OXFORD JOURNAL
OF STUDENT
SCHOLARSHIP

AI Assisted Computational Methods for Efficient Aerospace System Design

Arnav Dipak Patil
coolarnavpatil@gmail.com

ABSTRACT

Aerospace engineering is a field of engineering that involves the use of simulations, testing, and computational methods. To improve the efficiency and accuracy of Aerospace engineering, it is a viable option to consider the use of Artificial intelligence. This paper reviews the implementation of AI assisted computational methods in order to achieve an efficient Aerospace system design. In particular, this paper discusses and analyses AI implementation in Topology Optimization, Finite Element Method, Controls, and Computer vision. Furthermore, this paper emphasizes how AI can be used in traditional engineering to reduce computational cost and time, and how to create an effective method to produce more accurate results quicker. From the research done, incorporating AI improves efficiency and allows for more complex structures to be used without requiring an excessive amount of time and energy. These developments demonstrate that AI can play an important role for the future of aerospace engineering.

Keywords

Artificial Intelligence; Aerospace Engineering; Topology Optimization; Finite Element Method; Controls; Computer Vision; Traditional Systems; Computational Methods; Simulations; Efficiency; Black Boxes; Fluid Structure Interaction; Predictor Evaluator Network (PEN); Computational Costs; Coupling Parameter; Micropolar-Elasticity SIMP Algorithm; Density Field; Feedforward Neural Network; Convolutional Neural Network; Generative Adversarial Network; Degrees Of Freedom; Reduced Order Models; Proper Orthogonal Decomposition

INTRODUCTION

The Concept of Aerospace Engineering

Aerospace engineering involves some of the most technical systems that have been designed. These systems range from aircrafts operating at extreme conditions to computing complex physics simulations. Traditionally, solving aerospace problems required an extensive amount of time and cost, which can prove to be a limiting factor in the efficiency of getting research done. Thus, as technology grows more advanced, there are ways in which we can improve on aerospace system designs.

Understanding Artificial Intelligence (AI)

In recent years, Artificial intelligence has been growing more advanced in various fields and emerged as a powerful tool that can optimize the time that is needed to solve various problems. Its ability to recognize patterns and solve complex equations can prove useful in helping advance traditional aerospace methods rather than replacing them. In aerospace, fields such as mechanics, controls and topology optimization can have AI incorporated in them.

AI Assisted Computational Methods

AI assisted computational methods refer to the combining of artificial intelligence with traditional engineering methods. These methods can be used in various cases, such as in finite element modeling, control system optimization, topology optimization, and computer vision analysis.

Finite element modeling is a way for engineers to see how various structures are impacted under forces, stresses and the overall environment. Traditionally, FEM struggles with complex structures that have millions of degrees of freedom, which would need a massive amount of power to solve. These complex structures include the surplus of structures that are used in aerospace domains, such as wings and planes. By implementing AI, we are accelerating the simulations to require less time to compute and have efficient calculation to get the best results.

Topology optimization is a technique that is used to find out what is the most efficient way of distributing a material to achieve the task that we desire. In aerospace engineering, finding the most efficient way of distributing a material is incredibly important due to the fact that any excess material adds mass to the overall structure, and with this extra mass other components will change. For example, with extra mass the amount of fuel and thrust needed for an aircraft would increase, which would also increase costs. Traditional methods of topology optimization, such as SIMP, function by repeatedly calculating equations to output geometrically efficient components, which requires days to simulate. This extra time poses a problem since any minor error can lead to dangerous consequences, thus having more time to recheck and retest is crucial. By introducing AI, topology optimization has the ability to learn from itself and previous data to figure out patterns to generate optimized structures with less computational cost.

Control systems are able to design models that regulate the behavior of dynamic systems that can be witnessed in various aircrafts. Within aerospace spacecrafts, like rockets, they require precise operations in order to ensure safety. For example, mapping out a rocket landing would require navigating the trajectory with 6 degrees of freedom at relatively high speeds. This is done with time constraints that require fast computation, which traditionally control systems might not keep up with. With the integration of AI, systems can model and adjust their behavior within the given time. This can overall improve the accuracy and safety of aircrafts.

On another level, computer vision involves the use of algorithms to make sense of the world and enable individuals to interpret visual data, such as images from sensors and cameras. In aerospace this is seen within spacecrafts that orbit Earth. Due to the fact that these spacecraft are far away, there are communication delays that make it difficult for Earth to remotely give and receive data in important real-time scenarios. Thus, the use of AI methods can allow autonomous structural monitoring and defect detection, which reduces the risk of damaging spacecrafts.

Aerospace System Design

Aerospace system design involves making systems that ensure safety and efficient operations. Engineers rely on computational tools to test designs before they make their physical prototypes. Thus, the increasing complexity of modern aerospace systems would need efficient computational methods. This is where the integration of AI is valuable.

The Importance of the Paper

In engineering, reducing simulation time and computational costs is important to improve on prototypes and allow faster innovation cycles. With the use of AI we are able to improve in areas such as spacecraft design and advanced structural systems in which reliability is critical. Moreover, by improving efficiency, we enable researchers with the freedom to explore more complex and challenging designs that were once considered time consuming. Thus, by understanding how AI can improve on traditional methods we know today, we support a future that will allow more technological advancements to be made and improved on.

Aims and Goals of the Study

This research report explores how AI can be used to solve key problems within aerospace domains. It will examine specific examples for each field where AI enhances performance and reduces costs to run.

LITERATURE REVIEW

From recent research, the interest in utilizing AI to reduce high computational cost and time of aerospace system design has increased. In papers, such as Tang (2025), we are able to see methods such as machine learning, surrogate models, and evolutionary algorithms play a role in AI for aerodynamic and structural design. Furthermore to get around high computational demands of finite element analysis solvers and computational fluid dynamics, there has been studies where researchers would use data-driven surrogate frameworks in order to evaluate complex wing designs Kiyik et al., (2026), improve high speed turbomachinery Pierson et al. (2025), or use advanced decomposition methods in order to break down complex math equations into smaller chunks at a fast rate even as data grows Wu et al., (2025) .

There has also been research improving upon traditional methods. Traditional solvers such as Solid Isotropic Material with Penalization (SIMP) repeatedly solve finite element equations, which can be computationally expensive. To solve this issue there has been a study by Sosnovik and Oseledets (2017), where the material layout is treated as a grid or black and white pixels. Each pixel represents the presence of the material, with black being solid material and white being empty space. With this, Convolutional Neural Networks can be used to draw out an optimized shape. Furthermore, Rasulzade et al., (2022) demonstrated that optimized neural configurations like Res-U-net have the ability to add parameters onto final shapes in thousands of seconds. Which shows how there are frameworks that can pose a highly efficient alternative to traditional solvers. To avoid massive amount of data collection in topology optimization, there has also been research on how an AI model can test itself using the Predictor-Evaluator Network in order to train and get a better accuracy without requiring pre-optimized datasets Halle et al., (2021).

The integration of AI in trajectory optimization also plays a key role, as studies show how by implementing AI, researchers are able to have efficient real time guidance for descending vehicles Wang et al., (2019); Jung et al., (2024); Prous et al., (2025). These frameworks run relaxed and primary mathematical solvers to ensure the flight computer would always make a valid path without freezing during landing.

More application specific AI methods have also been studied. In a paper by Shen et al., (2022) a proposed AI assisted approach for a real time powered landing guidance was discussed that combined neural networks with convex optimization. Neural networks are machine learning models that have layers of connected nodes in which they learn patterns to form data to form decisions. Convex optimization is a mathematical method that is utilized to find the best solution for a problem that has a single global minimum. In the method discussed, a neural network makes an initial estimate which provides a good starting point to cut the amount of time that the programming algorithm has to work to find the solution. This is an effective approach, but it is limited to the specific guidance problem and relies a majority on trained neural networks. This shows that current incorporating AI may lack scalability when applied to more broad aerospace design, such as aircraft aerodynamic design and structural design.

Another area of research focuses on reducing computational time and cost by having machine learning and traditional numerical methods work together. In a study by Lu et al., (2021), it showed how hierarchical deep-learning neural networks can be combined with techniques that reduce model complexity such as Proper Generalized Decomposition (PGD). PGD simplifies simulations by breaking complex problems into smaller functions. This would allow simulations to run faster while having a similar accuracy. However, the authors mention how the performance does depend a majority on the reduction strategy used and would require careful integration with other solvers if it were to be a method that could be widely used.

METHODOLOGY

This study uses research papers and reviews of existing work in computer vision, topology optimization, finite element analysis, and controls. These papers were gathered from databases such as Google Scholar, arXiv, and SpringerLink. Key words used in searching for these papers were combining AI terms with domain specific terms. These AI terms included: Machine learning, Artificial intelligence, Deep learning, and Neural Networks. Domain Specific terms included: pose estimation, topology optimization, finite element analysis, optimal control with aerospace, and spacecraft computer vision.

Inclusion Criteria

During research, the papers that were considered were ones that dated from 2016 onwards. This was due to the significant advancements that were made from 2016 onwards, which made more recent studies more relevant to the topic of AI implementation in aerospace system design.

Exclusion Criteria

The reasoning behind the exclusion of papers was if the content of the paper lacked a strong connection with the topic on AI utilised in traditional methods, as it would have been extraneous information. Furthermore, papers were excluded if they had conclusions that were unclear or didn't say much about the overall topic.

Analysing the Papers

In order to analyze the studies, each paper was analyzed in terms of the problem that they wished to solve with AI, or what traditional method they wished to improve with AI. The use of AI was considered by identifying whether the papers had shown AI as a complete replacement, or as an aid to the traditional methods. Within each domain, the analysis was carried out and compared. For example, papers that used neural networks to boost FEM simulations were compared with other methods of reducing computational time and were ranked based on efficiency and accuracy.

Efficiency: The efficiency in computational methods refers to the reduction in computational time, the complexity of the model, and the number of iterations required to converge to a design compared to a traditional method.

Accuracy: The accuracy of computational methods refers to the amount of error that the method produces compared to the established baseline.

For computer vision, different methods of understanding the position of a component of a spacecraft with reference to its body were compared.

Screening Process

The initial amount of papers gathered was a total of 45 papers. From reading the title and evaluating the abstract of each paper, the number of papers that remained was 30 papers, which needed to be fully read. After reading these papers fully and excluding the papers that didn't meet the inclusion criteria, the total number of papers that remained was 20 that would be included as sources.

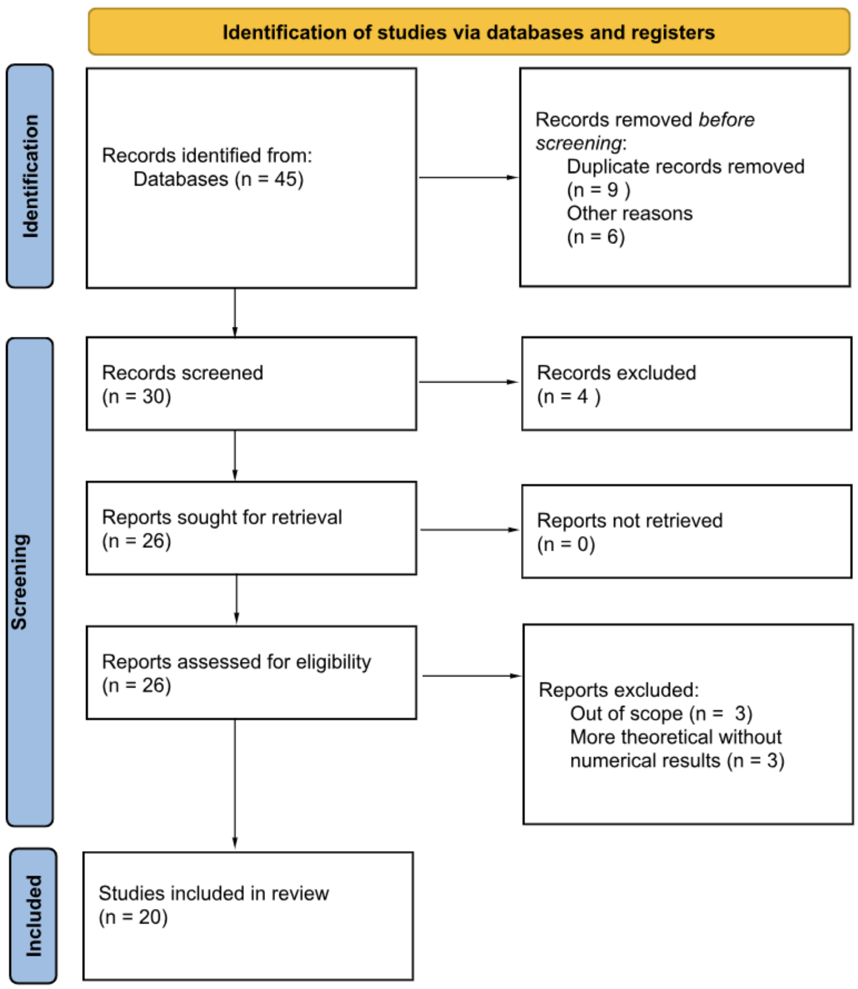

Figure 1: PRISMA flow diagram on the screening process for papers

FINDINGS AND DISCUSSIONS

Topology Optimization

Traditional Topology Optimization methods, such as Solid Isotropic Material with Penalization (SIMP), are computationally expensive. This is due to the fact that SIMP works iteratively. This means that it repeatedly solves finite element equations until they join together to an optimal structure.

In the past, there have been implementations with SIMP mixed with AI in order to speed up topology optimization. However, in order for their AI to work with SIMP, it requires large datasets of pre optimized geometries to train and build on. This process of getting and running large datasets is time consuming, often even requiring 100,000 samples to be trained on. In addition, these samples slightly restrict the AI, making it such that it cannot well handle the boundary conditions that it had not seen in its training data.

In order to solve this conundrum, we can allow the AI to test itself. This is done by using the Predictor-Evaluator Network (PEN), an architecture discussed in a paper by Halle et al., (2021). The PEN utilizes two components, The Predictor and the Evaluator. The Predictor is a neural network that is able to be trained. A problem for the Predictor to solve is done by giving different conditions for it to work around:

1. Kinematic boundary conditions: where the structure is fixed/supported
2. Static boundary conditions: where and how forces are applied
3. Target degree of filling: what fraction of the design should be filled with material

With this input the Predictor outputs a density map. This is a grid of values that are between 0 and 1, which denote how much material should be present in that area on the overall shape. With this density map, it could be compared with known good solutions and tweak the neural network from there. However, in PEN, they utilized four fixed mathematical evaluations to score each geometry:

1. Compliance: structural stiffness
2. Degree of filling: how close the material fraction is to the target
3. Filter: Penalizes unphysical alternating patterns
4. Uncertainty: Penalizes intermediate density values

These four fixed mathematical evaluations are what the Evaluator grades the produced density map on. The Predictor is then tweaked to be improved by the next set of new boundary conditions it would solve. Then, the process starts over again. As no fixed dataset was added in, the network is able to process millions of training samples during training.

Halle et al., (2021) were able to produce results that tell us more about how PEN worked with itself.

- Training time: ~3.25 hours (one-time cost)
- Inference speed: ~7.3 ms per geometry
- Conventional SIMP speed: ~1.9 seconds per geometry
- Speedup at inference: ~259×

In order to see another way AI can be used in topology optimization it is important to understand the flaws of Standard machine learning topology frameworks.

Standard machine learning topology frameworks have two flaws that act as limitations:

1. Physics fidelity: Nearly all TO methods rely on classical elasticity, which is a standard physics model that predicts how materials bend, stretch or compress. However this model has no length scale parameter, which means that it ignores size at small scales and cannot capture microstructure dependent behavior. Microstructure dependent behavior is important for different compositions such as, lattice structures, foams, and cellular solids. These compositions have size effects, where they behave differently depending on how small its internal build is. Thus if classical elasticity cannot determine one size to another, it will treat them the same even when they are not.
2. Computational cost: Even with the help of Machine learning, the Finite Element model is complex, which would require a high iteration count in order to get the desired topology optimization.

With these limitations in mind, one solution that Zhou et al., (2025) had done was replacing the classical elasticity with micropolar elasticity. Classical elasticity has each material point moving in two directions, either x or y. This means that it has only translational degrees of freedom. With micropolar elasticity we improve onto this by adding a third movement, micro rotation. This captures more physics scenarios in which materials can resist being twisted, rather than only resisting stretching and compressing. With this addon two new material parameters emerge:

1. Coupling parameter $\bar{\nu}$: controls how strongly micro rotation acts to affect the overall deformation.
2. Characteristic length l: how far the effects of rotation spread through the material.

When these parameters are nonzero, the material has bending and twisting resistance. This is what classical elasticity could not show before.

This alone makes the generated optimized topology change immensely compared to the classical elasticity result. The changes indicate the structural stiffness improving by up to 38% sometimes. With this, the first limitation is accounted for. However, the problem encountered is that this new improved model is now more expensive than it was before. The solution to this problem is to stop the optimization when it has already come up with the main body. Through the iterations that the model takes to come up with the optimized topology, the first 10-20% is used to establish the main skeleton, while the remaining 80-90% is used for refining. Thus, instead of having the algorithm do the remaining 80 to 90%, we can have a more cost efficient simple ML give the final output.

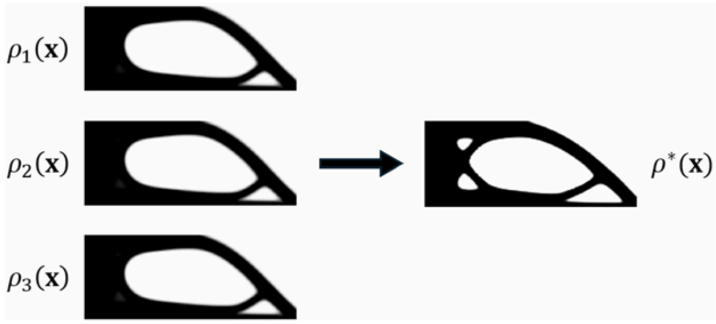

Figure 2:Three density layouts that are processed by ML to form final density field
Zhou et al., (2025)

To do this, we can split into two stages:

1. The micropolar-elasticity SIMP algorithm runs till $\delta \leq 10\%$, in which δ defines the change in material density between two consecutive iterations. Then, instead of having one output, the paper has the last 3 density maps outputted: $p_1(x)$, $p_2(x)$, $p_3(x)$. This would tell the ML of the direction and rate that the algorithm was heading towards.
2. Then, the ML processes these 3 density maps to give a final density field p*(x)

With this final density field, the study has addressed the two limitations which are originally present. The study proposes testing out different MLs to find which one is better suited for the final stretch. Three ML models include:

Case	(1)	(2)	(3)	(4)
TO Truth	$t = 23.95$s $w = 2769$J	$t = 37.60$s $w = 4025$J	$t = 41.33$s $w = 4852$J	$t = 39.69$s $w = 4153$J
FFNN	$t = 5.21$s $w = 883$J	$t = 7.12$s $w = 1210$J	$t = 7.80$s $w = 1367$J	$t = 7.22$s $w = 1259$J
CNN	$t = 5.81$s $w = 887$J	$t = 7.64$s $w = 1240$J	$t = 8.39$s $w = 1370$J	$t = 7.92$s $w = 1269$J
GAN	$t = 5.73$s $w = 880$J	$t = 7.83$s $w = 1243$J	$t = 8.26$s $w = 1370$J	$t = 8.05$s $w = 1250$J

Figure 3: Table showing time and energy consumed for each ML compared to the original
Zhou et al., (2025)

Figure 2 illustrates the time taken and the energy consumed for each ML model that is used for the remaining 80 to 90% to get the final density field. By looking at the traditional topology optimization method (TO truth), it can be witnessed that it takes on average 35 seconds to get the density field, and consumes an average of 3950 J of energy. This can be compared to the three ML methods below which demonstrate less than 10 seconds and consume around 1200 J of energy.

1. FFNN (Feedforward Neural Network): takes input data as a flattened vector, which is taking a matrix and turning it into a list of numbers. This makes it easier to use. However, it can be less effective for seeing geometric patterns.
2. CNN (Convolutional Neural Network): processes data in its grid form and uses filters to detect patterns. This allows it to capture features and makes it well suited for TO.
3. GAN (Generative Adversarial Network): uses a generator to create outputs and a discriminator to evaluate them, by using competition to improve results. It can produce detailed outputs. However, training can be unstable.

Controls

Controls are the inputs that are given to a spacecraft or aircraft on what actions it must take to reach an intended outcome. One such problem where controls are tested immensely is the 6 degrees of freedom (6-DoF) powered landing. In this problem we are trying to guide a rocket safely from an altitude to the ground.

This itself requires many calculations and overcoming many challenges. While coming down the vehicle must control its 3D position (x, y ,z) and its rotation (yaw, roll, pitch) while taking care of other factors. These factors include its thrust magnitude, its tilt as it comes down, and the amount of fuel it has. The calculations and the nature behind all of these challenges are highly nonlinear as it involves quaternion kinematics, aerodynamic drag, and rotational motion.

In order for the rocket to safely land on the ground it must compute all of these calculations in under one second, or else it wouldn't have enough time to respond to the disturbances that could happen, making the landing unsafe. This was discussed in a paper by Shen et al., (2022).

To solve this problem of whether we can solve the trajectory in time, Shen et al., (2022) first incorporated Sequential Convex Programming (SCP). From SCP they convert the highly dense nonlinear problem into smaller chunks of convex optimization subproblems, which each can be solved using modern solvers.

With the full 6 DoF nonlinear problems, which are nonconvex, we linearize them around a reference trajectory using Taylor expansion. This converts the nonlinear problems into a much simpler approximation equation. With this approximation equation the study introduces two additional factors to handle any sources of error and issues:
1. A trust region is established to ensure there is a proper limit as to how far each iteration's solution can be from the current reference. This makes sure that the linearization stays true.
2. A virtual control term is established to ensure that despite how off the initial reference trajectory is, there would always be a solution to keep the loop going. This is done by allowing some violations in the process of making a solution. Within multiple iterations the need of the virtual control term should go to zero, as to notify that no violations in physics are needed for this solution.

This works in simplifying the initially hard nonlinear problem we had. However, the main fault in this method is that it relies on the fact that our initial reference trajectory should be sufficiently accurate. This is because with a poor initial guess, the linearization of the reference trajectory would lead to an approximation that is far from the solution and with a high error. This would mean a high amount of iterations using the two factors listed above, which would require extra time. This time could easily exceed one second.

To solve this the paper decides to incorporate AI. One way to do this is to allow AI to control the rocket, however this is actually more dangerous. This is because AI neural networks can be “black boxes” in which we don’t necessarily know what the AI is thinking and what it might do to the rocket at any time. The AI can be unpredictable with full control. Thus, a more safer approach is to allow the AI to generate a good initial reference trajectory and let the SCP to work from there.

To do this we need to train the neural network to learn how to make good initial reference trajectories. The study does this by allowing SCP to run and using a variety of randomized conditions. This includes varying the aircraft's starting place, its velocity, and its rotation. With this the study produces 48,333 trajectories, in which 45,000 are used to train the AI while 3,333 are used to test it. One of the key parts of the study is that instead of outputting the whole landing trajectory at once, the AI predicts each step one at a time. This means that the network predicts the rocket's next state and next control command for each time it takes to complete the trajectory. This sequential solving is helpful because it makes sure the AI is showing how each step blends with the next and how motion is naturally evolving.

With this we solve the problem of getting the trajectory of a rocket landing within a second. This method that the study shows has an around 40% less computational time. Thus, making it a good solution.

Adding onto this solution, to have these control systems work in rugged and harsh environments, there has been newer studies on using Gated Transformer networks as shown in papers like Jain et al., (2025). With these networks, we have the history of the rocket's path influence our final generated trajectory instead of only looking at each step of the flight independently. This allows the computer to adapt to changes in the environment, like wind or rough weather, without slowing down its own calculations.

Finite Element Methods

Finite Element Methods (FEM) refers to the methods of dividing a physical object into various small shapes, forming a mesh of elements. With this mesh it then solves for unknown field quantities, such as displacement and stress. It solves these quantities in places on the object known as nodes. Calculating all of these quantities requires time and computational cost. This cost is directly proportional to the DoF that each node has. In aerospace, this computational cost can be a problem. This is due to the sheer size of aircraft components, since a huge mesh is required to cover all parts of the component, causing elements in the order of millions. Another reason is due to the amount of times the calculation for each unknown field quantity must be done for various different conditions and parameters.

To solve this, we first use neural networks. In a neural network, there are weights and biases. Weights signal the network, telling which inputs are actually important. Whereas, biases are numbers that are added to the weights to shift it up or down. These two quantities are random numbers at first. In saying that, during training they can change until the output matches with the data desired.

FEM has nodes and in between these, there are shape functions. These functions are curves that connect to each node and dictate the characteristics of what happens between each node based on the value at the node. Usually, these shapes functions would have to be manually entered in order to provide the curve. However, using neural networks, we can generate these shape functions instead. This is done by tying the neural network parameters to the position of the nodes. This makes it so that the neural network behaves like a shape function. Thus, by training the neural network it would move the node position to reduce error. For example, if there are nodes that are evenly spaced on an object, the neural network can begin to inform where to add or remove nodes to better fit the object, so that it is more accurate. This is helpful for refining the mesh. However, the problem of computational cost is still present. This is further solved by using PGD. The idea of PGD is to split the 3D solution into a sum. Each term of the sum is the product of 1D functions in each direction (x, y, and z):

$$u^h(x, y, z) \approx \sum_{q=1}^{Q} u_x^{(q)}(x) \cdot u_y^{(q)}(y) \cdot u_z^{(q)}(z)$$

Each term is called a mode and Q is the amount of modes needed. With this equation the number of modes (Q) needed would be small.

To understand this concept we can imagine a room where we want to measure the temperature. We could find the temperature for each x, y, and z or we could multiply the patterns for each direction. This multiplication of patterns is called a mode. With the number of modes (Q) we can refine the temperature of the room. When Q is 1 we get the general temperature of the room without much detail. If there is a heater in the corner of the room we can add another mode (Q = 2) to add another pattern so that it accounts for the increased temperature in the corner. In reference to this problem, people can look at the example given in the study when compared to a full 3D mesh.

If N is the number of Nodes in one direction, the full 3D mesh is

$$1)\ (N \times N \times N) + N^3 = 2N^3 = \mathrm{DoF}$$

(an extra N^3 gets added to represent the node locations at each coordinate axis)

Then, by using the above equation, the answer would be:

$$2)\ Q \times 3N + 3N = \mathrm{DoF}$$

(add an extra 3N to represent the node locations at each coordinate axis)

Take N = 100 and Q = 5

$$1)\ 2N^3 = 2(100)^3 = 2000000\ DoF$$

$$2)\ 5 \times 3(100) + 3(100) = 1800\ DoF$$

By using PGD, we reduce the DoF drastically:
The above calculations are explained in greater detail in the paper done by Lu et al., (2021).
Then, combine these two methods by applying the shape functions created by the neural network to represent each 1D function in PGD. This means replacing variables, such as $u_x^{(q)}(x)$ from the PGD equation with the shape functions at that point. From this, the problem of having a high computational cost can be solved.

In FEM there exists another method that is used to reduce large complex math while keeping important physics behavior. This is done by making a smaller version of the model that still functions and behaves like the original. This is known as Reduced-Order Models (ROMs). Usually ROMs use a method called Proper Orthogonal Decomposition (POD). POD works by finding important patterns within the FEM simulation and keeps them while discarding the rest. These important patterns are also referred to as modes.

POD does indeed work well for simple behaviors. However, when the complexity of the model is high, meaning it is highly nonlinear or chaotic, then many modes are required to have sufficient accuracy. This means that PODs are held only to a certain extent. This problem was observed by researchers and was improved by introducing Neural Networks. This formed a method known as PROM-ANN. This works by splitting the modes into two parts. One part being Primary modes. While the other part became Secondary modes. The Primary mode captures large important patterns and the Secondary mode captures small, nonlinear details. By splitting these modes the neural network learns how the secondary modes depend on the primary modes. From this we are able to produce a reconstruction of the snapshot from the modes.

This has worked significantly and is often accurate. However there was one problem that still lingered, the neural network functions by trying to reconstruct the proper FEM from the modes. What we really want is to not just try to make a copy of the original snapshot but to also reduce the residuals of the snapshot. This is where Sibuet et al. (2025) formed a way to fix this problem.

First, it is important to understand what a residual is. It is a measure of how much error there is in a solution. With 0 Residual means that there is no error. Whilst a number not equal to 0 indicates an error. Then, the neural network is trained to look at patterns that it thinks look correct, and residual behavior to

see what changes make the solution correct or incorrect. This is easy to implement since FEM software already computes residuals. Therefore, the FEM solver can be plugged into the training loop. In saying that, one problem witnessed is that POD mode values can range drastically. They can range from being very large or small. This is a problem due to the nature that neural networks often ignore small values. To solve this issue, the paper discusses rescaling each mode. Thus, making everything around the same size. This can immensely improve performance, by having neural networks take in each value. With this new method, the study has tested it by using a rubber cantilever beam. This is a beam that is flexible with one end fixed while the other is loaded. This causes the beam to bend. From this situation, the results of the model displayed have improved simulation accuracy and computational cost.
It is important to realize that a rubber cantilever beam does not connect to aerospace components directly, however it can serve as a benchmark for the basic functions of an aerospace structure. An example of this is an aircraft wing, which is connected to a fuselage and experiences force from aerodynamic loads. Thus, the results of the test with the rubber cantilever beam can relate to structures we see in aerospace domains.
By fixing scaling, the error dropped a decent amount which is a slight, but consistent improvement for accuracy of the final output FEM.

This broader trend of using physics equations to reduce estimation errors from the AI is further supported by the recent development of FE-PINNs in papers such as Sunil and Sills (2024). These are similar to the residual checking discussed, since they use a neural network to obey rules to ensure that AI's designs are always physically accurate when engineers are checking aerospace structures.

Computer Vision

When a spacecraft is far from Earth, the signals that are sent to it for communication can take a long time to reach. This includes anywhere from several hours to a day. This is for high Earth orbit spacecrafts. This causes delays in the response time of the spacecraft. This poses a problem because if something went wrong with the spacecraft, whether it was internal or external, any communication to save it would not be received in time. This means that these situations would be handled by the spacecraft alone. The traditional way that was used to design new components included human engineers manually designing the component in software, such as computer-aided design (CAD). However, this method is slow and relies too much on having work be done on Earth and sending it to the spacecraft.

Ali et al., (2024) created a method that can address this problem of communication. This method includes having an automated system that uses cameras to scan physical objects on the spacecraft and build a 3D model from the images taken. With this 3D model, the spacecraft uses math to optimize the shape of the structure to improve its vibration. This is all done without having human input. This leads to a solution that can run in space to produce ready made components which can be printed. To capture the structure of the object which is desired, the spacecraft uses cameras. This is done by using a technique called photogrammetry. It is a technique that makes a 3D model by having a camera capture 2D images of the object from different angles. For example, mapping the moon's surface and studying Mars's landscape involve photogrammetry.

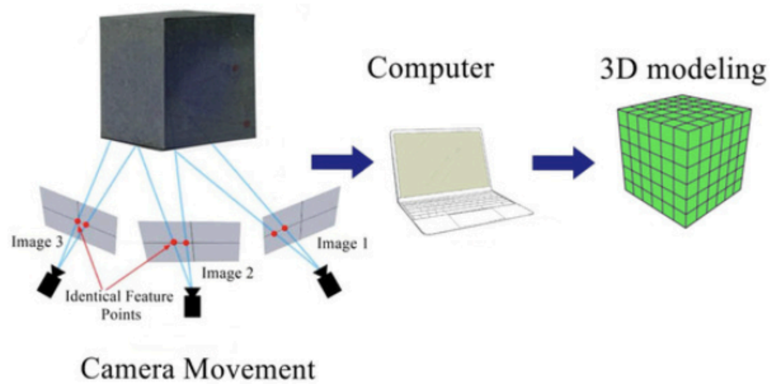

Figure 4: Process of making a 3D object using photogrammetry Ali et al., (2024)

From Figure 3, the setup has a camera that can go around the object to take a photo from every angle possible. Then, the software identifies matching points across these images and uses geometry to calculate the real-world 3D positions of each point. This produces a dense point cloud of the object's surface, which is converted into a mesh and saved. The important aspect here is that the system takes in its environment and makes its own design input automatically by using AI. Thus, in a real life scenario, a robotic system could scan available materials or existing structures and immediately begin the design process.

This method gives us the ability to see different components on our spacecraft. Adding onto this, it is equally as important to look at the bigger picture of the exact positions and orientations of spacecrafts in space. To do this we would need to utilize a different method of using AI

Another way in which we use AI in computer vision is in determining pose estimation. Pose estimation is the process of figuring out where a spacecraft is currently at and which direction it is facing in a given time (6 DoF). Knowing this is important in space because the location of a spacecraft helps in determining orbital rendezvous, docking, and debris capture. If the position of the aircraft isn't known, problems like crashing and missing might occur.

One may think that using the technology we use on Earth for location, like GPS, could just be applied to spacecraft as well. This statement is true, however it doesn't apply to all situations. Some satellites which are in Earth's low orbit do indeed use GPS to determine location, like the ISS. However, the main problem arises when dealing with precise spacecraft operations. When on Earth, GPS is usually accurate within 3-5 meters. While driving this accuracy is good enough to get where you need to go. In space, this 3-5 meters is a drastic difference, since crashes could easily happen within the 3-5 meters. Spacecraft operations, like docking, require centimeter accuracy.

This is why it is better to have technology on the spacecraft determine its own position. To do this we need to find a piece of technology that can function on a spacecraft, which is difficult. This is because spacecraft use little power, only around a few watts (~2 watts). Thus the question we want to find is. Is it possible to have a small commercial chip run AI fast enough to work in space with little power?

In a paper done by Leon et al. (2024) they utilized the Vision Processing Unit of the Intel Myriad X. This is a process that is able to handle images and AI. One key feature of this chip is that it has a Heterogeneous Architecture, this means that instead of having one processor handle all of the thinking it splits tasks between multiple specialized ones. One of these specialized processors is the Neural Compute Engine (NCE). The NCE is able to run deep neural networks and use them to find patterns in images and predict satellite positions. Generally having a neural network perform would take a lot of power. However, this unit was built specifically for AI. Thus, it is able to work a lot faster and with less energy.

According to research, using GPUs and FPGAs would prove to be another great solution. This would be true if the conditions were not restrictive. GPUs are powerful and would definitely give people faster results for efficient space operations. In saying that, the need for power and a cooler make it not ideal for space. FPGAs are flexible hardware that are used significantly in aerospace. Therefore, using it for this situation would make sense for space. However, due to the amount of time needed to input all the programming and work on the development, it falls short with the chip. The chip is a balance between the GPU and FPGA, which is often the best solution to have.

With this chip, two pipelines can be utilised. One pipeline is used to find the target spacecraft that it wants to know the location and position of. First, this pipeline takes a picture using a good camera to produce a clear picture, like a 1 megapixel RGB picture. Then, this picture is compressed to a smaller size in a process called downsampling. It is used so that less energy is needed to send and process the image. Next, the image is sent to an AI model which predicts its position and its orientation. This prediction is done through using a model called UrsoNet. From this, people can obtain the satellites x, y and z along with their rotation.

With the satellite spotted, computer vision is used to start tracking it using the second pipeline. Instead of switching to computer vision, AI takes a picture constantly and finds the position for every image. However, this method wastes a lot of energy and time for something that just needs tracking. This tracking that the second pipeline does uses:

1. Edge Detection
2. Depth Rendering
3. Edge Matching
4. Pose Refinement

This pipeline is much faster than the first pipeline since between each frame that the target satellite moves. The second pipeline only does small changes from the previous frame to track its position rather than starting from the beginning with its position.

This exact method of using multiple steps to track an object in space is seen throughout aerospace engineering. For example, in a paper by Chen et al., (2019), they had a much similar approach. They utilized a specialized AI model called HRNet (High-Resolution Network). This network first scans a photo to locate the key landmarks. After, it uses a geometric math program to calculate how the satellite will turn in space. Along with this, another paper by Phisannupawong et al., (2020) showed that with a single photo, an AI can figure out a satellite's complete position and tilt. Both of these methods demonstrate how utilizing advanced visual networks allows us to pinpoint accurately the position of spacecraft.

Cross domain Synthesis

From the four domains analyzed of topology optimization, finite element analysis, controls, and computer vision, a trend of how AI is best utilized in modern aerospace domains can be seen. This best method is having AI function as an "intelligent accelerator", instead of being a replacement for traditional computational methods. The fact that the internal thinkings of AI cannot be fully understood and predicted to a point where it can be completely trusted (described as a black box) would put aerospace systems at risk of getting false information that could lead to dangerous consequences. An example of this is seen when looking at 6 DoF powered landings, where the neural networks are limited to generating the initial reference trajectory, allowing the SCP to finalize the mathematics to create the landing pathway. If the final trajectory was trusted completely on AI, there is a greater chance that there could be a mistake which would lead to a crash. In a similar sense, when looking at topology optimization with micropolar elasticity, a classical algorithm is used for the initial 10-20% of iterations to develop the main skeleton while the ML model computes the remaining 80-90% to reduce the high computational costs.

By looking at these domains for how AI is being used, there is also a visible difference between operations that are not in real-time (FEM and Topology Optimization) and real-time operations (Controls and Computer Vision). Within operations that are not in real time, AI is used in reducing the complexity of calculations in order to obtain a complex FEM mesh. While for real-time operations in environments such as space, there are more physical limits that the AI has to navigate around. A clear example of this is when looking at computer vision in space environments, where the computer vision chips only have around 2 watts of power to utilize. Furthermore, for rocket control systems there is a time restriction of under one second to calculate the math for a landing to avoid crashing. In these real-time operations the AI is being used more than just for saving time on a computer, but rather making it possible to have a spacecraft function by itself.

With these in mind it is also critical to look at the setbacks that each of the four areas have in common. One of the biggest setbacks that is seen is the massive amounts of data to train the AI. These datasets can have a huge range depending on the model, like 100,000 samples to train topology optimizations and 45,000 different trajectories to teach AI how to form a good flight path for a rocket to land. This setback is a challenge for many AI models as to get an accurate model we would need to put more data, thus reducing data would reduce accuracy, which as seen in space like environments can be catastrophic. One solution that was used for this issue was the PEN being utilized in topology optimization, which allowed the AI to test itself by generating samples for it to process and evaluate its own accuracy based on rules. This reduces the need for finding pre-existing data. Learning from this, having similar self evaluating models in domains like FEM and controls could give a solution for this data training across aerospace systems.

CONCLUSION

From the research about AI assisted computational methods for aerospace system design, it clearly demonstrates that by integrating AI in traditional engineering methods, people are able to improve computational time, computational cost and accuracy. This was witnessed from studies about different fields, such as Finite Element Method, control systems, computer vision and topology optimization.

In terms of topology optimization, it can be observed that by incorporating AI, individuals have the option to get rid of the need to have data to train the AI on and instead use PEN. Which has shown a 259x speed up during inference. This saves the time that is needed to find data to train the AI on and it does not put a limit in regards to how much people can train the AI. Furthermore, with the implementation of AI, businesses are able to replace classical elasticity with micropolar elasticity, whilst still being efficient with computational time and cost. This is done by having MLs do 80 to 90% of the remaining computation after the initial density field is made. This approach maintains efficiency while having an improvement in structural stiffness by up to 38%.

The study has stressed that the Finite Element Method is improved by AI to be able to handle large sizes of components, without requiring high computational cost and time. This was achieved by utilizing neural networks to find the best ways to reduce the complexity of the FEM. This is seen when using PGD, which can reduce the complexity of a 3D mesh from 2,000,000 DoF to just 1,800 DoF. Specifically, this is done by having the weights of the neural network tied to nodal positions of an FEM, and by having AI detect patterns to split for reduced complexity.

Similarly, control systems benefit from AI by acting like a helper rather than a complete replacement. For tasks like 6 DoF powered landings, AI is used to generate the initial trajectory for SCP to build off of. This resulted in a 40% reduction in computational time, to ensure that the calculations are done fast enough to ensure the trajectory is made to safely land.

Furthermore, AI in computer vision allows for improved efficiency in getting information and performing tasks. By incorporating AI, individuals are able to use cameras to take pictures of spacecraft components, and have the AI create a model of the part to send back to Earth easily. Chips can be utilized that have an AI component in them in order to have a spacecraft track other spacecrafts. Therefore, the evidence that has been collected supports the hypothesis that AI-assisted computational methods help improve efficiency in aerospace system design.

Taking this information, it can be said that learning deeper about how AI can further be used in Aerospace system design would be incredibly rewarding. This is because it allows people to get calculations for situations that were seen as too complex to solve. Future researchers should consider not having AI perform entire calculations and do everything by itself. This is due to the fact that AI is still not completely reliable and can be prone to error, if not trained correctly.

Overall, AI integration in aerospace system design is improving current engineering methods and makes a foundation for innovations which may come in the future for how system design is made and tested.

References

[1] Ali, M., Shimoda, M., & Naguib, M. (2024, June 2). Toward Automated Structural Design for Controlled Vibration Characteristics Using Topology Optimization and Computer Vision in Space Missions. MDPI; MDPI. https://www.mdpi.com/2076-3417/14/15/6786

[2] Chen, B., Cao, J., Parra, A., & Chin, T.-J. (2019, August 30). Satellite Pose Estimation with Deep Landmark Regression and Nonlinear Pose Refinement. CVF; Computer Vision Foundation. https://openaccess.thecvf.com/content_ICCVW_2019/papers/R6D/Chen_Satellite_Pose_Estimation_with_Deep_Landmark_Regression_and_Nonlinear_Pose_ICCVW_2019_paper.pdf

[3] Halle, A., Campanile, L., & Hasse, A. (2021, September 27). An Artificial Intelligence–Assisted Design Method for Topology Optimization without Pre-Optimized Training Data. MDPI; MDPI. https://www.mdpi.com/2076-3417/11/19/9041

[4] Jain, A., Rodriguez-Fernandez, V., & Linares, R. (2025, November 14). Multi-Phase Spacecraft Trajectory Optimization via Transformer-Based Reinforcement Learning. Arxiv; arxiv. https://arxiv.org/html/2511.11402v1

[5] Jung, K.-W., Lee, S.-D., Jung, C.-G., & Lee, C.-H. (2024, May 2). Model Predictive Guidance for Fuel-Optimal Landing of Reusable Launch Vehicles. Arxiv; arxiv. https://arxiv.org/html/2405.01264v1

[6] Kiyik, H., Kaya, M., & Mahouti, P. (2026, April 3). Analysis of Wing Structures via Machine Learning-Based Surrogate Models. MDPI; MDPI. https://www.mdpi.com/2226-4310/13/4/338

[7] Leon, V., Minaidis, P., Lentaris, G., & Soudris, D. (2024, September 19). Accelerating AI and Computer Vision for Satellite Pose Estimation on the Intel Myriad X Embedded SoC. Arxiv; arxiv. https://arxiv.org/pdf/2409.12939

[8] Lu, Y., Li, H., Saha, S., & Mojumder, S. (2021, November). Reduced Order Machine Learning Finite Element Methods: Concept, Implementation, and Future Applications. Researchgate; Tech Science Press. https://www.researchgate.net/publication/354640767_Reduced_Order_Machine_Learning_Finite_Element_Methods_Concept_Implementation_and_Future_Applications

[9] Phisannupawong, T., Kamsing, P., Torteeka, P., Channumsin, S., Sawangwit, U., Hematulin, W., Jarawan, T., Somjit, T., Yooyen, S., Delahaye, D., & Boonsrimuang, P. (2020, August 30). Vision-Based Spacecraft Pose Estimation via a Deep Convolutional Neural Network for Noncooperative Docking Operations. MDPI; MDPI. https://www.mdpi.com/2226-4310/7/9/126

[10] Pierson, K., Gillespie, J., Ha, M., & Stuckner, J. (2025, May 30). Structural Optimization of Turbomachinery Rotors: A Study of Machine Learning Surrogate Models. NTRS - NASA Technical Reports Server; Glenn Research Center. https://ntrs.nasa.gov/citations/20250005698

[11] Prous, G., Grustan-Gutierrez, E., & Felicetti, L. (2025, January 31). A Decreasing Horizon Model Predictive Control for Landing Reusable Launch Vehicles. MDPI; MDPI. https://www.mdpi.com/2226-4310/12/2/111

[12] Rasulzade, J., Rustamov, S., Akhmetov, B., Maksum, Y., & Nogaibayeva, M. (2022, December 29). Computational Acceleration of Topology Optimization Using Deep Learning. MDPI; MDPI. https://www.mdpi.com/2076-3417/13/1/479

[13] Shen, Z., Zhou, S., & Yu, J. (2022, October 14). Real-time computational powered landing guidance using convex optimization and neural networks. Arxiv; arxiv. https://arxiv.org/pdf/2210.07480

[14] Sibuet, N., Ares de Parga, S., Bravo, J., & Rossi, R. (2025, May 19). Discrete Physics-Informed Training for Projection-Based Reduced-Order Models with Neural Networks. MDPI; MDPI. https://www.mdpi.com/2075-1680/14/5/385

[15] Sosnovik, I., & Oseledets, I. (2017, September 27). Neural networks for topology optimization. Arxiv; arxiv. https://arxiv.org/abs/1709.09578

[16] Sunil, P., & Sills, R. (2024, December 10). FE-PINNs: finite-element-based physics-informed neural networks for surrogate modeling. Arxiv; arxiv. https://arxiv.org/pdf/2412.07126

[17] Tang, H. (2025). A Review of Aircraft Aerodynamic and Structural Optimization Method Based on Artificial Intelligence. Science, Engineering and Technology Proceedings, 4, 59-68. https://cpcig-conferences.com/index.php/SETP/article/view/87

[18] Wang, J., Cui, N., & Wei, C. (2019, January 29). Optimal Rocket Landing Guidance Using Convex Optimization and Model Predictive Control. American Institute of Aeronautics and Astronautics; Aerospace Research Central. https://arc.aiaa.org/doi/10.2514/1.G003518

[19] Wu, R., Kovachki, N., & Liu, B. (2025). A Learning-based Domain Decomposition Method. Arxiv; arxiv. https://arxiv.org/abs/2507.17328

[20] Zhou, H., Shaat, M., & Gao, X. (2025). Machine learning-assisted topology optimization framework for designing 2D structures based on micropolar elasticity | Acta Mechanica | Springer Nature Link. SpringerLink. https://link.springer.com/article/10.1007/s00707-025-04380-z