



*An Online PDH Course
brought to you by
CEDengineering.com*

Generative AI vs. Agentic AI in Engineering

Course No: W02-001

Credit: 2 PDH

Brian Lisiewski, P.E.



Continuing Education and Development, Inc.

P: (877) 322-5800

info@cedengineering.com

Table of Contents

Introduction and Course Overview: 1

What is Generative AI?..... 2

Applications of Generative AI Across Engineering Disciplines 4

What is Agentic AI?..... 7

 Applications of Agentic AI Across Engineering Disciplines 9

Comparing Generative AI and Agentic AI 13

Ethical and Professional Considerations in Using AI (Generative & Agentic) in Engineering ... 17

Conclusion 21

Generative AI vs. Agentic AI in Engineering

Introduction and Course Overview:

Artificial Intelligence (AI) is transforming engineering practice across all disciplines. Two emerging paradigms of AI with significant impact are **Generative AI** and **Agentic AI**. Essentially, Generative AI produces content—including designs, code, text, and images—by leveraging patterns identified within data, whereas Agentic AI operates autonomously to pursue specified objectives. This course will explain these concepts in clear terms, highlight how each is deployed in various engineering fields (civil, mechanical, electrical, software, etc.), and discuss ethical considerations for their use. By the end, engineers will understand the differences between generative and agentic AI, recognize examples of each in their field, and be aware of professional responsibilities when integrating AI into engineering projects.

Generative AI – *The Creative Engine*

Role: Produces new content (designs, text, images, code) in response to prompts or data. It's like a digital creator that *generates* outputs but doesn't act on its own beyond that creation.

Agentic AI – *The Autonomous Agent*

Role: Takes action to meet goals. Given an objective, it *plans and executes* multi-step tasks with minimal human intervention, making decisions and adapting along the way. Think of it as an independent digital assistant that not only generates content but also acts on it.

Figure 1: Key Roles of the Generative AI and Agentic AI

Key distinction: Generative AI focuses on **content creation** upon request, while Agentic AI centers on **autonomous decision-making and execution**. For instance, generative AI may design a component or draft an email when prompted, whereas agentic AI can not only design the component but also initiate testing procedures, compile analytical reports, and modify parameters automatically to achieve predetermined performance objectives—all without requiring step-by-step guidance.

What is Generative AI?

Generative AI uses algorithms, often built on deep learning models like neural networks, to learn from existing data and create new, original content. This can include text, images, designs, music, computer code, or other outputs that mimic human creativity. Unlike traditional programs that operate with fixed instructions, generative AI analyzes patterns in its training data and produces unique combinations based on what it has learned. However, generative AI only acts when given a prompt or input—it doesn't start tasks on its own or make changes without being asked.

How Generative AI Works (Briefly): Most generative AI systems use advanced machine learning techniques. A common approach is through large models trained on huge datasets: for example, a language model trained on millions of pages of text can generate new text, or a model trained on mechanical part designs can propose new part configurations. The model essentially tries to predict the next element (word, pixel, etc.) in a sequence, given what it has seen so far. By stringing together many such predictions, it produces a complete output. For instance, if asked to “*generate a blueprint for a pedestrian bridge with certain span and load characteristics,*” a generative AI could produce a conceptual design drawing or parameters that meet those criteria (drawing from knowledge of many bridge designs it was trained on).

The generative process can be guided by constraints. In engineering, one might constrain the output to meet specific requirements (e.g., maximum stress or weight limits). Generative AI can rapidly **explore design spaces**, offering numerous possible solutions for an engineer to evaluate. However, it's crucial to note the AI doesn't *know* if a design is optimal or safe beyond the patterns it has learned – human expertise is needed to verify and refine its outputs.

Visualization – Generative AI Process:

The following diagram illustrates a simple generative AI workflow, from user input to AI model to output:

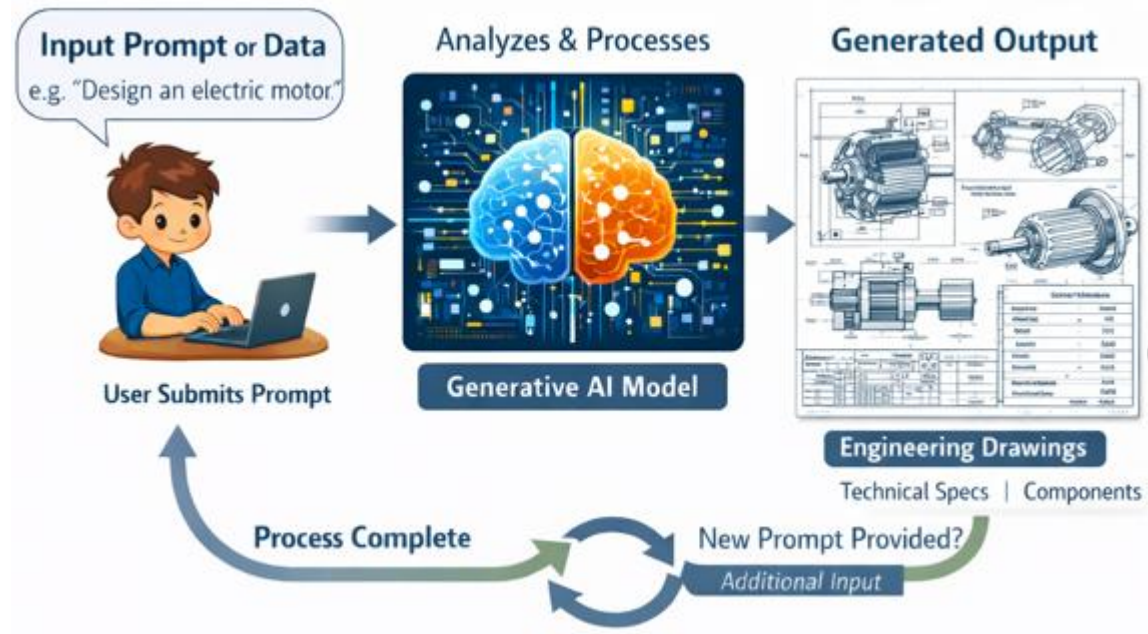


Figure 2: A Simple Generative AI Workflow

In summary, **Generative AI acts as a creative tool**. In an engineering context, think of it as a supercharged brainstorming assistant or design generator that can produce draft solutions or content which you can then evaluate and refine. It can speed up early-stage work (like concept design or documentation) by offering a plethora of options.

Applications of Generative AI Across Engineering Disciplines

Generative AI is already being applied (or experimented with) in various engineering fields to enhance creativity, automate drafting tasks, and optimize designs:

- **Civil Engineering:** Generative AI contributes to *generative design* in civil projects. For example, AI tools can propose multiple architectural layouts for a building or optimal shapes for a bridge truss, all within specified constraints like footprint, materials, and load conditions. A civil engineer might input design goals (e.g. a building's required capacity, site boundaries, and lighting needs), and the AI generates design alternatives that satisfy those parameters, offering innovative forms an engineer might not think of alone. Additionally, generative AI can assist in urban planning by generating city block arrangements, road network designs, or flood control layouts given certain criteria. In construction, language models can generate project plans or safety checklists from past data. While these AI-generated outputs can be very creative and efficient, engineers must verify that any design meets building codes and safety standards – the AI's suggestions jump-start the process, but professional judgment is needed to finalize them.
- **Mechanical Engineering:** In the fields of mechanical and aerospace engineering, generative AI is transforming component and product design. Engineers employ generative design software to automatically generate intricate geometries for parts—such as aircraft brackets, engine components, and consumer product casings—that optimize weight and material efficiency while preserving structural integrity. For instance, given the functional requirements (attachment points, forces, allowed materials), a generative AI can output dozens or hundreds of possible geometries optimized for those requirements – often resulting in organic-looking structures that a human might not traditionally draw. This technique is very useful for *lightweighting* and optimizing designs for additive manufacturing (3D printing), where complex geometries are feasible. Aside from physical parts, generative AI can help mechanical engineers by generating simulations or test scenarios (imagining various operating conditions) or even by writing preliminary **CAD scripts/code** to automate part of the design process. The result is a set of design candidates or datasets that engineers can then put through detailed analysis. As always, these suggestions require validation: AI might propose a shape that passes basic criteria but could have unforeseen issues (e.g., a resonance problem). Thus engineers treat generative outputs as proposals to investigate, not final solutions.
- **Electrical and Electronics Engineering:** Electrical engineers are using generative AI for designing circuits and systems. One emerging application is in **PCB (printed circuit board) layout and chip design** – generative algorithms can suggest component placements or routing schemes that minimize interference or signal delay, after learning from thousands of existing layouts. For instance, if designing a complex circuit, an

engineer could specify the circuit's functional blocks and constraints (like required connections, thermal limits), and a generative program might propose an initial board layout or wiring diagram fitting those needs. Similarly, in VLSI (chip design), AI can assist by generating floorplans for integrated circuits, arranging modules on a chip efficiently. Another use is creating *electrical schematics* from high-level descriptions: you describe what the circuit should do, and generative AI drafts a possible schematic. In power engineering, generative AI could be used to model and predict demand patterns or generate strategies for load balancing under different scenarios. It might also help generate synthetic data for testing (e.g., creating many possible fault scenarios in a power grid for training protective systems). Again, human engineers must verify all such AI outputs for correctness and safety – e.g., ensuring a generated circuit meets all reliability and compliance criteria before actual fabrication or deployment.

- **Software Engineering (and Computer Engineering):** This is one domain where generative AI is already widely adopted. Tools like code assistants (e.g., GitHub Copilot or similar AI) can generate code snippets or even entire functions based on a description, effectively acting as a junior programmer that writes boilerplate or suggests solutions. A software engineer might write a comment “// function to calculate stress on beam given load” and the AI can produce a plausible function implementation. Generative AI also helps in creating test cases, user interface layouts (given a description of an app screen, it can generate the HTML/CSS or a prototype), and even in producing documentation or help content from code. Beyond coding, generative AI can produce *textual content* useful for engineers: for instance, generating a first draft of a technical report, summarizing requirements, or writing emails to stakeholders based on bullet points. In systems and computer engineering, generative models can simulate network traffic patterns or generate synthetic sensor data to test systems before real data is available. The key benefit in software is boosting productivity – routine sections of code or documentation can be drafted by AI, allowing engineers to focus on higher-level logic and critical review of the AI's output. However, relying on generative code demands caution: the code might contain bugs or not handle edge cases, so engineers must thoroughly review and test AI-generated code (just as they would code written by a human junior engineer).
- **Other Fields:** Generative AI's impact spans **all engineering disciplines**. In chemical engineering, AI models propose new chemical compound structures or process optimizations (essentially “generating” new formulas or plant designs). In biomedical or materials engineering, generative models suggest material microstructures or biomedical device designs with desired properties. In architectural engineering, AI generates floor plans or facade designs. Across these, the pattern is similar – generative AI accelerates the creative and iterative process by producing candidate solutions, but the engineer guides the process with inputs and constraints, then evaluates and filters the results.

Benefits of Generative AI in Engineering: It can **significantly reduce the time spent on initial design iterations** and broaden the exploration of solutions. Engineers might quickly get 10 viable design ideas where traditionally they might manually draw 2 or 3. It's also helpful for tackling complex, high-dimensional problems (like optimizing many parameters at once) – the AI can churn through combinations faster than a human. Generative AI can incorporate past knowledge; for example, by training on many known good designs, it carries forward best practices embedded in the data. This can inspire innovative approaches (sometimes surprising ones that a conventional approach might miss).

Limitations to remember: Generative AI *does not understand* engineering principles in a human way – it doesn't *reason* about physics or purpose; it merely follows learned patterns. Thus, it might produce outputs that look plausible but are flawed or impractical. Perhaps it suggests a beam layout that technically supports the load in a simulation but is impossible to construct, or it writes code that superficially works but has security holes. Moreover, generative models can inherit biases or errors present in their training data. If an AI was trained on designs that all used a particular material, it might overlook better novel materials. There are also intellectual property concerns (if the training data included proprietary designs, the AI's output might inadvertently resemble them too closely). **Engineers must treat generative AI as an assistant, not an oracle** – its outputs are starting points requiring scrutiny. Good practice is to have rigorous validation steps for any AI-generated design or content, just as you would cross-check a human junior engineer's work. When used wisely, generative AI can enhance creativity and efficiency, but it should be paired with expert oversight.

What is Agentic AI?

Agentic AI describes systems that go beyond simply creating content on demand—they can act independently to accomplish goals or complete tasks. Think of agentic AI as a virtual partner that observes its surroundings or situation, makes choices based on those observations, and takes actions—all geared toward achieving objectives set by either a user or its programming. Importantly, after receiving a general goal, an agentic AI can work with little human input, determining and handling the necessary steps on its own. This autonomy (which gives it the name "agentic") means it is capable of deciding its next moves within the limits of what it's designed to do.

Agentic AI Operational Framework (Decision Loop): Numerous agentic AI systems operate through a structured cycle of **perception, decision-making, action, and learning**:

1. **Perceiving:** The agent collects information about its current context, whether through sensors, APIs, or other data sources. For instance, an AI agent managing a project reviews emails or reports, while a drone uses camera and lidar to survey its environment.
2. **Reasoning/Planning:** The AI uses what it perceives to decide its next steps toward a goal. It may break the goal into sub-tasks, consider strategies, or predict outcomes, often employing planning algorithms or reinforcement learning.
3. **Acting:** The agent carries out actions as determined by its strategic plan. These actions may include physical operations (such as a robot manipulating an object), digital activities (for example, transmitting an email or modifying a design parameter), or utilizing other tools or models (such as engaging a generative AI system to create required content).
4. **Learning/Adapting:** Following each action, the agent receives feedback or observes changes within the environment, subsequently updating its knowledge base or strategy as needed. An intelligent agentic AI continually refines its approach based on outcomes—reinforcing strategies that yield progress toward its objectives, and modifying those that result in unexpected or suboptimal results. This iterative process enhances the agent's effectiveness over time, enabling it to respond to evolving circumstances. While some agentic systems operate according to predetermined rules, advanced models incorporate learning mechanisms to facilitate greater adaptability.

This loop repeats as needed until the goal is achieved or a stopping condition is met. Agentic AI thus operates in a continuous or multi-step fashion, unlike generative AI which typically gives a one-off output and waits for the next prompt.

Visualization – Agentic AI Loop:

The diagram below illustrates a simplified agentic AI interaction between an agent and its environment:

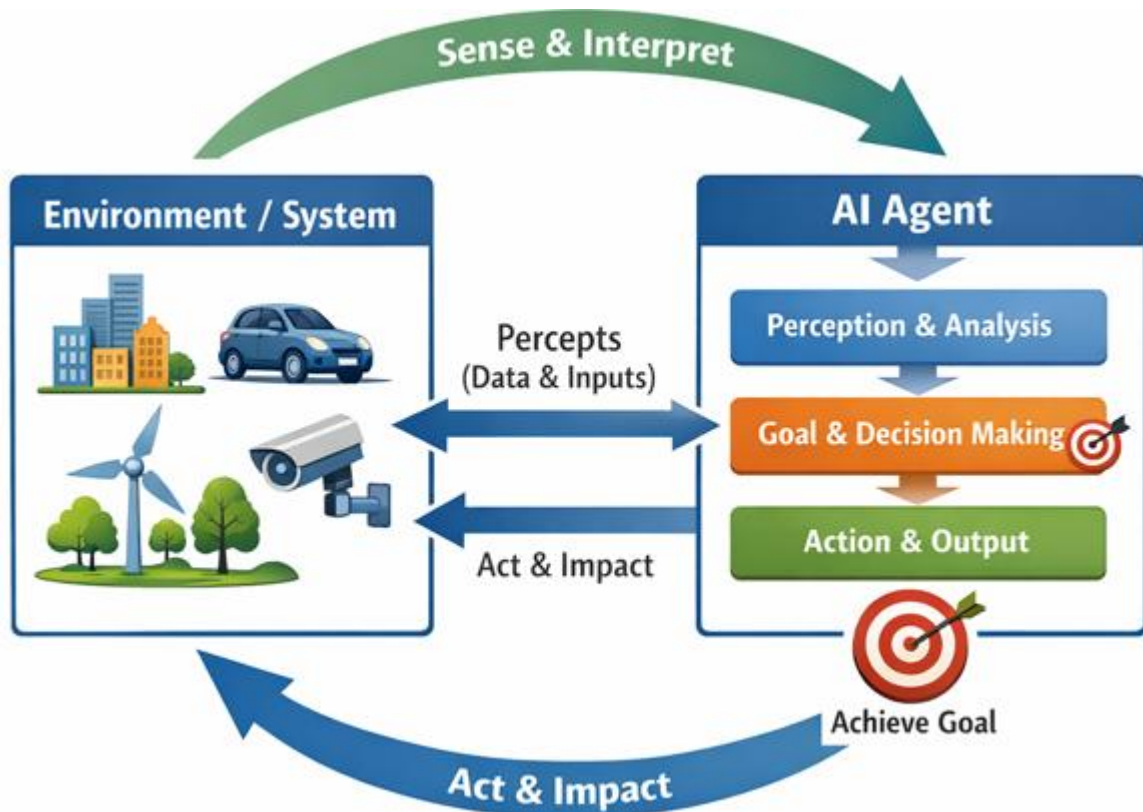


Figure 3: Simplified Agentic AI Interaction Between an Agent and its Environment

A key point is that **agentic AI often uses generative AI or other AI tools internally**. For example, an agentic system might use a generative model to create a piece of content as one step, but then it will proceed to do something with that content as part of a bigger task. Agentic AI is more about the orchestration of tasks and decision-making, potentially calling on various sub-AI capabilities (which could include generative, predictive, etc.) to accomplish its objectives.

In sum, Agentic AI is like a proactive assistant. Rather than waiting for detailed instructions for each step, you can tell it *what* you want to achieve, and it will figure out *how* to get there (within its domain of ability). It's designed to handle complexity and make decisions when faced with choices, much as a human agent might. For instance, if a human manager is tasked with "improve the energy efficiency of this building," they would gather data, evaluate options (upgrade HVAC? install solar panels? adjust operating schedules?), take actions, monitor results, and adjust the plan. An agentic AI aimed at the same goal would attempt to do a similar multi-step process autonomously: auditing energy data, suggesting changes, perhaps directly controlling some IoT devices, then checking if the changes helped.

Applications of Agentic AI Across Engineering Disciplines

Agentic AI is emerging in various engineering fields, especially as systems become more complex and autonomous operation is desired. Here are examples of how agentic AI can be deployed across disciplines:

- **Civil Engineering and Infrastructure:** Civil engineers deal with large-scale systems (cities, transportation networks, utilities) where autonomous decision-making can provide huge benefits. One application is in **intelligent transportation systems** – imagine a city traffic control AI that adjusts traffic light timings on the fly to optimize flow. A traditional system might be pre-programmed on a schedule, but an agentic AI traffic manager could perceive real-time traffic camera feeds and sensor data, detect an accident causing congestion, and autonomously reroute vehicles by changing signal patterns or variable message signs, coordinating with other city systems (like emergency services or public transit schedules) to manage the situation. Similarly, in infrastructure maintenance, agentic AI can be used for *structural health monitoring*: sensors on bridges or buildings feed data to an AI agent that monitors structural integrity. If the agent detects anomalies (say, unusual vibration patterns or stress loads), it can autonomously decide to flag an alert, lower the speed limit on a bridge, or even dispatch a drone to inspect the structure more closely – all without waiting for a human to initiate these steps. In construction project management, an AI agent could manage logistics: tracking supply deliveries, adjusting schedules when there are delays (e.g., autonomously reordering tasks if a shipment is late), and notifying the team of changes. These agentic systems effectively automate what a human controller might do, following pre-set rules but also learning from patterns (e.g., learning which suppliers are often late and adjusting orders proactively). Engineers in civil projects still oversee these AI, but the AI agent can handle the minute-by-minute decisions to keep the project or system optimally running.
- **Mechanical Engineering and Robotics:** Mechanical engineering often intersects with robotics and automation – a ripe area for agentic AI. Modern manufacturing systems use **autonomous robots and cobots (collaborative robots)** on assembly lines. Each robot can be controlled by agentic AI that allows it to adapt to changes. For instance, if one robot on a production line slows down due to a fault, an AI agent controlling upstream machines might autonomously buffer or reroute work to prevent a pile-up, or call a maintenance routine. Agentic AI in robotics enables *autonomous navigation and task execution*: consider warehouse robots that move materials. An agentic AI system can manage a fleet of robots, assigning tasks (like “Robot A, go fetch pallet X from shelf Y”), planning paths to avoid collisions, and adapting when new orders come in or an obstacle appears. In the field, autonomous vehicles (like self-driving cars or drones) use agentic AI to sense their environment (through cameras, lidar) and make driving/flying decisions in real-time to

reach a destination safely. That's a quintessential example of agentic AI in mechanical/electrical systems – the car's AI perceives road conditions, plans its route and maneuvers, acts by steering/braking, and learns continuously from its experiences to handle situations better (e.g., improving how it reacts to construction zones after encountering a few). In product design, you might have agentic AI algorithms that automatically tune machine parameters: say a smart HVAC system (heating, ventilation, air conditioning) that's tasked with minimizing energy use while keeping indoor climate comfortable. The agent monitors temperatures and occupancy (perception), decides on optimal settings (perhaps lowering airflow in unused rooms), takes actions by adjusting fans and dampers, and learns daily patterns to refine its controls. These examples show agentic AI handling interactive, ongoing tasks in the mechanical realm.

- **Electrical Engineering and Systems Control:** Electrical engineering systems, especially power and telecom networks, can benefit from agentic AI for real-time control and optimization. One prominent application is **smart grid management**. Instead of manually controlling power generation and distribution, utilities are deploying AI agents that autonomously balance supply and demand in the electric grid. Picture an agentic AI monitoring a regional power grid: it perceives data like current electricity demand, grid frequency, energy prices, weather forecasts (for renewable generation), and maybe even signals from smart appliances. It has the goal to maintain stability and efficiency. The AI can decide to reroute power flows, engage battery storage, or request increased generation from a power plant if it predicts a shortfall. It can also shed non-critical loads (turn off or reduce certain consumers) temporarily if needed to prevent blackouts. All of this can happen with minimal human intervention, following guidelines set by grid engineers. In telecommunications (another electrical/computer engineering area), agentic AI is used for network management. An AI agent might manage a data network's traffic: when one route gets congested, it autonomously reroutes packets through alternative paths; if a network node fails, it detects the outage and reconfigures the network topology on the fly to maintain service. It may also orchestrate self-healing: diagnosing issues and possibly initiating corrective actions (like cycling power on a malfunctioning device). Another example: **adaptive control systems** in electronics – say, an agentic AI controlling a complex chemical process in an industrial plant (overlap of electrical control and chemical engineering). The AI constantly adjusts valve positions, heater settings, etc., to maintain optimal reaction conditions, learning from past data to improve stability and yield. Such a controller goes beyond a fixed PID control by actually planning ahead and adapting to unexpected changes (like a catalyst degrading or an impurity spike). In summary, agentic AI here works as an autonomous control engineer, making sure systems run smoothly, efficiently, and safely within set bounds.

- **Software Engineering and Project Coordination:** In software and IT, agentic AI can take on the role of an **intelligent assistant or DevOps agent**. For example, consider the scenario of software deployment and monitoring. An agentic AI could be given the goal: “Keep this web service running and performant.” It could monitor server metrics (CPU, memory, response times), automatically scale resources up or down, deploy updates when they become available during off-peak hours, and even run tests after deployment to ensure everything works – all actions that a site reliability engineer might normally script or perform. If an incident occurs (like a server going down), the AI agent can detect it and reroute traffic to backup servers, send alert notifications, or attempt a fix (restart a service) on its own, following a learned playbook. In project management, we now see AI bots that can coordinate some tasks: for instance, analyzing a project’s progress and sending reminders to team members for overdue tasks, updating task boards, or scheduling meetings when it finds open calendar slots – acting like a project coordinator who proactively keeps everyone on track. Software development itself might see agentic AI that goes beyond just generating code; for instance, an AI that not only writes code (using generative AI) but also *decides* when to refactor code, sets up its own development environment, runs the code to verify outputs, and debug errors it finds – essentially an autonomous junior developer that only asks for help when truly stuck or when a decision beyond its scope (like a feature decision) is needed. This level of autonomy is cutting-edge and not yet commonplace, but prototypes exist. For example, some AI research initiatives (often called “AutoML” or automated machine learning pipelines) allow an AI to try different model designs and training methods on its own to achieve the best performance on a dataset, without a human manually tuning each experiment. This is agentic behavior applied to the software engineering of AI itself.
- **Multidisciplinary Engineering Systems:** Many modern engineering challenges are multidisciplinary (combining mechanical, electrical, software, etc.), and agentic AI plays a key role in such integrated systems. Think of **smart cities** or **industrial IoT (Internet of Things) systems**: an AI agent could coordinate various subsystems – traffic, energy, water, communications – to optimize overall performance. Another example is in healthcare technology (biomedical engineering): an agentic AI could manage a network of medical devices in a hospital, ensuring that patient monitoring devices, medication pumps, and diagnostic tools operate in sync and alert staff of any anomalies, effectively acting to maintain patient safety across the system by adjusting alarms or device settings as needed. In aerospace, a spacecraft or satellite operates largely on agentic AI when far from Earth – it must perform navigation, system checks, and course corrections autonomously based on onboard sensor data, since real-time human control is impossible due to communication delays. These scenarios highlight how agentic AI can assume duties where continuous decision-making is required and human intervention may be impractical in real-time.

Benefits of Agentic AI in Engineering: Agentic AI can **greatly enhance automation**, taking over complex coordination tasks or real-time control that would be tedious or impossible for humans to manage at scale or speed. For engineers, this means mundane or highly granular decisions can be offloaded. This can improve efficiency and consistency – an AI agent doesn't get tired or overlook things, and it can react within milliseconds to conditions (crucial for fast systems like electrical grids or high-speed manufacturing). Agentic systems can also find optimizations that humans might miss, because they can juggle many variables at once and learn from continuous data. Importantly, agentic AI can keep systems running optimally even in changing conditions, by constantly adapting (a traditional program would not adapt unless explicitly re-programmed for each scenario). In project contexts, having an agent handle routine follow-ups or minor decisions frees human engineers to focus on creative problem-solving and big-picture decisions.

Limitations and Cautions: With autonomy comes risk. An agentic AI's **decision quality is only as good as its algorithms and training**. If it's not designed with all necessary safeguards, it might take inappropriate actions. For example, a grid-control AI might decide to shut off power to a sector to prevent a larger outage – technically achieving a goal of grid stability, but maybe it shuts down power to a hospital if not properly constrained. Therefore, agentic AI usually needs carefully defined operating boundaries (guardrails) and sometimes a human-in-the-loop for critical decisions. Another challenge is debugging and transparency: when an AI agent makes a complex sequence of decisions, it can be hard to trace why it did what it did, which is problematic if something goes wrong or if a regulator asks for an explanation. Ensuring **accountability** is thus a key concern – engineers need to design agentic systems such that they can report their reasoning or at least operate under policies set by humans. Additionally, agentic AI might struggle in situations that involve nuanced judgment beyond data (ethical dilemmas, novel scenarios with no precedent in training). They excel in well-defined domains but can fail strangely outside those bounds. As with generative AI, **extensive testing** is required. For a self-driving car AI, for instance, it must be tested across millions of scenarios to be trusted on the roads, and even then one needs safety fallbacks. **Reliability** is paramount – in engineering, if an autonomous agent fails unpredictably, the consequences could be serious. So while agentic AI can take on a lot of work, engineers must supervise these agents, especially early in their deployment, and treat them as evolving aides that need tuning and possibly a hand on the emergency brake if they go off course.

Comparing Generative AI and Agentic AI

Both generative and agentic AI are powerful, but they serve different purposes and operate in different ways. Here we distill the key differences and relationships between them:

- **Core Function:** Generative AI acts as a **creator**, producing new outputs like designs or text only in response to specific prompts. In contrast, agentic AI functions as an **actor/decision-maker**, taking steps and making choices to achieve a set goal. While generative AI requires prompting to act, agentic AI pursues its objective independently once activated.
- **Initiative:** Generative AI has no initiative of its own; it's reactive. Agentic AI has **autonomy** – after receiving a goal, it can initiate steps without additional user prompts at each step.
- **Mode of Operation:** Generative AI typically operates in a single step or a short session – you give input, it gives output, end of story (until the next input). Agentic AI operates over *multiple steps in a loop*, continuously perceiving and acting until the goal is achieved or time runs out.
- **Output vs. Outcome:** Generative AI's output is usually some content (an artifact like a design or text). It doesn't ensure any real-world outcome beyond that output being produced. Agentic AI's "output" is an **achieved outcome or completed task** – it's judged by the result (e.g. task completed, system optimized), which might involve a series of outputs and actions along the way.
- **Examples in Practice:** For illustration – In an engineering firm, a **Generative AI** tool might generate a draft blueprint for a new product. An **Agentic AI** system might oversee the whole product development process: gathering project requirements, scheduling design reviews, triggering the generative AI to create a draft, then integrating feedback and maybe even initiating prototyping via a 3D printer. The generative AI is one component; the agentic AI is the manager orchestrating tasks.
- **Technical Underpinnings:** Generative AI often relies on large trained models (like neural networks) that are optimized to produce realistic outputs. Agentic AI might incorporate such models but also relies on decision-making algorithms, which could include planning algorithms, utility optimization, or reinforcement learning that lets it learn from trial and error. Memory is another aspect: a generative model doesn't have long-term memory of past interactions (each prompt-output is largely independent, aside from short context memory in some cases), whereas an agentic AI usually retains memory of what it has done or observed as it proceeds, to inform future decisions.
- **Use Cases Where Each Excels:** Generative AI shines when you need *creative generation*: designing, brainstorming, drafting, or when automating a well-defined output (like writing a section of a report). Agentic AI excels in *process automation*: handling the nitty-gritty of

workflows, real-time control, and scenarios where decisions need to be made in sequence or continuously.

- **Interdependency:**
- These two types of AI serve complementary functions. In practice, many advanced AI systems integrate both: agentic AI may invoke generative AI for specific tasks (for instance, generating a design for a subcomponent as part of a broader workflow). Conversely, generative AI applications can be embedded within an agentic framework to enable automated refinement of outputs based on predefined criteria, eliminating the need for manual prompting at each stage. Collectively, generative and agentic capabilities facilitate seamless end-to-end workflows—from initial concept development through to execution and deployment. For example, in engineering design, generative AI suggests potential designs, agentic AI evaluates them using simulation, selects the optimal solution, and can progress to manufacturing procedures.

The table below summarizes some key differences:

Table 1: Key Differences Between Generative AI and Agentic AI

Aspect	Generative AI	Agentic AI
Primary Role	<i>Content Creator</i> – Produces new data/content from inputs (e.g. designs, text, images).	<i>Autonomous Agent</i> – Executes actions to meet goals, making decisions along the way.
Trigger	Activated by a user prompt or request. It won't act unless asked.	Activated by setting a goal/task. Once started, it continues on its own (within its scope).
Operation Mode	Single-step or session-based output generation. (One prompt → one output, possibly iterative with user guidance for refining.)	Continuous, looped operation (perceive → plan → act → repeat), handling multi-step tasks over time.
Example Output	A set of design options for a bridge given design criteria; a block of code implementing a function described in words; a simulation scenario data set.	Adjusted traffic light timings throughout a city for the next 2 hours to reduce congestion; an automated sequence of tests and bug fixes applied to a software project overnight.
Adaptability	Limited to patterns in training data – doesn't adapt on the fly during an output (though can be steered with prompts). If the initial output isn't right, user must prompt again or fine-tune.	Highly adaptive – observes results of its actions and changes strategy accordingly (if designed with learning). Can handle dynamic situations within its trained experience.
Scope of "Intelligence"	Focused on <i>creative pattern generation</i> and often narrow (e.g. generate text that looks like Shakespeare, or design a beam under certain load).	Holistic <i>problem-solving</i> within a domain, often integrating multiple functions (could involve planning, using other tools, communications, etc.).
Human Interaction	Human provides input and then usually evaluates or edits the output; interactive but human-guided at each step.	Human sets objectives and constraints up front, then the AI agent may work autonomously for extended periods. Human may intervene or get updates, but not necessarily at each step.
When to Use	When you need a creativity boost, a draft or multiple options quickly, or automation of content creation. Particularly useful in design, documentation, option exploration.	When you need to automate a process or decision loop, manage a system in real-time, or offload a sequence of tasks. Useful for control systems, project management, complex decision-making scenarios.
Example Tools	<i>Generative:</i> GPT-based chatbots, image generators (DALL·E, etc.), generative design software (for CAD), code generators.	<i>Agentic:</i> Autonomous robots/drones, intelligent control systems (smart grid controllers, HVAC optimizers), AI assistants that can execute sequences (AutoGPT-like systems), adaptive scheduling software.

As shown, **generative AI and agentic AI differ fundamentally in their level of autonomy** and the nature of their outputs. An easy way to remember: *Generative = generates, Agentic = acts*. For an engineer, the choice between using generative AI or agentic AI (or both) depends on the problem at hand. If the task is “I need a bunch of innovative ideas or a first draft of something,” generative AI is the tool. If the task is “I need this process handled end-to-end automatically,” an agentic AI approach is needed. Often, engineering projects will involve both needs, so these AI techniques complement each other.

Ethical and Professional Considerations in Using AI (Generative & Agentic) in Engineering

The integration of AI into engineering brings not just technical challenges but also important **ethical and professional responsibilities**. Licensed engineers, bound by codes of ethics to hold paramount the safety, health, and welfare of the public, must carefully consider how using generative or agentic AI aligns with these obligations. Below are key considerations and best practices:

1. Safety and Reliability: In engineering, safety is non-negotiable. Any AI-generated design or AI-driven decision can impact real lives if it concerns infrastructure, machines, or systems people use. **Generative AI** might propose a design that hasn't been proven safe; if an engineer were to accept it without proper verification, the results could be catastrophic (imagine a building component that fails under stress because the AI's suggestion had a hidden flaw). Therefore, every AI-generated output used in engineering must be **validated and tested** using traditional engineering analysis and simulation. On the **Agentic AI** side, an autonomous system controlling something like traffic or equipment must have thorough fail-safes. Engineers should ensure that agentic AI controllers have bounds — e.g., an autonomous construction robot should shut down or hand over control if it detects conditions outside its safe operating envelope. Redundancies and manual override options are wise. Essentially, **human oversight** remains crucial: AI doesn't get a free pass on safety. A professional engineer might use AI to aid decision-making, but the engineer is ultimately accountable for the outcome. This aligns with existing engineering ethics: you can delegate tasks, but not responsibility.

2. Accountability and Transparency: When AI is involved in creating a design or making a decision, who is accountable if something goes wrong? Engineers must not fall into the trap of offloading blame to “the AI”. Ethically, if you use an AI tool, its output becomes part of *your* professional work product. It's important to document the use of AI and ensure traceability. For generative AI, keep records of how an AI-assisted design was generated and what modifications the human team made. For agentic AI, maintain logs of the agent's decisions and actions. In safety-critical domains, regulators and clients may require explanations for decisions – if an agentic AI makes a complex decision (like an autonomous vehicle swerving in a certain way during an accident), there should be a way to reconstruct why (through data logs or an internal reasoning trace). Transparency is also about being honest with stakeholders: If an AI was used to draft a report or design, should the client be informed? Generally, transparency builds trust. However, overly detailed disclosure can confuse; so one approach is saying “We used advanced software tools (including AI) to aid our design, and all outputs were thoroughly reviewed and validated to meet engineering standards.” This sets the context that AI was a tool, and the engineers still ensured standards compliance.

3. Bias and Fairness: AI systems can inadvertently introduce or perpetuate **bias**. In engineering, bias might not be about social issues only (though that is also possible, e.g., if AI prioritizes projects in wealthy areas over poor ones due to biased training data) – it can also be technical bias, like favoring familiar solutions. If a generative model was trained mostly on conventional designs, it might under-suggest novel approaches (conservative bias). Conversely, it might have biases from data that are not obvious (perhaps training data had many failures that were labeled successes incorrectly). Engineers should be aware of the training data behind generative models: is it comprehensive, up-to-date, and relevant to the specific context? For example, a bridge design AI trained on only short-span bridges might not be reliable for a long-span suspension bridge project. If using agentic AI, consider if its decision-making could unfairly disadvantage certain groups – e.g., an AI that allocates city resources should not underserve particular neighborhoods consistently. Ethical use means actively checking for bias in AI outputs. Sometimes this involves diverse testing scenarios or peer review of AI-involved decisions. Inclusivity in design is a growing concern: ensure AI suggestions or decisions align with fairness (for instance, an AI traffic system shouldn't always route traffic through one community causing them disproportionate noise and pollution). Engineers, guided by ethics, should mitigate bias by either tuning the AI, using diverse training sets, or simply recognizing and correcting biased outputs.

4. Intellectual Property (IP) and Attribution: Generative AI often learns from existing designs, many of which may be proprietary or copyrighted. If an AI-generated design is very similar to something in its training data, there could be legal or ethical issues. Engineers should ensure that using an AI tool does not accidentally *steal* someone else's design. This is tricky because the internal workings of AI are complex; however, measures can be taken such as using AI models that are trained on open data or company-owned data for sensitive design work. If an AI tool is cloud-based (e.g., a generative design service online), check the terms of service: who owns the output? Some tools might claim rights or expose your prompt and output to others – which may be unacceptable for confidential projects. Prefer tools that explicitly state the user retains IP of outputs. From an ethics perspective, if an AI significantly contributed to a piece of work, professionals might debate whether and how to acknowledge that. In publications, crediting the use of AI is appropriate, but in design documents, it might not be customary. The safe approach is to treat AI as part of the toolbox – you wouldn't list “used finite element software X” on a drawing, nor would you list “used AI tool Y,” but you remain responsible for the results of using those tools correctly.

5. Privacy and Data Security: Agentic AI systems, in particular, often need a lot of data to make decisions – some of which could be sensitive (e.g., an AI that monitors emails to coordinate a project, or building sensors that track occupants for energy optimization). Engineers implementing such AI need to safeguard personal or sensitive data. Ethical guidelines and laws (like GDPR for personal data in the EU) may apply. If, say, an engineering firm uses a generative AI to analyze or generate a client report that contains proprietary information, putting that into a third-party AI service could inadvertently leak data. Always consider where the AI runs (local vs cloud) and what

happens to the input data. Use enterprise or on-premises versions of AI for confidential data if possible. Also, inform people if an AI system is monitoring them – for instance, if construction workers are being monitored by an AI for safety, privacy rights might require that they know this and that surveillance is only for safety purposes.

6. Competence and Training: The engineering code of ethics typically requires engineers to work only in areas of their competence. Does using AI extend one’s competence or mask a lack of it? An ethical engineer shouldn’t use AI to produce something beyond their understanding and just stamp it. For example, an engineer not experienced in seismic design should not just take an AI-generated seismic analysis and sign off on it. You must either get the requisite human expertise or learn how to verify the AI’s output competently. Professional development (like this course) is crucial so engineers understand what the AI is doing and where it might err. In other words, using AI doesn’t relieve an engineer from understanding the fundamentals – in fact, it requires **more diligence** to ensure the technology is applied correctly. There’s also an ethical angle of telling the truth about AI’s role: if an AI helped create a design, the engineer shouldn’t claim sole credit deceitfully – nor should they hide AI use if transparency is needed. Conversely, they shouldn’t downplay their responsibility by saying “the AI did it, not me.”

7. Impacts on Society and Employment: Broadly, engineers should contemplate the societal impact of widespread AI deployment. Will generative and agentic AI displace jobs or certain tasks? Likely, yes for some routine tasks – but it also creates new opportunities. Ethically, companies and engineers should strive to use AI to **augment human capabilities** rather than simply replace humans for cost-cutting on critical tasks that require judgment. Many engineering ethics codes emphasize improving society. AI can indeed improve safety and efficiency (fewer human errors, ability to handle more complex data), which is a positive. But if misused (like relying on AI without checks or using it to cut corners), it could lead to harm or erosion of trust in engineers. Maintaining public trust means being proactive about these consequences: for example, if an autonomous vehicle controlled by agentic AI gets into accidents, it could set back public acceptance not only of AI but of the engineering field’s ability to protect the public. Thus, engineers should advocate for and adhere to standards and maybe even contribute to developing **regulations or standards for AI in engineering** (much like we have building codes, we might need AI safety codes).

8. Environmental Considerations: This is a subtle one that engineers are starting to consider – large AI models can be energy-intensive to train and run. A responsible engineering firm might consider the carbon footprint of using a massive generative model for trivial reasons. While one engineer’s use of AI is a drop in the bucket, at scale these decisions matter. The mitigation could be using more efficient models or running on renewable energy-powered cloud services. This is not yet a formal part of ethics codes, but it falls under sustainability, which many engineers value.

Best Practice Guidelines:

- *Augmentation, Not Autonomy (yet):* Treat AI as an assistant to **augment human decision-making**, not replace it. Especially for critical decisions, have a human in the loop.
- *Validation and Testing:* Set up a protocol to rigorously test AI outputs. For generative designs – run simulations, check against hand calculations. For agentic AI – simulate its operation in a safe environment (digital twin) before live deployment, and include emergency stop logic.
- *Continual Monitoring:* Don't "set and forget" an agentic AI. Monitor its performance. Use dashboards or alarms to track what the AI is doing. This allows catching any errant behavior early.
- *Ethical Training:* Make sure engineering teams are trained about AI ethics. It may be useful to have interdisciplinary oversight – e.g., involve an AI ethics specialist or committee when rolling out something like an autonomous urban system to get perspectives beyond pure engineering.
- *Documentation:* Keep documentation of AI usage in your projects. This helps with transparency, future troubleshooting, and learning. It's similar to documenting calculations – if a design decision was influenced by an AI recommendation, note it and how it was verified.
- *Stay Within Regulations:* Follow any emerging regulations about AI. For example, some jurisdictions might introduce rules for autonomous vehicles or AI-designed medical devices. Ensure compliance with both engineering regulations and AI-specific ones.

By proactively addressing these considerations, engineers can harness generative and agentic AI effectively **and** responsibly. These technologies, when guided by ethical professional practice, can lead to safer designs (AI might catch issues humans miss), more efficient processes, and innovative solutions to engineering challenges. The goal is to leverage the strengths of AI while upholding the standards of our profession.

Conclusion

Artificial Intelligence is rapidly expanding what's possible in engineering. **Generative AI** empowers engineers with a creative partner that can produce designs, code, and content at an unprecedented speed and breadth, helping to explore possibilities and automate routine creative tasks. **Agentic AI** introduces a proactive agent that can take on the burden of execution and control, handling complex sequences of actions to meet engineering goals, whether it's optimizing a system's performance in real time or managing project workflows. Each brings unique advantages – generative AI adds inspiration and efficiency, agentic AI adds autonomy and scalability – and together they can transform engineering workflows from conception to operation.

However, as we implement these powerful tools across civil, mechanical, electrical, software, and other engineering disciplines, it is our responsibility as professionals to remain vigilant about the outcomes. AI doesn't replace the need for engineering judgment; rather, it elevates the importance of our judgment in guiding and validating AI. The **engineer's role is evolving**: in addition to designing and problem-solving, it now involves curating and overseeing AI contributions. By understanding the differences between generative and agentic AI, engineers can better choose the right tool for the task, harness their capabilities, and mitigate their risks.

In your practice, you might start by using generative AI to draft a few design ideas or a preliminary report – you'll save time, but remember to double-check everything. Or you might deploy an agentic AI system to monitor a process – you'll gain efficiency, but keep an eye on it and be ready to step in if needed. Over time, these AIs can become trustworthy extensions of our teams, handling the heavy lifting of computation and coordination. This leaves engineers more time to do what humans excel at: innovation, critical thinking, and making the value-based decisions that machines alone can't.

Embracing AI in engineering is not just about adopting new tools, but also about **continuously learning** – technologies will improve, new methods will emerge, and ethical norms will develop. As licensed professionals, staying informed (through courses like this and beyond) is part of our duty. This ensures we maintain competence in the face of change and continue to uphold the safety and well-being of the public.

The future of engineering will likely see generative and agentic AI integrated into standard practice, much like CAD and simulation tools are today. Those who understand and manage these tools well will lead the way in innovation. As you finish this course, consider how these AI techniques could specifically address challenges in your day-to-day work, and what steps you can take to implement them responsibly. By combining human expertise and creativity with AI's capabilities, we stand to reach new heights in engineering achievement, safely and ethically.