C5	C6	C7	C8	C9	C10
	StdOrder	RunOrder	CenterPt	Blocks	A	B	C	D		
1	11	1	1		1	1	1			
2	6	2	1		1	-1	1			
3	2	3	1		1	-1	-1			
4	16	4	1		1	1	1			
5	5	5	1		-1	-1	1			
6	7	6	1		1	1	-1			
7	1	7	1		-1	-1	-1			
8	13	8	1		-1	-1	1			
9	16	9	1		1	1	1			
10	12	10	1		1	1	-1			
11	3	11	1		-1	1	-1			
12	7	12	1		-1	1	1			
13	14	13	1		1	-1	1			
14	10	14	1		1	-1	-1			
15	8	15	1		1	1	1			
16	9	16	1		-1	-1	-1			
17										

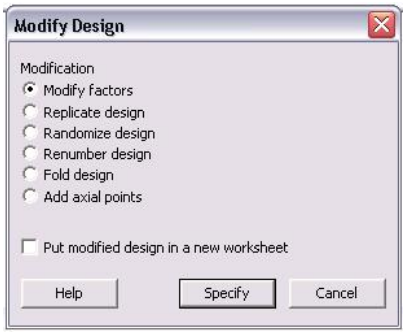
The above table contains all the conditions required for a full factorial DOE. Minitab displays the standard order and randomized run order in columns C1 and C2, respectively. Columns A-D are the factors. The first run (as specified by the random run order) should be performed at the low levels of A and C and the high levels of B and D. A total of 16 runs are required to complete the DOE.

9.7.1.2.4.2: MODIFYING DOE TABLE

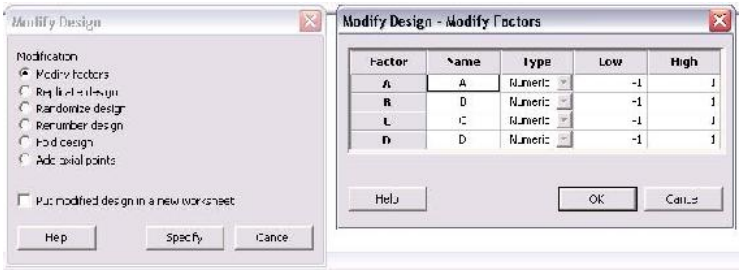
Once a table of trials for the DOE has been created, additional modifications can be made as needed. Some typical modifications include modifying the name of each factors, specifying the high and low level of each factor, and adding replicates to the design. To being modifications of a current design, go to Stat>DOE>Modify Design... as seen in the figure below.



The following menu is displayed for modifying the design.

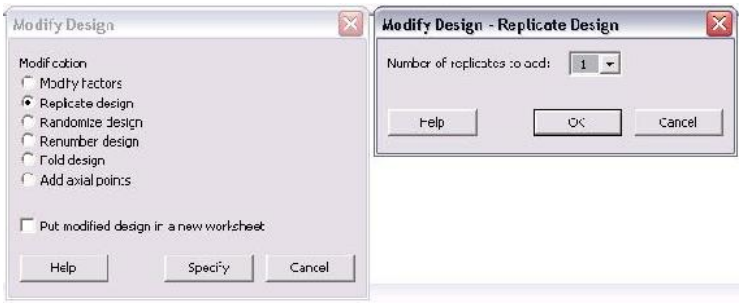


In the "Modify Design" menu, users can modify factors, replicate design, randomize design, renumber design, fold design, and add axial points. Additionally, any changes made can be put into a new worksheet. To change the factors, click the "Modify factors" radio button and then "Specify" to see the following options menu.



The default factors are named "A", "B", "C", and "D" and have respective high and low levels of 1 and -1. The name of the factors can be changed by simply clicking in the box and typing a new name. Additionally, the low and high levels for each factor can be modified in this menu. Since the high and low levels for each factor may not be known when the design is first created, it is convenient to be able to define them later. Click "OK" after modifications are complete.

Another typical modification is adding replicates to a design. Replicates are repeats of each trial that help determine the reproducibility of the design, thus increasing the number of trials and accuracy of the DOE. To add replicates, click the "Replicate design" radio button in the "Modify Design" menu. The following menu will be displayed.

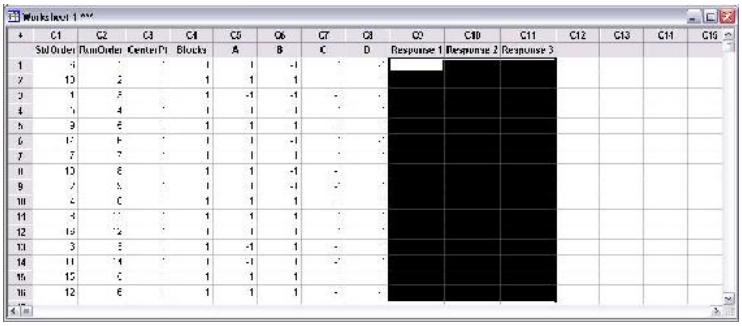


The only option in this menu is the number of replicates to add. The number ranges between 1 and 10. To have a total of 3 trials of each, the user should add 2 replicates in this menu. If 4 replicates are added, there will be a total of 5 trials of each. Typically, if the same experimentation will occur for 3 lab periods, 2 replicates will be added.

Additional modifications to the design include randomizing and renumbering the design. These are very straightforward modifications which affect the ordering of the trials. For information about the "Fold design" and "Add axial points", consult the "Help" menu.

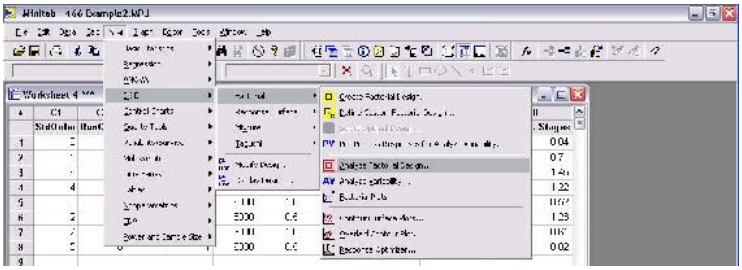
9.7.1.2.4.3: ANALYZING DOE RESULTS

After the complete DOE study has been performed, Minitab can be used to analyze the effect of experimental results (referred to as responses) on the factors specified in the design. The first step in analyzing the results is entering the responses into the DOE table. This is done much like adding data into an Excel data sheet. In the columns to the right of the last factor, enter each response as seen in the figure below.

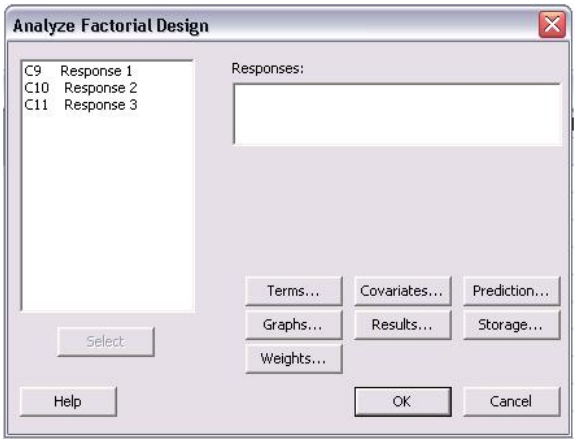


The above figure contains three response columns. The names of each response can be changed by clicking on the column name and entering the desired name. In the figure, the area selected in black is where the responses will be inputted. For instance, if the purity, yield, and residual amount of catalyst was measured in the DOE study, the values of these for each trial would be entered in the columns.

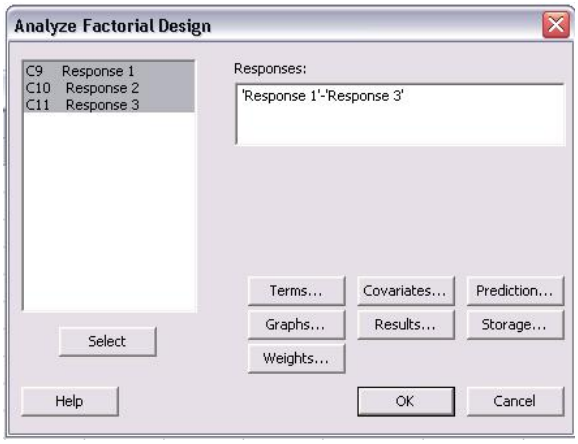
Once the responses are entered, statistical analysis on the data can be performed. Go to Stat>DOE>Factorial>Analyze Factorial Design... as seen in the following image.



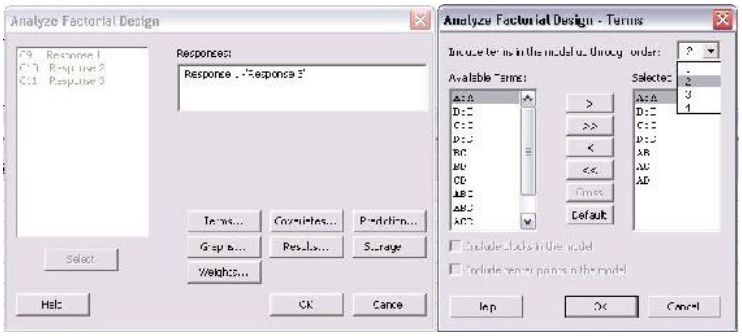
The menu that appears for analyzing factorial design is shown below.



In the "Analyze Factorial Design" menu, the responses are shown on the left of the screen. The first step is to choose the responses to be analyzed. All of the responses can be chosen at once or individually. To choose them, click (or click and drag to select many) and then click "Select" to add them into the "Responses:" section as seen below.

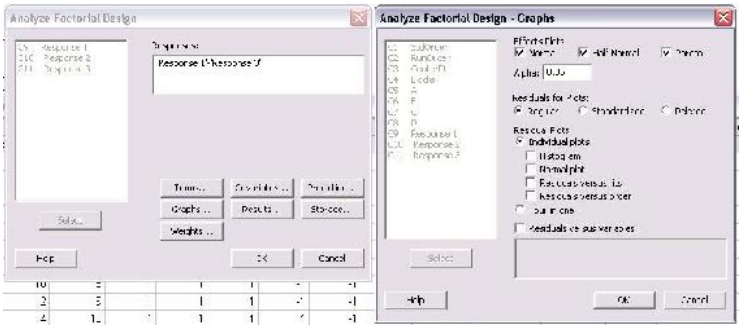


The next step is selecting which terms will be analyzed for the responses. To do this, click on "Terms..." and the following menu will appear.



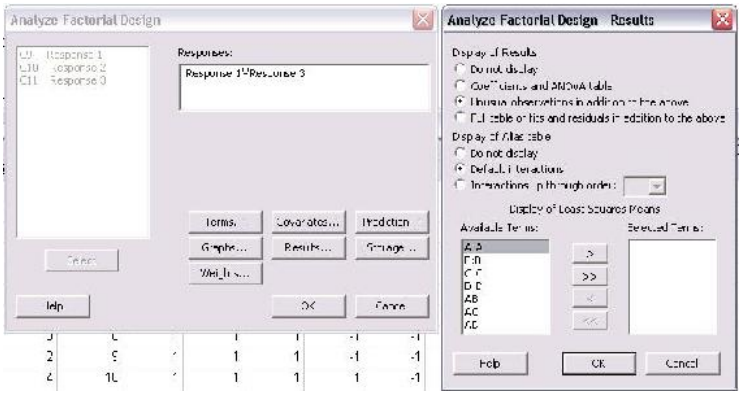
The types of interactions between factors are chosen in this menu. For a first order model which excludes all factor-to-factor interactions, "1" should be chosen from the drop-down menu for "Include terms in the model up through order:". To include higher order terms and account for factor interactions, choose 2, 3, or 4 from the drop-down menu. Unless significant factor-to-factor interactions are expected, it is recommended to use a first order model which is a linear approximation.

Once the terms have been chosen, the next step is determining which graphs should be created. The types of graphs can be selected by clicking on "Graphs..." in the main "Analyze Factorial Design" menu.



In the Graphs menu shown above, the three effects plots for "Normal", "Half Normal", and "Pareto" were selected. These plots are different ways to present the statistical results of the analysis. Examples of these plots can be found in the Minitab Example for Centrifugal Contactor Analysis. The alpha value, which determines the limit of statistical significance, can be chosen in this menu also. Typically, the alpha value is 0.05. The last type of plots that can be chosen is residual plots. A common one to select is "Residuals versus fits" which shows how the variance between the predicted values from the model and the actual values.

The final option that must be specified is results. Click "Results..." from the "Analyze Factorial Design" menu to see the following screen.



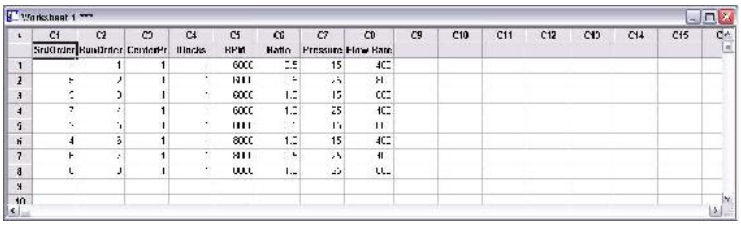
In this menu, select all of the "Available Terms" and click the ">>" button to move them to the "Selected Terms". This will ensure that all the terms will be included in the analysis. Another feature that can be selected from this menu is to display the "Coefficients and ANOVA table" for the DOE study.

Other options can be selected from the "Analyze Factorial Design" menu such as "Covariates...", "Prediction...", "Storage...", and "Weights...". Consult the "Help" menu for descriptions of the other options. Once all desired changes have been made, click "OK" to perform the analysis. All of the plots will pop-up on the screen and a text file of the results will be generated in the session file.

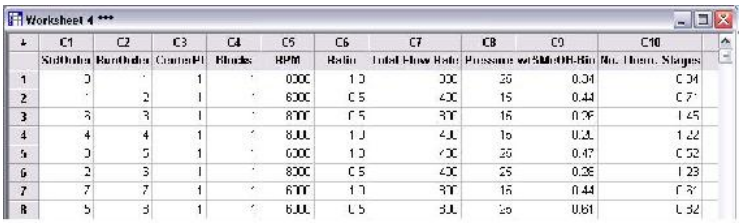
9.7.1.2.4.4: MINITAB EXAMPLE FOR CENTRIFUGAL CONTACTOR ANALYSIS

Centrifugal Contactors, also known as Podbielniak (POD) centrifugal contactors, are used to purify a contaminated stream by counter-current, liquid-liquid extraction. Two immiscible fluids with different specific gravities are contacted counter-currently and the solute from the dirty stream is extracted by the clean stream. A common use for PODs methanol removal from biodiesel by contacting the stream with water. The amount of methanol remaining in the biodiesel (wt% MeOH) after the purification and the number of theoretical stages (No. Theor. Stages) obtained depend on the operating conditions of the POD. The four main operating parameters of the POD are rotational speed (RPM), ratio of biodiesel to water (Ratio), total flow rate of biodiesel and water (Flow Rate), and pressure (Pressure). A DOE study has been performed to determine the effect of the four operating conditions on the responses of wt% MeOH in biodiesel and number of theoretical stages achieved. (NOTE: The actual data for this example was made-up)

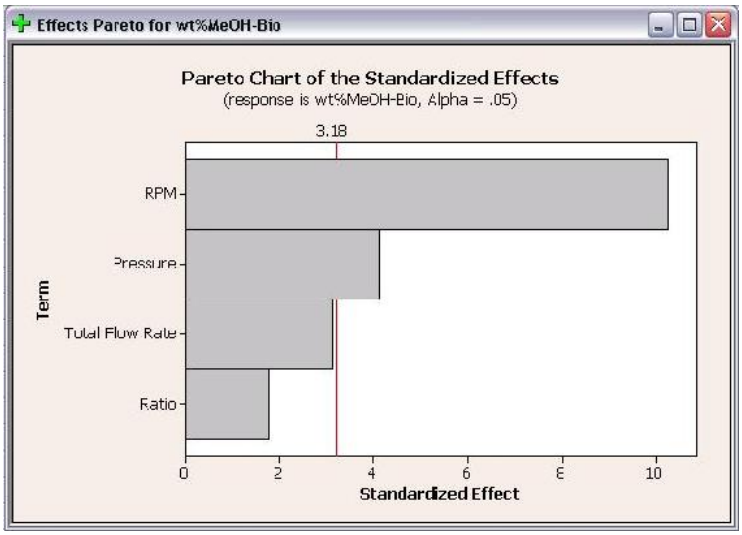
A 4-factor, 2-level DOE study was created using Minitab. Because experiments from the POD are time consuming, a half fraction design of 8 trial was used. The figure below contains the table of trials for the DOE.

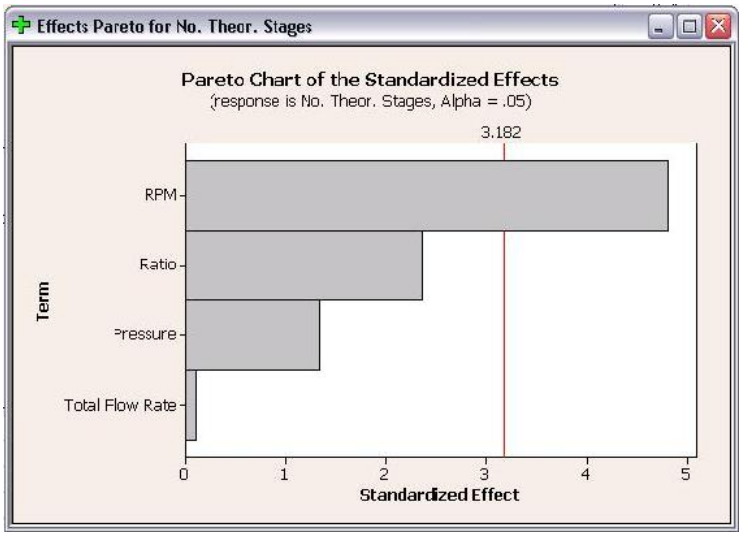


After all the trials were performed, the wt% methanol remaining in the biodiesel and number of theoretical stages achieved were calculated. The figure below contains the DOE table of trials including the two responses.



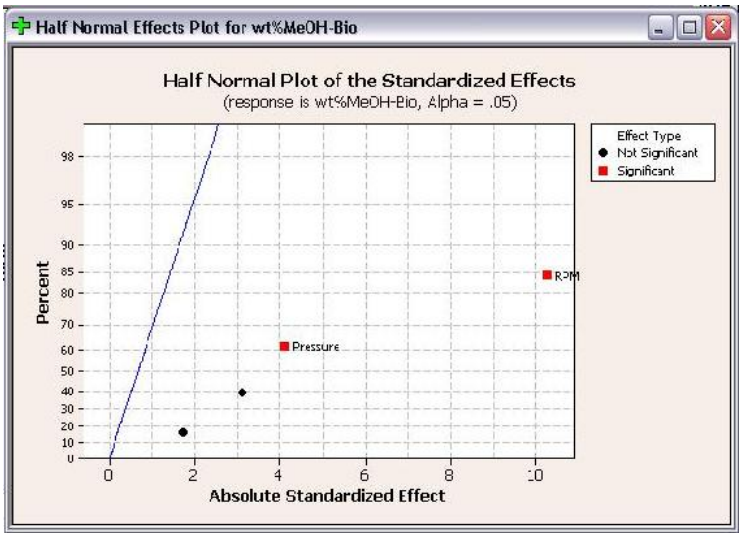
Analysis was performed on the DOE study to determine the effects of each factor on the responses. Only first order terms were included in the analysis to create a linear model. Pareto charts for both wt% MeOH in biodiesel and number of theoretical stages are shown below.

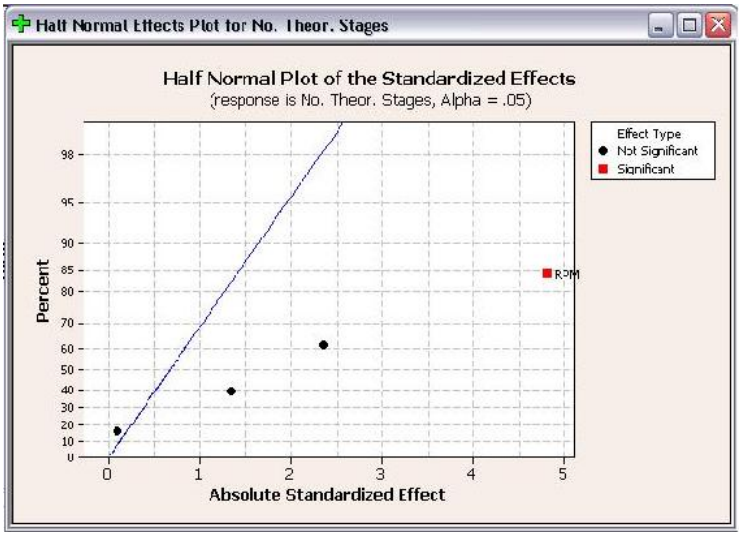




The Pareto charts show which factors have statistically significant effects on the responses. As seen in the above plots, RPM has significant effects for both responses and pressure has a statistically significant effect on wt% methanol in biodiesel. Neither flow rate or ratio have statistically significant effects on either response. The Pareto charts are bar charts which allow users to easily see which factors have significant effects.

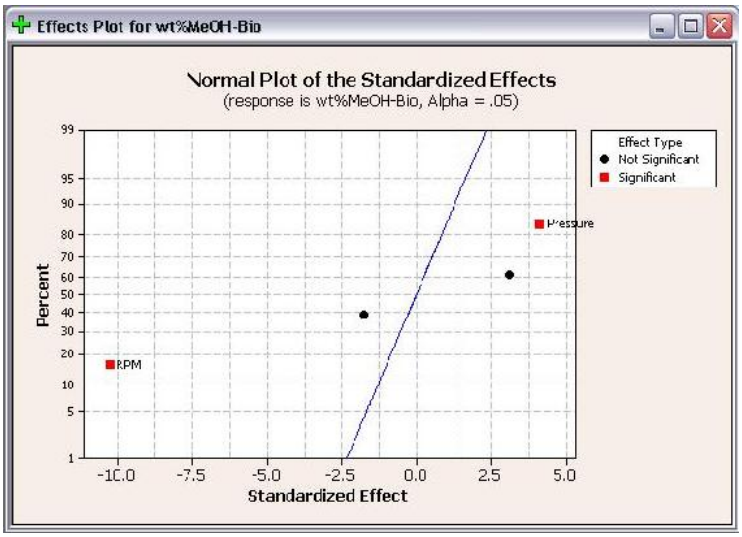
Half Normal Plots for wt% methanol in biodiesel and number of theoretical stages are shown below.

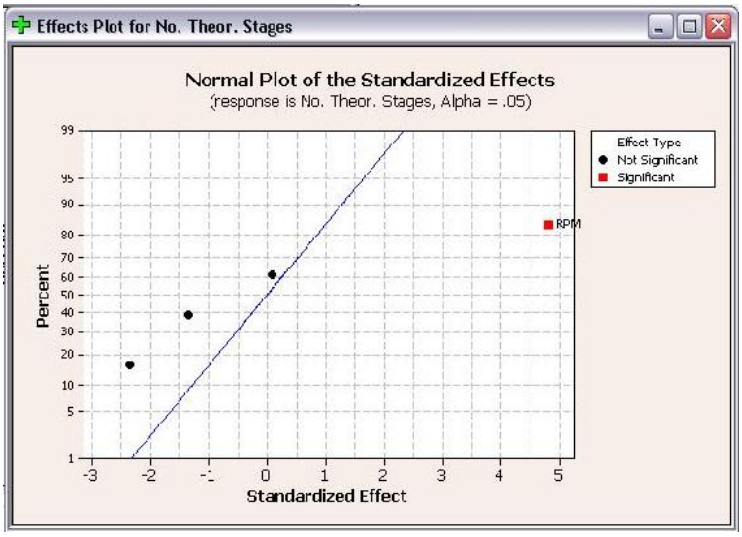




Like Pareto plots, Half Normal plots show which factors have significant effects on the responses. The factors that have significant effects are shown in red and the ones without significant effects are shown in black. The further a factor is from the blue line, the more significant effect it has on the corresponding response. For wt% methanol in biodiesel, RPM is further from the blue line than pressure, which indicates that RPM has a more significant effect on wt% methanol in biodiesel than pressure does.

The final plot created is the Normal Effect Plot. The Normal Plot is similar to the Half Normal plot in design. However, the Normal Plot displays whether the effect of the factor is positive or negative on the response. The Normal Plots for the responses are shown below.





As seen above, RPM is shown with a positive effect for number of theoretical stages, but a negative effect for wt% methanol in biodiesel. A positive effect means that as RPM increases, the number of theoretical stages increases. Whereas a negative effect indicates that as RPM increases, the wt% methanol in biodiesel decreases. Fortunately for operation with the POD, these are desired results. When choosing operating conditions for the POD, RPM should be maximized to minimize the residual methanol in biodiesel and maximize the number of theoretical stages achieved.

In addition to the above effects plots, Minitab calculates the coefficients and constants for response equations. The response equations can be used as models for predicting responses at different operating conditions (factors). The coefficients and constants for wt% methanol in biodiesel and number of theoretical stages are shown below.

Estimated Coefficients for wt%MeOH-Bio using data in uncoded units

Term	Coef
Constant	0.926833
RPM	-1.09592E-04
Ratio	-0.0748768
Total Flow Rate	0.000165648
Pressure	0.00879626

Estimated Coefficients for No. Theor. Stages using data in uncoded units

Term	Coef
Constant	-0.239346
RPM	0.000260115
Ratio	-0.509538
Total Flow Rate	0.000025577
Pressure	-0.0144769

Since this is a first order, linear model, the coefficients can be combined with the operating parameters to determine equations. The equations from this model are shown below.

$$wt\%MeOH = 0.927 - 1.096 \cdot 10^{-4}(RPM) - 0.075(Ratio) + 1.655 \cdot 10^{-4}(FlowRate) + 8.796 \cdot 10^{-5}(Pressure)$$

$$NoStages = -0.229 + 2.601 \cdot 10^{-4}(RPM) - 0.510(Ratio) + 2.558 \cdot 10^{-5}(FlowRate) - 0.014(Pressure)$$

These equations can be used as a predictive model to determine wt% methanol in biodiesel and number of theoretical stages achieved at different operating conditions without actually performing the experiments. However, the limits of the model should be tested before the model is used to predict responses at many different operating conditions.

EXAMPLE 9.7.1.2.1

You have been employed by SuperGym, a local personal training gym, who want an engineer's perspective on how to offer the best plans to their clients. SuperGym currently categorizes her clients into 4 body types to help plan for the best possible program.

- Type 1 - Very healthy
- Type 2 - Needs tone
- Type 3 - Needs strength
- Type 4 - Needs tone and strength

In addition, SuperGym offers 4 different workout plans, A through D, none of which are directly catered to any of the different types. Create an experimental factorial design that could be used to test the effects of the different workout plans on the different types of people at the gym.

Solution

In order to solve this problem, we need to determine how many different experiments would need to be performed. In order to solve this, we can see that we have two different factors, body type and workout plan. For each factor, there exist four different levels. Thus, we have a 4^2 factorial design, which gives us 16 different experimental groups. Creating a table of all of the different groups, we arrive at the following factorial design:

Solution

A1	B1	C1	D1
A2	B2	C2	D2
A3	B3	C3	D3
A4	B4	C4	D4

Where A-D is the workout plan and 1-4 is the types

EXAMPLE 9.7.1.2.2

Suppose that you are looking to study the effects of hours slept (A), hours spent with significant other (B), and hours spent studying (C) on a students exam scores. You are given the following table that relates the combination of these factors and the students scores over the course of a semester. Use the Yates method in order to determine the effect each variable on the students performance in the course.

Given Information

1	17	24	19	21	22	28	25	24
2	18.5	21	20	19	26	22	27	19
3	16.5	22.5	22	25	24	26	21	20
Total	52	67.5	61	65	72	76	73	63

Solution

Using the approach introduced earlier in this article, we arrive at the following Yates solution.

		Stage			Main Total
Combination	Total	1	2	3	Factorial Effect
$a_1b_1c_1$	52	119.5	245.5	529.9	Doesn't matter
$a_2b_1c_1$	67.5	126	284	13.5	A
$a_1b_2c_1$	61	148	19.5	-5.5	B
$a_2b_2c_1$	65	136	-6	-25.5	AB
$a_1b_1c_2$	72	15.5	6.5	38.5	C
$a_2b_1c_2$	76	4	-12	-25.5	AC
$a_1b_2c_2$	73	4	-11.5	-18.5	BC
$a_2b_2c_2$	63	-10	-14	-2.5	ABC

From this table, we can see that there is positive correlation for factors A and C, meaning that more sleep and more studying leads to a better test grade in the class. Factor B, however, has a negative effect, which means that spending time with your significant other leads to a worse test score. The lesson here, therefore, is to spend more time sleeping and studying, and less time with your boyfriend or girlfriend.

EXAMPLE 9.7.1.2.3

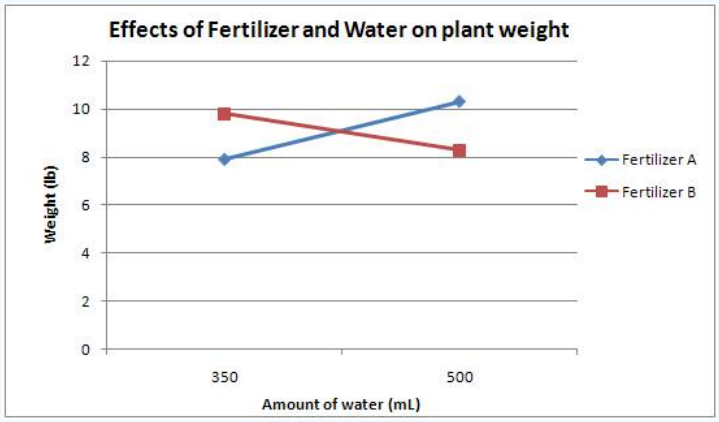
Your mom is growing a garden for the state fair and has done some experiments to find the ideal growing condition for her vegetables. She asks you for help interpreting the results and shows you the following data:

Plant No.	Fertilizer Type	Amount of water (mL)	Weight (lb)
1	A	350	7.9
2	B	350	9.8
3	A	500	10.3
4	B	500	8.3

Make plots to determine the main or interaction effects of each factor.

Solution

Here is the plot you should have gotten for the given data.



From this one can see that there is an interaction effect since the lines cross. One cannot discuss the results without speaking about both the type of fertilizer and the amount of water used. Using fertilizer A and 500 mL of water resulted in the largest plant, while fertilizer A and 350 mL gave the smallest plant. Fertilizer B and 350 mL gave the second largest plant, and fertilizer B and 500 mL gave the second smallest plant. There is clearly an interaction due to the amount of water used and the fertilizer present. Perhaps each fertilizer is most effective with a certain amount of water. In any case, your mom has to consider both the fertilizer type and amount of water provided to the plants when determining the proper growing conditions.

? EXERCISE 9.7.1.2.1

Which of the following is not an advantage of the use of factorial design over one factor design?

a. More time efficient
b. Provides how each factor effects the response
c. Does not require explicit testing
d. Does not require regression

Answer

TBA

? EXERCISE 9.7.1.2.2

In a 2^2 factorial design experiment, a total main effect value of -5 is obtained. This means that

a. there is a relative positive correlation between the two factors
b. there is no correlation between the two factors
c. there is a relative negative correlation between the two factors
d. there is either a positive or negative relative correlation between the two factors

Answer

TBA

9.7.1.2.5: REFERENCES

- Box, George E.P., et. al. "Statistics for Engineers: An Introduction to Design, Data Analysis, and Model Building." New York: John Wiley & Sons.
- Trochim, William M.K. 2006. "Factorial Designs." Research Methods Knowledge Base. <http://www.socialresearchmethods.net/kb/expfact.htm>
- Perez, Jose A., et. al. "Effect of process variables on liquid hot water pretreatment of wheat straw for bioconversion to fuel-ethanol in a batch reactor." Journal of Chemical Technology & Biotechnology. Volume 82, Issue 10, Pages 929-938. Published Online Sep 3, 2007.

This page titled 9.7.1.2: Design of experiments via factorial designs is shared under a CC BY 4.0 license and was authored, remixed, and/or curated by Jocelyn Anleitner, Stephanie Combs, Diane Feldkamp, Heeral Sheth, Jason Bourgeois, Michael Kravchenko, Nicholas Parsons, Andrew Wang, & Andrew Wang via source content that was edited to the style and standards of the LibreTexts platform.

- **14.2: Design of experiments via factorial designs** by Andrew Wang, Diane Feldkamp, Heeral Sheth, Jason Bourgeois, Jocelyn Anleitner, Michael Kravchenko, Nicholas Parsons, Stephanie Combs is licensed CC BY 3.0. Original source: https://open.umn.edu/opentextbooks/textbooks/chemical-process-dynamics-and-controls.

9.7.1.3: DESIGN OF EXPERIMENTS VIA RANDOM DESIGN

9.7.1.3.1: INTRODUCTION

Random design is an approach to designing experiments. As the name implies, random experimental design involves randomly assigning experimental conditions. However, numbers should not be picked without any thought. This type of experimental design is surprisingly powerful and often results in a high probability to create a near optimal design.

The simplified steps for random design include the following:

1. Choose a number of experiments to run (NOTE: This may be tricky to pick a number because it is dependent upon the amount of signal recovery you want.)
2. Assign to each variable a state based on a uniform sample. For instance, if there are 5 states, each state has a probability of 20%.

Random designs typically work well for large systems with many variables, 50 or more. There should be few interactions between variables and very few variables that contribute significantly. Random design does not work very well with relatively smaller systems. Generally speaking, Taguchi and random designs often perform better than factorial designs depending on size and assumptions. When choosing the design for an experiment, it is important to determine an efficient design that helps optimize the process and determines factors that influence variability.

There is more than one type of random design, randomized block design and completely randomized design. Randomized block design involves blocking, which is arranging experimental units into groups so they have a common similarity. The blocking factor is usually not a primary source of variability. An example of a blocking factor may include eye color of a patient, so if this variability source is controlled, greater precision is achieved. Completely randomized design is where the groups are chosen at random.

In various technological fields, it is important to design experiments where a limited number of experiments is required. Random design is practical for many design applications. Extensive mathematical theory has been used to explore random experimental design. Examples of random design include areas of data compression and medical imaging. The research conducted to support the practical application of random design can be found at <http://groups.csail.mit.edu/drl/journal_club/papers/CS2-Candes-Romberg-05.pdf>.

Other research has been conducted recently on random design, and more information can be found at: ieeexplore.ieee.org/xpls/abs_all.jsp?arnumber=1614066

9.7.1.3.2: COMPLETELY RANDOMIZED DESIGN (CRD)

9.7.1.3.2.1: DESCRIPTION OF DESIGN

Completely randomized design (CRD) is the simplest type of design to use. The most important requirement for use of this design is homogeneity of experimental units.

9.7.1.3.2.2: PROCEDURE FOR RANDOMIZATION

1. Assign treatments to experimental units completely at random.
2. Verify that every experimental unit has the same probability of receiving any treatment.
3. Perform randomization by using a random number table, computer, program, etc.

9.7.1.3.2.3: EXAMPLE OF CRD

If you have 4 treatments (I, II, III, IV) and 5 replicates, how many experimental units do you have?

{I} {IV} {III} {II} {II} {III} {III} {II} {I} {III} {I} {IV} {III} {IV} {I} {IV} {II} {I} {II} {IV} **=20 randomized experimental units**

9.7.1.3.3: RANDOMIZED BLOCK DESIGN (RBD)

9.7.1.3.3.1: DESCRIPTION OF DESIGN

Randomized block design (RBD) takes advantage of grouping similar experimental units into blocks or replicates. One requirement of RBd is that the blocks of experimental units be as uniform as possible. The reason for grouping experimental units is so that the observed differences between treatments will be largely due to “true” differences between treatments and not random occurrences or chance.

9.7.1.3.3.2: PROCEDURE FOR RANDOMIZATION

1. Randomize each replicate separately.
2. Verify that each treatment has the same probability of being assigned to a given experimental unit within a replicate.
3. Have each treatment appear at least once per replicate.

9.7.1.3.3.3: ADVANTAGES OF RBD

1. Generally more precise than the CRD.
2. Some treatments may be replicated more times than others.
3. Missing plots are easily estimated.
4. Whole treatments or entire replicates may be deleted from the analysis.

This page titled 9.7.1.3: Design of Experiments via Random Design is shared under a CC BY 4.0 license and was authored, remixed, and/or curated by Peter Woolf et al. via source content that was edited to the style and standards of the LibreTexts platform.

- **14.3: Design of Experiments via Random Design** by Peter Woolf et al. is licensed CC BY 3.0. Original source: https://open.umn.edu/opentextbooks/textbooks/chemical-process-dynamics-and-controls.

9.7.1.4: SUMMARY- SUMMARY ON CONTROL ARCHITECTURES' PHILOSOPHIES, ADVANTAGES, AND DISADVANTAGES.

9.7.1.4.1: SUMMARY ON CONTROL ARCHITECTURES

A summary of the philosophies, advantages, and disadvantages of the different control architectures is shown below in Table 1.

Table 1. Control architectures' philosophy, advantages, and disadvantages.

	Philosophy	Advantages	Disadvantages
Feedback	Control adjusts errors as they occur	• simple to design • no process model needed	• connects for error after they have already occurred • usually only takes input from one sensor
Feed Forward	Control connects errors they occur	• corrects potential errors before they occur • Ideally can produce a perfect control where there is never any offset	• infinitely accurate models are needed • infinitely accurate measurements are needed
Ratio	Controls connects two flow to maintain a constant ratio	• Generate defined ratio • Simple Model	• The flow you control will never truly measured and ratio model is a assumed to be accurate • does not take into consideration of varying pressure
Cascade	Controls set points of sensors by controlling other sensors to integrate multiple information	• Responds quickly to high frequency changes • Incorporate multiple sensor readings together (incidence graphs)	• can oe very complex model • tuning model is difficult due to multiple set points and parameters
Mixed	Control integrating combinations of control architectures	• pick the right combination of controls to accomplish main control in zo objective • incorporate any number of sensors in a logical manner	• even more complex model than cascade alone • tuning model is even more complex than cascade alone, Zeigler Nichols tuning won't work by itself

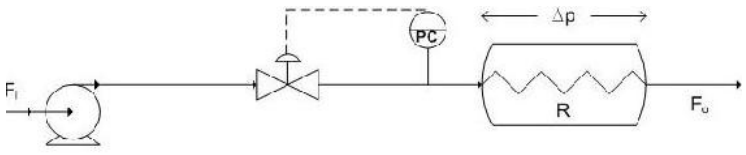
9.7.1.4.2: INTRODUCTION

Liquid level and liquid pressure control are fundamental aspects of many control processes because of the reliance on the stability and convenience of liquid usage. The following article will introduce the main concepts behind liquid pressure control, liquid level control, examining the loop characteristics, response, tuning and limitations. Two models are included to illustrate the application of a liquid pressure control loop and a level control loop in idealized systems.

9.7.1.4.3: PRESSURE CONTROL BASICS

Like its counterparts of temperature, level and flow, pressure is one of the most common process variables. Pressure is a key process variable as it provides a critical condition for processes such as any chemical reaction, extrusion, boiling, distillation, air conditioning and vacuuming. Poor pressure control may likely lead to problems associated with quality, safety and productivity. For example, the presence of high-pressure conditions in a sealed vessel may result in an explosion and thus it is imperative that pressure be kept under good control within its safety limits.

Liquid pressure control is one of the easier control loops in the sense that it is shares many of the same characteristics of a common flow loop. The aim of the liquid pressure loop is to control the pressure at the needed set point pressure by controlling the flow with changing process needs. Because dealing with a liquid, the flow present is that of an incompressible fluid and as a result the pressure changes very quickly leading to a fast-responding process with small dead time and capacitance. This process behaves as a fixed restriction where the change in pressure is a function of the flow through the process. Illustrated below is a diagram with liquid pressure control on the pipe leading to the process.



There are a few additional considerations for the aforementioned liquid pressure loop. Firstly the controller can be proportional plus integral (PI) or integral only (I-only) if $K_c < 2$ otherwise a proportional only (P-only) controller is to be used. This controller is tuned similarly to the flow controller. Another consideration is that process gain is not constant, a square-root extractor or the highest loop gain should be used in

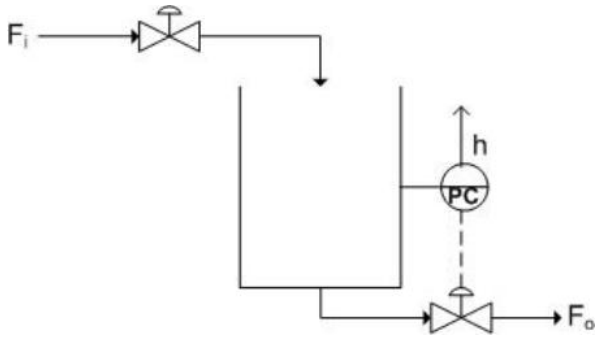
tuning the controller. This highest loop gain is employed to prevent the process loop from ever becoming unstable. Finally, this liquid pressure loop is noisy like its flow loop counterpart and as a result it is recommended that derivative action in the controller not be used.

If you wish to see the various issues associated with pressure controllers please refer to previous wiki articles.

9.7.1.4.4: LEVEL CONTROL BASICS

Controlling the level of liquid in a tank is one of the most basic requirements for a chemical process to run effectively. Most chemical engineering processes require some sort of liquid to be held for usage, and the most convenient way of ensuring the proper amount of liquid is available is through level control. A specific example of a process that requires level control is seen around the world. Water towers must have a designated amount of water within them to ensure sufficient water pressure to surrounding neighborhoods and hence liquid level control (LLC) is used.

Fortunately Liquid level control is one of the easier control loops available. The systems are fairly simple, a sensor reads the level of liquid within a tank, which then is conveyed to the controller. Subsequently, the controller changes a valve to increase or decrease the flowrate into or out of the tank, depending on the required action and controlled valve placement. Below is a figure with LLC on the pipe leaving the tank.



The ease of LLC also lies in its large capacitance and nominal dead time. Usually hold-up times vary from 5-15 minutes. Problems may arise when signal noise becomes a factor, as it usually does with level controllers. Refer to previous wiki articles to the issues associated with level controllers(), however it is important to note that a measurement of weight instead of directly measuring level can be a effective way of eliminating noise.

9.7.1.4.4.1: P-ONLY CONTROLLERS

Oftentimes using a P-only controller is the best way to ensure proper level control, in most cases only a small percent error will result, and it reduces problems associated with noise. P-only controllers should only be used when gain is very small and the tank has a large capacity. This will be the basis for the rest of the text.

A P-only controller works off the following principle of control:

$$\text{Output} = \text{Gain} \times \text{Error} + \text{Bias}$$

The output directly effects a valve to control the flowrate into or out of the tank. Refer to the previous wiki articles to learn more about P-only control.

9.7.1.4.4.2: LEVEL MEASUREMENT NOISE

When controlling liquid levels in drums or vessels, an important aspect to consider is that there may be noise in the level measurement due to disturbances such surface turbulence, boiling of the liquid, or agitation. Due to the existence of this noise, using a derivative action controller may not be appropriate. However, if this type of controller action is used, there are various methods that may be employed to minimize the noise.

Method 1: Use a displacer in a stilling well

- *Advantage:* Filters high frequency noise due to turbulence in the tank
- *Disadvantage:* Bobbing level due to the low-frequency movement of tank liquid into the well resulting from the formation of a U-tube between tank and well

Method 2: Use an ultrasonic level measurement with electronic filtering of the signal

- *Advantage:* Works well when the level response period is much lower than the noise frequency

Method 3: Use a tank weighing method

In the this method, a loading device is placed beneath the tank supports to measure the mass of the tank. Output weight values are averaged by a transmitter which are then sent to a converter which outputs the corresponding level to the controller.

- *Advantage:* Effectively eliminates noise in the level measurement due to turbulence

9.7.1.4.5: MODELS

The following are Excel models of Pressure and Level control in simple systems. Proportional only control is used primarily for these models. Controller noise has been eliminated for sake of simplicity.

9.7.1.4.5.1: LIQUID PRESSURE CONTROL MODEL

The model at hand illustrates both the feed backwards system of the liquid pressure in a pipline of a process and its process gain relationships. This liquid pressure control loop is regulated by the position of the valve on the pipeline leading to the process. A P-only control is used here to change the valve position resulting in the need response in attaining the set point value. The following are any assumptions made and the equations used in modeling this process. An assumption is made that the process behaves like a fixed restriction such as an orifice plate whose Dp is a function of flow through the process. Another assumption that should be made is that the valve responds linearly to the flow-rate through it. In determining the process gain the following equations where used.

$$P = \Delta p + P_0$$

where P_0 is the downstream pressure at zero flow

$$\Delta p = \frac{F^2}{R^2}$$

where R is the process flow resistance. As a result we obtain:

$$P = \frac{F^2}{R^2} + P_0$$

The gain of the process is then determined by the following expression:

$$K_p = \frac{dP}{dF} = \frac{2F}{R^2}$$

A model for the feed backwards system for liquid pressure control introduced in this article is very complicated and might possibly be out of the scope of our discussion. A possible alternative model would be very similar to the liquid level model presented below with the head in the tank determining the liquid pressure in the pipeline leaving the reactor.

9.7.1.4.5.2: LIQUID LEVEL CONTROL MODEL

The following model represents a feed backward system of a liquid holding tank. The control loop is regulated by the exiting valve position on the pipe leaving the tank. A maximal flowrate out is assumed and the height of water in the tank does not have any effect on the flowrate out of the tank. This assumption can be made in tanks which are sufficiently wider than the pipe exiting operating with little liquid level change, as often is the case. Another assumption made is the linear response of the valve to flowrate out.

The pipe entering the tank is set by the user at a fixed flowrate. The P-only control is used to change the exiting valve position in order to cause the desired response to obtain the setpoint value. The P control follows the equation listed above and the bias is designated by the point at which the valve is open to allow a steady state to occur. Gain in Level control systems is usually low, however this can be changed by the user as well. The setpoint for the tank is determined by the user, as are the entering liquid flowrates and the maximal exiting flowrate.

The user can input a set point change in order to view the corresponding response in the controller.

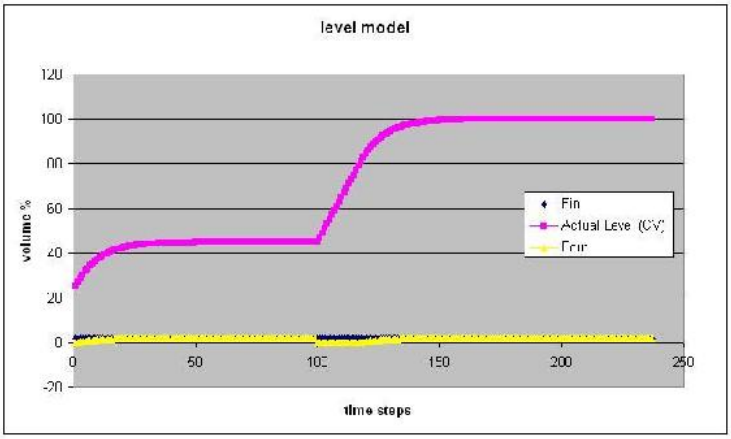
EXAMPLE 9.7.1.4.1

Using the Liquid Level control Excel Model, what do you find when you change the level set point at a given point in time?

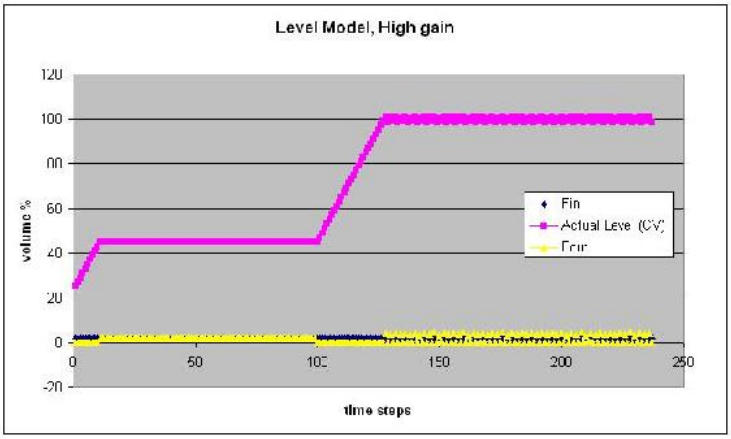
What happens if you change the gain to an large number?

Solution

When you change the level set point, you should notice a corresponding increase or decrease in the level as the P-control kicks in and beings to compensate for the error. The rate at which this set point is obtained is determined by the controller gain. At high controller gains the system responds extremely rapidly in an unreal fashion, this is why controller gain is usually small in large capacity tanks with flow control.



The above figure changes the setpoint to 100 @ t=100



The above figure uses a very high controller gain. Note the unreal response.

EXAMPLE 9.7.1.4.2

What is the primary difference between the model for the process gain of the pressure flow loop introduced earlier in the article and a typical head flow response curve?

Solution

The response of pressure to flow is shaped similarly to the response of the head flow device and as a result the rules that govern the liquid pressure process gain model are in essence the same for the head flow device response. The only difference between the two response curves is that whereas the pressure varies from Po to 100 per cent for the process gain curve for the pressure flow loop, the pressure varies from 0 to 100 percent for the head flow device.

Below is the response for the pressure loop followed by that of head flow device.

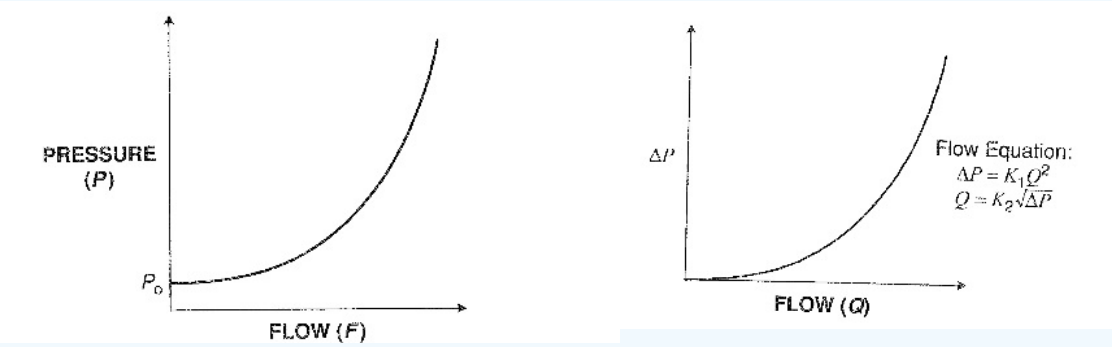


Figure **9.7.1.4.1**: esponse for the pressure loop followed by that of head flow device.

9.7.1.4.5.3: MULTIPLE CHOICE QUESTION 1

When should P-only controllers be used in Liquid Level control?

a. Whenever, because it's really easy.
b. Never because it is too simple.
c. Sometimes, when the gain is small and the tank capacity is large.
d. None of the above

9.7.1.4.5.4: MULTIPLE CHOICE QUESTION 2

Various methods may be used to minimize the noise in a level measurement when using which type of controller action:

a. None, you can never minimize noise in a level measurement
b. Integral
c. Derivative
d. PI-control

This page titled 9.7.1.4: Summary- Summary on Control Architectures' philosophies, advantages, and disadvantages. is shared under a CC BY 4.0 license and was authored, remixed, and/or curated by Jennifersue Bowker, George Cater, Kibwe John Date, & Kibwe John Date via source content that was edited to the style and standards of the LibreTexts platform.

- **14.4: Summary- Summary on Control Architectures' philosophies, advantages, and disadvantages.** by George Cater, Jennifersue Bowker, Kibwe John Date is licensed CC BY 3.0. Original source: https://open.umn.edu/opentextbooks/textbooks/chemical-process-dynamics-and-controls.

9.8: Chapter 8 Supplementary Reading Materials

9.8: Chapter 8 Supplementary Reading Materials is shared under a not declared license and was authored, remixed, and/or curated by LibreTexts.

Index

A

ADC

9.2.3.7: Discrete Time Processing of Continuous Time Signals
9.6.6.7: Discrete Time Processing of Continuous Time Signals

aliasing

9.2.3.1: Signal Sampling
9.2.3.5: Aliasing Phenomena
9.6.6.1: Signal Sampling
9.6.6.5: Aliasing Phenomena

aliasing phenomena

9.2.3: Sampling and Reconstruction
9.6.6: Sampling and Reconstruction

analog signals

9.2.1.1: Introduction to Digital Signal Processing

audio

9.2.3.2: Sampling Theorem
9.6.6.2: Sampling Theorem

B

bandlimited

9.2.3.5: Aliasing Phenomena
9.6.6.5: Aliasing Phenomena

bandlimited signals

9.2.3.3: Signal Reconstruction
9.2.3.4: Perfect Reconstruction
9.6.6.3: Signal Reconstruction
9.6.6.4: Perfect Reconstruction

C

causal system

9.2.3.7: Discrete Time Processing of Continuous Time Signals
9.6.6.7: Discrete Time Processing of Continuous Time Signals

communication

9.2.3.2: Sampling Theorem
9.6.6.2: Sampling Theorem

completely randomized design (CRD)

9.7.1.3: Design of Experiments via Random Design

computer technology

9.2.1.1: Introduction to Digital Signal Processing

continuous time

9.2.3.6: Anti-Aliasing Filters
9.6.6.6: Anti-Aliasing Filters

continuous time signal

9.2.3: Sampling and Reconstruction
9.2.3.3: Signal Reconstruction
9.6.6: Sampling and Reconstruction
9.6.6.3: Signal Reconstruction

continuous_time_fourier_transform

9.2.3.1: Signal Sampling
9.6.6.1: Signal Sampling

continuous_time_signals

9.2.3.1: Signal Sampling
9.6.6.1: Signal Sampling

D

data analysis

9.2.1.1: Introduction to Digital Signal Processing

digital

9.2.3.2: Sampling Theorem
9.6.6.2: Sampling Theorem

digital logic

9.2.3.7: Discrete Time Processing of Continuous Time Signals
9.6.6.7: Discrete Time Processing of Continuous Time Signals

digital signal processing

9.2.1.1: Introduction to Digital Signal Processing
9.2.2.6: Extended Topic - Digital Signal Processing
9.2.3.4: Perfect Reconstruction
9.2.3.6: Anti-Aliasing Filters
9.6.6.4: Perfect Reconstruction
9.6.6.6: Anti-Aliasing Filters

digitalcomputers

9.2.3.1: Signal Sampling
9.6.6.1: Signal Sampling

discrete signals

9.2.1.1: Introduction to Digital Signal Processing

discrete time

9.2.3.6: Anti-Aliasing Filters
9.6.6.6: Anti-Aliasing Filters

discrete time processing

9.2.3.4: Perfect Reconstruction
9.6.6.4: Perfect Reconstruction

discrete time signal

9.2.3: Sampling and Reconstruction
9.2.3.3: Signal Reconstruction
9.6.6: Sampling and Reconstruction
9.6.6.3: Signal Reconstruction

discrete_time_signals

9.2.3.1: Signal Sampling
9.6.6.1: Signal Sampling

DSP

9.2.2.6: Extended Topic - Digital Signal Processing

E

error energy

9.2.3.6: Anti-Aliasing Filters
9.6.6.6: Anti-Aliasing Filters

F

factorial design

9.7.1.2: Design of experiments via factorial designs

filtering

9.2.3.4: Perfect Reconstruction
9.6.6.4: Perfect Reconstruction

filters

9.2.3.6: Anti-Aliasing Filters
9.6.6.6: Anti-Aliasing Filters

Fourier transform

9.2.3.5: Aliasing Phenomena
9.6.5: Fourier Transform
9.6.6.5: Aliasing Phenomena

Fourier Transforms

9.2.1.1: Introduction to Digital Signal Processing

frequency

9.2.3.2: Sampling Theorem
9.6.6.2: Sampling Theorem

frequency_domain

9.2.3.1: Signal Sampling
9.6.6.1: Signal Sampling

friction

9.5.3: Friction (Coulomb) Damped Free Vibrations

I

ideal lowpass filter

9.2.3.4: Perfect Reconstruction
9.6.6.4: Perfect Reconstruction

impulse response

9.2.3.7: Discrete Time Processing of Continuous Time Signals
9.6.6.7: Discrete Time Processing of Continuous Time Signals

interpolation

9.2.3: Sampling and Reconstruction
9.2.3.3: Signal Reconstruction
9.2.3.5: Aliasing Phenomena
9.6.6: Sampling and Reconstruction
9.6.6.3: Signal Reconstruction
9.6.6.5: Aliasing Phenomena

L

linear filtering

9.2.1.1: Introduction to Digital Signal Processing

lowpass

9.2.3.6: Anti-Aliasing Filters
9.6.6.6: Anti-Aliasing Filters

lowpass filter

9.2.3.3: Signal Reconstruction
9.2.3.7: Discrete Time Processing of Continuous Time Signals
9.6.6.3: Signal Reconstruction
9.6.6.7: Discrete Time Processing of Continuous Time Signals

N

Nyquist

9.2.3.2: Sampling Theorem
9.2.3.5: Aliasing Phenomena
9.2.3.6: Anti-Aliasing Filters
9.6.6.2: Sampling Theorem
9.6.6.5: Aliasing Phenomena
9.6.6.6: Anti-Aliasing Filters

Nyquist rate

9.2.3.4: Perfect Reconstruction
9.6.6.4: Perfect Reconstruction

nyquist_shannon_sampling_theorem

9.2.3.1: Signal Sampling
9.6.6.1: Signal Sampling

O

orthogonal arrays

9.7.1.1: Design of Experiments via Taguchi Methods - Orthogonal Arrays

P

Power spectral density

9.2.1.1: Introduction to Digital Signal Processing

processing

9.2.3.2: Sampling Theorem
9.6.6.2: Sampling Theorem

Q

quantization

9.2.1.3: Amplitude Quantization
9.2.3.7: Discrete Time Processing of Continuous Time Signals
9.6.6.7: Discrete Time Processing of Continuous Time Signals

quantization interval

9.2.1.3: Amplitude Quantization

R

random design
- 9.7.1.3: Design of Experiments via Random Design

randomized block design (RBD)
- 9.7.1.3: Design of Experiments via Random Design

Reconstruction
- 9.2.3: Sampling and Reconstruction
- 9.2.3.5: Aliasing Phenomena
- 9.2.3.6: Anti-Aliasing Filters
- 9.6.6: Sampling and Reconstruction
- 9.6.6.5: Aliasing Phenomena
- 9.6.6.6: Anti-Aliasing Filters

reconstruction filters
- 9.2.3.3: Signal Reconstruction
- 9.6.6.3: Signal Reconstruction

reconstruction process
- 9.2.3.3: Signal Reconstruction
- 9.6.6.3: Signal Reconstruction

S

sample reconstruction
- 9.2.3.7: Discrete Time Processing of Continuous Time Signals
- 9.6.6.7: Discrete Time Processing of Continuous Time Signals

sampling
- 9.2.3.1: Signal Sampling
- 9.2.3.2: Sampling Theorem
- 9.2.3.3: Signal Reconstruction
- 9.2.3.5: Aliasing Phenomena
- 9.2.3.6: Anti-Aliasing Filters
- 9.6.6.1: Signal Sampling
- 9.6.6.2: Sampling Theorem
- 9.6.6.3: Signal Reconstruction
- 9.6.6.5: Aliasing Phenomena
- 9.6.6.6: Anti-Aliasing Filters

sampling rate
- 9.2.3.4: Perfect Reconstruction
- 9.6.6.4: Perfect Reconstruction

sampling theorem
- 9.2.3: Sampling and Reconstruction
- 9.6.6: Sampling and Reconstruction

sampling_period
- 9.2.3.1: Signal Sampling
- 9.6.6.1: Signal Sampling

sampling_rate
- 9.2.3.1: Signal Sampling
- 9.6.6.1: Signal Sampling

Shannon
- 9.2.3.2: Sampling Theorem
- 9.2.3.5: Aliasing Phenomena
- 9.6.6.2: Sampling Theorem
- 9.6.6.5: Aliasing Phenomena

sign function
- 9.5.3: Friction (Coulomb) Damped Free Vibrations

signal
- 9.2.3.2: Sampling Theorem
- 9.6.6.2: Sampling Theorem

signal processing
- 9.2.1.1: Introduction to Digital Signal Processing
- 9.2.3.3: Signal Reconstruction
- 9.2.3.5: Aliasing Phenomena
- 9.2.3.7: Discrete Time Processing of Continuous Time Signals
- 9.6.6.3: Signal Reconstruction
- 9.6.6.5: Aliasing Phenomena
- 9.6.6.7: Discrete Time Processing of Continuous Time Signals

signal reconstruction
- 9.2.3.3: Signal Reconstruction
- 9.2.3.4: Perfect Reconstruction
- 9.6.6.3: Signal Reconstruction
- 9.6.6.4: Perfect Reconstruction

signal sampling
- 9.2.3: Sampling and Reconstruction
- 9.6.6: Sampling and Reconstruction

sinc function
- 9.2.3.4: Perfect Reconstruction
- 9.6.6.4: Perfect Reconstruction

spectrum
- 9.2.3: Sampling and Reconstruction
- 9.2.3.5: Aliasing Phenomena
- 9.6.6: Sampling and Reconstruction
- 9.6.6.5: Aliasing Phenomena

T

Taguchi loss function
- 9.7.1.1: Design of Experiments via Taguchi Methods - Orthogonal Arrays

Taguchi method
- 9.7.1.1: Design of Experiments via Taguchi Methods - Orthogonal Arrays

telecommunications
- 9.2.1.1: Introduction to Digital Signal Processing

theorem
- 9.2.3.2: Sampling Theorem
- 9.6.6.2: Sampling Theorem

V

viscous damping
- 9.5.2: Viscous Damped Free Vibrations

viscous damping constant
- 9.5.2: Viscous Damped Free Vibrations

W

windowing
- 9.2.3.7: Discrete Time Processing of Continuous Time Signals
- 9.6.6.7: Discrete Time Processing of Continuous Time Signals

Y

Yates algorithm
- 9.7.1.2: Design of experiments via factorial designs

Glossary

Sample Word 1 | Sample Definition 1

Detailed Licensing

Overview

Title: Experimental Design and Evaluation: A structured outline

Webpages: 93

Applicable Restrictions: Noncommercial

All licenses found:

- CC BY-NC 4.0: 46.2% (43 pages)
- CC BY 4.0: 25.8% (24 pages)
- Undeclared: 11.8% (11 pages)
- CC BY-NC-SA 4.0: 10.8% (10 pages)
- CC BY-SA 4.0: 5.4% (5 pages)

By Page

- Experimental Design and Evaluation: A structured outline - *CC BY-NC 4.0*
  - Front Matter - *CC BY-NC 4.0*
    - TitlePage - *CC BY-NC 4.0*
    - InfoPage - *CC BY-NC 4.0*
    - Table of Contents - *Undeclared*
    - Preface and Licensing - *CC BY-NC 4.0*
    - Introduction - *Undeclared*
  - 1: Making Measurement Connections - *CC BY-NC 4.0*
  - 2: Periodic Signals and Data Acquisition - *CC BY-NC 4.0*
    - 2.1: Features and Fundamental Equations - *CC BY-NC 4.0*
    - 2.2: Sampling Experimental Data - *CC BY-NC 4.0*
  - 3: Uncertainty Analysis - *CC BY-NC 4.0*
    - 3.1: Component Uncertainty - *CC BY-NC 4.0*
    - 3.2: Combining Uncertainty in Function - *CC BY-NC 4.0*
  - 4: Technical Writing - *CC BY-NC 4.0*
  - 5: Calibration and Dynamic Response - *CC BY-NC 4.0*
    - 5.1: Fitting with Weighted Least Squares - *CC BY-NC 4.0*
    - 5.2: Dynamic Response from Linear ODEs - *CC BY-NC 4.0*
  - 6: Fourier Analysis - *CC BY-NC 4.0*
    - 6.1: Descriptions - *CC BY-NC 4.0*
    - 6.2: Conversion of Function - *CC BY-NC 4.0*
    - 6.3: Discrete data conversion and filtering - *CC BY-NC 4.0*
  - 7: Probability and Statistics - *CC BY-NC 4.0*
    - 7.1: Contextual background resulting in histogram - *CC BY-NC 4.0*
    - 7.2: Central moments of probability density function - *CC BY-NC 4.0*
    - 7.3: Probability from Gaussian distribution - *CC BY-NC 4.0*
    - 7.4: Analysis with finite statistics - *CC BY-NC 4.0*
  - 8: Laboratory Exercises - *CC BY-NC 4.0*
    - 8.1: Digital data sampling - *CC BY-NC 4.0*
    - 8.2: Stress and strain - *CC BY-NC 4.0*
    - 8.3: Temperature and Heat Transfer - *CC BY-NC 4.0*
    - 8.4: Fluid Flow and Forces - *CC BY-NC 4.0*
  - 9: Supplementary Material Section - *Undeclared*
    - 9.1: Chapter 1 Supplementary Reading Materials - *Undeclared*
    - 9.2: Chapter 2 Supplementary Reading Materials - *CC BY-NC 4.0*
      - 9.2.1: Discrete-Time Signal Processing - *CC BY-NC 4.0*
        - 9.2.1.1: Introduction to Digital Signal Processing - *CC BY-NC 4.0*
        - 9.2.1.2: The Sampling Theorem - *CC BY-NC 4.0*
        - 9.2.1.3: Amplitude Quantization - *CC BY-NC 4.0*
        - 9.2.1.4: Discrete -Time Signals and Systems - *CC BY-NC 4.0*
      - 9.2.2: Analog-to-Digital-to-Analog Conversion - *CC BY-NC-SA 4.0*
        - 9.2.2.1: Introduction to Analog to Digital to Analog Conversion - *CC BY-NC-SA 4.0*
        - 9.2.2.2: The Sampling Theorem - *CC BY-NC-SA 4.0*

- 9.2.2.3: Resolution and Sampling Rate - *CC BY-NC-SA 4.0*
- 9.2.2.4: Digital-to-Analog Conversion Techniques - *CC BY-NC-SA 4.0*
- 9.2.2.5: Analog-to-Digital Conversion - *CC BY-NC-SA 4.0*
- 9.2.2.6: Extended Topic - Digital Signal Processing - *CC BY-NC-SA 4.0*
- 9.2.2.7: Summary - *CC BY-NC-SA 4.0*
- 9.2.2.8: Problems - *CC BY-NC-SA 4.0*

9.2.3: Sampling and Reconstruction - *CC BY 4.0*

- 9.2.3.1: Signal Sampling - *CC BY 4.0*
- 9.2.3.2: Sampling Theorem - *CC BY 4.0*
- 9.2.3.3: Signal Reconstruction - *CC BY 4.0*
- 9.2.3.4: Perfect Reconstruction - *CC BY 4.0*
- 9.2.3.5: Aliasing Phenomena - *CC BY 4.0*
- 9.2.3.6: Anti-Aliasing Filters - *CC BY 4.0*
- 9.2.3.7: Discrete Time Processing of Continuous Time Signals - *CC BY 4.0*

9.3: Chapter 3 Supplementary Reading Materials - *Undeclared*

- 9.3.1: Importance of Statistics and Experimental Measurements - *CC BY-NC 4.0*
- 9.3.2: Total Uncertainty, Bias Error, and Propagation of Error - *CC BY-NC 4.0*

9.4: Chapter 4 Supplementary Reading Materials - *Undeclared*

9.5: Chapter 5 Supplementary Reading Materials - *Undeclared*

- 9.5.1: Undamped Free Vibrations - *CC BY-SA 4.0*
- 9.5.2: Viscous Damped Free Vibrations - *CC BY-SA 4.0*
- 9.5.3: Friction (Coulomb) Damped Free Vibrations - *CC BY-SA 4.0*
- 9.5.4: Undamped Harmonic Forced Vibrations - *CC BY-SA 4.0*
- 9.5.5: Viscous Damped Harmonic Forced Vibrations - *CC BY-SA 4.0*

9.6: Chapter 6 Supplementary Reading Materials - *Undeclared*

- 9.6.1: Introduction to the Frequency Domain - *CC BY 4.0*
- 9.6.2: Complex Fourier Series - *CC BY 4.0*
- 9.6.3: Classic Fourier Series - *CC BY 4.0*
- 9.6.4: Fourier Series Approximation of Signals - *CC BY 4.0*
- 9.6.5: Fourier Transform - *CC BY-NC-SA 4.0*
- 9.6.6: Sampling and Reconstruction - *CC BY 4.0*
  - 9.6.6.1: Signal Sampling - *CC BY 4.0*
  - 9.6.6.2: Sampling Theorem - *CC BY 4.0*
  - 9.6.6.3: Signal Reconstruction - *CC BY 4.0*
  - 9.6.6.4: Perfect Reconstruction - *CC BY 4.0*
  - 9.6.6.5: Aliasing Phenomena - *CC BY 4.0*
  - 9.6.6.6: Anti-Aliasing Filters - *CC BY 4.0*
  - 9.6.6.7: Discrete Time Processing of Continuous Time Signals - *CC BY 4.0*

9.7: Chapter 7 Supplementary Reading Materials - *Undeclared*

- 9.7.1: Design of Experiments - *CC BY-NC 4.0*
  - 9.7.1.1: Design of Experiments via Taguchi Methods - Orthogonal Arrays - *CC BY 4.0*
  - 9.7.1.2: Design of experiments via factorial designs - *CC BY 4.0*
  - 9.7.1.3: Design of Experiments via Random Design - *CC BY 4.0*
  - 9.7.1.4: Summary- Summary on Control Architectures' philosophies, advantages, and disadvantages. - *CC BY 4.0*

9.8: Chapter 8 Supplementary Reading Materials - *Undeclared*

Back Matter - *CC BY-NC 4.0*

- Index - *CC BY-NC 4.0*
- Glossary - *CC BY-NC 4.0*
- Detailed Licensing - *CC BY-NC 4.0*
- Detailed Licensing - *Undeclared*

Detailed Licensing

Overview

Title: Experimental Design and Evaluation: A structured outline

Webpages: 93

Applicable Restrictions: Noncommercial

All licenses found:

- CC BY-NC 4.0: 46.2% (43 pages)
- CC BY 4.0: 25.8% (24 pages)
- Undeclared: 11.8% (11 pages)
- CC BY-NC-SA 4.0: 10.8% (10 pages)
- CC BY-SA 4.0: 5.4% (5 pages)

By Page

- Experimental Design and Evaluation: A structured outline - *CC BY-NC 4.0*
  - Front Matter - *CC BY-NC 4.0*
    - TitlePage - *CC BY-NC 4.0*
    - InfoPage - *CC BY-NC 4.0*
    - Table of Contents - *Undeclared*
    - Preface and Licensing - *CC BY-NC 4.0*
    - Introduction - *Undeclared*
  - 1: Making Measurement Connections - *CC BY-NC 4.0*
  - 2: Periodic Signals and Data Acquisition - *CC BY-NC 4.0*
    - 2.1: Features and Fundamental Equations - *CC BY-NC 4.0*
    - 2.2: Sampling Experimental Data - *CC BY-NC 4.0*
  - 3: Uncertainty Analysis - *CC BY-NC 4.0*
    - 3.1: Component Uncertainty - *CC BY-NC 4.0*
    - 3.2: Combining Uncertainty in Function - *CC BY-NC 4.0*
  - 4: Technical Writing - *CC BY-NC 4.0*
  - 5: Calibration and Dynamic Response - *CC BY-NC 4.0*
    - 5.1: Fitting with Weighted Least Squares - *CC BY-NC 4.0*
    - 5.2: Dynamic Response from Linear ODEs - *CC BY-NC 4.0*
  - 6: Fourier Analysis - *CC BY-NC 4.0*
    - 6.1: Descriptions - *CC BY-NC 4.0*
    - 6.2: Conversion of Function - *CC BY-NC 4.0*
    - 6.3: Discrete data conversion and filtering - *CC BY-NC 4.0*
  - 7: Probability and Statistics - *CC BY-NC 4.0*
    - 7.1: Contextual background resulting in histogram - *CC BY-NC 4.0*
    - 7.2: Central moments of probability density function - *CC BY-NC 4.0*
    - 7.3: Probability from Gaussian distribution - *CC BY-NC 4.0*
    - 7.4: Analysis with finite statistics - *CC BY-NC 4.0*
  - 8: Laboratory Exercises - *CC BY-NC 4.0*
    - 8.1: Digital data sampling - *CC BY-NC 4.0*
    - 8.2: Stress and strain - *CC BY-NC 4.0*
    - 8.3: Temperature and Heat Transfer - *CC BY-NC 4.0*
    - 8.4: Fluid Flow and Forces - *CC BY-NC 4.0*
  - 9: Supplementary Material Section - *Undeclared*
    - 9.1: Chapter 1 Supplementary Reading Materials - *Undeclared*
    - 9.2: Chapter 2 Supplementary Reading Materials - *CC BY-NC 4.0*
      - 9.2.1: Discrete-Time Signal Processing - *CC BY-NC 4.0*
        - 9.2.1.1: Introduction to Digital Signal Processing - *CC BY-NC 4.0*
        - 9.2.1.2: The Sampling Theorem - *CC BY-NC 4.0*
        - 9.2.1.3: Amplitude Quantization - *CC BY-NC 4.0*
        - 9.2.1.4: Discrete -Time Signals and Systems - *CC BY-NC 4.0*
      - 9.2.2: Analog-to-Digital-to-Analog Conversion - *CC BY-NC-SA 4.0*
        - 9.2.2.1: Introduction to Analog to Digital to Analog Conversion - *CC BY-NC-SA 4.0*
        - 9.2.2.2: The Sampling Theorem - *CC BY-NC-SA 4.0*

- 9.2.2.3: Resolution and Sampling Rate - *CC BY-NC-SA 4.0*
- 9.2.2.4: Digital-to-Analog Conversion Techniques - *CC BY-NC-SA 4.0*
- 9.2.2.5: Analog-to-Digital Conversion - *CC BY-NC-SA 4.0*
- 9.2.2.6: Extended Topic - Digital Signal Processing - *CC BY-NC-SA 4.0*
- 9.2.2.7: Summary - *CC BY-NC-SA 4.0*
- 9.2.2.8: Problems - *CC BY-NC-SA 4.0*

9.2.3: Sampling and Reconstruction - *CC BY 4.0*

- 9.2.3.1: Signal Sampling - *CC BY 4.0*
- 9.2.3.2: Sampling Theorem - *CC BY 4.0*
- 9.2.3.3: Signal Reconstruction - *CC BY 4.0*
- 9.2.3.4: Perfect Reconstruction - *CC BY 4.0*
- 9.2.3.5: Aliasing Phenomena - *CC BY 4.0*
- 9.2.3.6: Anti-Aliasing Filters - *CC BY 4.0*
- 9.2.3.7: Discrete Time Processing of Continuous Time Signals - *CC BY 4.0*

9.3: Chapter 3 Supplementary Reading Materials - *Undeclared*

- 9.3.1: Importance of Statistics and Experimental Measurements - *CC BY-NC 4.0*
- 9.3.2: Total Uncertainty, Bias Error, and Propagation of Error - *CC BY-NC 4.0*

9.4: Chapter 4 Supplementary Reading Materials - *Undeclared*

9.5: Chapter 5 Supplementary Reading Materials - *Undeclared*

- 9.5.1: Undamped Free Vibrations - *CC BY-SA 4.0*
- 9.5.2: Viscous Damped Free Vibrations - *CC BY-SA 4.0*
- 9.5.3: Friction (Coulomb) Damped Free Vibrations - *CC BY-SA 4.0*
- 9.5.4: Undamped Harmonic Forced Vibrations - *CC BY-SA 4.0*
- 9.5.5: Viscous Damped Harmonic Forced Vibrations - *CC BY-SA 4.0*

9.6: Chapter 6 Supplementary Reading Materials - *Undeclared*

- 9.6.1: Introduction to the Frequency Domain - *CC BY 4.0*
- 9.6.2: Complex Fourier Series - *CC BY 4.0*
- 9.6.3: Classic Fourier Series - *CC BY 4.0*
- 9.6.4: Fourier Series Approximation of Signals - *CC BY 4.0*
- 9.6.5: Fourier Transform - *CC BY-NC-SA 4.0*
- 9.6.6: Sampling and Reconstruction - *CC BY 4.0*
  - 9.6.6.1: Signal Sampling - *CC BY 4.0*
  - 9.6.6.2: Sampling Theorem - *CC BY 4.0*
  - 9.6.6.3: Signal Reconstruction - *CC BY 4.0*
  - 9.6.6.4: Perfect Reconstruction - *CC BY 4.0*
  - 9.6.6.5: Aliasing Phenomena - *CC BY 4.0*
  - 9.6.6.6: Anti-Aliasing Filters - *CC BY 4.0*
  - 9.6.6.7: Discrete Time Processing of Continuous Time Signals - *CC BY 4.0*

9.7: Chapter 7 Supplementary Reading Materials - *Undeclared*

- 9.7.1: Design of Experiments - *CC BY-NC 4.0*
  - 9.7.1.1: Design of Experiments via Taguchi Methods - Orthogonal Arrays - *CC BY 4.0*
  - 9.7.1.2: Design of experiments via factorial designs - *CC BY 4.0*
  - 9.7.1.3: Design of Experiments via Random Design - *CC BY 4.0*
  - 9.7.1.4: Summary- Summary on Control Architectures' philosophies, advantages, and disadvantages. - *CC BY 4.0*

9.8: Chapter 8 Supplementary Reading Materials - *Undeclared*

Back Matter - *CC BY-NC 4.0*

- Index - *CC BY-NC 4.0*
- Glossary - *CC BY-NC 4.0*
- Detailed Licensing - *CC BY-NC 4.0*
- Detailed Licensing - *Undeclared*