Introduction to Civil Information Systems

Mark A. Austin

University of Maryland

austin@umd.edu

ENCE 201, Spring Semester 2025

January 31, 2025

Overview

Modern

Civil Infrastructure

Systems

Modern Civil Infrastructure Systems

Various Sources (Google, ScienceDirect):

- Civil Infrastructure Systems provide for human activity, ranging in scale from buildings to cities.
- Includes supporting infrastructure: water supply networks; energy networks; transportation systems, communication systems.

Support Human Needs:

- Basic: Access to clean air and clean water.
- Health: Access to good medical services.
- Economic: Affordable low maintenance housing.
- Security: Protections against crime, environmental attack.

Modern Civil Infrastructure Systems

- Transportation: Good roads; parking; fast access to work.
- Educational: Access to good schools.
- Green Spaces: Access to parks, bike paths, etc.
- Retail: Access to shopping; reliable supply chains.
- Lifestyle: Access to social and recreational spaces.

Urban Planning and Engineering Concerns:

- Understand short- and long-term planning needs.
- Efficiency in design – aesthetically pleasing design.
- Efficiency in operations – better use of limited resources.
- Improved response to unexpected events.

Framing the Opportunity

Metrics of Good Engineering Design:

- A good engineering design works correctly, has good performance, and is economical.
- Functionality and performance are resilient to uncertainties.
- System can be easily upgraded to take advantage of new technologies.

Metrics of Good Systems Operation:

A well-run system has situational awareness and handles unexpected events:

- Sense the system state and surrounding environment,
- Look ahead and anticipate events, and
- Take action to control system behavior.

Framing the Opportunity

We seek:

- Data-driven approaches to measurement of performance in the building environment and identification of trends and patterns in behavior.
- Solutions that account for unique physical, economic, social and cultural characteristics of individual cities.

Sources of Complication:

- Multiple domains; multiple types of data and information.
- Network structures that are spatial and interwoven.
- Behaviors that are distributed and concurrent.
- Many interdependencies among coupled urban subsystems.

Framing the Opportunity

Systems Perspective:

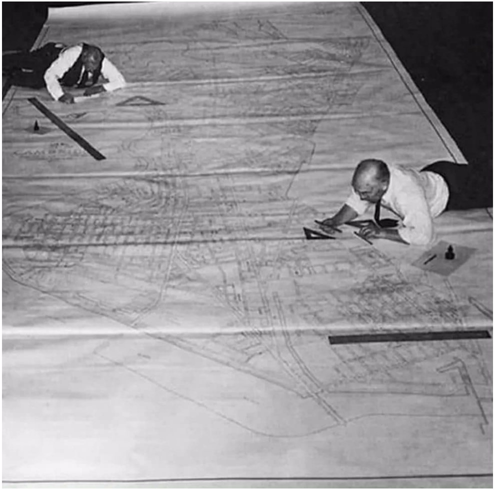
- Entities in the infrastructure environment have both system structure and system behavior ....

Decision makers use behavior modeling to understand:

- Levels of attainable performance.
- Sensitivity of systems to model parameter choices.
- Influence of resource constraints.
- Potential emergent interactions and propogation of cause-and-effect relationships.
- Identification of parts of the systems that are vulnerable.

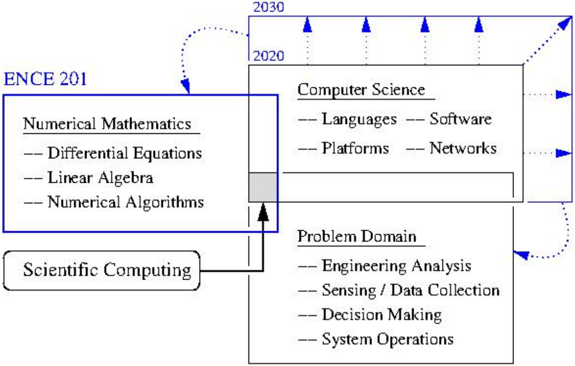
Framing the Opportunity

ENCE 200: Way back in time (pre- 1980):



Framing the Opportunity

ENCE 201: Foundations for Scientific Computing ...

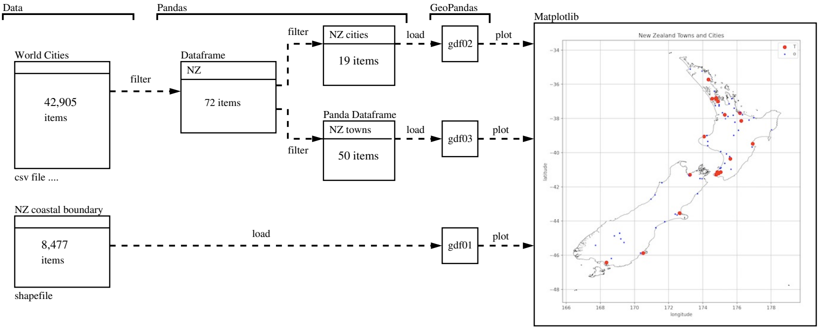


Scientific computing lies at the intersection of computer science, numerical mathematics, and domain-specific problem solving.

Framing the Opportunity

ENCE 201: Learn to work with data ...

Data Processing Pipeline Example: Use sequence of filters to specialize views of data ...



A Little History

Pathway Forward $\rightarrow$ Look to the Past

What is Civil Engineering?

Civil Engineering deals with (Civil Engineering, Wikipedia) ...

.

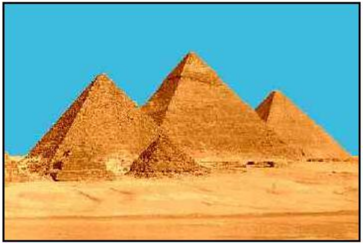
.. the design, construction, and maintenance of the physical and naturally built environment, including roads, bridges, canals, dams, and buildings.

After military engineering, civil engineering is the oldest engineering profession.

Goals during Early Civilization (4000 BC – 6000 BC)

- Problems of survival and basic systems were solved.
- Design and construction methods evolved.

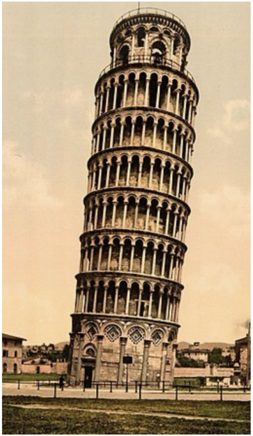
Exemplars of Early Work



- Great Pyramid of Giza, Egypt (20 year construction; finished 2556 BC).
- The Parthenon in Ancient Greece (447-438 BC).
- Construction of the Great Wall of China (220 BC).
- The Romans developed civil structures throughout their empire, including especially aqueducts, insulae (apartment buildings).

Exemplars of Early Work

Leaning Tower of Pisa (12th Century)



- Designed to be the tallest bell tower in Europe.
- Construction: Three stages over 199 years (1173-1372).
- Constructed from white marble.
- Tower leans because of weak unstable subsoil.
- It once leaned at 5.5 degrees.
- Currently leans at 3.99 degrees.
- Has survived 4 earthquakes –ironically, weak subsoil conditions work to protect Pisa from ground accelerations.

Industrial Revolution

Fast forward to the Industrial Revolution: (1760 – 1840).

Year	Milestone
1692	Languedoc Canal. 240 miles long. 100 locks.
1708	Tull's mechanical seed sower → large-scale planting.
1765	Spinning jenny/wheel automates weaving of cloth.
1775	Watt's first efficient steam engine.
1801	Robert Trevithick demonstrates a steam locomotive.
1821	Faraday, electro-magnetic rotation → electric motor.
1834	Babbage analytic engine → forerunner of the computer.
1903	Wright brothers make first powered flight.
1908	Henry Ford mass-produces the Model T.

Industrial Revolution

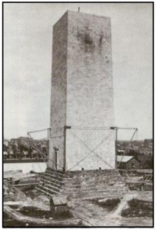
Industrial Revolution Actually Changed the World!

Characteristics	**Stage 1** **Mechanical Era**	**Stage 2** **Electrical Era**
Onset in the U.S.	Late 1700s.	Late 1800s.
Economic Focus	Agriculture/Mining	**Manufacturing**
Productivity Focus	Farming	Factory
Underlying Technologies	**Mechanical Tools**	**ElectroMechanical**
Product Lifecycle	Decades	Years
Human Contribution	Muscle Power	Muscle/Brain Power
Living Standard	Subsistence	**Quality of Goods**
Geographical	Family/Locale	Regional/National

Industrial Revolution

Advances in Civil Engineering

Year	Milestone
1854	Bessemer invents steel converter.
1849	Monier develops reinforced concrete.
1863	Siemens-Martin makes steel available in bulk.



The Eiffel Tower - April 1888

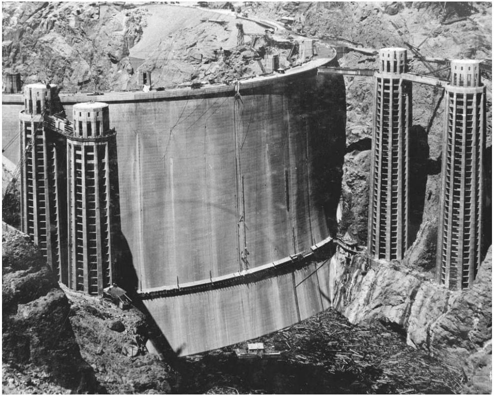
Post- Industrial Revolution

Skyscrapers: Construction of the Empire State Building (1930-1931)

- New materials → design of tall structures having large open interior spaces.
- Elevators (1857) → vertical transportation building occupants.
- Mechanical systems → delivery of water, heating and cooling.
- Collections of skyscrapers → high-density CBDs/commuter society.

Post- Industrial Revolution

Dams: Construction of the Hoover Dam (1931-1935)



Hydroelectric power for Arizona, Nevada, Southern California; controls floods; provide irrigation water.

Post- Industrial Revolution

Bridges: Construction of the Golden Gate (1933-1937)

Post- Industrial Revolution

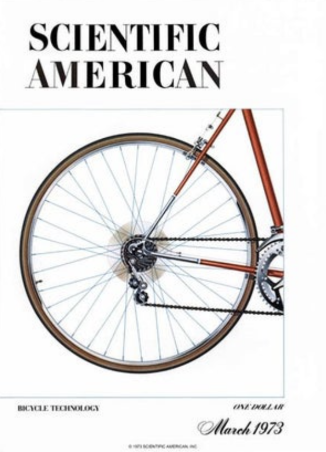
Bridges: Construction of the Golden Gate (1933-1937)

Post- Industrial Revolution

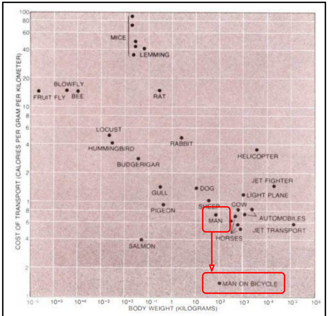
Bridges: Golden Gate Bridge (May 27, 1937)

Post- Industrial Revolution

Bicycles: Mass-produced for personal transportation.



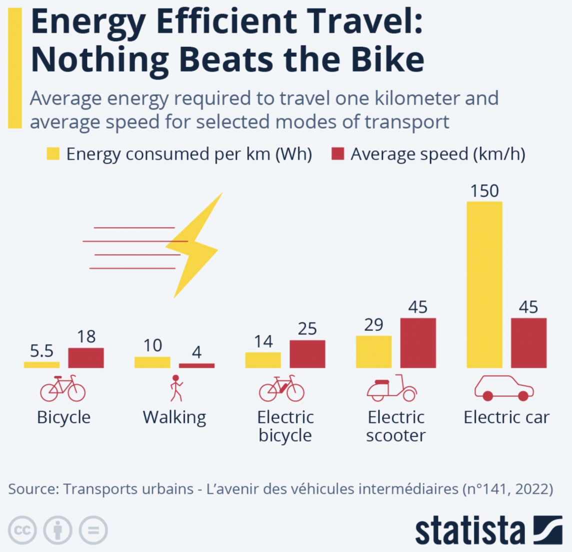
Energy Efficiency of Locomotion in Animals / Machines



MAN ON A BICYCLE ranks first in efficiency among traveling animals and machines in terms of energy consumed in moving a certain distance as a function of body weight. The rate of energy consumption for a bicyclist (about .15 calorie per gram per kilometer) is approximately a fifth of that for an unaided walking man (about .75 calorie per gram per kilometer). With the exception of the black point representing the bicyclist (*lower right*), this graph is based on data originally compiled by Vance A. Tucker of Duke University.

Source: Wilson S.S., Bicycle Technology, Scientific American, March, 1973.

Post- Industrial Revolution



Post- Industrial Revolution

Bicycles ⟶ Electric Cars ⟶ Flying Cars:

From an energy consumption standpoint, improvements in transportation performance and convenience come at a huge cost.

Transition to Information Era

Characteristics	**Stage 2 Electrical Era**	**Stage 3 Information Era**
Onset in the U.S.	Late 1800s.	**Late 1900s.**
Economic Focus	Manufacturing	Services
Technologies	ElectroMechanical	**Information**
Product Lifecycle	Years	Months
Living Standard	Quality of Goods	**Quality of Life**
Human Contribution	Muscle/Brain Power	Enhanced Brain Power
Geographical Impact	Regional/National	**Global**

Enhanced Brain Power: We seek computational engines that work like a bicycle for the mind, providing support for the synthesis and solution of problems.

Transition to Information Era

Motivation (Why?): State-of-the-art systems – planes, trains and automobiles – rely on human involvement as a means for sensing and controlling behavior, e.g.,

- Driving a car,
- Manual collection of road tolls,
- Traffic controllers at an airport,
- Manual focus of a camera.

Systems work, but:

- Humans are slow.
- Humans make mistakes.
- They also easily tire.

Transition to Information Era

Premise of Information-Age System Design:

- Advances in computer software, sensing, and wireless networking technologies can work together to expand the functionality and performance of systems.

Emergence of Automation:

- New types of systems where human involvement for management of system functionality is replaced (or partially replaced) by software automation.

Civil Engineering Applications:

- Automated road toll collection (Rt. 200), baggage handling systems at airports.
- Self-driving cars, smart cities, etc ...

Near-Term Challenges

Civil Engineers need to create the infrastructure for citizens of the Information Era

Crisis in US Infrastructure Investment

Exemplars of Work from the 1800s and 1900s

From the 1800s	From the 1900s
Erie Canal (1825)	New York City Subway (1904)
Transcontinental Railroad (1869)	The Panama Canal (1914)
Brooklyn Bridge (1883)	Holland Tunnel (1927)
Washington Monument (1884)	Empire State Building (1931).
	Hoover Dam (1936).
	Golden Gate Bridge (1937)
	Interstate Highway System (1956)

Source: Celebrating the Greatest Profession, Magazine of the American Society of Civil Engineers, Vol. 72, No. 11, 2002.

Crisis in US Infrastructure Investment

Universal Observations:

- Aging infrastructure becomes expensive to maintain.
- New (replacement) infrastructure is very expensive.
- Politicians are eager to talk up Infrastructure Investment , but slow to deliver ….

Bottom line:

- Critical infrastructure is taken for granted and not a national priority (ASCE, IEEE).

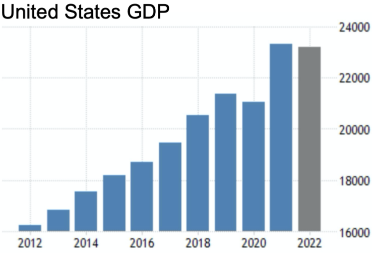
Delay, delay, delay ….

Bangkok, Thailand

Crisis in US Infrastructure Investment

Statistics:

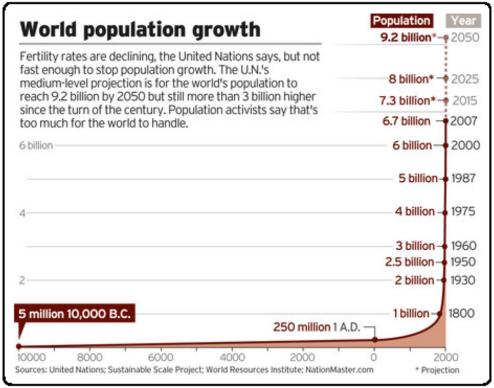
- US: Post World-War II (1950-1970): 3% of Gross Domestic Product (GDP)
- US: 1980-present: 2% of GDP.
- China: 5% GDP.
- India: 9% GDP.



Infrastructure Investment and Jobs Act (2021).

- Invest $1.2T over 10 years.
- Sounds like a lot – but is it too low, too high?
- Increases investment by 0.5% of GDP.

World Population Forecasts



Increasing Population → Increased Demand on Limited Resources → Increasing need for Improvements to System Efficiency.

Urbanization and Sustainable Cities

Urbanization in America:

- In 2010, 82 percent of Americans lived in cities.
- By 2050 it will be 90 percent.

Cities are responsible for:

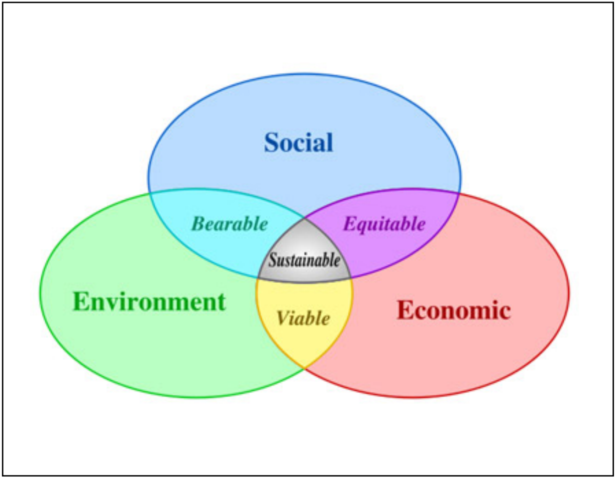
- Two thirds of the energy used,
- 60 percent of all water consumed, and
- 70 percent of all greenhouse gases produced worldwide.

Sustainable cities (SIEMENS, Sustainable Cities, USA):

- Environmentally friendly infrastructures;
- Improved quality of life for residents;
- Good economics.

Sustainable Urban Systems

Sustainability involves physical, organizational and social systems.



Sustainable Urban Systems

Urban systems are like plants in your garden:

- Cities are defined by their emergent properties (e.g., beautiful flower ⇔ New York City Skyline).
- Cities grow and fourish based on societal and economic stimulus, and fall into decay when stimulus is absent.

But sustainability is a tough problem:

- Many of the world's large urban areas – so-called mega-cities – are in poor economic shape.

Cities are system of systems:

- Subsystems have a preference to operate as independently as possible from the other subsystems.
- Strategic collaborations needed to raise levels of attainable performance and limit cascading failures.

Urban Infrastructure Protection and Recovery

Case Study. Cascading Failures in Hurricane Katrina (2005)

- Hurricane Katrina caused a storm surge which, in turn, resulted in the failure of levees around New Orleans.
- This is a failure in the waterway network.
- A more conservative (expensive) design might have prevented this failure.
- But the failure didn't stop there.

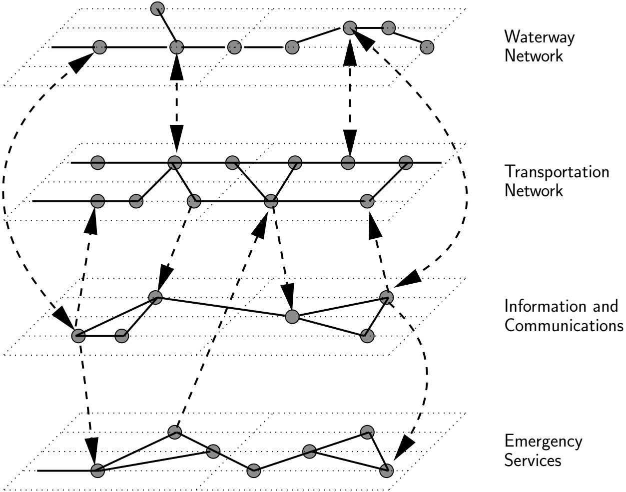
Urban Infrastructure Protection and Recovery

Cascading Failures in Hurricane Katrina:

- **Waterway system failure.** The levees were insufficint to resist the storm surge.
- **Highway and electrical power system failures.** Flooding resulted in failure of the electrical power and highway systems.
- **Federal emergency failures.** Inhabits had to flee their homes, but few plans were in place for their orderly evacuation.
- **Social network failures.** After the inhabitants left their homes, looters stole property from evacuated properties.
- **Political system failures.** ...

Urban Infrastructure Protection and Recovery

Dependencies Among Urban Networks:



Urban Infrastructure Protection and Recovery

Lessons Learned:

Cascading failures of this type indicate that:

- There is a need to understand and manage interactions among infrastructure networks and organizational and societal factors.

Basic Questions

- What kinds of dependencies exist between the networks?
- How will a failure in one network impact other networks? These are so-called cascading failures.
- What parts of a system are most vulnerable?

We need to look at interactions between network models.

Urban Infrastructure Protection and Recovery

Looking Ahead:

- Need situational awareness to understand what is actually happening (or about to happen) in a city.
- Sense the spatial, temporal, and intensity aspects of environmental phenomena (e.g., fires, flooding) and their impact on natural (e.g., air quality) and man-made systems (e.g., transportation networks, food chains).

Goal and Approach:

- Connect measurements and behavior modeling to planning of protection mechanisms and relief actions.
- Create warning systems that can look ahead and predict likely future states of the urban system.

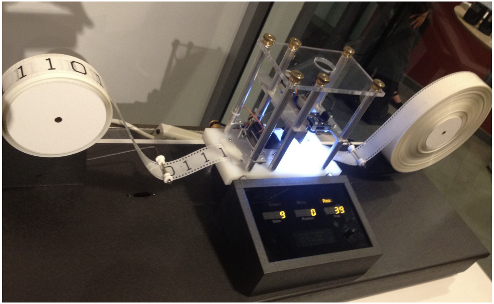
Features of Modern Computing

Key Question: **How can we use modern computing technologies to improve Civil Engineering Systems?**

Early Models of Computing

Turing Machine Model: 1930s ...

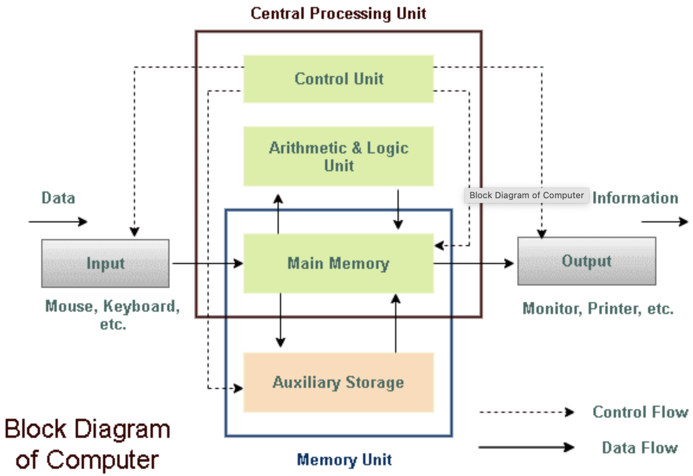
- Alan Turing (1936) created the Turing machine that included the idea of a computer program.



- Turing showed that you can compute anything using only 6 primitives: right, left, print, scan, erase, nothing.

Early Models of Computing

Block Diagram of a Computer: 1980s ...



State-of-the-Art Computing

What does a modern computer do?

- Performs calculations – billions (sometimes even trillions) of calculations per second.
- Remembers results – gigabytes and terabytes of storage.

Modern Programming Languages

- Modern programming languages have a more convenient set of primitives.
- Can abstract methods to create new primitives (e.g., user-defined objects).
- Anything computable in one language is computable in any other programming language.

Man and Machine (Traditional View)

Man	Machine
• Good at formulating solutions to problems. • Can work with incomplete data and information. • Creative. • Reasons logically, but very slow. • Performance is static. • Humans break the rules.	• Manipulates Os and 1s. • Very specific abilities. • Requires precise decriptions of problem solving procedures. • Dumb, but very fast. • Performance doubles every 18-24 months. • Machines will follow the rules.

Sensible Problem Solving Strategy

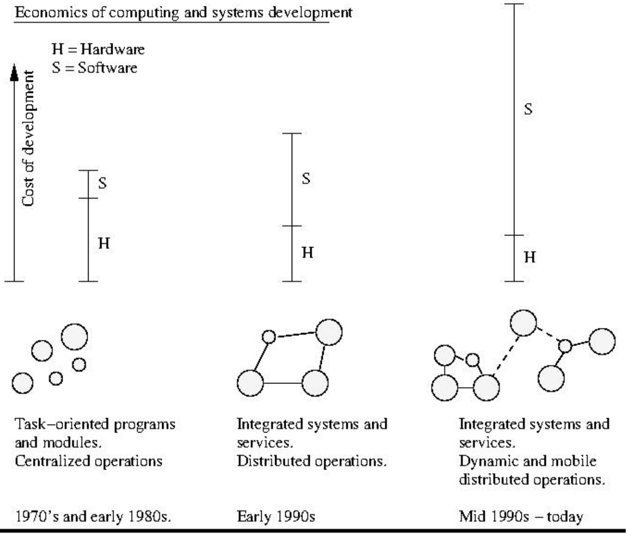
Let engineers and computers play to their strengths:

- Accelerates the solution procedure.
- Enables the analysis of problems having size and complexity beyond manual examination.
- Adds value in areas that will lead to long-term economic growth.

Getting things to work We need to:

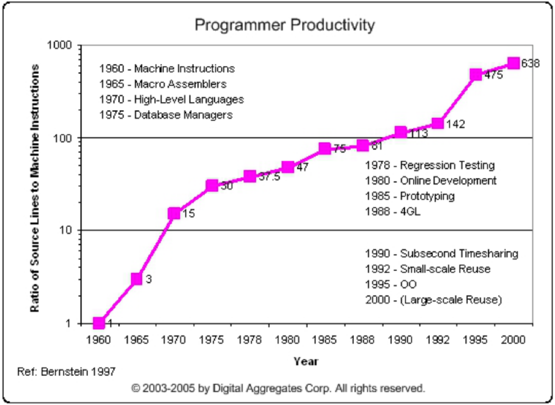
- Describe to the computer solution procedures that are completely unambiguous.
- Look at data, organization and manipulation of data, and formal languages.

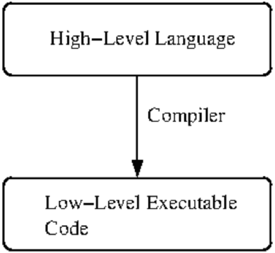
Expanding Expectations of Computing



Pathway to Improved Programmer Productivity

Increasing System Complexity: Software programmers need to find ways to solve problems at high levels of abstraction.





Evolution of Computer Languages

Computer Languages. Formal description – precise grammar – for how a problem can be solved.

Evolution. It takes about a decade for significant advances in computing to occur:

Capability	1970s	1980s	1990s
Users	Specialists	Individuals	Groups
Usage	Numerical computations	Desktop computing	E-mail, web, file transfer.
Interaction	Type at keyboard	Screen and mouse	audio/voice.
Languages	Fortran, C	MATLAB	Python, Java

Popular Computer Languages

Tend to be designed for a specific set of purposes:

- FORTRAN (1950s – today). Stands for formula translation.
- C (early 1970s – today). New operating systems.
- C++ (early 1970s – today). Object-oriented version of C.
- MATLAB (mid 1980s – today). Stands for matrix laboratory.
- Python (1990s – today). Object-oriented scripting language.
- HTML (1990s – today). Layout of web-page content.
- Java (1994 – today). Object-Oriented language for network-based computing.
- XML (late 1990s – today). Description of data on the Web.

Post- 2000 Era

Imagine: What if COVID-19 had arrived in 2000?

- No iPhone, No iPad, No iTunes.
- No Facebook, No Instagram, No WhatsApp.
- No Google Maps, No Google Streetview.
- No Dropbox, No Zoom.

Recent Advances in Technology:

- Average internet speeds: In 2000, 0.07 Mbs; In 2009, 5-7 Mbs; In 2020, 100-200 Mbs; 5G, 1000-2000 Mbs.
- Cloud-based data storage and computational services (AWS).
- New languages: Swift $\rightarrow$ App development on iPhone/iPad.
- Many new types of sensors and methods of data collection.

Post- 2000 Era

New Infrastructure → New Architectures, Languages, ...

Capability	2000-present	2020-2030
Users	Groups of people, sensors and computers.	Integration of the cyber and physical worlds.
Usage	Mobile computing. Control of physical systems. Social networking.	Embedded real-time control of physical systems.
Interaction	Touch, multi-touch, proximity.	....
Languages	XML, RDF, OWL.	New languages to support time-precise computations.

Post-2010 Era → Emergence of AI

State-of-the-Art Implementation (2020, Google, Siemens, IBM)

- AI and ML will be deeply embedded in new software and algorithms.

Artificial Intelligence:

- Knowledge representation and reasoning with ontologies and rules. Semantic graphs. Executable event-based processing.

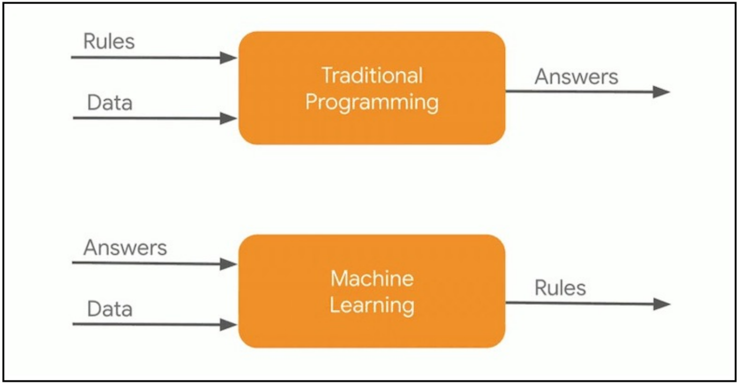
Machine Learning:

- Modern neural networks. Input-to-output prediction.
- Data mining.
- Identify objects, events, and anomalies.
- Learn structure and sequence. Remember stuff.

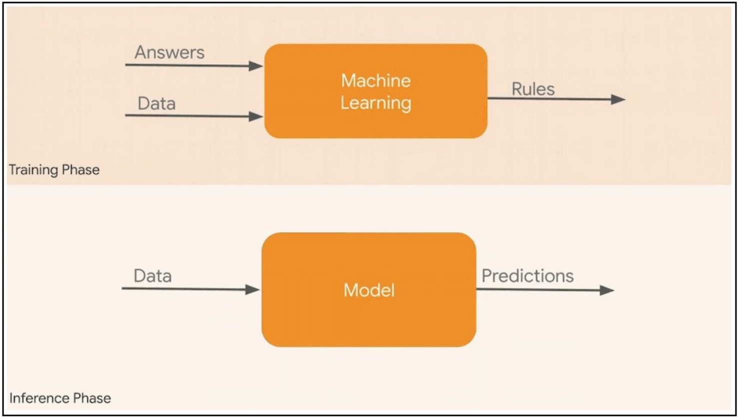
Man and Machine (AI-ML View)

Man	AI-ML Machine
• Good at formulating solutions to problems. • Can work with incomplete data and information. • Creative. • Reasons logically, but very slow. Forgetful. • Performance is static. • Humans make the rules, then they break them.	• Manipulates Os and 1s. • Can work with incomplete data and information. • Creative. • Fast logical reasoning. • Performance doubles every 18-24 months. • Data mining can discover the rules.

Traditional Programming vs AI-ML Workflow

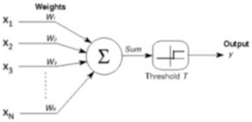


Traditional Programming vs AI-ML Workflow



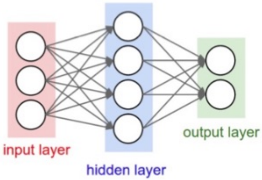
Machine Learning Capabilities (1980-1990)

Expressive Power of a Neural Network

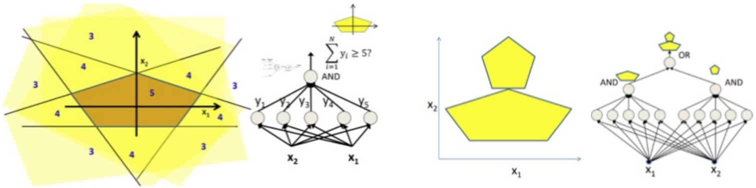


$$y = \begin{cases} 1 & \text{if } \sum_{i=1}^{d} w_i x_i \geq T \\ 0 & \text{else} \end{cases}$$

Neural Network with Single Hidden Layer



Approximation of Functions / Boolean Logic



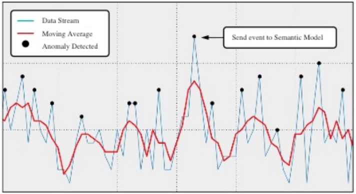
Machine Learning Capabilities (1997-2014)

Learning Streams of Text

- Download complete works of Shakespeare (5.4 million characters)
- Train machine to remember text.
- Write new Shakespeare!

```
Sample after 1000 Iterations
----------------------------

  OTHERS. Allow up, whe, Will a moutice,
  Reason thot boy Eckents alliem; comn jusgius,
  And encarnembover hath last)!
  TASTOR. When no; devires at though me beligo jody?
  MANCELOT. Trom juther and bur itnot spock,
  That as take have wendisho

Sample after 2000 Iterations
----------------------------

  VIRGTARD. Walk this boy and door as am 'stone!
  NTRURLEN. Being entainure af Haton.
  His that by much that, I mion's now, who make foll the kidl!
  CLOTE. Why which Hamm'd?
  QUICKLY. And stand quast of I; this Fi

Sample after 2500 Iterations
----------------------------

  FORD. Nay, You're, excount: and now did yet.
  PAROLLES. Take DUTBY
  This who is begin Gnoban a bows; but yet which that have be,
  Oll, thou stam, and me not ready withered gids
  And he in the pleasues or pardon us.
  Mer. I pray you, how can, and tu

Sample after 3000 Iterations
----------------------------

  Hume, 'tigoning, dear?
  My lam, an hour, chork'd more in my grain,
  I am thing forent innonion, madam! I thank I do speak you?
  ABESSITGBO. But his bosines, giving to know; foward to
  the distvesail.' The to you well know yea, my lovi.
  SECOND CMONBOR. He needs, for the reforeds are;
```

Time Series Anomaly Detection



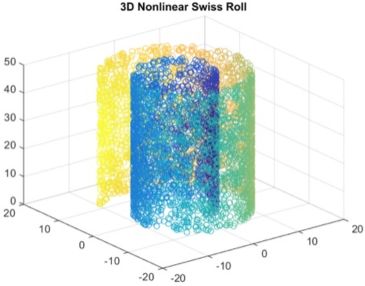
Time Series Prediction

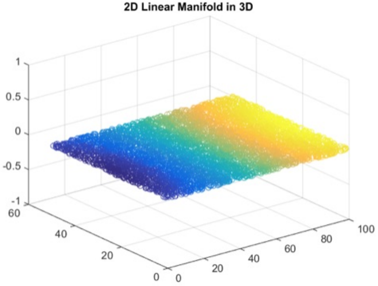
Classification of Machine Learning Problems

Dimensionality Reduction

Strategies of dimensionality reduction involve transformation of data to new (lower) dimension in such a way that some of the dimensions can be discarded without a loss of information.

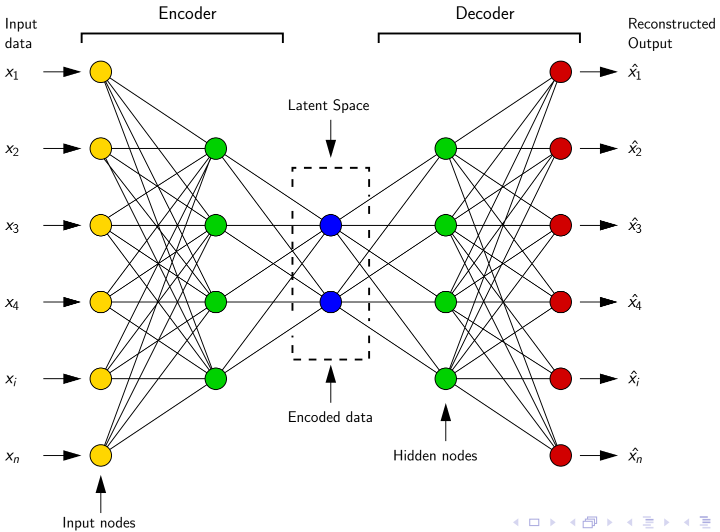
Example: Projection of Swiss Roll data in 3D to 2D ...





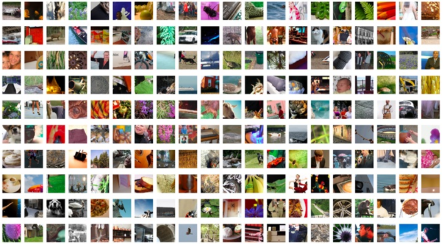
Classification of Machine Learning Problems

AutoEncoder (Encoder-Decoder-Reconstruction)



Classification of Machine Learning Problems

ImageNet and Deep Learning (2009-present)



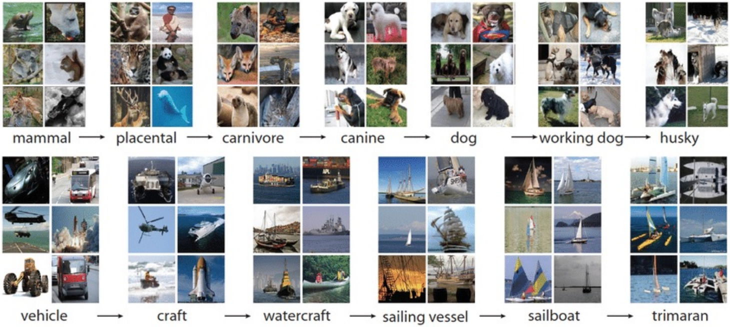
Indexed Database of 14.2 million Images

- Project initiated by Fei Fei Li in 2006
- Image annotation process crowd sourced via Amazon's Mechanical Turk. Categories derived from WordNet.
- Well organized $\rightarrow$ supervised machine learning.

Classification of Machine Learning Problems

ImageNet and Deep Learning Capabilities:

- Identify objects in an image.
- 27 high-level categories; 21,800 sub-categories.



ImageNet and Deep Learning

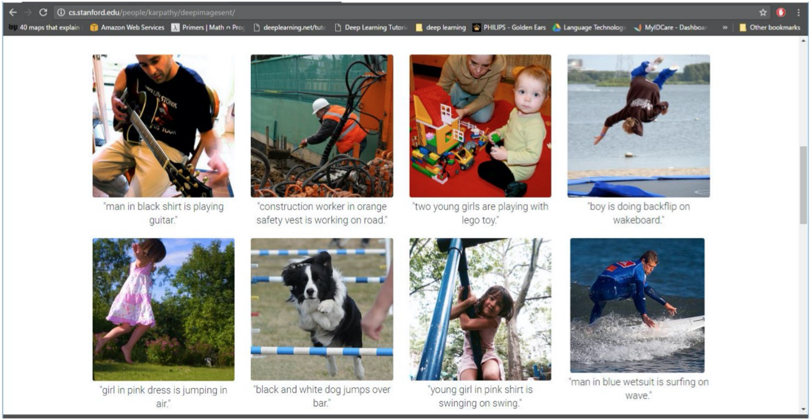
Capabilities (2018):

- Identify relationship among multiple objects in a image.

Example. Dog riding skateboard

ImageNet and Deep Learning

Captions generated by a neural network:



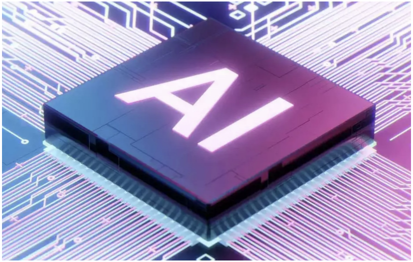
Machine Learning at Scale

Object-recognition module:

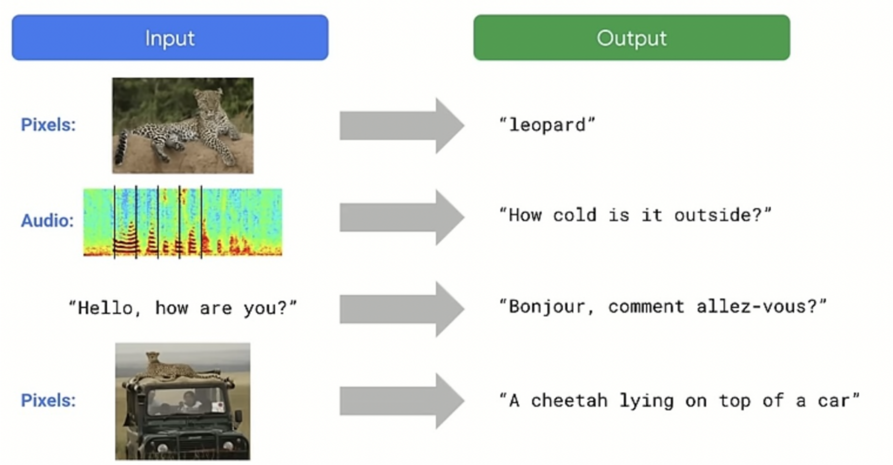
- 24 million nodes; 140 million parameters; 15 billion connections.

Source: Fei Fei Li, TEDTalk, YouTube 2015.

AI Chips: Nvidia, Google TPUs, etc ...

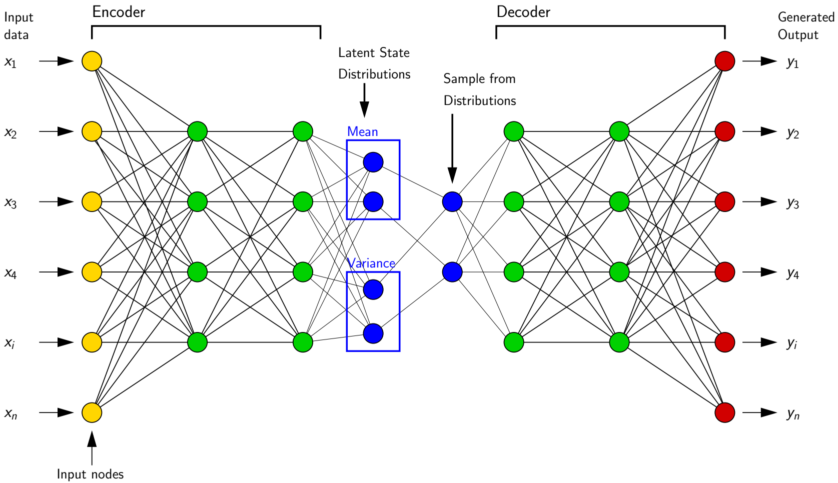


Machine Learning at Scale (Transformers)



Post-2020 Era → Explosion of Generative AI

Variational AutoEncoders (Generative Models)

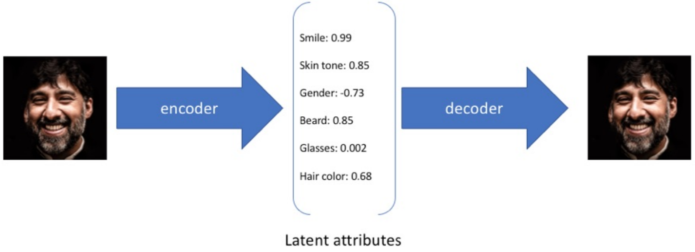


Post-2020 Era → Explosion of Generative AI

Standard Autoencoders vs. Variational Autoencoders:

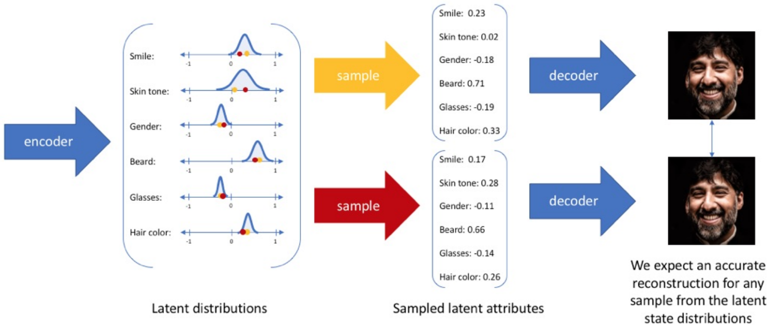
- A standard autoencoder outputs a single value for each encoding dimension.
- Variational autoencoders provide a probability distribution for each latent attribute.

Example: Single value representations for latent attributes:



Post-2020 Era → Explosion of Generative AI

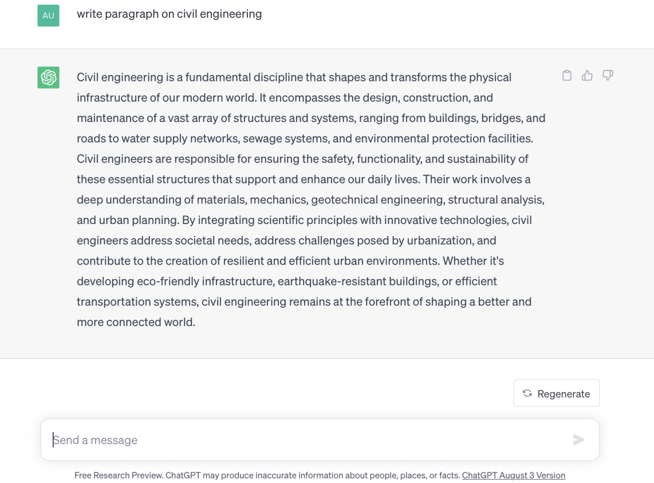
Image Reconstruction: sampled from latent distributions ...



Source: Jordan J., Variational Autoencoders, Data Science, March 2018.

Post-2020 Era → ChatGPT

ChatGPT: Generative Pre-Trained Transformer



Post-2020 Era → ChatGPT

AU integrate sin(3x)*sin(2x) dx

To integrate the expression ∫sin(3x) * sin(2x) dx, we can use the product-to-sum trigonometric identity, which states that:

sin(A) * sin(B) = (1/2) * [cos(A - B) - cos(A + B)].

Using this identity, we can rewrite the integral as:

∫sin(3x) * sin(2x) dx = (1/2)∫[cos(3x - 2x) - cos(3x + 2x)] dx
= (1/2)∫[cos(x) - cos(5x)] dx.

Now, we can integrate each term separately:

∫cos(x) dx = sin(x) + C1,

∫cos(5x) dx = (1/5)sin(5x) + C2.

Putting it all together:

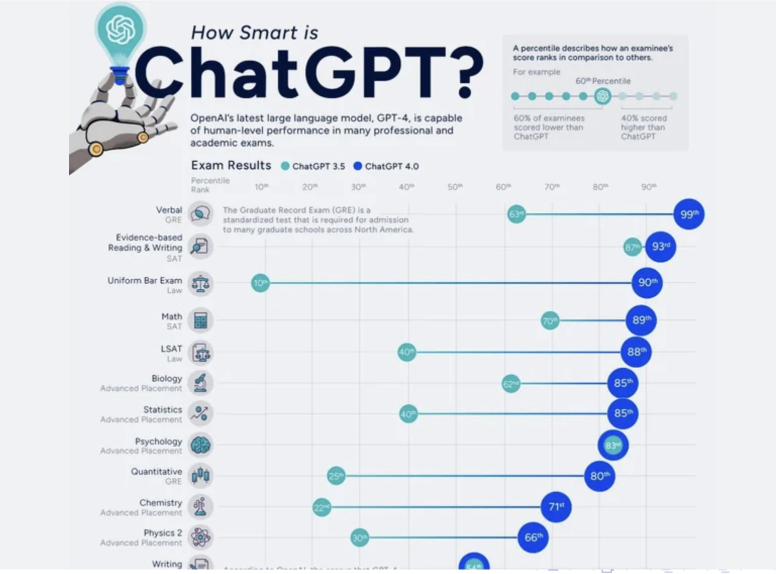
∫sin(3x) * sin(2x) dx = (1/2) * [sin(x) + C1 - (1/5)sin(5x) - C2]
= (1/2) * [sin(x) - (1/5)sin(5x)] + (C1/2 - C2).

So, the integral of sin(3x) * sin(2x) with respect to x is:

∫sin(3x) * sin(2x) dx = (1/2) * [sin(x) - (1/5)sin(5x)] + C,
where C = C1/2 - C2 is the constant of integration.

Regenerate

Post-2020 Era → ChatGPT



Post-2020 Era → AI Generated Architecture

Convergence: Engineering-Architecture-AI

AI-generated art ...

AI-generated building architecture

Post-2020 Era → AI Generated Presentation

RT
Russian TV channel unveils AI weather ...

DayFR Euro
AI weather forecaster

AI and Machine Learning

in CEE

Machine Learning in CEE

Opportunities for Machine Learning in CEE:

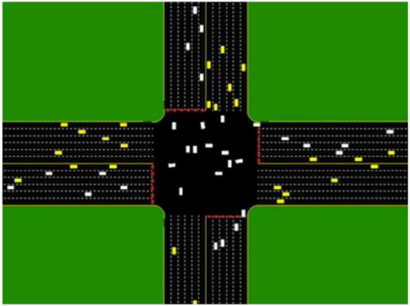
- Predicting system response and performance.
- Interpreting data and formulating models to predict component and subsystem-level properties.
- Information retrieval from images and text.
- Recognizing patterns in streams of sensed data.

Economic Considerations (CS Perspective):

- Urban infrastructure is permanent/semi-permanent and very expensive to build and maintain.
- Prioritize improvements to efficiency by identifying and removing bottlenecks in performance.
- Use AI-ML to identify events, cause-and-effect relationships, and design of actions that enhance system performance.

AI-ML Enabled Decision Making (Self-Driving Cars)

Goal. Improve performance by removing bottlenecks → no human driver; no traffic lights.



Remark: 95% of the requirements are for the system software.

Source: ISR visitor from GM Research.

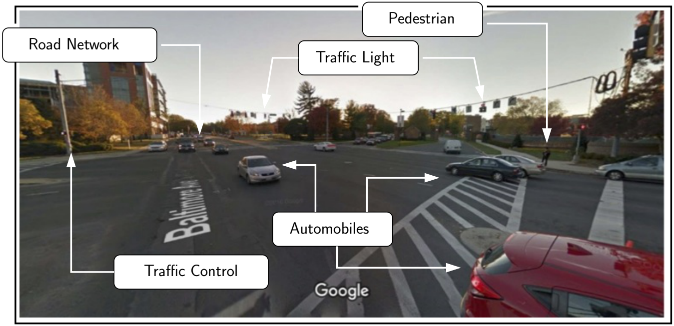
Remark: Tesla will produce self-driving cars by 2016.

Source: Elon Musk.

Stop signs and traffic lights are replaced by mechanisms for vehicle-to-vehicle communication (Adapted from http:citylab.com).

AI-ML Enabled Decision Making (Self-Driving Cars)

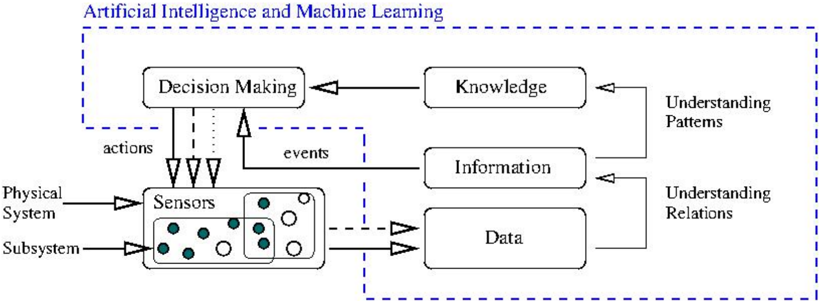
Goal. How to traverse traffic intersection safely and without causing an accident?



Required Capability. Observe, evaluate, reason, take actions.
Challenges. Multiple domains, multiple streams of heterogeneous data, event-driven behavior, dynamic, time critical.

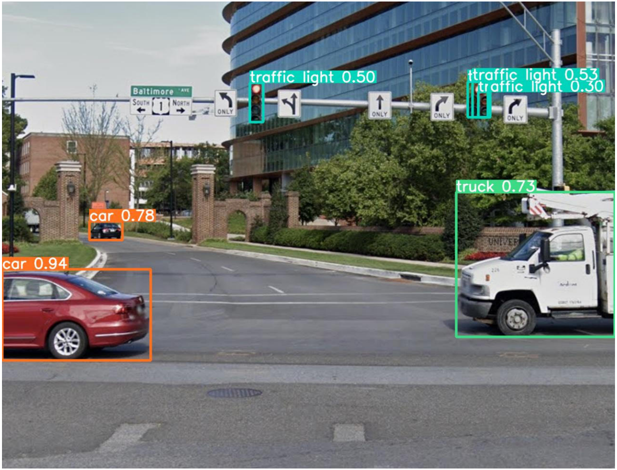
AI-ML Enabled Decision Making (Self-Driving Cars)

Solution Procedure. Pathway from sensing and data collection to ... action ... improved performance, now enabled by AI and ML capabilities:

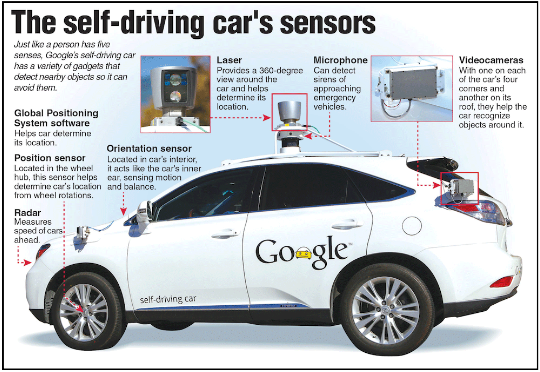


AI-ML Enabled Decision Making (Self-Driving Cars)

Ainur's Experiments with Computer Vision (OpenCV):



AI-ML Enabled Decision Making (Self-Driving Cars)



Today: Modern automobiles → 100 million lines of software.
Tomorrow: Self-Driving automobiles → 200-300 million lines of software.

AI-ML Enabled Decision Making (Self-Driving Cars)

Navigating a Busy Traffic Intersection:

- Identify various kinds of objects (e.g., vehicles, crosswalk).
- Predict what objects will do next.
- Conduct safety assessment.
- Take action.

Google DeepMind (2018-2020)

Teach Self-Driving Cars to Navigate a City without a Map

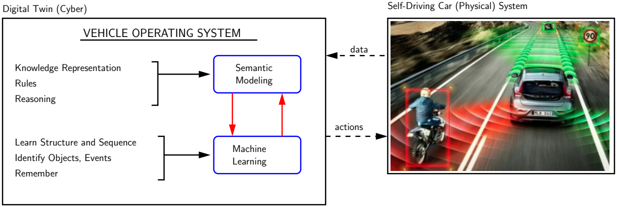
Test Cities: London, Paris, New York.

Digital Twin Systems

New Computing Infrastructure $\rightarrow$ New System Abstractions

Digital Twins (2000-today)

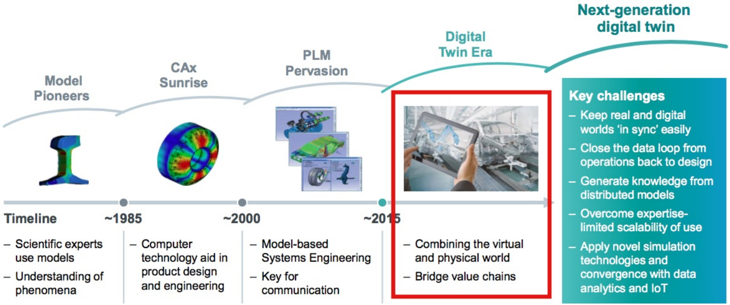
Definition. Virtual representation of a physical object or system that operates across the system lifecycle (not just the front end).



Required Functionality

- Mirror implementation of physical world through real-time monitoring and synchronization of data with events.
- Provide algorithms and software for observation, reasoning, and physical systems control.

Digital Twins (Business Case + Applications)



Many Applications

- NASA Spacecraft
- Manufacturing processes
- Building operations
- Personalized medicine
- Smart Cities
- ... etc.

Digital Twins (Technical Implementation)

Technical Implementation (2023, Google, Siemens, IBM)

- AI and ML will be deeply embedded in new software and algorithms.

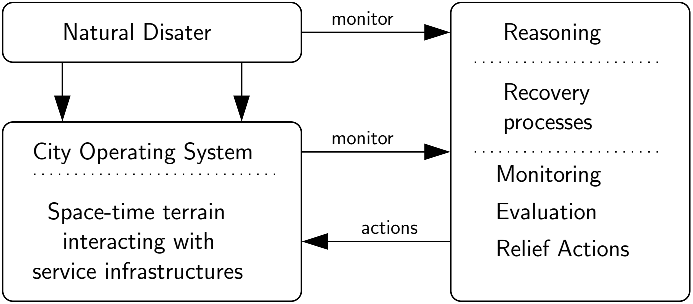
Artificial Intelligence:

- Knowledge representation and reasoning with ontologies and rules. Semantic graphs. Executable event-based processing.

Machine Learning:

- Modern neural networks. Input-to-output prediction.
- Data mining.
- Identify objects, events, and anomalies.
- Learn structure and sequence. Remember stuff.

Digital Twin: City Operating Systems

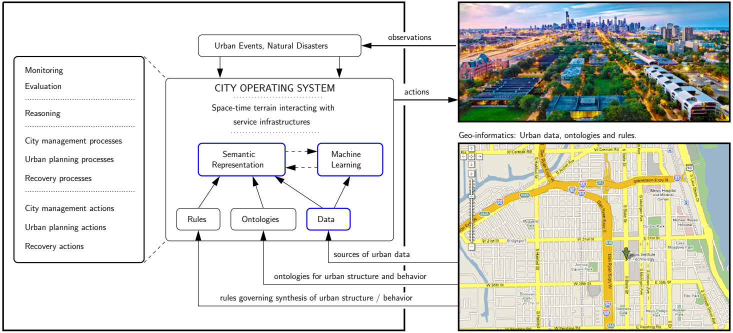


Smart City Digital Twins (2018-2019)

Smart City Digital Twin Architecture

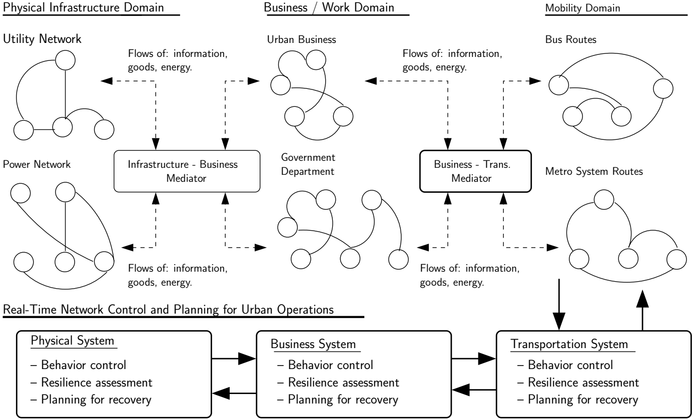
Downtown, Chicago, USA

Geo-informatics: Urban data, ontologies and rules.



Required Capability. Monitoring and control of urban processes.
Complications. Potentially, a very large number of digital twins. Distributed decision making.

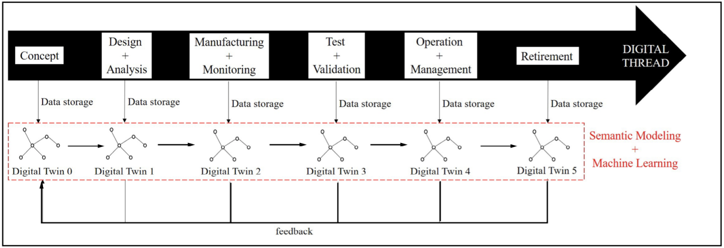
Smart City Digital Twins (2018-2019)



Requirements. Support for digital twin individuals and digital twin communities.

Digital Thread Systems

Digital Threads: (Cradle-to-Grave Lifecycle Support) ...

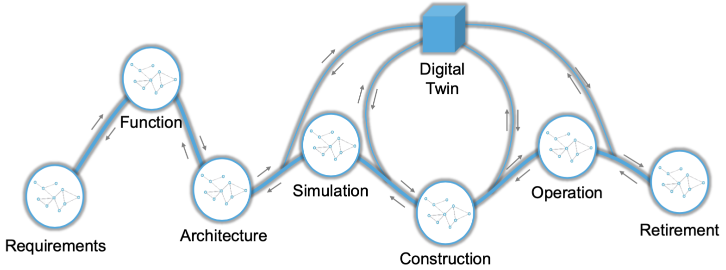


Graph-based Approach

A lot of model-centric engineering boils down to representation of systems as graphs and sequences of graph transformations punctuated by decision making and work/actions.

Digital Thread Systems

Digital Thread System at INL: (Conceptual Model) ...



Def'n: A digital thread is an interconnected software data exchange used to enable digital engineering and digital twinning systems ...

Source: Coelho and Browning, INL, 2022.

Urban Applications

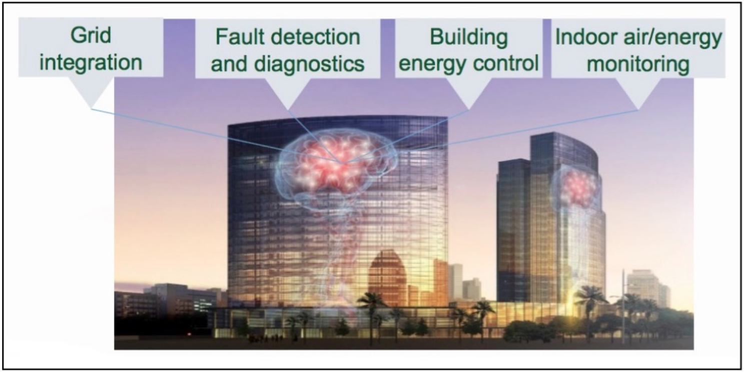
How do buildings work?

Array of Things: Sensing behavior in Chicago

SONYC: Noise Pollution in NYC

Modern Buildings (Vision for Future)

Example 1: Buildings that Think! (Work at NIST / UMD, 2017)



Research Question: How to use AI / Semantics to bring data, context and algorithms together for decision making?

Legend: data = building geometry; context = occupant behavior; algorithms = reasoning.

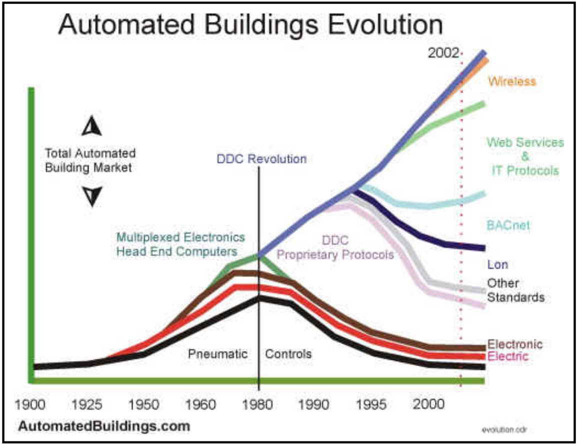
Modern Buildings (Key Features)

Modern buildings are:

- Advanced, self-contained and tightly controlled environments design to provide services (e.g., transportation, lighting, etc).
- Large size (e.g., 30,000 occupants, thousands of points of sensing and control for air quality and fire protection).
- Many stakeholders; highly multi-disciplinary.
- Building have networks for: arrangement of spaces; fixed circulatory systems (power, hvac); dynamic circulatory systems (flows of energy).
- Many sources of heterogeneous data.
- Necessity of performance-based design and real-time management.
- System functionality controlled by software!

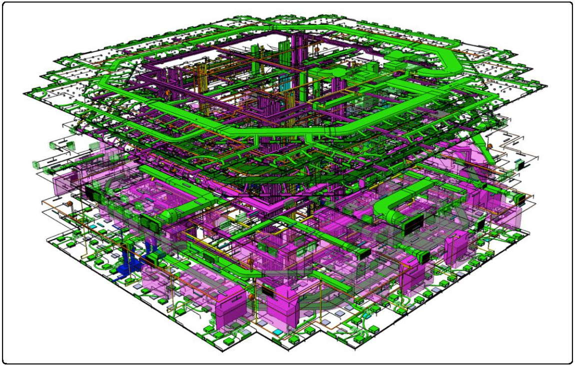
Modern Buildings (Key Features)

Large-scale building systems are packed with automation:



Modern Buildings (Key Features)

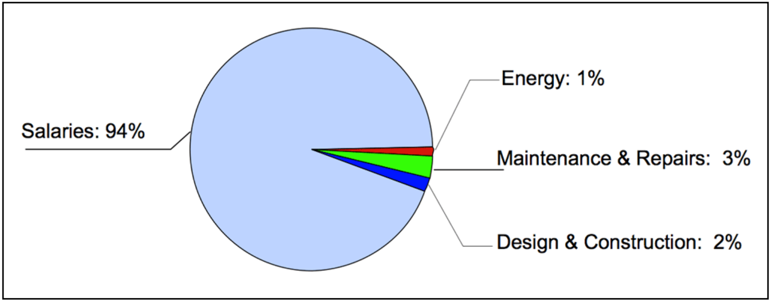
Large-scale building systems are intertwined networks of networks:



Understanding the relationships among the networks and their combined behaviors can be very challenging.

Modern Buildings (Economics)

Lifecycle costs in office buildings over a 30-Year period:



Energy systems have a huge impact on building occupant comfort and indoor air quality which, in turn, affects salary performance.

Source: United Technologies Research Center, 2009.

Modern Buildings (Integrated Energy Systems)

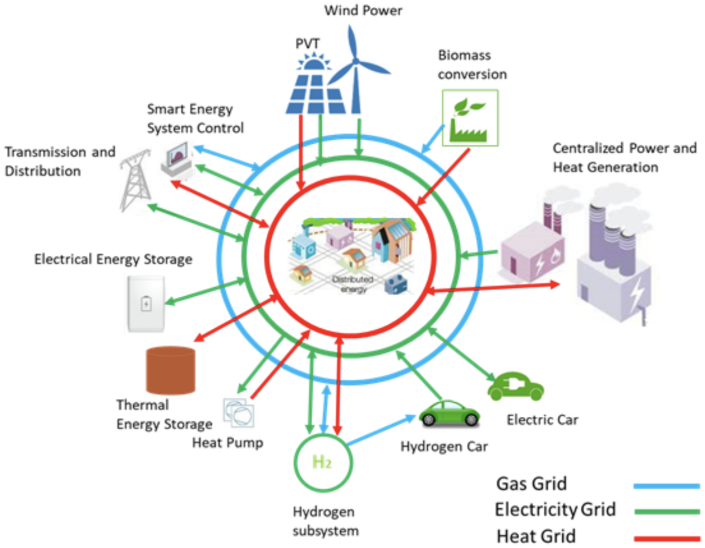
Trend toward Integrated Energy Systems:

- Commercial and residential buildings consume 1/3 of the world's energy.
- And by 2025, buildings will consume more energy than the transportation and industrial sectors combined.
- Standard models of building operation rely on centrally produced power as a source of high-grade energy.
- Advances in technology allow for consideration of alternatives, such as local production of power.

Examples:

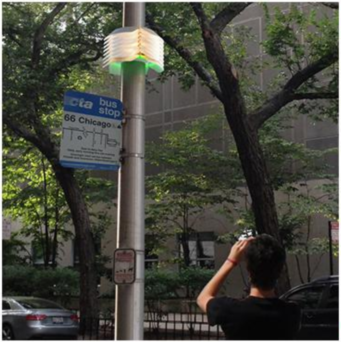
- Solar power; small-scale combined heat and power systems.
- Electricity production through use of ducted wind turbines.

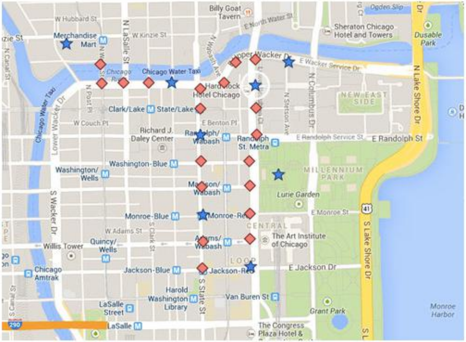
Integrated Energy Systems (Proposed)



Smart Cities: Urban Sensing in Chicago

Array of Things, Chicago (EOL 2022). Modular sensor boxes will collect real-time data on the city's environment, infrastructure and activity.





Basic Questions. How is the city used? What is going on?

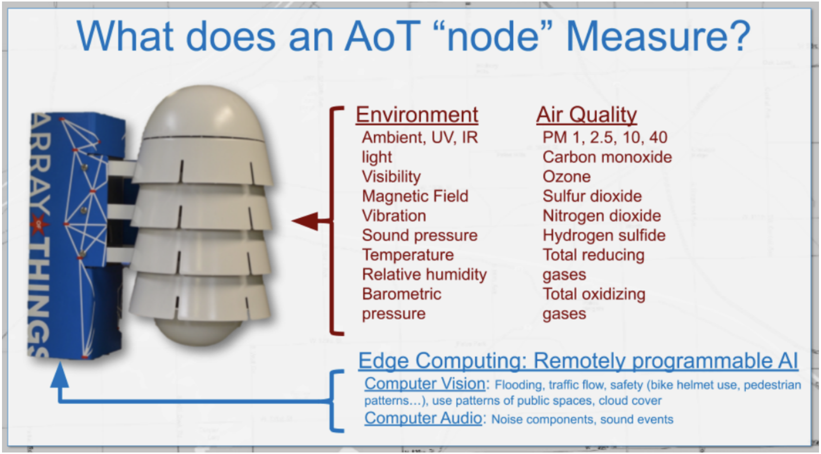
Smart Cities: Urban Sensing in Chicago

What Data is Collected?

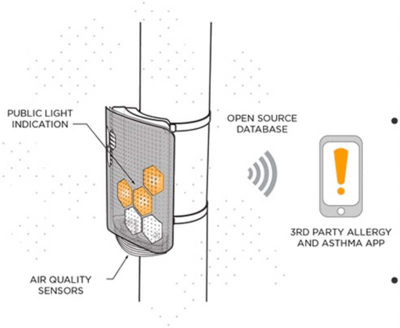
The nodes will initially measure temperature, barometric pressure, light, vibration, carbon monoxide, nitrogen dioxide, sulfur dioxide, ozone, ambient sound intensity, pedestrian and vehicle traffic, and surface temperature. Continued research and development will help create sensors to monitor other urban factors of interest such as flooding and standing water, precipitation, wind, and pollutants.

Array of Things is interested in monitoring the city's environment and activity, not individuals. In fact, the technology and policy have been designed to specifically avoid any potential collection of data about individuals, so privacy protection is built into the design of the sensors and into the operating policies. Array of Things will not collect any personal or private information.

Smart Cities: Urban Sensing in Chicago



Smart Cities: Urban Sensing in Chicago



What Can be Done with this Data?

Potential applications of data collected by the Array of Things include:

- Sensors monitoring air quality, sound and vibration (to detect heavy vehicle traffic), and temperature can be used to suggest the healthiest and unhealthiest walking times and routes through the city, or to study the relationship between diseases and the urban environment.
- Real-time detection of urban flooding can improve city services and infrastructure to prevent property damage and illness.
- Measurements of micro-climate in different areas of the city, so that residents can get up-to-date, high-resolution "block-by-block" weather and climate information.
- Observe which areas of the city are heavily populated by pedestrians at different times of day to suggest safe and efficient routes for walking late at night or for timing traffic lights during peak traffic hours to improve pedestrian safety and reduce congestion-related pollution.

SONYC: Sounds of New York City

SONYC. A system for monitoring, analysis and mitigation of urban noise pollution.

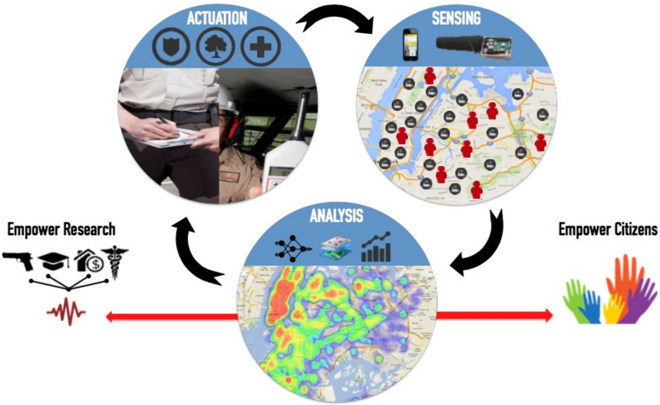
Motivation. Over 70 million people in US are exposed to noise levels beyond the limit of EPA considers to be harmful.

Short-term Problems. Sleep disruption.

Long-term Problems. Hypertension, heart disease, hearing loss.

SONYC: Sounds of New York City

Complaints. NYC authorities receive more than 800 noise-related complaints per day!



SONYC: Sounds of New York City

Noise Analytics. Analyze and understand noise pollution at a city-scale.

Summary

Recurring Themes and Key Points

Recurring Themes

- Information-age systems offer enhanced functionality and better performance, but their design is more difficult than in the past.
- Physical systems and computational systems fail in completely different ways.
- Sensor networks will form the eyes and ears of complex control and information systems.
- As system complexity increases, more and more of the functionality will be managed by software!

Key Points for Building Better Systems

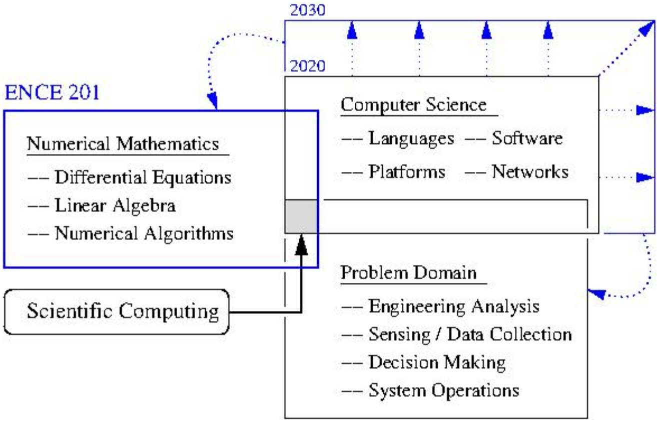
Looking Forward

Use sensing and software to build better systems:

- Improve situational awareness – to understand what is actually happending a building or city?
- Connect sensor measurements to short- and long-term urban needs (e.g., decisions on a bus stop; longer term urban planning).
- Capture the spatial, temporal, and intensity aspects of environmental phenomena (e.g., fires, flooding) and their impact on natural (e.g., air quality) and man-made systems (e.g., transportation networks, food chains).
- Look ahead and forecast future states of the system?

Central Role of Scientific Computing

Scientific computing lies at the intersection of computer science, numerical mathematics, and domain-specific problem solving.



Central Role of Scientific Computing

Computer Science and Software:

- Very fast computations.
- Mass collection of data.
- Rapidly growing importance of data sciences.
- Artificial Intelligence and Data Mining.
- Machine Learning.

Numerical and Applied Mathematics:

- Differential Equations
- Numerical Analysis and Linear Algebra.

Central Role of Scientific Computing

Large-Scale Simulation:

- Improved protection of buildings from extreme environmental loadings (e.g., earthquakes, fire, tsunamis, blast).
- To understand how consequences of global warming (e.g., sea-level rise; wild fires) will impact cities.

Improved Management of Urban Processes:

- Network systems analysis and optimization.
- New strategies for data-driven management of interdepenent urban networks.
- Prevention of cascading failures.

Computer Language for ENCE 201

Getting Started. We need learn to walk before we can run:

Capability	1970s	1980s	1990s
Languages	Fortran, C	MATLAB	Python, Java

Why Python?

- Not too difficult – it's a reasonable place to start learning.
- Good support for data analysis and data analytics.
- Good support for numerical calculations.
- Provides a stepping stone to other languages.

References

- Array of Things: See https://arrayofthings.github.io
- Austin M.A., Delgoshaei P., Coelho M. and Heidarinejad M. , Architecting Smart City Digital Twins: Combined Semantic Model and Machine Learning Approach, Journal of Management in Engineering, ASCE, Volume 36, Issue 4, July, 2020.
- Bello J.P. et al., SONYC: A System for Monitoring, Analyzing, and Mitigating Urban Noise Pollution, Communications of the ACM, 62, 2, 2019, pp. 68-77.
- Coelho M., and Browning L.S., INL Digital Engineering: Model-Based Design, Digital Threads, Digital Twins, Artificial Intelligence, and Extended Reality for Complex Energy Systems, INL/CON-22-69247, Idaho National Laboratory, Idaho Falls, Idaho 83415, September, 2022.
- Jordan J., Variational Autoencoders, Data Science, March 2018.
- Leveson N.G., A New Approach to Software Systems Safety Engineering, System Safety Engineering: Back to the Future, MIT, 2006.
- Tien J.M., Toward a Decision Informatics Paradigm: A Real-Time Information-Based Approach, to Decision Making, IEEE Transactions on Systems, Man, and Cybernetics – Part C: Applications and Reviews, Vol. 33, No. 1, February, 2003.

Appendices

How do Physical Systems Fail?

Physical System Concern

- Design success corresponds to notions of enhanced performance, resilience and reliability.
- Behavior is constrained by conservation laws (e.g., conservation of mass, conservation of momentum, conservation of energy, etc..).
- Behavior often described by families of differential equations.
- Behavior tends to be continuous – usually there will be warning of imminent failure.
- Behavior may not be deterministic – this aspect of physical systems leads to the need for reliability analysis.
- For design purposes, uncertainties in behavior are often handled through the use of safety factors.

How do Software Systems Fail?

Software System Concerns

- Design success corresponds to notions of correctness of functionality and timeliness of computation.
- Computational systems are discrete and inherently logical. Notions of energy conservation ...etc... and differential equations do not apply.
- Does not make sense to apply a safety factor. If a computational strategy is logically incorrect, then "saying it louder" will not fix anything.
- The main benefit of software is that functionality can be programmed and then re-programmed at a later date.
- A small logical error can result in a system-wide failure.