Artificial Intelligence in Engineering Design

by

Mark Ludwigson, P.E.

Course Outline:

AI Overview
Forms of AI
Recent Advances
AI in Transportation Design
AI in Structural Design
AI in Mechanical Design
AI in Electrical Design
Responsible Charge
Helpful References
Examination

AI Overview

Artificial Intelligence (AI) is the re-creation of human cognitive functions by artificial means involving computer software.

AI in Design

Engineering design work requires advanced cognitive skills such as defining design criteria, gathering relevant information, applying technical skills, planning, decision-making, communicating effectively, and producing a unique set of deliverables such as drawings, specifications, reports, cost estimates, and schedules. At a glance, it seems a computer is not up to the task. But with recent developments in AI, a computer can indeed assist an engineer in increasingly more design tasks.

AI can help engineers during design by:

- Gathering and analyzing reference information,
- Generating design ideas and alternatives,
- Optimizing parameters and complex combinations,
- Creating more efficient designs,
- Designing more quickly than a human,
- Creating layouts and 3D models,
- Programming design tools, and
- Quality reviews of designs.

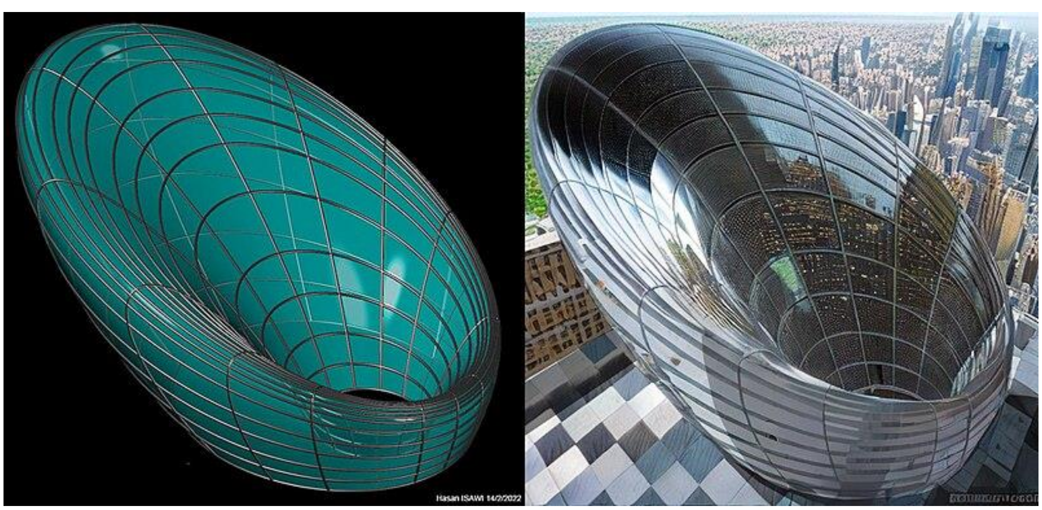


Figure 1: A 3D rendering developed by AI.
Source: commons.wikimedia.org/wiki/File:Serve-tanta-logica-geometrica.jpg, Hasanisawi, CC-BY-SA-4.0

Traditional Algorithms

Traditional computer programs complete functions based on inputs and comprehensive rules (such as logic and formulas) entered by the programmer. The programmer must consider all potential scenarios and hard-code numerous rules into algorithms. The programmer is either an expert in the subject matter or works closely with an expert.

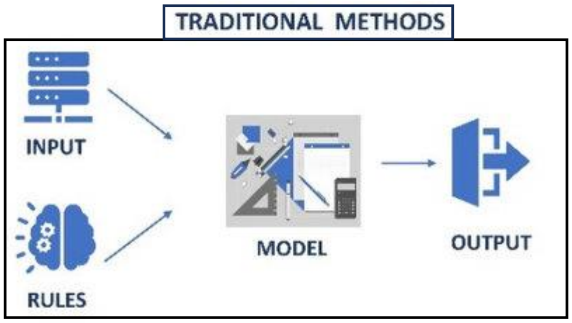


Figure 2: Traditional algorithms use mathematical equations and coded logic (rules) that are applied to a limited set of inputs to produce a model and solutions (output).
Source: https://ui.adsabs.harvard.edu/abs/2022RemS...14.3377K/abstract, CC-4.0

AI Algorithms

AI based programs don't need all the rules to be defined since AI algorithms can utilize input data to determine the rules (step 1), and then apply the rules to new situations (step 2). The programmer does not need to consider all potential scenarios nor be an expert in the subject matter. AI algorithms are particularly useful when dealing with large datasets or complex processes.

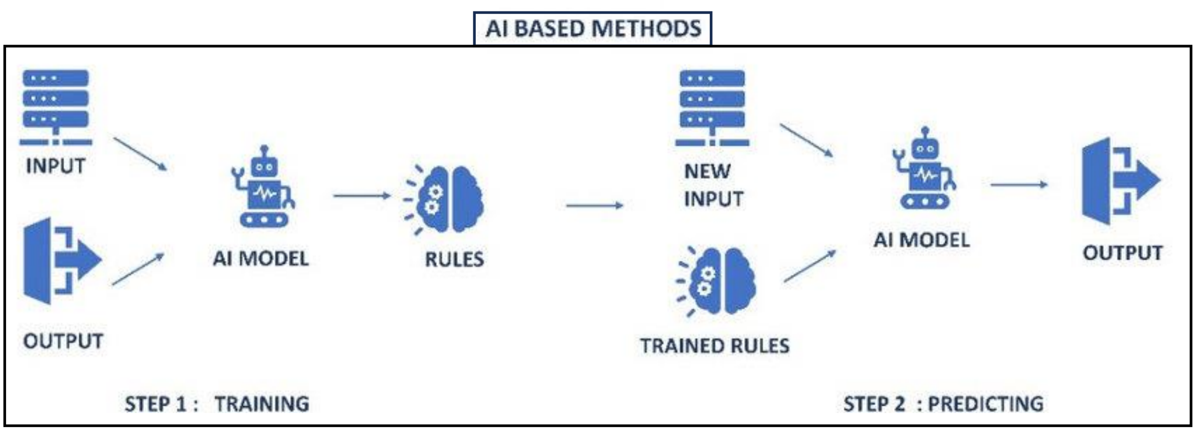


Figure 3: AI-based algorithms analyze initial datasets to find patterns and create rules (step 1). These trained rules are applied to new inputs to provide solutions (step 2).
Source: https://ui.adsabs.harvard.edu/abs/2022RemS...14.3377K/abstract, CC-4.0

Example Application

For example, create a program that sizes pipe diameters with the following inputs and outputs:

- Inputs: Flow Rate, Liquid Type, and Pipe Material
- Outputs: Nominal Pipe Diameter

Traditional Programming:

- Create a database with the following information:
  - Unit conversion rates for flow rates (gpm, MGD, cuffs, cfm, etc.)
  - List of different liquids (potable water, wastewater, ferric chloride, etc.)
  - List of ideal velocities for each liquid (5 fps for potable water, etc.)
  - List of nominal pipe diameters for different pipe materials
- Create a formula for calculating the idealized pipe diameter based on ideal velocities in the database.
- Creates a formula that will select the next largest nominal pipe diameter based on values in the database.
- User enters inputs and program outputs a result.

AI Programming:

- Start a new AI algorithm that accepts inputs and displays outputs.
- Define the input and output options.
- Step 1: Training
  - Give AI program access to reliable design information such as:
    - Example pipe installations,
    - Example pipe design problems with solutions,
    - Chemical compatibility charts,
    - Product data from pipe suppliers, and
    - Unit conversions.
  - AI algorithm will analyze the data and define initial rules.
  - User tests the AI program and supplies additional information as needed.
- Step 2: Predicting
  - User enters new inputs and program outputs a result.
  - User can indicate if result is incorrect, and AI modifies the rules to avoid making the same mistake in the future.

Neural Networks

A neural network is a form of AI that is inspired by the human brain. Neural networks are also known as simulated neural networks (SNNs) or artificial neural networks (ANNs). Neural networks involve applying weights to inputs to help determine the most important factors in solving a problem. Iterations can also be included to optimize the results. Figure 4 shows an example of a simple neural network that can recognize objects in an image. Figure 5 depicts a single neuron as part of a larger neural network.

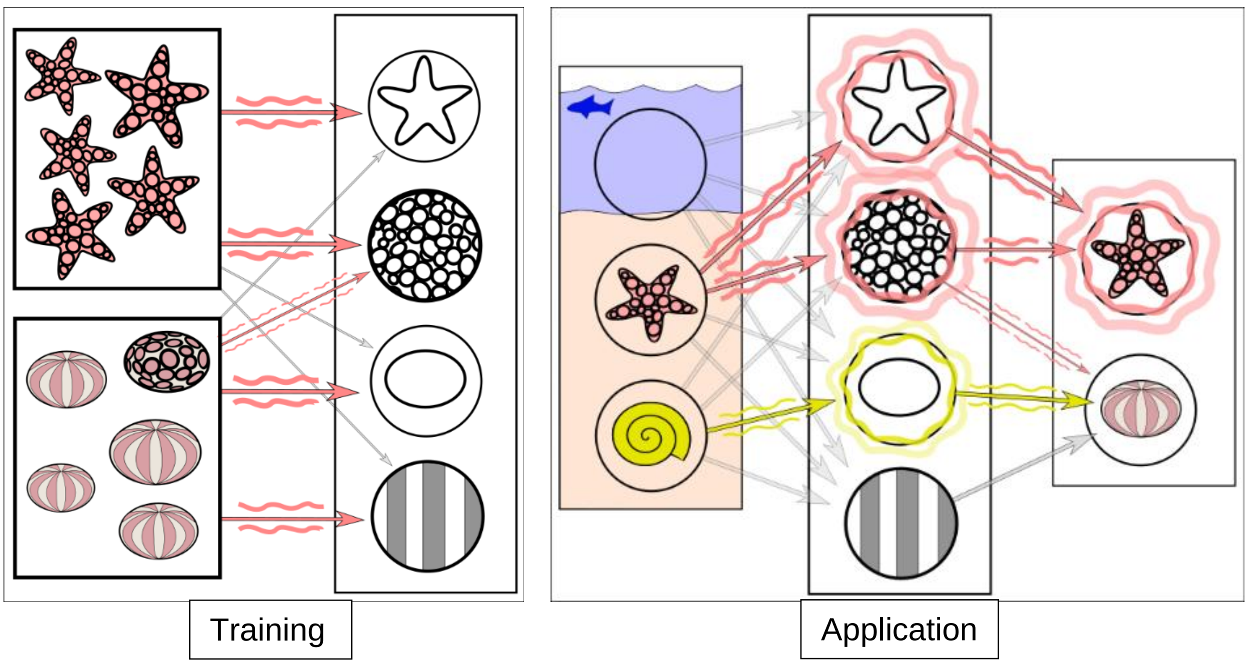


Figure 4: Example AI neural network program for recognizing starfish and sea urchins in images. The program is first trained with example images (left) to find patterns. The program is then applied to new images (right). The yellow shell is circular like a sea urchin so there is a weak correlation. Applying higher weights for striped texture and less for circular shape can help avoid a false positive.

Source: https://en.wikipedia.org/wiki/File:Simplified_neural_network_example.svg, Mikael Häggström, M.D., public domain

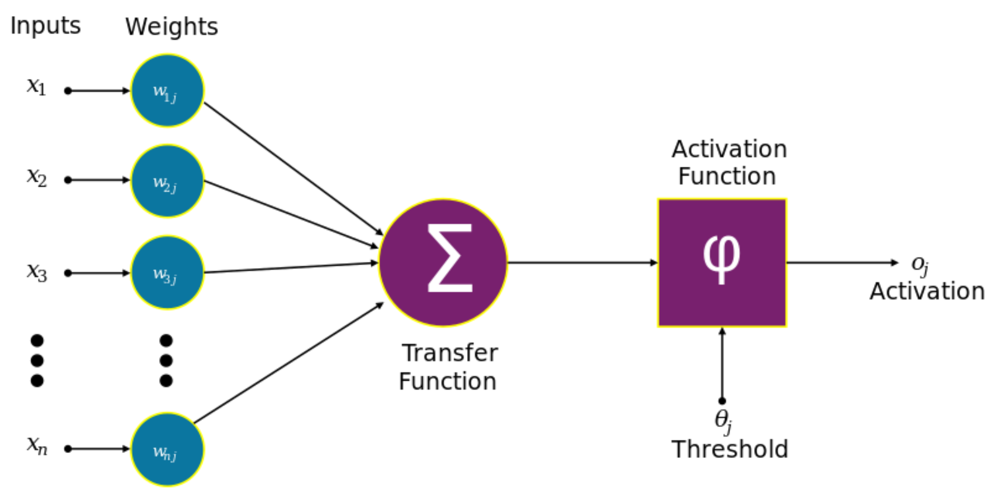


Figure 5: Depiction of an artificial neuron. The "activation" on the right is the output.
Source: commons.wikimedia.org/wiki/File:Artificial_neuron_structure.svg, public domain

Evolutionary Algorithms

Some AI programs use evolutionary algorithms which are modeled after biological evolutionary processes. The goal is to find and optimize a sufficiently good solution based on multiple factors. This approach can be well suited for design tasks in which there are multiple possible solutions. The algorithm can optimize the solution by running and comparing many scenarios, as shown in Figure 6.

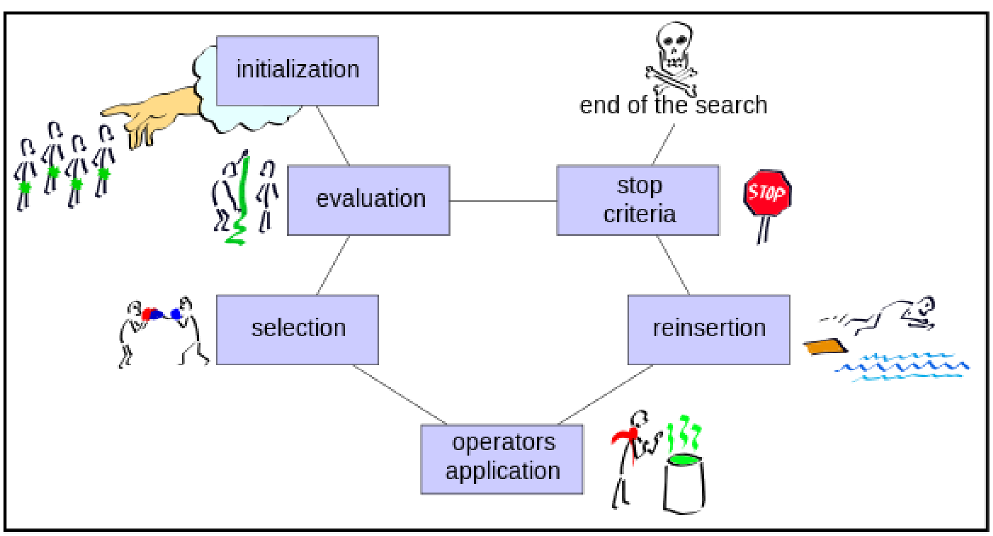


Figure 6: Basic steps in an evolutionary algorithm that compares and selects one person based on multiple objectives.
Source: commons.wikimedia.org/wiki/File:Evolutionary_Algorithm.svg, Jorge.maturana, CC-BY-3.0

Forms of AI

Table 1 provides an overview of common categories (or forms) of AI. New categories are likely to emerge as AI is further developed for vastly different applications.

Table 1: Categories of AI Systems

Category	Intelligence	Learning	Rules	Examples	Notes
Expert Systems	Limited or None	None or Limited	Fixed	KBES, Fuzzy Logic, Diagnostics	Built for specific tasks, AI features are optional
Machine Learning	Cognitive	Supervised, Unsupervised or Reinforced	Self-modified	Deep Blue, Voice-to-Text	Algorithms can learn from new data
Large Language Models	Cognitive	Supervised, Unsupervised or Reinforced	Self-modified	ChatGPT, Chatbots, Search Engines	Perform natural language processing (NLP) tasks, large data
Deep Learning	Cognitive	Neural Network	Self-modified	Autonomous Vehicles, Python, Facial Recognition	Advanced type of machine learning, complex decision-making abilities
Generative AI	Cognitive	Various	Self-modified	Generative Adversarial Networks, Variational Autoencoders	Design examples: structural optimization, shape synthesis
Strong AI	Cognitive & Emotional	Independent	Self-created	None (Theoretical)	Creative abilities, optical computing
Super AI	Cognitive, Emotional, & Social	Superior to Human Learning	Not Required	None (Theoretical)	Self-aware, beyond most human abilities, artistic creativity

Expert Systems	
Description	**Examples**
• The term "expert systems" covers a vast array of computer programs, many with simple input to output algorithms for specific technical tasks. • Expert systems are a good fit when there is a small amount of data and the rules are well understood. • Expert systems can solve very complex problems and use advanced mathematical computations to arrive at a design solution. • There are many engineering design programs that can be considered expert systems. For example, a program that can choose the most efficient structural beam based on a unique set of inputs such as loads, dimensions, connection type, and applicable building codes. • There is an "expert" that creates the initial rules and the initial knowledge base.	• The first expert system was developed in 1965 at Stanford University, called DENDRAL, which stands for "DENDRitic ALGORITHM". The program analyzes mass spectroscopy results to determine the chemical compound being analyzed. • Many expert systems have been developed for technical, scientific and engineering tasks.

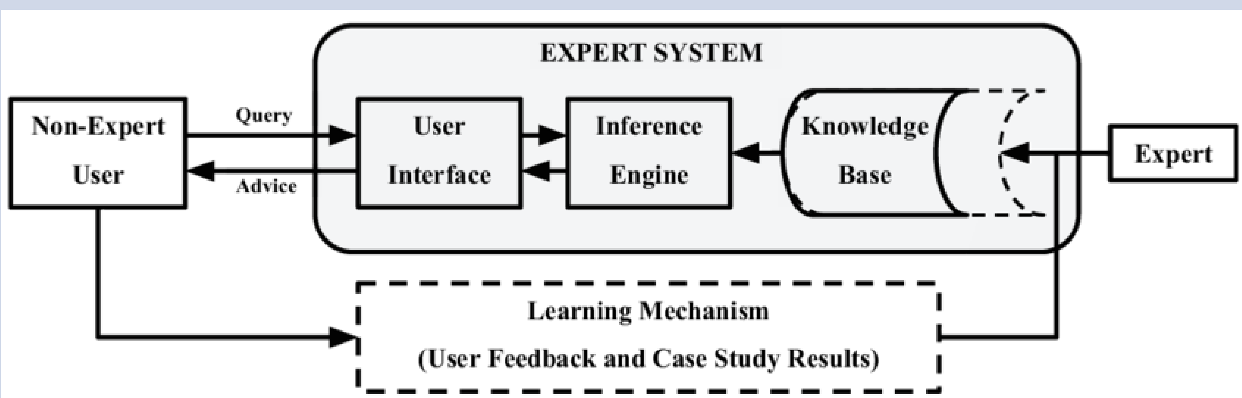


Figure 7: Typical Architecture of an Expert System
Source: https://doi.org/10.1016/j.proeng.2016.04.127, CC BY-NC-ND 4.0

Machine Learning	
Description	**Examples**
• In machine learning (ML), AI algorithms help computers learn from new data and past performance and then modify the functions to increase performance.	• Image Recognition

ML has three methods for "learning":

1. Supervised Learning:
   - Inputs with labels build knowledge about a given task, similar to how humans use "past experience". Desired outputs are given for each example input.
   - The program creates a set of rules so that important input features are recognized and consistently produce desired outputs.
   - Many examples are required to achieve accuracy; significantly more than a person would require when learning a new subject.
   - The learning process employs statistical methods and confidence levels.
   - Manual adjustments of the program are usually not required.
2. Unsupervised Learning:
   - Input data does not need to have labels defined by a user.
   - An algorithm groups the data and features according to unknown properties.
   - Unsupervised programs can either cluster the input data, find an anomaly in it, or reduce the number of dimensions in which to express the dataset.
   - Data points are grouped or clustered based on shared features without knowing what labels or categories are present.
   - Pattern recognition, anomaly detection, and failure detection programs use unsupervised learning to recognize important features and classify data as either part of the set or an anomaly.
3. Reinforcement Learning
   - The program is provided with an input and a reward function that gives an indication of how well the program is doing.
   - The algorithm learns how to maximize the reward.
   - The program is periodically reviewed and modified based on results achieved.
   - Reinforcement learning is used in situations where it is difficult to input a set of perfectly correct labels that will cover potential new data.

Deep Learning	
Description	**Examples**
• Deep Learning (DL) is an advanced type of ML based on artificial neural networks (as explained in the previous section) to process and classify data. • Capable of complex decision-making. • The leading programming language is Python, which is open source.	• Autonomous vehicles use deep learning algorithms to make real-time decisions based on the environment around the vehicle, as depicted in Figure 8.

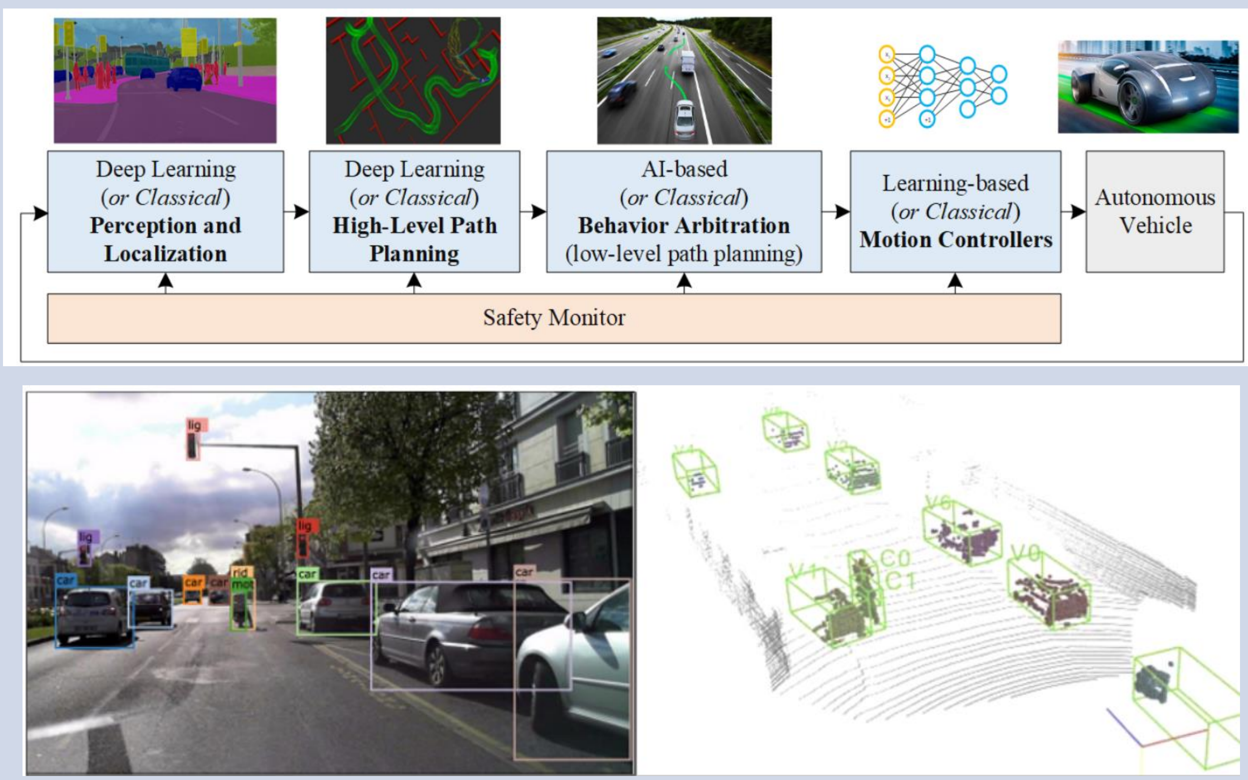


Figure 8: Top: Architecture for self-driving cars with deep learning.
Bottom left: Scene detection with 2D images from bumper.
Bottom right: Scene detection with 3D data from LiDAR on the roof.
Source: https://arxiv.org/pdf/1910.07738, Grigorescu et. al.

Generative AI	
Description	**Examples**
• Generative AI is also called Deep Generative AI or Gen AI. • Gen AI models learn patterns from initial training and apply them to new situations in order to create complex digital deliverables. • Gen AI allows engineers to focus more on project planning, creative problem-solving, and higher-level decision-making. • See Figure 9 for how Gen AI is often considered a subset of deep learning.	• As an example, information and project requirements can be input into a Gen AI program to create a building layout in CAD. • Gen AI uses deep learning programming to produce deliverables, such as text, images, videos, 2D layouts, 3D models, calculation summaries, and unique user specified outputs.

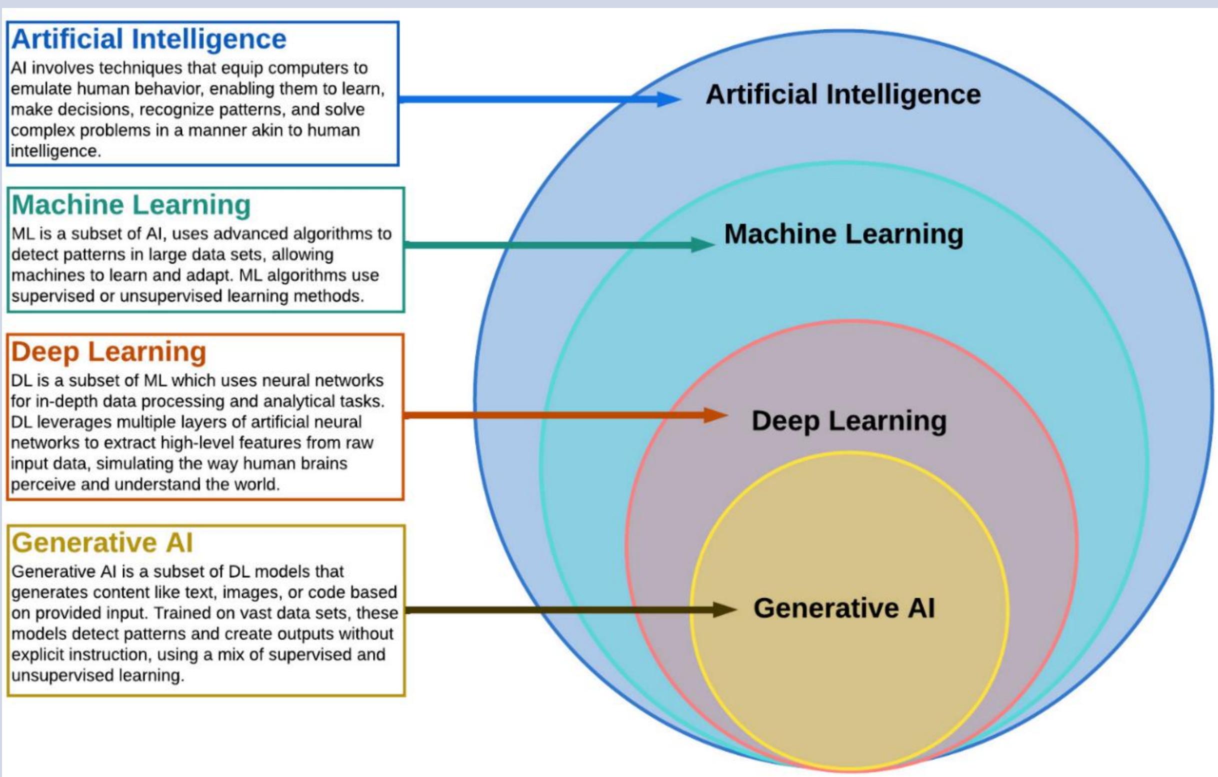


Figure 9: Venn diagram with Generative AI.

commons.wikimedia.org/wiki/File:Unraveling_AI_Complexity_-_A_Comparative_View_of_AI, PopovaZhuhadar, CC-BY-SA-4.0

Recent Advances

The following are recent AI advances related to engineering.

Predictive Maintenance	
Description	**Examples**
• Predictive maintenance (PdM) uses AI algorithms to predict failure or detect the first signs of failure of physical parts such as machines or electrical parts. • Early detection allows maintenance or replacement prior to a costly failure. • Predictive maintenance uses data analysis to continuously monitor operating conditions and working environments to detect early clues for degradation of equipment.	• Accruent • IBM Maximo® Predict, • InfluxData, Fiix • Limble CMMS, • LLumin, MaintainX, • MATLAB, UpKeep, • Reftab, Samsara, • Siemens MindSphere

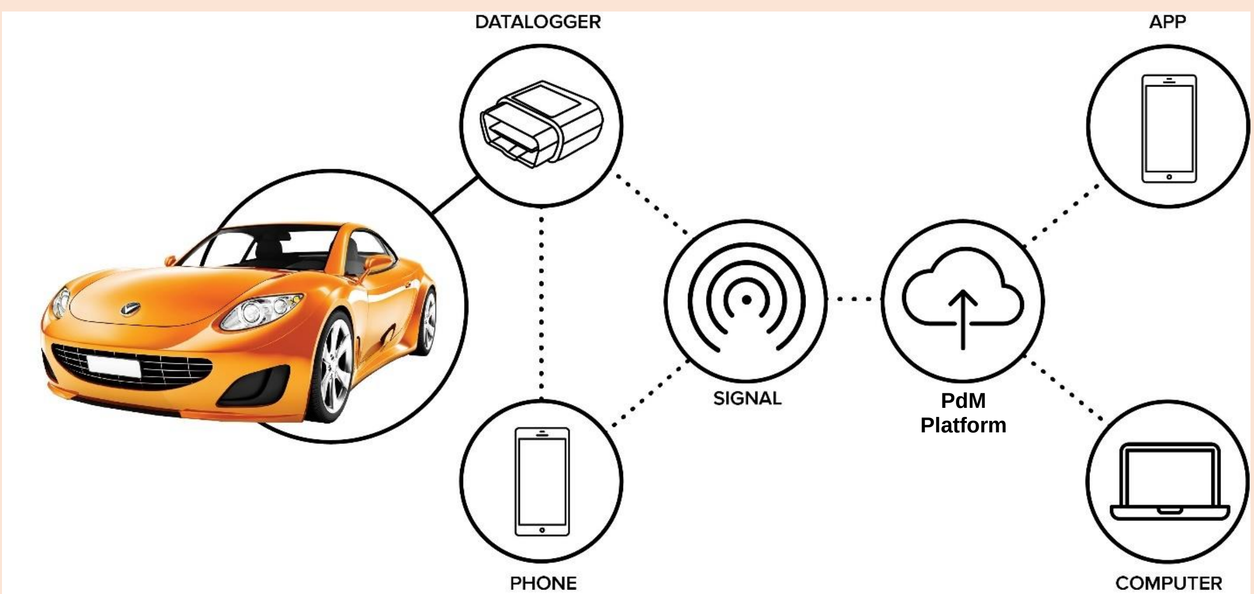


Figure 10: Communication paths for automobile preventative maintenance (PdM).
Source: commons.wikimedia.org/wiki/File:DataLogger_Platform-_How_it_Works.jpg, Danlaw Inc

<table>
<tr><th colspan="2">Digital Twins</th></tr>
<tr><th>Description</th><th>Examples</th></tr>
<tr><td>• A digital twin is a model of a physical system with a boost of AI and real operations data to help simulate dynamic real-world scenarios.
• Uses both deep learning and generative AI.
• Digital twins are built using the current asset condition and relevant historical data about the asset.
• Digital twins can be used to optimize the current condition, optimize operations and maintenance, and predict future behaviors.
• The models can reflect ongoing operations by direct streaming of data into the AI algorithms.</td><td>• Example applications include machinery, engines, pumps, production equipment, assembly lines, packaging systems, power plants, or an entire fleet of vehicles.
• Bentley’s iTwin</td></tr>
</table>

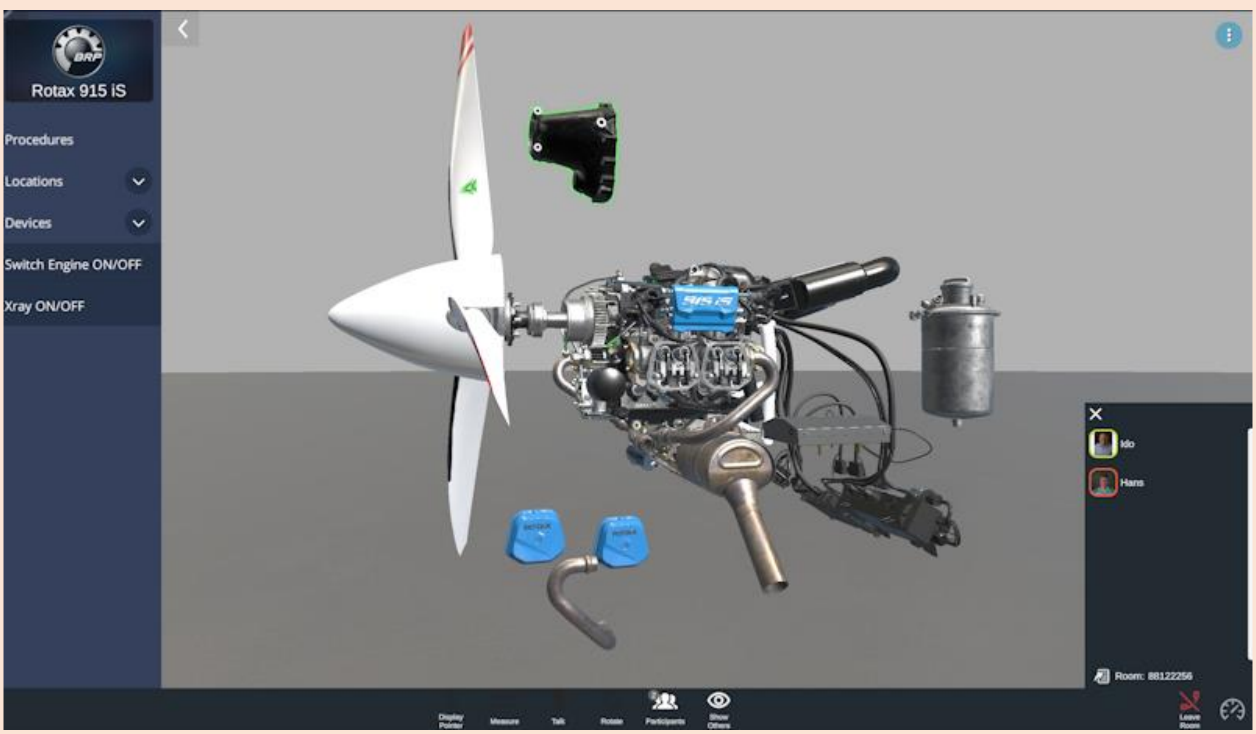


Figure 11: Example digital twin model of a propeller engine.
Source: commons.wikimedia.org/wiki/File:Training_with_digital_twin.jpg, Iditsulkin, CC-BY-SA-4.0

Data Mining	
Description	**Examples**
• Application of ML algorithms to find patterns and relationships in very large data sets. • Common techniques include data classification, association analysis, and regression analysis. • The most widely used data mining method is called CRISP-DM (Cross-Industry Standard Process for Data Mining). • For design, often there is a lack of existing data since: 1) design firms do not openly share project data, 2) human design decisions are not fully documented, and 3) not digitally organized.	• Programing tools include SQL, Python, Hadoop, and Apache Spark. • Data analysis software: GCP Dataflow, GCP Vertex AI, Azure Data Factory, Azure ML Studio, AWS Glue, and AWS Sagemaker. • Existing project databases can be mined for lessons learned.

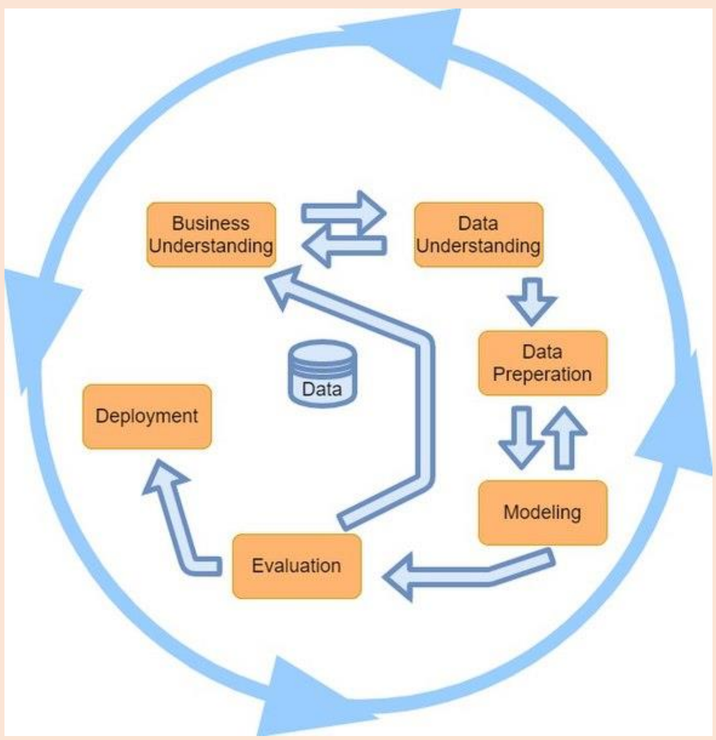


Figure 12: Depiction of CRISP-DM approach. AI is used in the modeling step to extract useful information and make conclusions.

Source: wikimedia.org/wiki/File:CRISP_DM_Data_mining_management_process.jpg, A. Schröder, CC-BY-SA-4

Virtual Assistants	
Description	**Examples**
• Virtual assistant (VA) is a large language model (LLM) that interacts in a human-like manner. • Can perform a range of tasks or services based on user input such as commands or questions. • Typically utilizes online chat (chatbot) capabilities to simulate human conversation. This is also called a visual dialog model. • Some virtual assistants can interpret human speech and respond via synthesized voice. • Can assist engineers in finding information and making informed decisions.	• Apple's Siri • Apple Intelligence • Amazon Alexa • Google Assistant • Microsoft Copilot • Mycroft (open source) • Samsung's Bixby • Slackbot

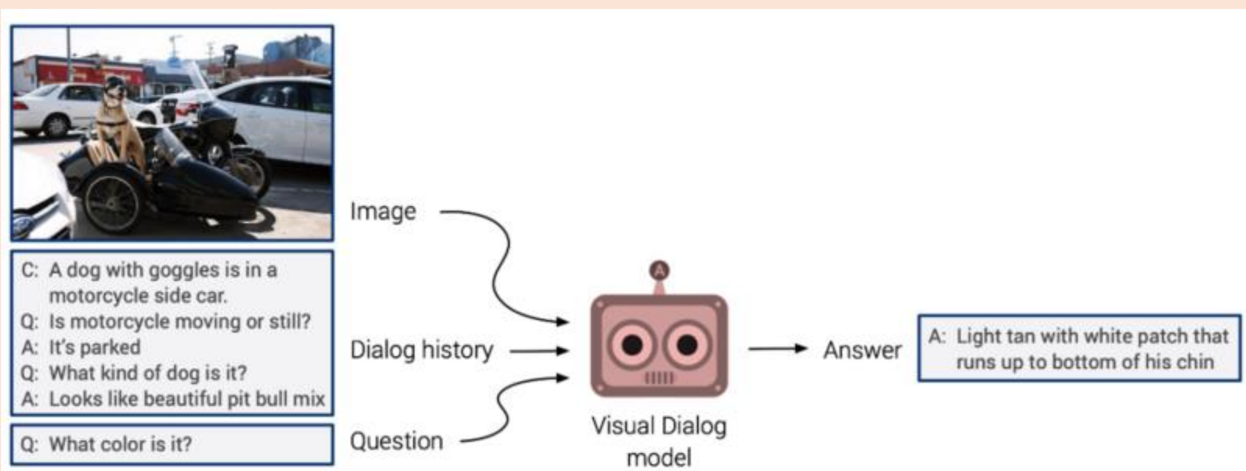


Figure 13: Example of a chatbot answering questions about a unique picture.
Source: commons.wikimedia.org/wiki/File:Visual-Dialog.png, Abhishek Das, CC-BY-SA-4

Smart Infrastructure	
Description	**Examples**
• IoT (Internet of Things) devices linked to a central system with AI for real-time monitoring and control. • Instrumentation and controls systems can become "smarter" with the addition of deep learning or Gen AI. • AI can review trends, monitor current conditions and make adjustments to achieve energy efficiency or other goals. • Some systems sold with label "smart" do not actually have AI programming.	• Utilities infrastructure (water meters, electric meters, monitoring, etc.) • HVAC controls • Security systems • Surveillance systems • Smart home system

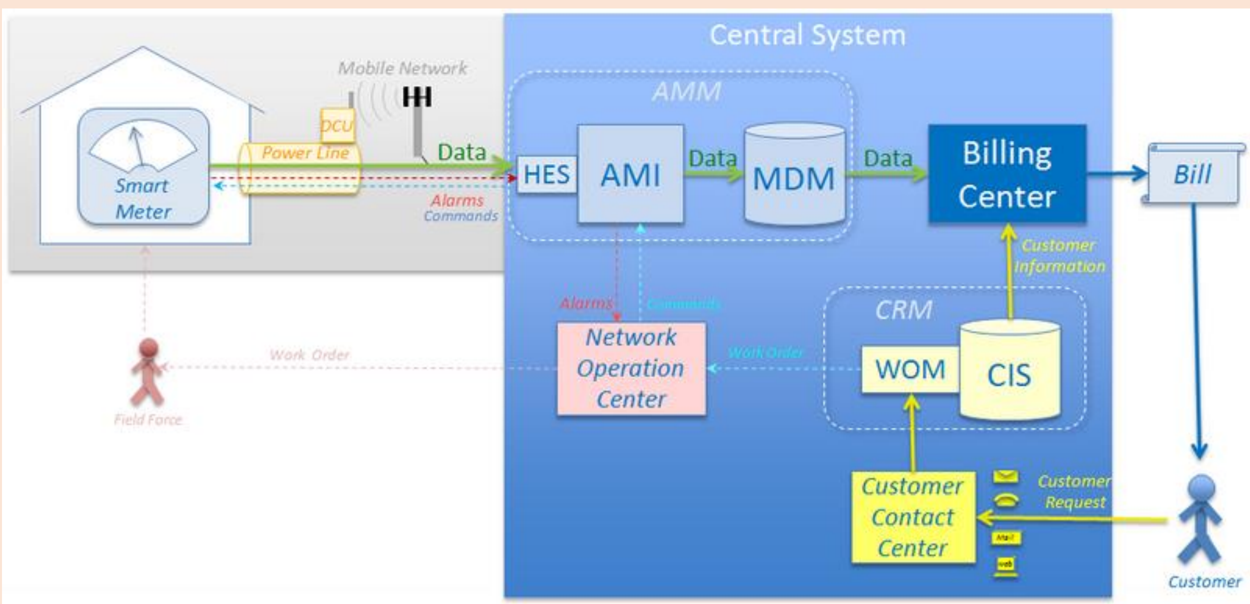


Figure 14: Smart meter system for billing customers.
AMM = Advanced Meter Management (software and hardware bundle)
HES = Head End System (sends and receives smart meters signals)
AMI = Advanced Metering Infrastructure (controls the smart meters, AI potential)
MDM = Meter Data Management (data storage & calculations, AI potential)
Source: commons.wikimedia.org/wiki/File:Smart_Meter_Infrastructure.png, Skyhead, CC-BY-SA-4

Hardware-based AI	
Description	**Examples**
• Hardware-based AI is specifically designed to operate physical components. • Collaborative robots, called cobots, are becoming common in the workplace, working alongside humans in various applications, especially where physical items are being manufactured. • Robots excel in tasks that require high levels of precision, speed, and consistency. • Robotic systems are also equipped with advanced sensors and AI algorithms, enabling them to adapt to their human counterparts' actions and learn from their experiences. • AI programming allows robots to learn and thus become more reliable and efficient with time.	• Autonomous vehicles • Drones • Robots (Cobots, Delta, Educational, Humanoid, Industrial, etc.)

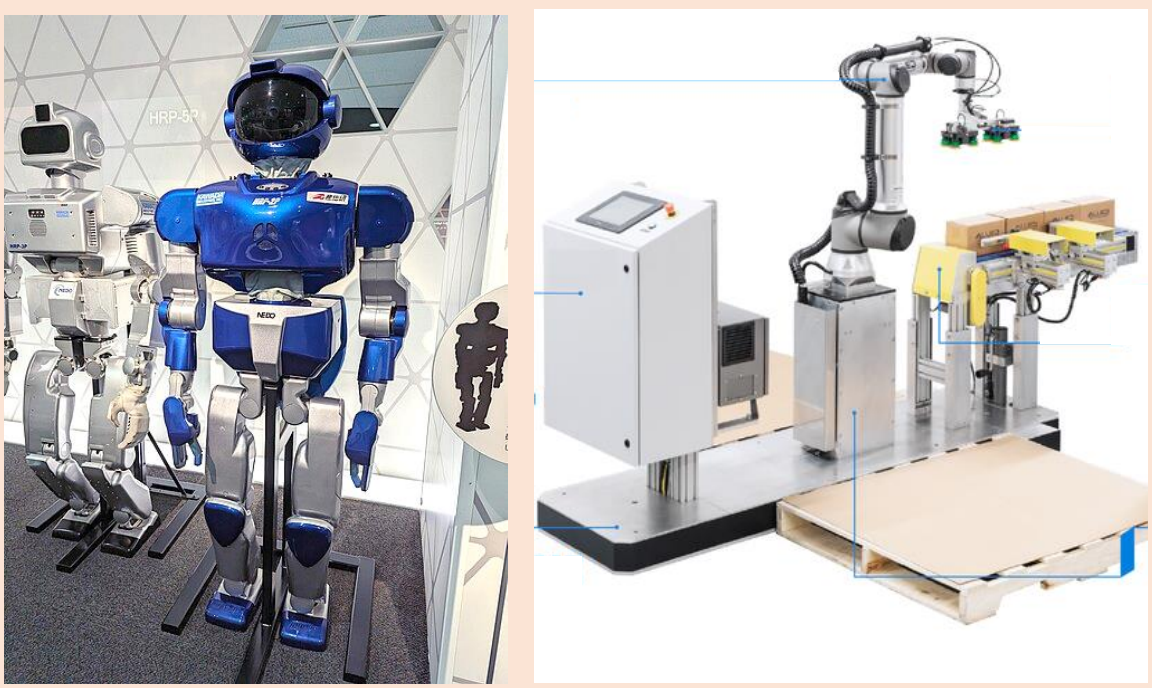


Figure 15: left) humanoid robots; right) industrial packaging cobot.
commons.wikimedia.org/wiki/File:Collaborative-Palletizer-AAA20-RAAS-Universal_Robot-2.png, CollaborativePalletizer, CC-BY-SA-4

AI Code Generation	
Description	**Examples**
• AI Code Generation (also called AI Coding Assistance, Automated Code Generation, or ACG) is the use of an AI program to create code based on a user's conversational prompts. • ACG generates code, writes functions, and aids in code completion. • ACG uses publicly available source code to generate new code. • ACG can create AI-based design tools for specific engineering applications.	• Amazon Q Developer (w/ CodeWhisperer) • Gemini Code Assist • GitHub Copilot • OpenAI ChatGPT • Variety of programming language options • Cloud based or local

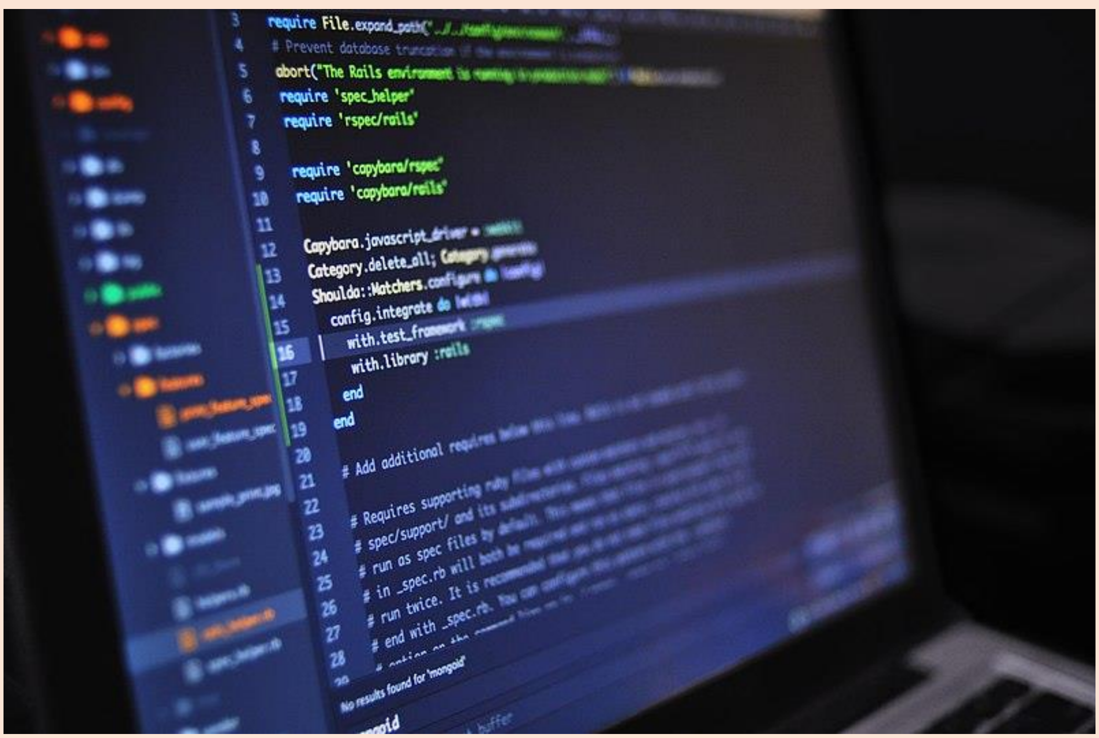


Figure 16: Code like this can be written by AI for engineering applications.
Source: www.pexels.com/photo/close-up-photo-of-programming-of-codes-546819, public domain

Explainable AI (XAI)	
Description	**Examples**
• Explainable Artificial Intelligence (XAI) is a transparent AI program that can clearly explain design decisions, actions taken, and model results to human users. • XAI enables human users to understand, trust, and effectively manage the AI program, the automated design process, and the model results. • Utilizes modified machine learning techniques that produce more explainable models with graphical interfaces and savable outputs in formats like pdf. • XAI opens up the "black box" of AI, allowing users to understand, verify, and challenge decisions made.	• Financial Services: Paypal, ZestFinance, BlackRock • Healthcare: IBM Watson, Google DeepMind, PathAI • Legal: SpeedLegal, Goldman Sachs • DARPA XAI program • Not to be confused with the company xAI

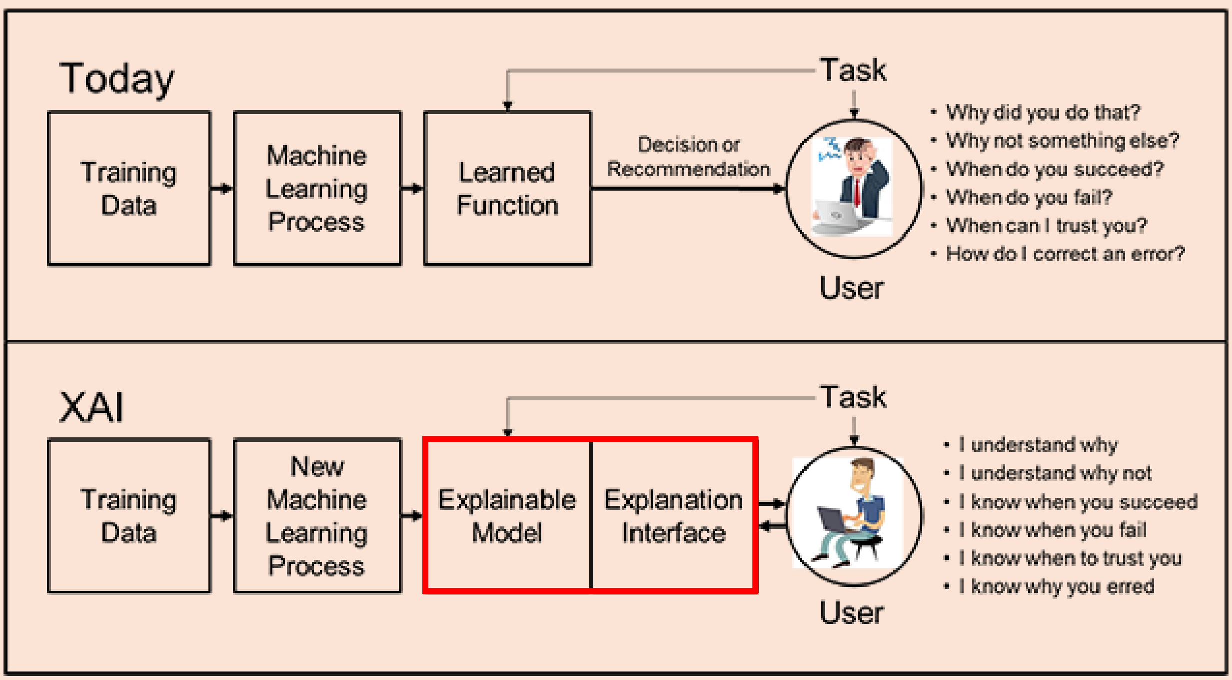


Figure 17: Normal AI (top) versus XAI (bottom) with key differences in **red**.
Source: www.darpa.mil/program/explainable-artificial-intelligence, public domain

AI in Transportation Design

AI has had a big impact in the transportation sector. Automated vehicles are an incredible development made possible in large part through AI programming. Here are some ways AI is influencing transportation design.

Intelligent Transportation Systems

Intelligent Transportation Systems (ITS) leverage AI to enhance traffic management, safety, and efficiency. Vast amounts of data are collected and analyzed through sensors and cameras to identify congestion, areas of speeding, predict traffic patterns, and optimize traffic flow. ITS can adjust traffic lights and reroute vehicles to minimize congestion. ITS is a proactive approach to traffic management with the following goals:

- Reduce travel time
- Minimize the likelihood of accidents
- Create a safer travel environment
- Provide alerts for emergency response
- Provide traffic insights for design purposes

Automated Highway Systems

Automated Highway Systems (AHS) utilize AI technologies to control vehicle movements based on real-time information on surroundings. Automated vehicles function best when the road is designed with consistent features and there are ways to communicate or “connect”. Connected vehicles can make driving more reliable and efficient. Here are the main ways vehicles can communicate:

- V2V (Vehicle to Vehicle) - see Figure 18
- V2I (Vehicle to Infrastructure)
- V2P (Vehicle to Pedestrian)
- V2G (Vehicle to Grid)
- V2D (Vehicle to Device such as smartphone)

Figure 18: Depiction of vehicle to vehicle (V2V) communication as part of an AHS.
. Source: commons.wikimedia.org/wiki/File:ITS_image_V2V_communication_on_freeways_will_help_to_prevent_crashes.jpg

Smart Highways
Having both ITS and AHS allows for an overall “smart highway”. Smart highways have interconnections between several items including the following:

- traffic lights
- road signs
- vehicles
- pedestrians
- bicycles
- buses
- bus stops
- monitoring stations
- rescue services,
- patrols
- construction traffic control items
- parking meters
- parking lot controls
- weather instruments

All these connections and communications allow for dynamic, real-time responses to changing traffic and weather conditions. Traffic signal times can be adjusted based on current traffic flow. Drivers and autonomous vehicles can receive instant updates about road conditions ahead. Accidents can be addressed, and traffic diverted more quickly.

Road Design

Roadway design can involve a lot of math and trigonometry. Before computer aided drafting (CAD) and transit surveying, engineers were trusted to solve complex equations such as horizontal curves and vertical curves. Here are examples of popular CAD software that has helped create numerous road designs:

- 1982 - AutoCAD
- 1990 - Bentley InRoads
- 2005 - AutoCAD Civil 3D
- 2012 - Bentley OpenRoads

Engineers can use CAD software and digital tools to analyze and visualize alternative layouts, identify conflicts, coordinate with other disciplines, calculate cut and fill, calculate material quantities for cost estimating, simulate traffic flow, and generally advance a design more quickly. In the last few years, CAD software and traffic software have had aspects of machine learning added, which is now further empowering engineers. It is anticipated that Civil 3D, OpenRoads, and other civil software will soon have AI toolkits released.

For example, Softree RoadEng Optimal has an AI toolbar that can help make design decisions and make adjustments to a previous design to achieve project objectives. The software can quickly optimize routes between points, keep grades within maximum and minimum ranges, minimize cut and fill quantities, avoid no-go zones, and adjust for crossings.

<u>Traffic Simulation</u>

Traffic Simulation Software with AI includes Trafficware by Cubic and PTV Vissim. These software packages use AI to test traffic flow scenarios and alternative arrangements for infrastructure projects. This enables engineers to visualize traffic issues and make informed decisions to optimize traffic management. A large scale example is shown in Figure 19.

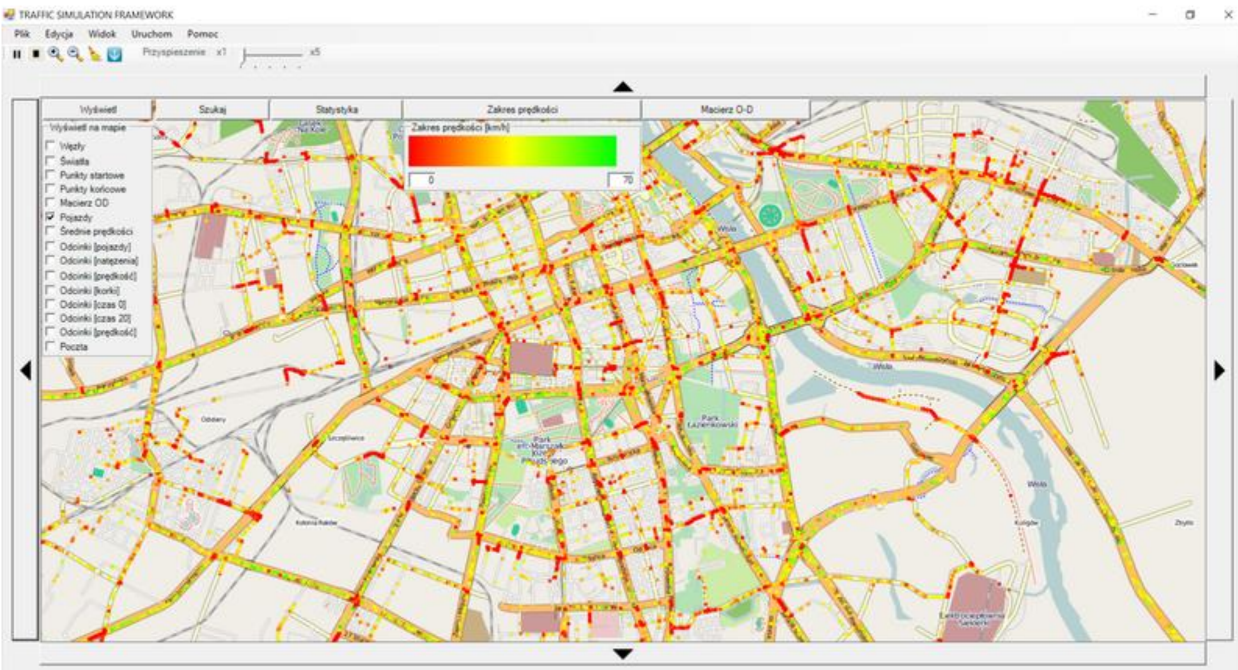


Figure 19: Software for simulating traffic in Warsaw.

AI in Structural Design

Structural engineers often utilize software for design analysis and calculations. Examples include:

- ANSYS
- Enercalc
- ETABS
- RAM
- RISA-3D
- RFEM
- SAFE
- SAP2000
- STAAD

Most of these popular structural software packages already have aspects of machine learning and AI. Likely these programs will continue to incorporate more and more AI features resulting in more design decisions being driven by AI.

AI structural design software can perform a variety of tasks, including the following:

- AI can optimize a structural design based on desired objectives such as minimizing cost, weight, or footprint.
- AI can also create or modify a design based on project limitations such as only using A572 steel material or only using certain beam shapes.
- AI can identify problems/failures in the design and modify the design accordingly. For example, if a member is found to have excess stress near a connection, the program can evaluate options such as softening the corners, adding stiffener plates, changing connection type, or increasing the member size. The chosen approach is then incorporated into the model.
- Some programs have feedback or reinforced learning in which the user can identify areas where the program did not make a desirable design decision. Henceforth, the program will attempt to avoid making the same poor decision. This feature has been called Human-Informed Topology Optimization. An open-source (free) version for 2D-design is located here (3D under development):
    - https://github.com/datquocha/CarstensenGroup_Public

Design Optimization

Structural design optimization involves the following three approaches:

1) Topology, with decisions on the number and connectivity of members,
2) Shape, with decisions on the location of elements and the layout of joints; and
3) Sizing, with defining member cross sections.

AI programs can tackle various design optimization approaches, with examples shown in Figure 20.

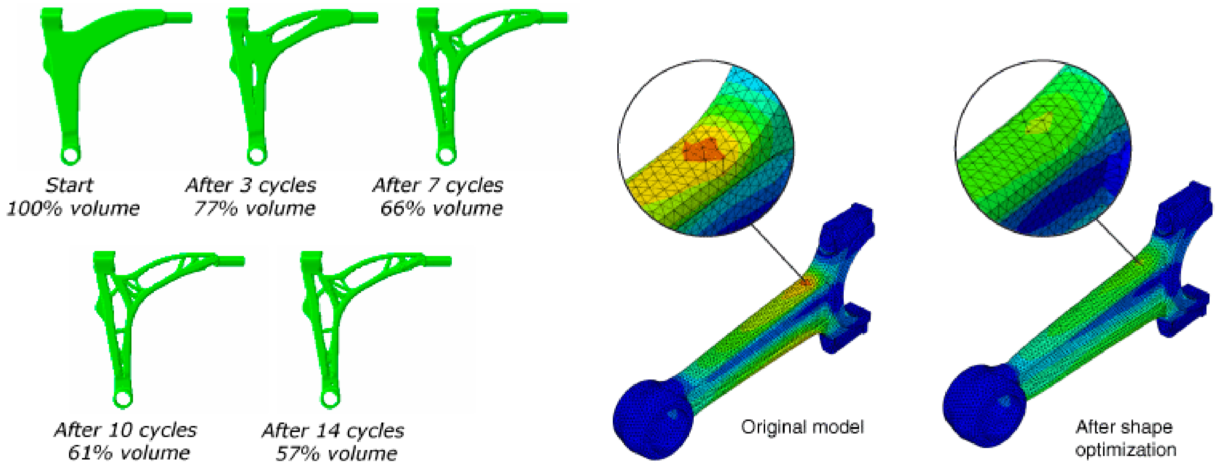


Figure 20: Examples of structural design optimization approaches.
Left) Topology optimization of an automotive control arm using a Python script.
Right) Shape optimization of a finite element analysis (FEA) model.
Sources: docs.software.vt.edu/abaqusv2022/English/SIMACAEEXARefMap/simaexa-c-controlarm.htm
docs.software.vt.edu/abaqusv2022/English/SIMACAEANLRefMap/simaanl-c-optover.htm

The following are example applications where AI software is currently being utilized.

<u>Automotive</u>
AI can design and optimize the shape of vehicles to maximize aerodynamic efficiency and reduce drag forces. AI is used to create digital twins and run model simulations for dynamic real-world assessments of engines and other complex systems.

<u>Aerospace</u>
AI is used in a variety of aerospace applications to reduce weight while still meeting structural strength and deflection requirements. Similar to automotive applications, AI is used with digital twins and model simulations, or to optimize a complex design.

Buildings, Bridges, and Other Structures

AI software for the design of structures can perform a variety of tasks, including:

- Design from the ground up based on initial directions such as building purpose, number of occupants, location, space available, height restrictions, budget, etc. This is a very advanced form of AI programming and is currently limited to specific standard structure types.
- Identify horizontal and vertical irregularities and evaluate modification options to remove them. Such structural irregularities often exist due to architectural and aesthetic objectives, in which case human input is required.
- STAAD Pro (Structural Analysis and Design Professional) utilizes AI for complex load analysis, including running various seismic and wind load combinations and analyzing the results for each structural member. The program learns with each iteration of calculations so that the next iteration will be more efficient.

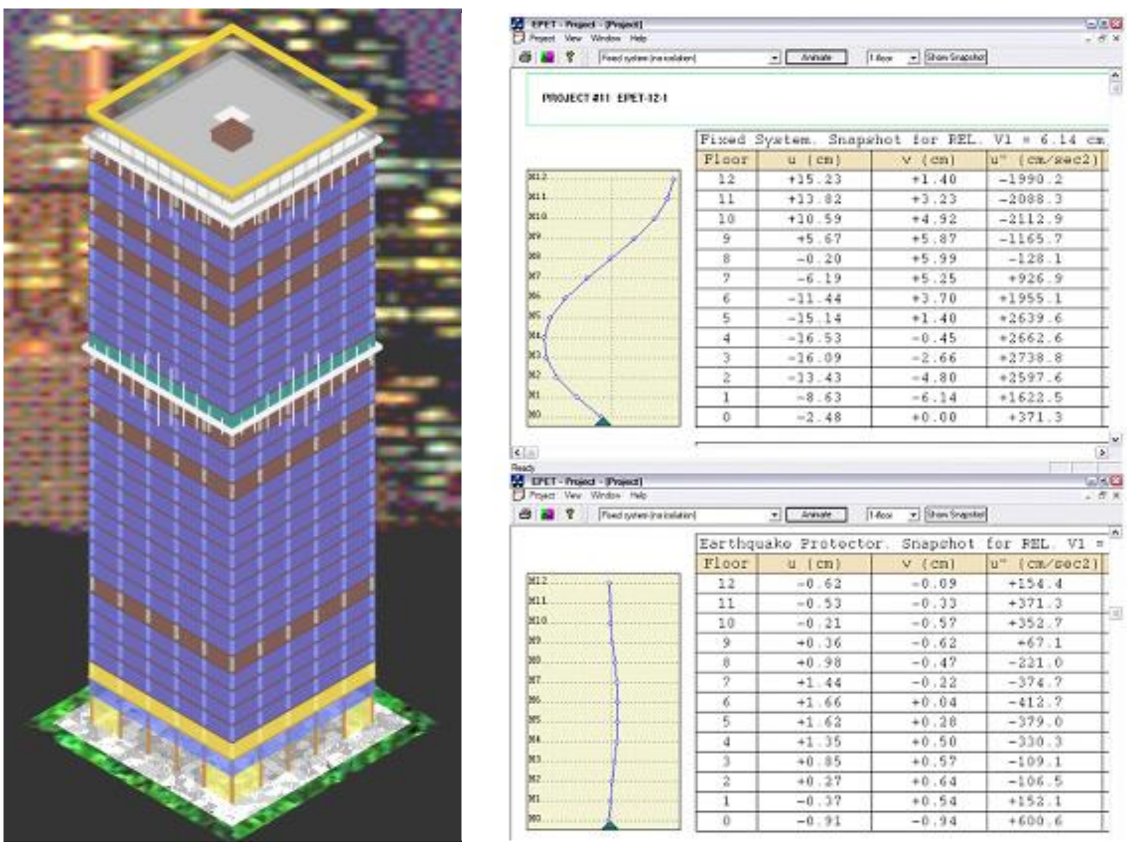


Figure 21: 3D model and seismic computation results in which AI could be helpful.

Sources: commons.wikimedia.org/wiki/File:Multi-Frequency_Quieting_Building_System.jpg
commons.wikimedia.org/wiki/File:EPET_print_out.jpg, Shustov, CC-BY-SA-3.0

AI in Mechanical Design

Mechanical engineers utilize specialized software for a variety of calculations. The software is often very specific to the application. Two common applications are below.

CFD Modeling

Computational fluid dynamics (CFD) modeling helps civil, process, mechanical, and biomedical engineers simulate designs involving moving liquids and gases. CFD can solve for hydraulic grade, energy grade, headloss, mixing rate, surface pressure, drag coefficient, flow rate, chemical concentration, velocity, airfoil efficiency, and temperature change. Example software includes the following:

- Ansys Fluent
- COMSOL Multiphysics
- OpenFOAM
- PowerFLOW
- SimScale
- Solidworks
- STAR-CCM
- XFlow

AI algorithms allow new approaches to CFD analysis. AI allows engineers to simulate more iterations in shorter times and view the results very quickly. If repeating similar models, AI can be trained with previous models to more quickly build a new model, solve with less iterations, tweak the design, and show only the relevant results.

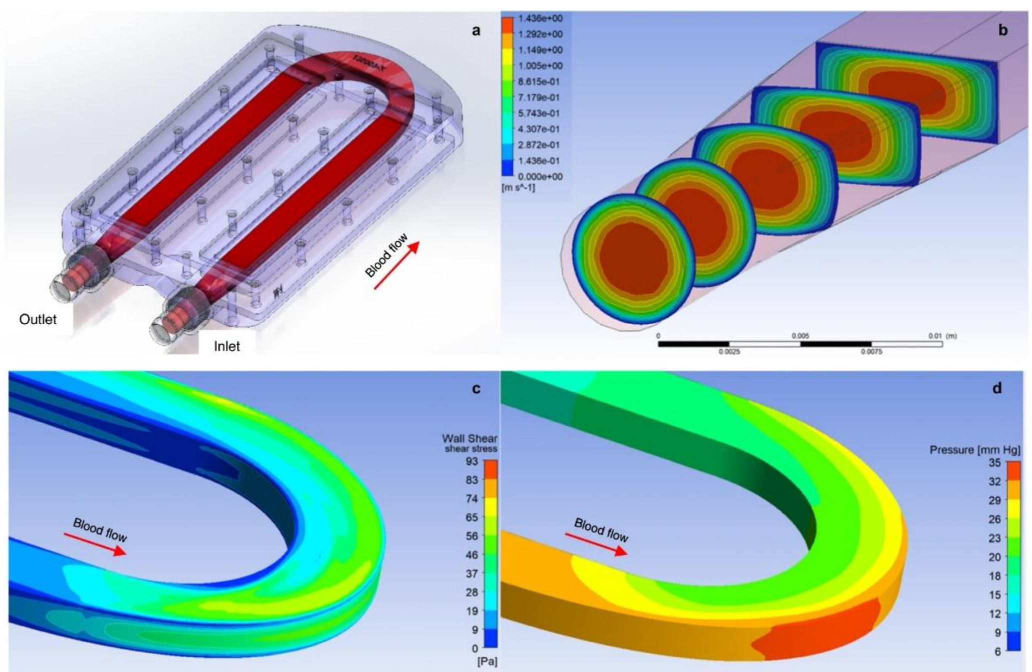


Figure 22: CFD model results for an implantable bioreactor for renal cell therapy.

Source: commons.wikimedia.org/wiki/File:In_silico_optimization_of_the_blood_flow_path_of_bioartificial, Various CC-BY-4.0

<u>Machines and Engines</u>

Software used for machine and engine design includes:

- Ansys CFX
- Aveva
- Catia
- CamQuest
- Blender
- Engine Analyzer Pro
- EngineBuilder Professional
- EngineSim
- Fusion 360
- GT-POWER
- ProCalc
- ProTune
- Simcenter

Some of these mechanical software packages have aspects of machine learning and AI. And some engineers have developed creative ways to use AI to allow more efficient and effective use of software that is currently without AI. No doubt software will incorporate more and more AI features in the near future.

Digital twin models and other forms of simulation have proven helpful to anticipate weaknesses and make design modifications to make engines more robust. For example, turbomachines (turbines and compressors) are complex systems exposed to a wide range of operating conditions. It is now commonplace to create a simulation of the turbomachine and run the model with the full range of anticipated conditions. See Figure 23 for an example model. AI has the potential to handle the complexity and vast amount of information involved in a full engine simulation.

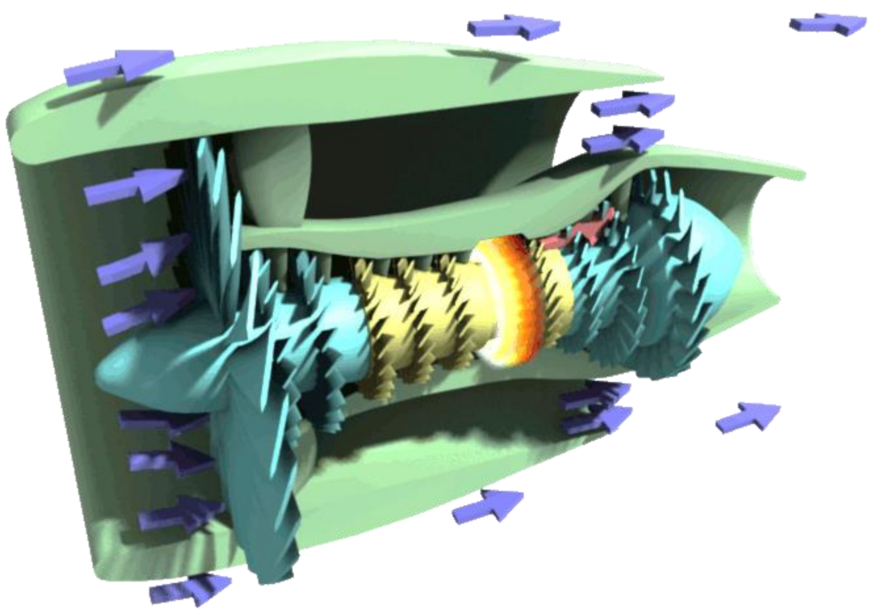


Figure 23: An 3D model of a turbofan engine, made in Blender.

Source: https://en.wikipedia.org/wiki/File:Turbofan3_Unlabelled.gif, Richard Wheeler (Zephyris), CC-BY-SA-3.0

AI in Electrical Design

AI models can simulate complex electrical systems, predict performance, and provide insights and tools that guide the design of more efficient and robust systems.

Software with AI
Software for electrical design with AI include the following:

- Altair HyperWorks employs AI to optimize design and simulate advanced analytics for electrical engineering projects.
- Ansys Electronics Desktop (AEDT) is a platform of electromagnetics simulation solutions such as Ansys HFSS, Ansys Maxwell, Ansys Q3D Extractor, Ansys SIwave, and Ansys Icepak using electrical CAD (ECAD) and mechanical CAD (MCAD) workflows.
- AutoCAD Electrical Toolset and Autodesk AI combine to provide powerful features for creating electrical drawings, electrical schematics, wiring diagrams, and many other forms of electrical design. Advanced programming assists with sizing of conductors, conduits, breakers, grounding checks, creating schedules, and load lists.
- MATLAB Simscape Electrical helps develop control systems and test system-level performance. MATLAB Simulink allows computations and validation for motor drives, power converters and Battery Management Systems (BMS). See Figure 24 for an example.
- Siemens NX integrates AI to streamline electrical design and simulation. The software provides for a transition from conceptual design to detailed analysis, with a focus on complex electrical projects.
- Valispace facilitates collaborative systems and requirements engineering for electrical hardware and software with AI-assisted tools.

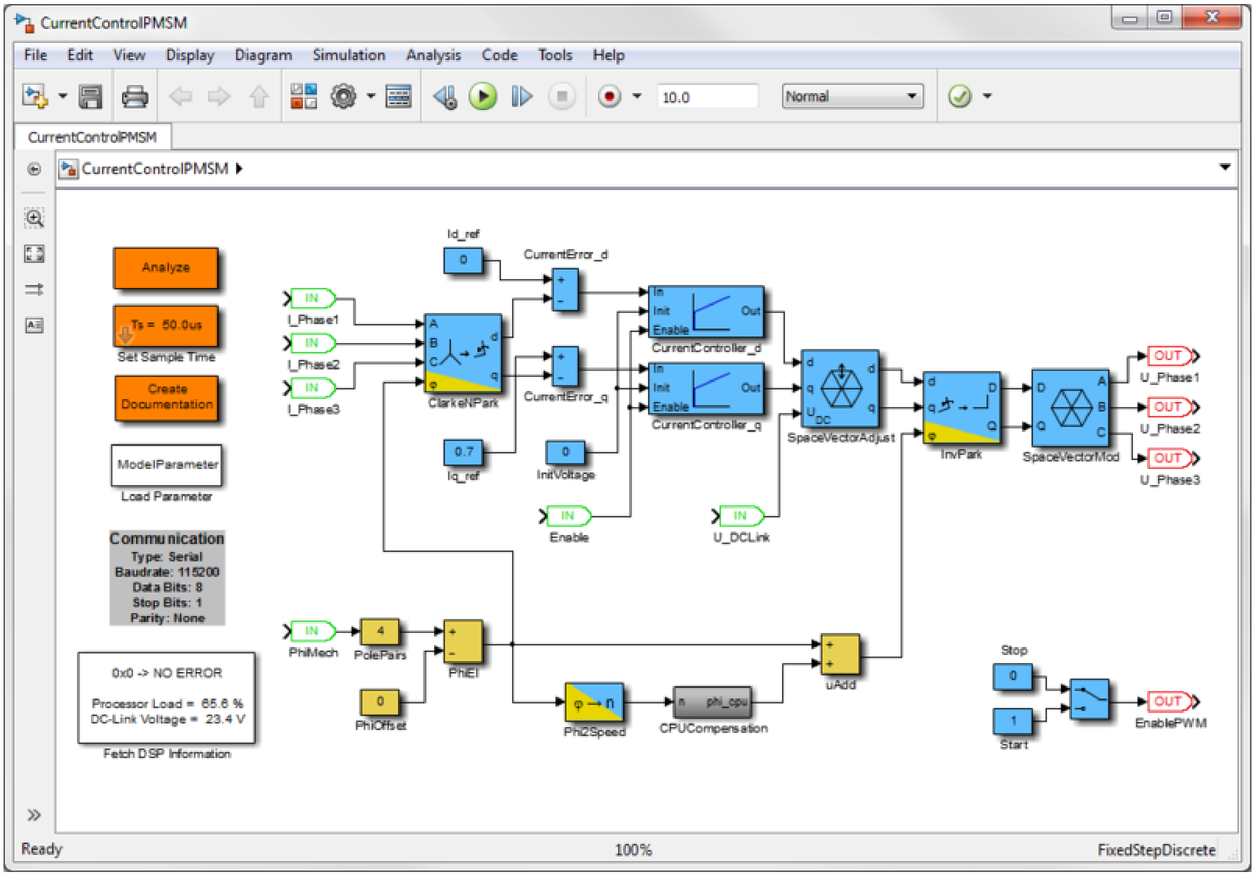


Figure 24: Design of a permanent-magnet synchronous motor (PMSM) block in Matlab Simulink. A series of such blocks can make up a control structure. AI can efficiently design these blocks based on input requirements.
Source: commons.wikimedia.org/wiki/File:CurrentControllerPMSM_Simulink.png, Bbergmair, CC-BY-SA-4.0

Circuit Design

AI to automatically generate circuit schematics and layout designs, and how to use simulation tools to test and evaluate circuit performance. Some example software is as follows:

- Cadence has created several controls and circuit board programs including a large language model (LLM) chatbot that can drive chip design and software for printed circuit board (PCB) design.
- Siemens Solido Design Environment is an AI-powered, cloud-based integrated circuit (IC) design and verification software. Solido can optimize the chip's power, performance, layout, and yield. view waveforms, run simulations, and perform regression analysis on analog and digital custom IC designs.
- SnapMagic Copilot uses generative AI with a digital query interface, much like ChatGPT. SnapMagic Copilot can optimize the bill of materials (BOM) and determine current availability and costs for components.
- Synopsys.ai Copilot utilizes generative AI to enhance productivity for semiconductor and electronics design. The software collaborates with Az Azure OpenAI Service, bringing the power of generative AI into the design process. The knowledge query system pulls from Synopsys resources such as application notes, product user manuals, videos, and drawings.

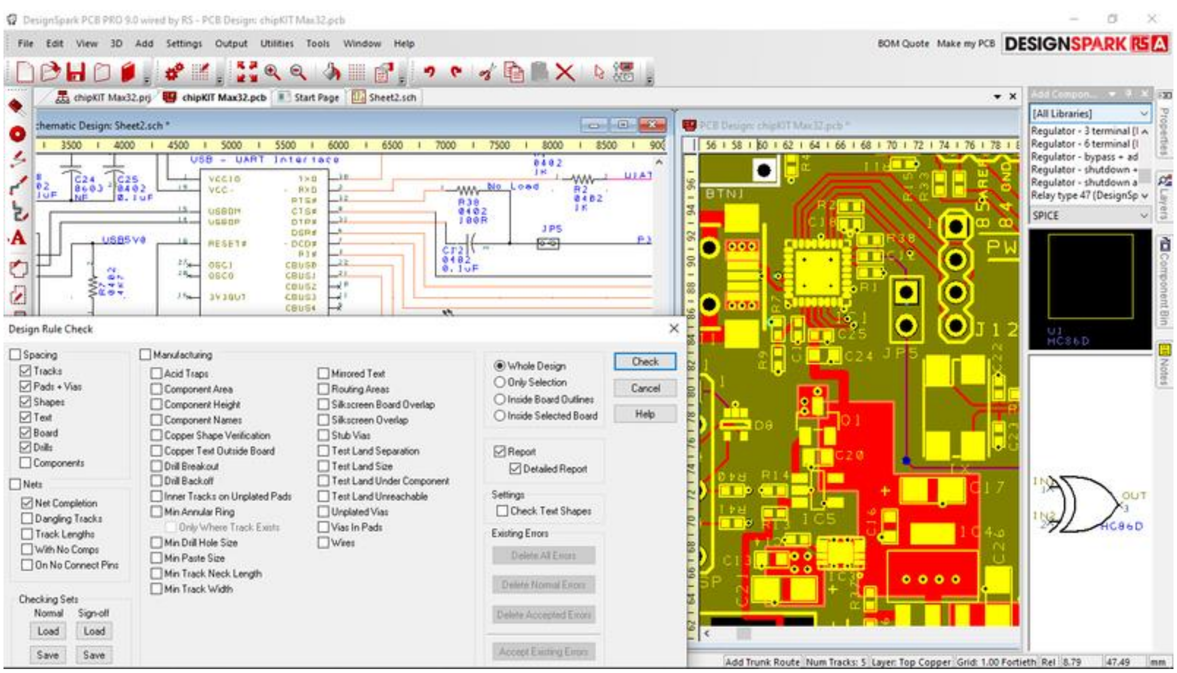


Figure 25: Example software for printed circuit board design.

Source: commons.wikimedia.org/wiki/File:DSPCB_Pro_9.0_screenshot.png, Rds10, CC-BY-SA-4.0

Predictive Maintenance
AI driven software can continuously analyze electrical installations to predict potential malfunctions before they occur. Predictive maintenance (PdM) is becoming common in a variety of electrical systems and is now being specified and designed by electrical engineers. PdM is a proactive approach that avoids surprising failures, extends the lifespan of equipment, minimizes downtime, optimizes performance, and reduces resources.

Many electrical components come with PdM software. For example, Omron Automation makes the following with built-in AI:

- K6CM Motor Condition Monitor
- K6PM Thermal Condition Monitor
- S8VK-X Power Supply

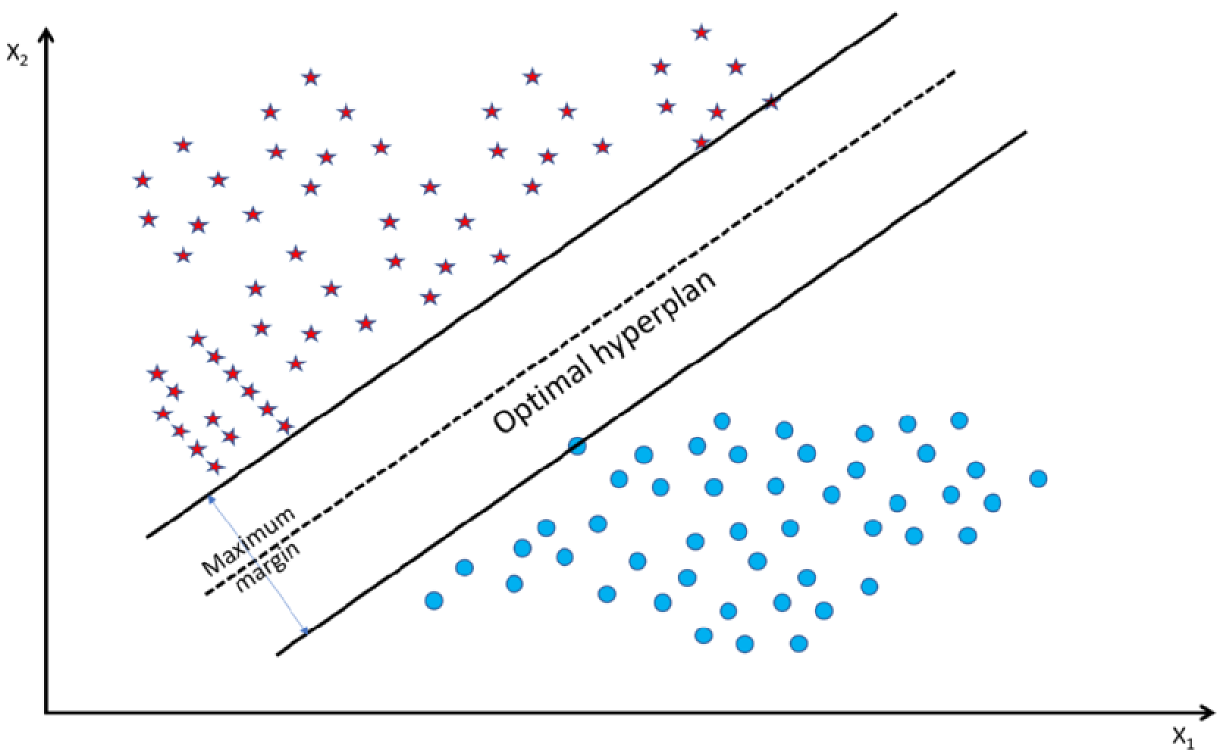
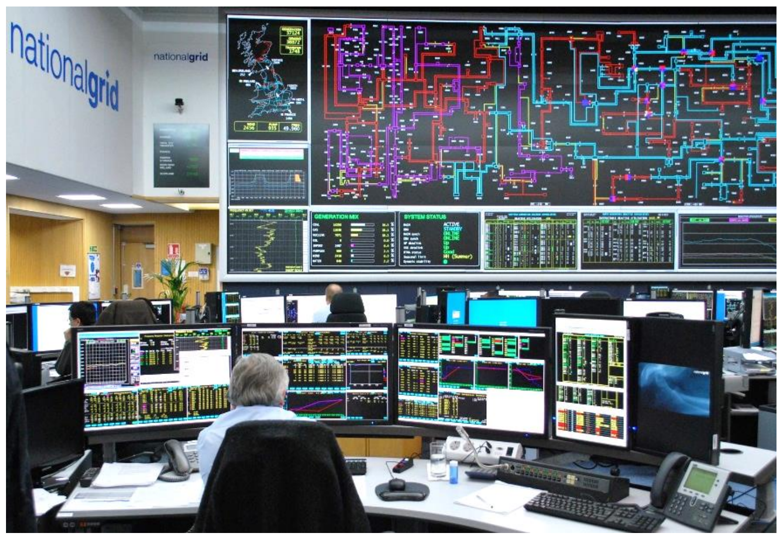


Figure 26: PdM approach called Support Vector Machine (SVM) which can detect outliers (potential failures) by using data to define an “optimal hyperplane” which distinguishes normal conditions (stars) and abnormal conditions (circles).
Source: www.researchgate.net/figure/Support-Vector-Machine-SVM-based-classification_fig3_362604337

Smart Grid Management

AI is being used to oversee and optimize electricity distribution. Remote terminal units are installed on various electrical lines in the distribution system to collect real-time data from the electrical grid. With smart grid management, AI software with predictive analytics and machine learning combine to achieve the following objectives:

- Forecast demand
- Adapt to supply changes
- Prevent outages
- Resource reduction



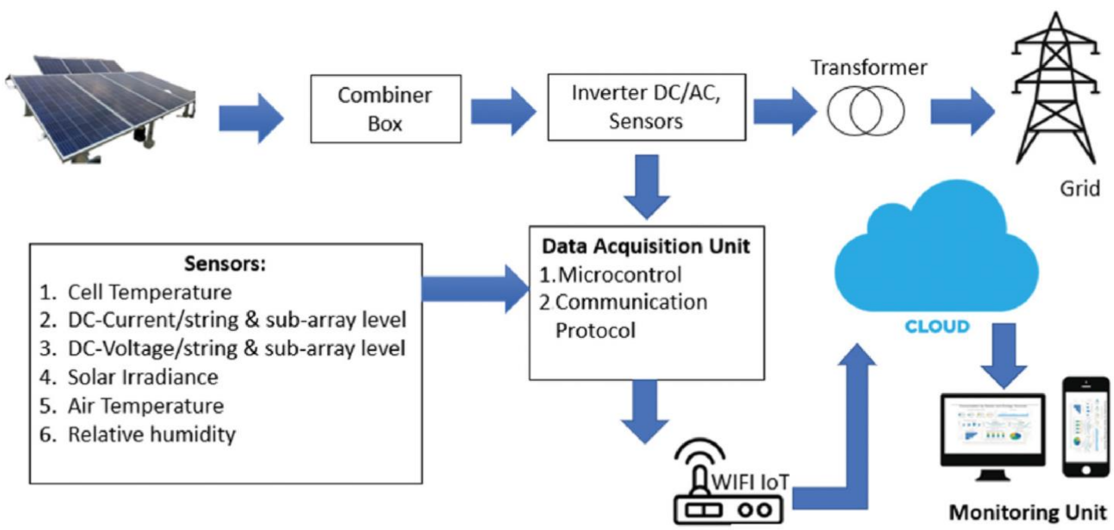


Figure 27: Top) National Grid Control Centre in England.
Bottom) Smart grid arrangement for a solar panel feeding a grid.

. Source: www.flickr.com/photos/deccgovuk/8725424647, CC-BY-SA-4.0
Source: commons.wikimedia.org/wiki/File:Graphical_User_Interface_of_the_Traffic_Simulation_Framework.png, Pgora

Responsible Charge

This course covers major AI design software that is currently assisting engineers. The trend to utilize AI is likely to continue, and at some point, one may ask: Is there a limit to how much design control can be given to AI? And is a human engineer really needed? The answer is: Yes, and Yes! There must be a licensed engineer (only humans can be licensed) with "responsible charge" as explained in this section.

EOR

Each engineering work is to be designed under the responsible charge of one **human** professional engineer (P.E.) who is certified/licensed in the state where the work is to be installed. After signing and sealing engineering documents, that professional engineer becomes the engineer of record (EOR) for the associated design work. The term EOR is common in the building permitting and construction phases of capital improvement projects and may not apply to engineering studies or other projects that don't involve construction. The EOR often needs to confirm the installation conforms to the engineering plans and specifications and sign and seal record drawings. EOR responsibility and liability for the design continues after the installation is placed into occupancy or service.

Other engineers or designers can work under the direction of the EOR. However, the EOR must remain in "responsible charge".

Note that there can be multiple EORs on a single project, each responsible for different portions of the work. Often each EOR is responsible for the work in their particular discipline (civil, structural, plumbing, electrical, etc.). However, it is possible for there to be more than one EOR within the same discipline. For example, if there are multiple buildings being designed on a property, there may be a different structural EOR for each building. Each structural EOR takes responsibility for their assigned building.

Humans Only

No state allows an AI robot or AI software to be a P.E. since only humans can be licensed. Therefore, P.E.'s needs to take full responsibility for their design work, regardless of the software utilized to help produce the design. Although engineering software has been used for decades, AI enhanced software can make design decisions that historically only people made.

Definition of Responsible Charge

The most commonly accepted definition of "responsible charge" is from the National Council of Examiners for Engineering and Surveying (NCEES) Model Law document, which includes the following excerpts:

- The term "**Responsible Charge**," as used in this Act, shall mean direct control and personal supervision of engineering work
- Licensed engineers shall be in **responsible charge** of all engineering design of buildings, structures, products, machines, processes, and systems that can affect the health, safety, and welfare of the public
- Any jurisdiction, county, or local government agencies or authorities, or officials or employees thereof, shall not engage in the practice of engineering or surveying involving either public or private property without the project being under the **responsible charge** of a professional engineer for engineering projects

A National Society of Professional Engineers (NSPE) Position Statement also defines "Responsible Charge" as the direct control and personal supervision of engineering work. However, it also goes into more detail:

- The professional engineer in **Responsible Charge** is actively engaged in the engineering process, from conception to completion.
- Engineering decisions must be personally made by the professional engineer or by others over which the professional engineer provides supervisory direction and control authority.
- Reviewing drawings or documents after preparation without involvement in the design and development process does not satisfy the definition of **Responsible Charge**.

Based on these definitions for responsible charge, it is clear that there are limits to how much design control should be given to AI. This is especially true when the engineering work has impact on the health, safety, and welfare of the public.

Some states and local jurisdictions have additional requirements for "responsible charge" and, thus, licensees should review governing laws and regulations based on the project location and regulatory agencies involved.

Direct Control

"Direct Control" implies the ability to give direction and have the "final say" in engineering decisions. This includes the right to say "no" to using AI.

Ways to maintain direct control when using AI:

- Make design decisions and adjust the design accordingly, either within the AI software or after exporting the AI results, but prior to producing final drawings, specifications, and calculations.
- Do not sign and seal drawings from a program that produces drawings in a single step. There should be multiple design steps, or stage gates, so an engineer can review design progress, make decisions, give directions, or manually modify the design prior to final drawings. Use XAI software when possible.
- Ensure the AI program follows directions given. The engineer's direction should not be optional.
- Document directions given and design decisions made.

Personal Supervision

"Personal Supervision" means that the P.E. observes the design work, is actively engaged throughout the design, reviews any work done by others, and guides the work of others.

Ways to maintain personal supervision when using AI:

- Understand how the AI program works such as by reading the software user manual, completing the tutorial, or being trained in its use.
- Consider the AI program reputation, history, and experience related to the engineering tasks. Can it be trusted for this application?
- Review the inputs and outputs of the AI program. If the program was run multiple times, review the outputs of each major iteration. Use XAI when possible.
- Review calculations performed by AI. If the AI program involved many iterations, at least review the calculations for the final design result. In cases such as finite element analysis (FEA) or computational fluid dynamics (CFD), review of the calculations at each node may not be practical as there are often millions of nodes in a model. However, key parameters and results should be reviewed in depth such that the engineer is satisfied with the accuracy and confident in the overall design.
- Save the inputs, outputs, and calculations in an accessible format such as pdf.

Example Problem 1

Engineer Ivy is EOR for an HVAC design. Some design work was done by Toby, an engineer-in-training, under her supervision. Ivy kept the following log of activities she did during the design process. Which are primarily considered direct control versus personal supervision?

1. Reviewed applicable codes and permit requirements
2. Attended a design progress meeting
3. Sketched concept layout and explained it to Toby
4. Decided on an AI software program for air fan sizing and energy calculations
5. Reviewed program results and calculations
6. Gave red-line markups to Toby
7. Attended a design workshop
8. Discussed alternatives with the client
9. Client asked if the air fan can be eliminated; I told them "no"
10. Told Toby to add an automatic dampener to the design
11. Determined an additional section view is needed
12. Reviewed final plans and specifications

Solution:

Although some activities fall into both categories, the following colors show the **primary** category based on **direct control in red** and **personal supervision in green**.

1. Reviewed applicable codes and permit requirements
2. Attended a design progress meeting
3. Sketched concept layout and explained it to Toby
4. Decided on an AI software program for air fan sizing and energy calculations
5. Reviewed program results and calculations
6. Gave red-line markups to Toby
7. Attended a design workshop
8. Discussed alternatives with the client
9. Client asked if the air fan can be eliminated; I told them "no"
10. Told Toby to add an automatic dampener to the design
11. Determined an additional section view is needed
12. Reviewed final plans and specifications

Example Problem 2

Engineer Shay is to be the EOR for a new roadway and roundabout. She is busy and no one has time to help her. Shay asks ChatGPT to produce a road design and inputs the intersection location and required distance to the center of the proposed roundabout. The program responds with a set of drawings in pdf format. Why shouldn't Shay sign and seal these drawings?

Solution:

Shay did not have responsible charge because she did not have personal supervision of the design. The program made all the design decisions.

Shay did not have personal supervision because she was not involved in the design process, is unaware of which design decisions were made, and did not review the design calculations and drawings.

Helpful References

Ahmad, Afaq; et. al. (2023) “Artificial Intelligence and Machine Learning Techniques for Civil Engineering”, IGI Global. ISBN:9781668456446

Felzmann, H.; *et al.* (2020) “Towards Transparency by Design for Artificial Intelligence” Sci Eng Ethics 26, 3333–3361. https://doi.org/10.1007/s11948-020-00276-4

Kassabian, Paul; Lim, John (2024) “Seven Questions on Artificial Intelligence and its use by AEC Design Professionals”. https://www.sgh.com/insight/artificial-intelligence-and-its-use-by-aec-design-professionals/

Málaga-Chuquitaype, C (2022) Machine Learning in Structural Design: An Opinionated Review. Front. Built Environ. 8:815717. https://doi.org/10.3389/fbuil.2022.815717

National Council of Examiners for Engineering and Surveying (NCEES) (2021) “Model Law”. https://ncees.org/wp-content/uploads/Model_Law_2021_web-2.pdf

National Society of Professional Engineers (NSPE) “NSPE Position Statement No. 10-1778 - Responsible Charge”. https://www.nspe.org/resources/issues-and-advocacy/professional-policies-and-position-statements/responsible-charge

Neural Concept (2023) “How Is AI Used in Mechanical Engineering?” https://www.neuralconcept.com/post/how-is-ai-used-in-mechanical-engineering

O'Beirne, Kevin (2024) “Responsible Charge an Essential Concept for Design Professionals”. https://www.csiresources.org/blogs/kevin-obeirne-pe-fcsi-ccs-ccca-cdt1/2024/01/23/responsible-charge-an-essential-concept-for-design

Yüksel, Nurullah; *et al.* (2023) “Review of Artificial Intelligence Applications in Engineering Design Perspective”. Engineering Applications of Artificial Intelligence, Vol. 118, 105697, ISSN 0952-1976. https://doi.org/10.1016/j.engappai.2022.105697

