ASME Journal of Mechanical Design
Online journal at:
https://asmedigitalcollection.asme.org/mechanicaldesign

Soheyl Massoudi[1]
Laboratory for Applied Mechanical Design,
Ecole Polytechnique Fédérale
de Lausanne (EPFL),
CH-1015 Lausanne, Switzerland
e-mail: smassoudi@ethz.ch

Joseph Bejjani
Laboratory for Applied Mechanical Design,
Ecole Polytechnique Fédérale
de Lausanne (EPFL),
CH-1015 Lausanne, Switzerland
e-mail: joseph.bejjani@epfl.ch

Timothy Horvath
Laboratory for Applied Mechanical Design,
Ecole Polytechnique Fédérale
de Lausanne (EPFL),
CH-1015 Lausanne, Switzerland
e-mail: timothy.horvath@epfl.ch

Dogukan Üstün
Laboratory for Applied Mechanical Design,
Ecole Polytechnique Fédérale
de Lausanne (EPFL),
CH-1015 Lausanne, Switzerland
e-mail: dogukan.ustun@epfl.ch

Jürg Schiffmann
Laboratory for Applied Mechanical Design,
Ecole Polytechnique Fédérale
de Lausanne (EPFL),
CH-1015 Lausanne, Switzerland
e-mail: jurg.schiffmann@epfl.ch

DARTS-NETGAB: Design Automation and Real-Time Simulation Using Neural Network Ensembles for Turbocompressors on Gas-Bearings

DARTS-NETGAB is a unified framework for real-time simulation and automated design of gas-bearing supported turbocompressors, facilitating efficient transition from optimization to manufacturable designs. The framework integrates ensemble artificial neural networks (EANNs) trained on high-fidelity simulation data to predict performance metrics—including isentropic efficiency, pressure ratio, and rotordynamic stability—across various operating conditions and manufacturing tolerances. A user-friendly interface using Panel-Bokeh libraries allows dynamic design modifications and immediate visualization. The Paraturbo-CAD library automates the generation of detailed 3D computer-aided design (CAD) models from optimized design parameters. The surrogate models maintained prediction errors below 5% for isentropic efficiency and pressure ratio in most conditions, with errors up to 11% near choke limits. Real-time simulations were efficient, averaging 1 s for coarse discretization (6195 points) and 8.5 s for fine discretization (311,250 points). Automated CAD generation produced manufacturable 3D models in approximately 7 min per model, successfully translating optimized designs into detailed geometries suitable for production. DARTS-NETGAB enhances the efficiency and accuracy of the turbocompressor design process by unifying rapid performance prediction with automated CAD model generation. This integration enables rapid iterations and robust assessments of design sensitivity to manufacturing imperfections, addressing a critical gap in transitioning from optimization to practical, manufacturable designs. [DOI: 10.1115/1.4068091]

Keywords: surrogate models, real-time simulation, artificial neural networks, ensemble learning, parametric CAD, graphical user interface, Bokeh Python library, herringbone grooved journal bearings, gas-bearings, microturbomachinery, integrated design, robust design, machine learning in engineering, multidisciplinary design optimization

1 Introduction

1.1 Nature of the Issue. The engineering design process commences with the interpretation of technical specifications into a functional decomposition [1,2], which is then followed by the development of a morphological matrix to explore potential subsystems [3–5]. The chosen subsystems are modeled at various levels of fidelity, where surrogates often replace high-fidelity models to expedite the optimization process [6]. With the escalation of design complexities, comprehensive strategies such as multidisciplinary design optimization become indispensable [7–9]. Following multi-objective optimization, there is a necessity to convert solutions identified on the Pareto front into computer-aided design (CAD) models through an automated approach [10]. Despite the recent advances in enhancing the integration, efficiency, and automation of engineering workflows [11,12], there is still a gap in reliably converting an optimized solution into a practical 3D model.

The journey from identifying optimal solutions on the Pareto front to realizing manufacturable 3D models is a pivotal phase in the engineering design process. This phase often encounters hurdles such as increased costs and extended timelines due to the need for multiple design iterations to refine the initial solutions. Real-time simulation, as advocated by Odot et al. [13], presents a vital solution by facilitating swift and efficient model adjustments. Yet, the utility of such simulations extends beyond speed; they must also adapt broadly across the design spectrum to be genuinely transformative. Tools like Dash-Plotly offer interactive environments that could, theoretically, streamline this iterative phase by enhancing user engagement and integration into the design workflow.

[1]Corresponding author.

Contributed by the Design Automation Committee of ASME for publication in the Journal of Mechanical Design. Manuscript received October 2, 2024; final manuscript received February 16, 2025; published online April 3, 2025. Assoc. Editor: Chao Hu.

However, their potential is rarely realized in practice [14], with their use often limited to interactive plotting rather than full-scale computational analysis [15]. Addressing this underutilization could significantly refine the process of transitioning from Pareto-optimized solutions to finalized designs, marking a progressive step in design methodology.

The necessity of a 3D model for manufacturing is undisputed, especially for automated processes like laser engraving and five-axis computer numerical control machining through computer-aided manufacturing software packages. Beyond facilitating these processes, it serves as the foundation for technical drawings and the assignment of manufacturing tolerances essential for production technicians. However, transitioning from an optimized solution to a tangible 3D model is complex. Numerical models employed during optimization are often of lower fidelity to expedite the process, leaving the comprehensive modeling of the engineering system pending. In such instances, designs are typically refined using specialized CAD software, often involving multiple stakeholders, which can necessitate iterative adjustments or even reoptimization to meet revised constraints. This multidisciplinary, software-shifting process increases the risk of errors.

Recent explorations into artificial intelligence for direct CAD modeling have presented several approaches. Working directly with boundary representations offers one route [16–18], although they are still limited in the availability of the datasets [19] and the reconstruction of complex topologies. Alternatively, generative models like generative adversarial networks or variational autoencoders (VAEs), which reconstruct 3D geometries from 2D images, have shown promise. However, these methods often require design heuristics for 2D to 3D mapping [20] and postprocessing to eliminate AI-generated artifacts as they do not always yield perfectly accurate CAD reconstructions [21]. A novel approach involves directly manipulating CAD commands leveraging large datasets, using transformers and decoders [22,23]. This method accurately generates geometries by creating executable CAD commands within a genuine CAD kernel. The latest advancements in the field have developed a method that abstracts CAD model generation through a unique combination of transformers and VAEs. This approach, known as hierarchical neural coding, introduces a three-tier embedding into codebooks of CAD, allowing for a novel and controllable generation of CAD models [24]. Despite their potential, the effectiveness of transformers in direct CAD modeling relies heavily on the availability of large and diverse datasets. Methods have been developed to parse CAD instructions from existing B-Rep datasets [25]. At present, these datasets predominantly encompass 2D sketch extrusions, rendering them inadequate for complex CAD operations like lofting through cross sections—a technique essential for designing complex structures such as impeller blades or aircraft fuselages. This limitation significantly hampers their ability to accurately learn and replicate more sophisticated geometries.

The demand for a streamlined workflow that transitions from Pareto-optimal solutions to fully realized 3D models—supported by rapid visualization and assessment—is increasingly evident in contemporary design processes. While commercial platforms such as Siemens HEEDS, ANSYS Workbench, and Altair HyperStudy offer integrated environments for design exploration and optimization, they typically rely on external tools for parametric CAD modeling and often require manual oversight during geometry updates. This can slow the iterative cycle, especially when large or complex design changes are involved. Furthermore, although these platforms can automate simulation-driven studies, they generally do not provide real-time performance computations or fully interactive design modifications—capabilities that are crucial when manufacturing tolerances are tight or rapid design iterations are required. Recent developments, such as live links between CAD and computer-aided engineering (CAE) software (e.g., Dassault Systemes SIMULIA Isight, ESTECO modeFRONTIER) and specialized real-time solvers (e.g., ANSYS Discovery Live, Altair Inspire), partly address these limitations but remain supplemental rather than universally embedded in standard workflows. Likewise, integrated generative design tools (e.g., Autodesk Fusion 360, PTC Creo Generative Design) enable quicker geometry development, yet still face challenges in delivering high-fidelity, immediate feedback for more complex physics.

1.2 Goals and Objectives. The primary objective of this study is to develop and validate a comprehensive framework (DARTS-NETGAB) designed for the efficient and accurate analysis and modeling of gas-bearing supported turbocompressors. The framework is aimed at achieving the following key goals:

(1) *Unified framework development*: Create a unified workflow that integrates simulation and automated CAD generation to enhance the efficiency and accuracy of the design process, significantly reducing time-to-market.
(2) *Graphical user interface (GUI) design*: Develop an intuitive, user-friendly GUI that leverages advanced visualization tools for dynamic and interactive design adjustments, aimed at improving user engagement and decision-making efficiency.
(3) *Real-time simulation capability*: Implement an advanced simulation engine using artificial neural networks (ANNs) to provide real-time feedback for iterative design improvements.
(4) *Automated CAD generation*: Automate the generation of accurate and manufacturable CAD models directly from optimized design parameters, aiming to streamline the translation from design to production.

These goals are designed to collectively enhance the flexibility of the design process and accelerate the transition from concept to manufacturing readiness, positioning DARTS-NETGAB as a transformative and streamlined tool in engineering design.

1.3 Scope of the Article. This article introduces DARTS-NETGAB, a framework for designing gas-bearing supported turbocompressors. The framework leverages ensemble artificial neural networks (EANNs) for rapid performance predictions and integrates automated CAD generation for efficient design-to-manufacture transitions.

Section 2 outlines the framework's architecture, including its web-based interface, performance computation, and CAD generation. Section 3 describes the methodology, detailing case studies and the development of surrogate models. Section 4 evaluates the results through accuracy testing and real-time simulation benchmarks. Section 5 discusses the framework's contributions and implications, which is followed by a conclusion.

2 Architecture of the Framework

The DARTS-NETGAB framework offers a user-centric interface using interactive PYTHON libraries for real-time visualizations and dashboard functionality, streamlining the design-to-manufacture process. Users can adjust design parameters and operating conditions, prompting the system to compute performance maps and generate CAD models. The process culminates in the production of standard for the exchange of product model data (STEP) files for manufacturing and standard tessellation language (STL) files for visualization, embodying a cohesive solution for rapid design iteration.

2.1 Interactive Web-Based Interface. The DARTS-NETGAB framework integrates a user-centric interface, streamlining the turbocompressor design and manufacturing process. Bokeh facilitates interactive, JavaScript-powered visualizations right in the browser, enabling users to craft complex performance maps without deep diving into JavaScript coding [26]. Panel, part of the HoloViz ecosystem, extends this functionality, allowing for the construction of comprehensive dashboards and applications in PYTHON. This integration offers users an intuitive platform for:

- Modifying design parameters: Users have the capability to adjust key design elements and operating conditions, such as geometries, structural components, and environmental factors. These adjustments directly impact the system's overall design and operational efficiency.
- Visualizing performance and robustness: The framework offers advanced visualization tools for analyzing performance metrics and system robustness. This feature is crucial for identifying potential areas of improvement and understanding the effects of manufacturing variances in system functionality.
- Generating CAD models: Using a specialized library, users can convert detailed system designs into comprehensive 3D models in formats like STL/STEP. This integration streamlines the process from design optimization through to the manufacturing phase, enhancing workflow efficiency across various engineering domains.

This cohesive environment offers a blend of interactive visualization, real-time parameter adjustment, and direct CAD generation capabilities.

2.2 Performance Computation. To swiftly generate performance maps, our framework employs EANNs as a core computational engine. These ANNs effectively replace traditional, computationally intensive models, streamlining robustness evaluations for radial and axial bearings and overall compressor performance. Integrated seamlessly with the DARTS-NETGAB web interface, these networks activate upon user interactions, processing inputs such as geometry, materials, and operating conditions through interactive controls like widgets and sliders.

The ANN ensembles within the framework are rigorously trained on extensive datasets produced from high-fidelity models, equipping them with the ability to provide accurate and swift predictions for a variety of engineering systems [27,28]. Input parameters, encompassing system geometries and material properties, are normalized to create dimensionless groups that are representative of the different subsystems' characteristics. These processed inputs are fed into the ANNs, which then predict standardized outputs. These outputs are subsequently translated into dimensional forms suitable for a range of interpretative metrics relevant to the system's performance.

This process culminates in the dynamic update of performance visualizations within the interface, providing users with immediate feedback on design alterations. This integration not only underscores the framework's capability for rapid performance evaluation but also highlights the synergy between advanced computational techniques and user-friendly interfaces in modern engineering design tools.

2.3 Computer-Aided Design Generation. The CAD generation component of our framework is enhanced by CadQuery 2, an advanced PYTHON library dedicated to constructing parametric 3D CAD models. CadQuery 2 is designed to (1) leverage the high-level PYTHON language for scriptable model building, enabling users to grasp the creation process easily, (2) facilitate the crafting of customizable parametric CAD models, and (3) support the output of CAD formats such as STEP, in addition to the more traditional STL.

2.4 Workflow Integration. The workflow of the DARTS-NETGAB framework is described in Fig. 1. It begins when users initiate the web interface, typically by running a command such as "panel serve—show DARTS-NETGAB.py" from their terminal. Upon loading their rotor design in either .obj or .mat format, or constructing it within the interface, the framework's controls for geometry and operating parameters automatically update to reflect the new configuration, if a .obj file created with the framework is used. This ensures that performance predictions are accurately tailored to the current design.

As users adjust controls within the interface, the changes are captured and processed into PYTHON dictionaries containing material properties, subsystems geometries and locations, and operating conditions. These data undergo a transformation process—it is made dimensionless and standardized—before being fed into the ANN ensembles. These ensembles swiftly compute performance metrics, which are then reverse-standardized and re-dimensionalized for parameters such as load capacity and losses. The framework processes these metrics to update the visualization in real-time, providing immediate feedback on the performance impact of design modifications.

When users initiate the CAD generation process, the framework, using CadQuery 2, responds to the command by reading the specified design parameters. It then methodically constructs and assembles the subsystems of the engineering system into a comprehensive model. The finalized model is prepared as an STEP file, ready for manufacturing processes, and concurrently, it is meshed as an STL file for visualization purposes. This meshed model can be visualized in detail within the interface, which harnesses the capabilities of the visualization toolkit (VTK) library for rendering.

The framework offers an integrated environment for the design of engineering systems, spanning from the initial visualization of performance metrics to the generation and export of manufacturable CAD models. This comprehensive setup streamlines the workflow, enhancing efficiency for designers and engineers alike by seamlessly bridging the gap between conceptualization and production.

2.5 Technology Stack. The DARTS-NETGAB framework employs PYTHON 3.9.17 for its core programming needs, complemented by Bokeh 3.3.2 and Panel 1.3.4 for dynamic data visualization and web interface development. It leverages TensorFlow 2.10 for sophisticated machine learning tasks and ParaturboCAD[2] built on top of CadQuery 2.3.1 for accurate 3D CAD modeling. Data processing and numerical computations are adeptly handled by pandas 2.1.4 and NumPy 1.26.2. The framework's CAD modeling and visualization capacities are further enhanced by cqMore 0.1 and VTK 9.3.0, with PyVista 0.43 providing advanced 3D visualization features. Additionally, for superior STL meshing, DARTS-NETGAB integrates trimesh 4.0.5 and gmsh 4.11.1, ensuring high-quality mesh generation essential for detailed analysis.

3 Methods

3.1 The Case Study: A Gas-Bearing Supported Microcompressor for Heat Pump Applications. Gas-bearing supported turbocompressors yield key characteristics such as high power density, low mechanical losses, adaptability to high-speed rotation, and a high expected lifetime. These features minimize maintenance intervals and enhance overall system efficiency [29]. Typical applications include blowers for fuel cells [30], turbines for waste heat recovery [31], or compressors for heat pumps [32].

A typical turbocompressor spindle system is shown in Fig. 2. The key system is the centrifugal compressor impeller (COMP1) which provides the pressure rise. To do so, it needs power, that is provided from a brushless direct-current motor to the magnet (MAG). The power or torque is then transmitted through a rotor shaft (ROT). To rotate, the rotor shaft needs to be supported by radial bearings. Here, gas-bearings are used which are characterized by thin film gas lubrication and V-shape engraving of the journal part of the rotor shaft, so-called herringbone grooved journal bearings (HGJBs) [33]. Finally, to ensure axial alignment and load balancing, a pump-in spiral groove thrust bearing (SGTB) is used.

Beyond being composed of multiple systems, it is also multidisciplinary. Aerodynamics is considered in the computation of the compressor performance. Lubrication theory drives the fluid–structure interaction modeling the axial and radial gas-bearings. Rigid rotordynamics are considered to estimate the dynamic stability of the system as well as flexible dynamics to ensure operation below the first lateral bending eigenfrequency. Structural considerations

[2]https://github.com/SoheylM/paraturbo-cad

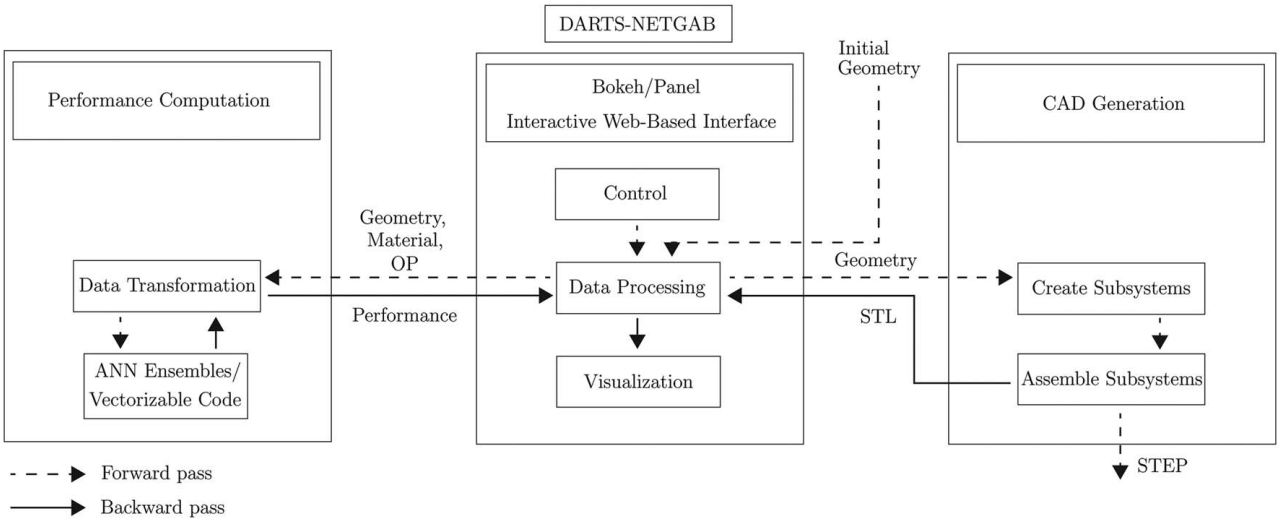

Fig. 1 DARTS-NETGAB framework architecture: a PYTHON-based interactive tool leveraging Bokeh and Panel for design and performance evaluation of engineering systems. It allows users to modify parameters, evaluates performance, and generates CAD models, producing STEP for manufacturing and STL for visualization, streamlining design and production processes.

are also taken into account with respect to centrifugal expansion and stresses. Furthermore, its millimetric scale compared to the micrometric clearance and groove depth of the bearing makes it particularly challenging to manufacture. Furthermore, the stability of the system is highly dependent on a tight tolerancing of the local bearing clearance.

To validate the framework, four turbocompressors were selected from four distinct all-at-once robust multidisciplinary design optimizations per the methodology described by Massoudi et al. [12]. They represent heat pump compressors functioning with vapor phase R134a, delivering the same pressure ratio but for increasing mass flows ranging from 5 g s^{-1} for T1 to 75 g s^{-1} for T4. These machines exhibit variations, including the number of impeller blades, the design of the SGTB, the HGJBs, and the rotor size.

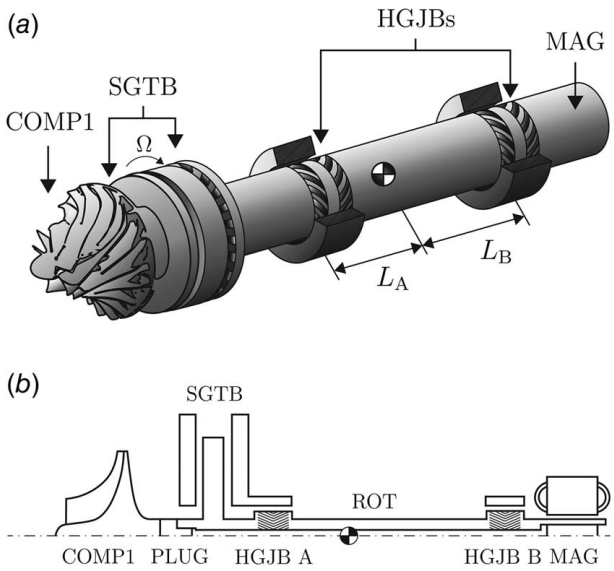

Fig. 2 Schematic diagrams illustrating a possible layout and arrangement of the turbocompressor spindle, including the positions of the centrifugal compressor impeller (COMP1), the plug (PLUG) connecting it to the rotor (ROT), the HGJB A and HGJB B, the SGTB, and the magnet (MAG): (*a*) 3D diagram of the turbocompressor spindle and (*b*) 2D axisymmetric projection showing the nomenclature of the subsystems

3.2 Bokeh Web Interface Implementation. DARTS-NETGAB provides an interactive and intuitive interface, leveraging advanced web-based visualization and dashboard-building capabilities. This interface enables seamless integration of design modification, performance evaluation, and CAD generation in a unified environment tailored for turbocompressor design.

The interface is structured into five distinct tabs highlighted in Fig. 3, each addressing critical aspects of the design process:

- Overview tab (1): Facilitates the loading of rotor designs and the adjustment of key operating conditions. Users can define material properties, rotor geometry, and performance ranges, with visual indicators highlighting regions of extrapolation to ensure accurate predictions.
- Compressor tab (2): Allows detailed exploration of compressor performance, visualizing isospeed lines, mass flowrates, and pressure ratios. Color-mapping techniques encode efficiency metrics, providing clear insights into performance trends.
- HGJB and SGTB tabs (3 and 4): Provide tools for evaluating bearing performance metrics such as stability, load capacity, and losses. Dynamic contour plots visualize the impact of variations in groove depth and bearing clearance, aiding in the identification of optimal configurations.
- CAD generation tab (5): Streamlines the transition from conceptual design to 3D models. Leveraging the ParaturboCAD library, this tab generates detailed STEP and STL files, visualized using PyVista.

3.3 Development and Implementation of Artificial Neural Networks. To facilitate real-time simulations within the DARTS-NETGAB framework, particularly for robustness computations and comprehensive compressor mapping, the deployment of fast surrogate models is crucial. For this case, the performance metrics to be predicted span across several components:

- For the rotor, predictions include windage losses, lateral and axial rigid dynamics, flexible dynamics, and structural stresses.
- For HGJBs and SGTB, stiffness and damping matrices, load capacity, and losses are evaluated.
- The electric motor (EM) assessments cover torque and losses.
- Compressor evaluations consider isentropic efficiency, pressure ratio, and operational state (functioning or not).

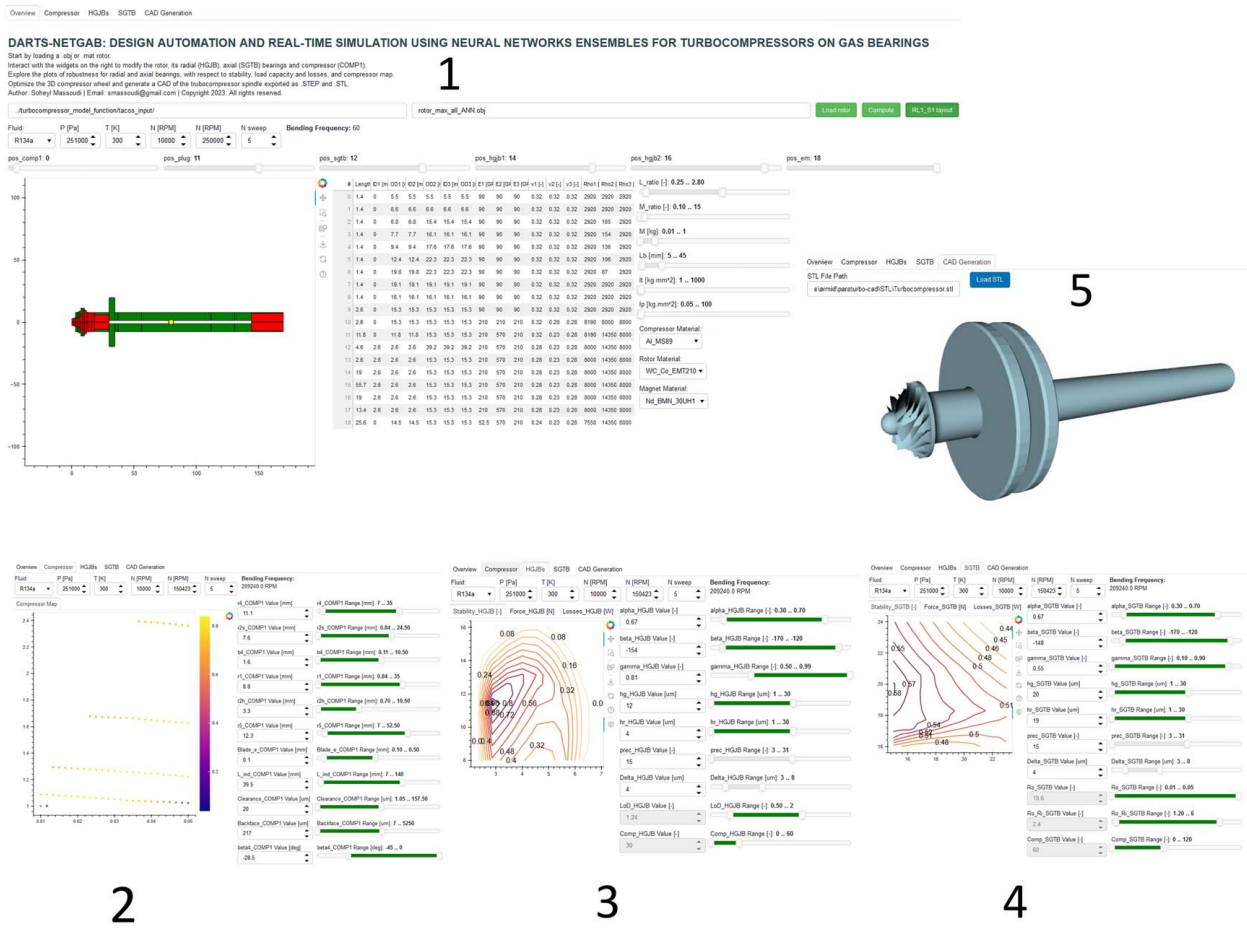

Fig. 3 DARTS-NETGAB interface overview. Developed in PYTHON with the Bokeh and Panel libraries, DARTS-NETGAB is an interactive, browser-based application. It features five distinct tabs for comprehensive rotor analysis and design. Tab (1) enables users to upload rotor data, adjust geometry, material properties, and operating conditions. Tab (2) facilitates the selection of compressor wheel parameters and visualizes compressor maps. Tabs (3) and (4) are dedicated to modifying bearing geometries, and assessing their stability, load capacity, and loss calculations. Tab (5) provides tools for generating and visualizing the turbocompressor spindle in a detailed 3D format.

While some baseline models, like those analyzing windage and bearing losses, are amenable to vectorization and effectively scale for computations across multiple samples, others face inherent limitations. Specifically, models dealing with aerodynamics, bearing stiffness, damping, and overall system dynamics yield computational bottlenecks. This is particularly evident in the comprehensive characterization of compressor maps, which demand analyses across varied mass flows and rotational speeds. Additionally, conducting sensitivity analyses to account for manufacturing deviations—such as variations in groove depth and bearing clearance for HGJBs and SGTB—places significant demands on computational resources.

To address these challenges, surrogate models have been developed, targeting not only compressor performance but also the underlying bearing and dynamics models. These models are designed to directly output system dynamics based on inputs like bearing and rotor geometries and operating conditions, eliminating the need for intermediary modeling steps and thus streamlining the computational process, enabling rapid computations that leverage the parallel processing capabilities of graphics processing units (GPUs). This approach allows for complex simulations that would traditionally require extended computation times to be executed swiftly. This streamlined performance is attributed to the model's reliance on matrix multiplications and activation functions—operations that are expedited on GPUs leveraging Nvidia's CUDA library. The only limiting factor in this process is the GPU's resources, specifically the number of CUDA and Tensor cores and the available video random access memory (VRAM). Provided these resources are not fully used, the computation time remains constant regardless of the batch size, highlighting the surrogate model's scalability and efficiency in handling large datasets.

Data are sampled as tabular paired input–output from the high-fidelity baseline models using a combinatorial sampling technique to map a broad range of operating conditions within the vapor phase regions of refrigerant working fluids, air, and steam. This ensures that critical thermodynamic properties, such as viscosity, are captured accurately. Inputs were selected to comprehensively represent the system's physical and operational characteristics, including dimensional parameters such as compressor, axial bearing, herringbone grooved journal bearing, and rotor geometries, along with rotational speed and mass flow. Outputs were chosen to capture essential performance metrics aligned with the framework's objectives, such as rotor dynamics (e.g., stability indicators), bearing properties (e.g., load capacity, losses), and compressor performance (e.g., isentropic efficiency, pressure ratio, and operational state). To ensure consistency and applicability across scenarios, inputs and outputs were nondimensionalized, removing biases from physical units and improving generalizability. Sampling included two Latin hypercube processes applied to dimensional and dimensionless geometries, generating datasets in the order of millions for the axial and lateral dynamics surrogate models and tens of millions for the compressor surrogate model. These datasets formed the basis for training feed-forward neural networks,

enabling the surrogate models to predict system behavior with high fidelity and efficiency.

Hyperparameter tuning of surrogate models is a critical step in their development. To optimize the performance of the artificial neural networks used to model the various outputs, each ANN is trained via gradient descent within a genetic algorithm loop [27,34]. The hyperparameters that govern the training process are chosen as decision variables to be optimized as presented in Table 5 located in the Appendix A. Hyperparameter tuning can be computationally intensive, and we use a genetic algorithm with a total of 5 epochs and an initial population size of 100 to efficiently search the hyperparameter space. Two types of ANNs are trained: regressors, which predict continuous outputs such as the isentropic efficiency of the compressor, and classifiers, which predict categorical variables such as the rotordynamics stability Γ of a given design or the functioning state of a compressor (surge, normal, choke). The choice of loss function depends on the type of output being predicted and includes mean squared error or mean absolute error for regressors, and categorical cross-entropy for classifiers. Classifiers are trained with larger batch sizes than regressors. The data are split into training, validation, and test sets in 86%, 8%, and 8% of the sampled dataset, respectively.

To enhance the precision and reliability of our surrogate model predictions, we employed an ensemble approach, training six variations of the optimally configured artificial neural network. Each variant of the ANN was initialized with different weight initialization strategies, specifically He Normal, Lecun Normal, Glorot Uniform, He Uniform, Lecun Uniform, and Glorot Normal [35–37]. This approach ensures diversity among models by leveraging the sensitivity of neural network training to initial weights, with each model exploring a different local minimum or saddle point in the loss landscape. The aggregate prediction of our model is derived from averaging the outputs of these six networks, effectively reducing variance, improving stability, and mitigating overfitting [38].

The accuracy of the surrogate models is demonstrated in the results section with selected designs. Although the prediction quality may vary from one sample evaluation to another, these models have been rigorously tested on datasets comprising over 500,000 samples, which were not used for training the ANNs. It was observed that the prediction of compressor performance, such as isentropic efficiency $\eta_{\text{is, COMP}}$, exhibited less than 3% relative error in more than 50% of the dataset. Similarly, the prediction of lateral rotor dynamics or axial dynamics, indicated by the respective logarithmic decrements Γ_{HGJB} and Γ_{SGTB}, showed less than 3% relative error in more than 80% of the test dataset. The same level of accuracy applies to the prediction of pressure ratio Π.

More details about the datasets and the training of the ensemble artificial neural networks can be found in Ref. [12].

3.4 ParaturboCAD Library Functionality. Building upon CadQuery 2, we developed the ParaturboCAD library specifically for transforming 1D turbocompressor geometries into their 3D counterparts.[3] ParaturboCAD automates the creation of CAD models from design parameters optimized through user interaction and ANN analysis. It models essential components such as the impeller wheel, spiral groove thrust bearing, herringbone grooved journal bearings, and the complete turbocompressor rotor. On command, it assembles detailed CAD models encapsulating all critical features—from the rotor and grooves to the impeller—ready for export as STEP and STL files. These files can then be seamlessly integrated and visualized within the framework itself, bridging the gap between design conceptualization and ready-to-manufacture models.

3.5 Validation and Testing Techniques. The framework's performance is evaluated using four single-stage compressors operating in R134a, with different nominal mass flows, labeled T1–T4. The validation process involves three stages:

(1) Surrogate model quality assessment: The accuracy of the surrogate model is determined by analyzing prediction errors in compressor maps and robustness maps for HGJBs and SGTB against manufacturing deviations compared to high-fidelity model prediction.
(2) Real-time simulation testing: The framework's capacity for rapid computational analysis is tested. The timings are recorded over 10 runs for both coarse and fine discretizations, along with the generation time of 3D models in STEP and STL formats, aiming for a generation under 10 min.
(3) Three-dimensional model qualitative analysis: The quality of the generated 3D models is evaluated through a triangular Delaunay surface mesh using PYTHON trimesh and Gmsh, and scaled 3D prints, assessing the quality of the mesh for potential fluid or structure simulations.

All tests are conducted on a desktop with an AMD Ryzen 3900X CPU and a Nvidia RTX 3080 GPU.

4 Results

4.1 Quality of the Surrogate Models. Figure 4 illustrates the surrogate model's prediction accuracy for the compressor map isentropic efficiency (η_{is}) and pressure ratio (Π), as well as the axial stability (Γ_{sgtb}) and lateral rotordynamic stability (Γ_{hgjb}). The first row of plots focuses on isentropic efficiency and pressure ratio predictions, with scatter points colored by their relative error compared to the baseline model. The x-axis represents the gas mass flow at the compressor impeller's inlet, while the y-axis in the top plots denotes the total outlet to inlet pressure ratio. Isospeed lines, indicating constant compressor rotational speeds in rpm, are shown as gray lines. The second row displays the rotor dynamics stability, considering manufacturing deviations in the HGJBs, followed by the third row, which presents the axial dynamics stability affected by deviations in the SGTB.

The results demonstrate that the relative error in predicting the pressure ratio and isentropic efficiency largely stays below 5% across all case studies, from T1 to T4. This accuracy persists except at the extreme right of the isospeed lines, where the compressor approaches the choke limit. Notably, the highest relative error is observed in case study T2, reaching 11% in pressure ratio and 9% in isentropic efficiency, as depicted in Fig. 4(*b*). This indicates a slight deviation in model performance at the operational boundaries.

The rotordynamics stability maps, reflecting manufacturing deviations of the nominal clearance and groove depth of the HGJB, reveal a strong correlation between the high-fidelity baseline model and the surrogate model composed of artificial neural networks. The contour plots are colored using ANN predictions, with black contour lines representing the surrogate model limits and white lines indicating the baseline model predictions. This comparison generally shows excellent agreement, particularly in predicting stability variations against local bearing clearance h_{r} and groove depth h_{g}. A notable exception is in Fig. 4(*e*), where a minor discrepancy is observed, although it does not significantly impact the stability map's boundary for the onset of instability, where the logarithmic decrement Γ_{hgjb} reaches zero.

The axial dynamics stability maps, considering manufacturing deviations of the SGTB, also exhibit a commendable match between the baseline 1D model and the ANN-based surrogate model. Despite this overall agreement, a slight offset in predictions is observed for cases T1, T2, and T3, as depicted in Figs. 4(*i*), 4(*j*), and 4(*k*), respectively. Notably, case T4, shown in Fig. 4(*l*), aligns more closely with the baseline model. These minor deviations are within acceptable margins, ensuring the capability to design robust systems with substantial safety factors in groove depth and bearing clearance.

[3]See Note 2.

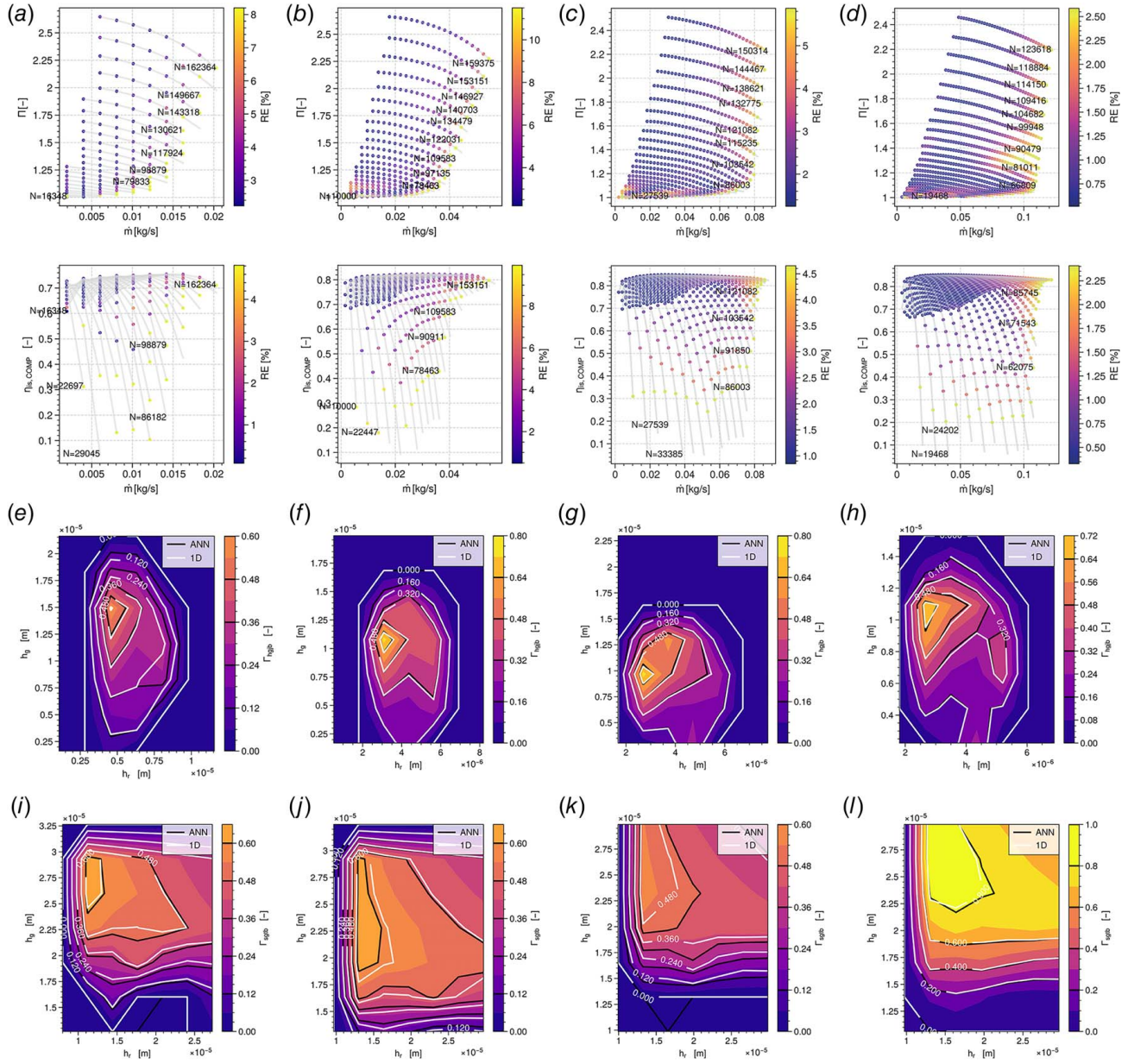

Fig. 4 Prediction quality of the surrogate model: (*a*) T1 compressor map, (*b*) T2 compressor map, (*c*) T3 compressor map, (*d*) T4 compressor map, (*e*) T1 stability map—HGJBs, (*f*) T2 stability map—HGJBs, (*g*) T3 stability map—HGJBs, (*h*) T4 stability map—HGJBs, (*i*) T1 stability map—SGTB, (*j*) T2 stability map—SGTB, (*k*) T3 stability map—SGTB, and (*l*) T4 stability map - SGTB.

4.2 Real-Time Simulation. The real-time simulation's efficacy is gauged across two scenarios: one with coarse and another with fine discretization. This involves varying the sample numbers for rotational speed N_k, inlet compressor mass flow $\dot{m}_k$, bearing clearances $h_{r,\,r,\,k}$ and $h_{r,\,a,\,k}$, and groove depths $h_{g,\,r,\,k}$ and $h_{g,\,a,\,k}$ for the robustness maps of HGJBs and SGTBs against manufacturing deviations. The resolutions for these discretizations in both scenarios are summarized in Table 1. This approach allows for a comprehensive evaluation of the framework's performance in simulating real-time scenarios under varying levels of detail. The tests are run on a Nvidia GPU RTX 3080 and AMD Ryzen 7 3700X.

The performance of DARTS-NETGAB in updating plots and recomputing turbocompressor performance is assessed by averaging the update time over ten iterations. This average time and its standard deviation are detailed in Table 2. The results indicate that simulation time is primarily influenced by the discretization level rather than the specific turbocompressor spindle type. Coarse resolution achieves faster evaluations, averaging around 1 s, compared to 8.5 s for the fine scheme. Interestingly, despite a 50-fold increase in the number of evaluated points, the increase in computation time is only by a factor of < 9.

Table 1 Discretization resolution for real-time simulation

	N_k	$\dot{m}_k$	$h_{g,r,k}$	$h_{r,r,k}$	$h_{r,a,k}$	$h_{g,a,k}$	Total
Coarse	5	25	13	13	13	13	6195
Fine	50	51	31	31	31	31	311,250

4.3 Three-Dimensional Generation. Table 3 presents the time required to generate the 3D models of the turbocompressor's subsystems and the complete assembly. This duration includes optimizing blade and splitter geometries to be straight at the impeller tip radius (lean angle of zero degrees), creating STEP files, and meshing STL files. The data indicate that, on average, CAD files

Table 2 Wall time of interface updates

	T1	T2	T3	T4
Coarse	0.95 ± 0.03s	1.05 ± 0.06s	1.0 ± 0.1s	0.98 ± 0.09s
Fine	8.8 ± 0.2s	8.93 ± 0.09s	8.5 ± 0.1s	8.4 ± 0.1s

Table 3 Three-dimensional generation time

	T1	T2	T3	T4
STEP and STL	460 ± 18s	412 ± 14s	445 ± 11s	446 ± 18s

for each turbocompressor are generated within 5–7 min. Notably, the generation time remains rather constant from T1 to T4, with an exception for T2, which takes slightly less time (412 ± 11s) than T1 (460 ± 10s).

In Fig. 5, the Delaunay mesh of model T3 is exhibited, with the STL mesh's conformity prominently visible at the impeller, specifically between the hub and the blades/splitters. This region, highlighted with a circle, showcases how each element's nodes align seamlessly with those of adjacent elements, ensuring the integrity of the mesh necessary for precision in downstream applications such as computational fluid dynamics (CFD), vibration mode analysis, and assessments of structural stresses or heat conduction.

Figure 6 in the Appendix B shows the 3D-printed models of turbocompressors using the automatically produced stl files, magnified to a 150% scale to ensure the details are discernible. To accommodate the printing capabilities and accuracy limitations of the 3D printer, certain features were intentionally exaggerated: the groove depths of both HGJBs and SGTB, along with the impeller blade thickness.

While the overall models are enlarged to 150%, exceptions include the plug and magnet, which are scaled to 145% to account for surface roughness and material retraction, and allow proper fitting between the different parts. The printing parameters were chosen as follows: a layer height of 0.16 mm and a 20% infill setting were used to balance detail resolution with material use efficiency, supported by additional structures to uphold the models' integrity during the printing process. These modifications were executed on a Creality 3D printer S1, resulting in a 17-h print time. This approach to 3D printing ensured that even the most delicate features, like grooves and blades, were rendered visible and tangible, making the models both a practical and insightful tool for examination.

The impellers have been correctly modeled and installed, with blades and splitters conforming to the pre-established thickness parameters. Notably, the lean angle at the outlet adheres to the design specification of zero degrees. Additionally, no discernible gaps are observed between these elements and the hub.

The overall assembly of the rotor presents different subsystems separated into distinct subassemblies without detectable gaps. These subassemblies can be individually manipulated in further CAD software interactions.

Following initial analyses of turbocompressors with nominal mass flows of 5, 25, 50, and 75 g/s, we extended our parametric study to incorporate higher mass flows of 25, 50, 100, and 200 g/s, designated as case 1 through case 4 [12]. This extension aimed to validate our design tool's effectiveness over a broader operational range, addressing compressor power levels ranging from 1 kW to 8 kW for operation in a heat pump with R134a refrigerant. This extensive study yielded a comprehensive dataset of 1188 models, publicly available and documented in the referenced dataset [39]. Each model, provided in both STEP and STL formats, represents individual subsystems and the assembled final turbocompressor. Designed for operation with R134a refrigerant, these models are optimized for various nominal mass flows, demonstrating the ParaturboCAD PYTHON library's capability to efficiently translate optimization parameters into practical CAD designs. Detailed generation time statistics are provided in Table 4. The majority of generation time is consumed by the impeller and rotor with herringbone grooved journal bearings, where the impeller blades require complex lofting operations and the HGJB involves numerous pocketing operations. Despite the complexity, the average total generation time per model is under 7 min, underscoring the efficiency and consistency of the ParaturboCAD tool in CAD generation and meshing, capable of handling complex geometries within a practical timeframe.

4.4 DARTS-NETGAB in Action. A video showcasing the functionality of DARTS-NETGAB is accessible through the provided link [40]. This framework demonstrates exceptional adaptability and user-friendliness, facilitating seamless modifications to geometries and instantaneous visualization of their performance impacts. Its intuitive user interface simplifies operations, including the direct manipulation of surrogate model parameters for potential extrapolation. Easy navigation across tabs and the web-based nature of the framework ensure its accessibility, even to nonexpert users familiar with basic web browsing. Usage of the framework does not require programming expertise, as all actions, from altering operating conditions to interpreting performance data via mouse hover on plots, are executed through graphical inputs, streamlining the computational process.

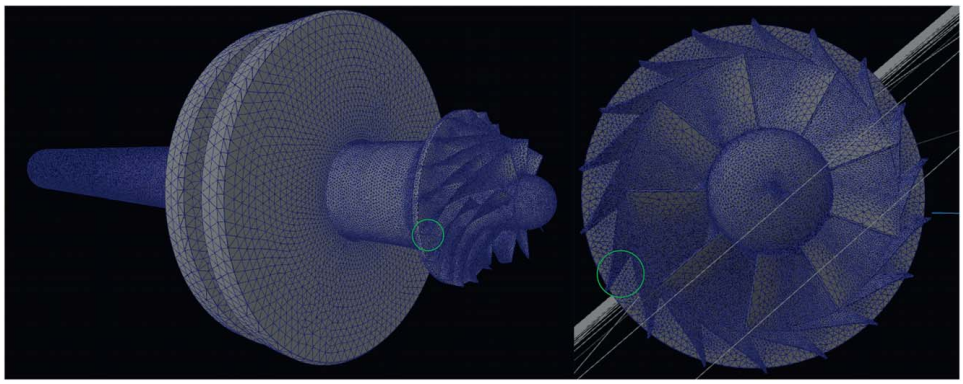

Fig. 5 STL mesh of the T3 model, featuring conformal Delaunay meshing with a focus on the impeller area, where hub-to-blade/splitter conformity is critical. A region highlighted in green emphasizes the precise node alignment between adjacent mesh elements, indicating a mesh of high integrity that is well-suited for advanced downstream applications such as CFD, modal vibration analysis, structural stress evaluations, and thermal conductivity studies.

Table 4 Statistics of generation time per turbocompressor

Metric	Mean time (s)	Standard deviation (s)
t_{SGTB}	20.75	0.71
$t_{\text{ROT+ HGJB}}$	140.69	35.36
t_{COMP}	160.47	18.62
t_{assembly}	83.13	13.51
t_{tot}	405.03	49.10

5 Discussion

5.1 Reflection on the Framework's Performance. The real-time simulation performance of DARTS-NETGAB demonstrates significant computational efficiency. Prediction times are maintained below 10s for detailed simulations on a single GPU, while traditional high-fidelity baseline models require nearly 2 h on a 16-core CPU for comparable analyses [12]. This improvement underscores the framework's ability to handle large-scale design exploration within practical time constraints.

Such speed gain, combined with the direct CAD generation capability through parametric CadQuery 2 scripting, significantly shortens the design cycle duration. The surrogate models not only expedite computations but also maintain a high accuracy level, contributing to fewer iterative cycles needed for design refinement.

The computational workflow is divided into offline and online components, ensuring both accuracy and responsiveness. The EANNs are trained offline using data sampled from high-fidelity baseline models, along with extensive preprocessing, hyperparameter tuning, and validation to ensure their reliability across the design space. Once trained, the EANNs are deployed for online use, where they facilitate rapid real-time evaluations of performance metrics and visualizations during user interactions. This separation allows the framework to achieve computational rigor while maintaining the interactivity necessary for an iterative design. The initial computational investment in training the EANNs is effectively amortized through a single turbocompressor optimization [12], where the selected Pareto-optimal design serves as a starting point for subsequent iterations in DARTS-NETGAB.

For example, in a prior application [41], the framework was used to redesign a rotor originally constructed from hollow tungsten carbide to a solid hardened steel shaft due to manufacturing constraints. This redesign required modifying both the material and geometry of the rotor while ensuring equivalent performance, particularly in terms of rotordynamic stability. Using DARTS-NETGAB, the material and geometric adjustments were implemented, and all relevant performance metrics were recomputed within approximately 10 min. This rapid iteration enabled the updated design to be promptly delivered to production, demonstrating the framework's capability to efficiently handle a real-world design challenge.

The EANNs were validated as global surrogate models, ensuring reliability across the design space. Their predictions are trusted for iterative design tasks, as the epistemic error is significantly reduced through ensemble averaging and extensive validation. However, higher prediction errors may occur near the boundaries of compressor maps. This is primarily due to the lack of training in these regions, where performance metrics like isentropic efficiency are constrained by physical limits and clipped at zero past the surge and choke lines. The sparsity of data and the rapid nonlinear changes in behavior near these boundaries contribute to increased uncertainty in predictions. To address this, postprocessing with the baseline high-fidelity models can be selectively used to reevaluate designs in critical regions, ensuring accuracy for parameters such as the achievable mass flow range between surge and choke or efficiency values near these operational limits.

5.2 Impact on Design Flexibility and Innovation. The interactive interface significantly streamlines the user experience and expedites the simulation process, allowing for a thorough examination of the design space. The versatility of the framework is exemplified by its ability to swiftly adapt to variations in operating fluids, thus facilitating the assessment of the turbocompressor's performance under diverse conditions. This capability extends to the modification of compressor speed and mass flow to achieve desired pressure ratios, further illustrating the system's flexibility. Moreover, the framework includes a scanning feature that probes the sensitivity of performance metrics to variations around a nominal design point, a critical function for evaluating design robustness. Such a feature is instrumental in identifying the optimal design parameters that ensure reliability and efficiency, even in the face of potential manufacturing deviations.

DARTS-NETGAB's advanced capabilities include the automatic generation of accurate and usable 3D CAD parts and assemblies directly from design iterations. This functionality facilitates a seamless transition from the optimization phase to the manufacturing environment, bypassing potential delays and material constraints. By swiftly accommodating design changes and rapidly producing accurate CAD models, the framework significantly reduces the need for repetitive engineering input on minor adjustments. This eases the import of designs into further high-fidelity analyses such as CFD or FE-based stress or thermal calculations. Such streamlined adaptability is invaluable, preventing the overuse of engineering resources and enhancing the efficiency of the design-to-manufacture process.

This innovative framework fosters a more dynamic and efficient design process, capable of accommodating complex and evolving engineering and manufacturing demands. Its ability to navigate within the landscape of turbocompressor design marks a significant leap forward in the field.

5.3 Contributions to the Field. DARTS-NETGAB represents a paradigm shift in the integration of computational tools and CAD generation within engineering workflows. Its novel architecture seamlessly bridges the gap between performance computation and manufacturable 3D CAD models, addressing critical bottlenecks in transitioning from conceptual design to production. This integration eliminates the traditionally disjointed nature of these processes, streamlining workflows and significantly reducing the time and effort required for design iterations.

While commercial CAD software often supports automation through PYTHON scripting, these tools are typically tied to proprietary ecosystems, limiting accessibility for academic research. In contrast, DARTS-NETGAB uses CadQuery, an open-source CAD tool, to seamlessly integrate performance computation and CAD model generation within a single PYTHON environment. Although CadQuery's Open CASCADE Technology backend may not rival the speed or advanced functionalities of proprietary kernels, it offers flexibility and accessibility, making it particularly well-suited for iterative design processes and academic workflows. This approach eliminates the need for manual intervention and enables efficient transitions between design iterations and CAD generation.

The framework's contributions extend beyond workflow efficiency. By leveraging scripted parametric CAD modeling, DARTS-NETGAB provides a robust method for generating high-quality datasets, particularly for domains like gas-bearing supported turbocompressors, which have historically lacked comprehensive geometric libraries. These datasets have the potential to inform and advance emerging techniques in AI-driven CAD generation.

6 Conclusion

The DARTS-NETGAB framework introduces a methodical approach to engineering design, prioritizing efficiency and user engagement from inputting specifications to automated CAD generation. Initially tailored to gas-bearing supported turbocompressors,

the platform employs neural network ensembles for swift and accurate performance predictions. The interactive dashboard enables instantaneous visualization of design modifications, reflecting a commitment to user-driven development.

This article highlights the adaptability of DARTS-NETGAB, illustrating its potential for broader application across diverse engineering systems. By integrating automated CAD generation with machine learning, the framework aims to streamlining the design process, offering a thoughtful contribution to the field of engineering design.

Acknowledgement

The authors wish to express their profound gratitude to the Laboratory for Applied Mechanical Design's engineers, Christophe Pham and Theo Muller, for their invaluable expertise and commitment. Their comprehensive workshops provided detailed insights into the challenges of CAD design for gas-bearing supported turbocompressors. The in-depth explanations concerning the herringbone grooved journal bearings, centrifugal impeller, and spiral groove thrust bearing have greatly augmented the foundation of this research. The authors acknowledge the significant enhancement these discussions brought to this study and appreciate their willingness to share such essential knowledge.

Conflict of Interest

There are no conflicts of interest.

Data Availability Statement

The datasets generated and supporting the findings of this article are obtainable from the corresponding author upon reasonable request.

Nomenclature

h = gap height (m)
D = bearing diameter (m)
L = length (m)
R = radius (m)

Greek Symbols

α = groove-ridge width ratio/lean angle (–/deg)
β = groove/blade angle (deg)
γ = grooved region ratio (–)
Γ = logarithmic decrement (–)
η = efficiency (–)
Π = pressure ratio (–)
Ω = angular velocity (rad/s^{-1})

Superscripts and Subscripts

A = first HGJB
a = axial
B = second HGJB
g = groove
in = inlet
is = isentropic
k = sample number
r = ridge
r = radial

Acronyms

AAO-RMDO = all-at-once robust multidisciplinary design optimization
ANN = artificial neural network
CAD = computer-aided design
CAE = computer-aided engineering
CAM = computer-aided manufacturing
CFD = computational fluid dynamics
CNC = computer numerical control
COMP = compressor
EANN = ensemble artificial neural network
FE = finite element
GAN = generative adversarial network
GPU = graphics processing unit
GUI = graphical user interface
HGJB = herringbone grooved journal bearing
MAE = mean absolute error
MAG = magnet
MDO = multidisciplinary design optimization
MSE = mean squared error
OCCT = open CASCADE technology
ROT = rotor
STEP = standard for the exchange of product model data
STL = standard tesselation language
rpm = rotations per minute
RTS = real-time simulation
SGTB = spiral groove thrust bearing
VAE = variational autoencoder
VRAM = video random access memory
VTK = visualization toolkit

Appendix A: Artificial Neural Network Hyperparmeter Optimization

Table 5 Description of the hyperparameters searched for the optimization of the feed-forward neural networks

Term	Symbol	Value
Number of neurons per hidden layer	n	16, 32, 64, 128, 256
Number of hidden layers	l	2, 3, 4
Activation	a	relu, selu, tanh, softplus, softsign
Optimizer	opt	Adam, Adamax, Adadelta, Adagrad
Batch size	bs	2^{12}, 2^{13}, 2^{14}, $2^{15}/2^{9}$, 2^{10}, 2^{11}, 2^{12}, 2^{13}, 2^{14}
Kernel initializer	ki	Glorot Normal (gn), He Normal (hn), Lecun Normal (ln), Glorot Uniform (gu), He Uniform (hu), Lecun Uniform (lu)
L2 penalization	β_{L2}	10^{-6}, 10^{-5}, 10^{-4}, 10^{-3}, 10^{-2}
Learning rate	α_{lr}	0.001, 0.01, 0.1
Decay steps	ds	10^{3}, 10^{4}, 10^{5}

Appendix B: Three-Dimensional Prints

Fig. 6 Three-dimensional printed turbocompressor models T1–T4, each enlarged to 150% scale for enhanced visibility. Key features like groove depths in the herringbone grooved journal bearings (HGJBs) and spiral groove thrust bearings (SGTBs), as well as impeller blade thickness, are amplified to 0.5mm to circumvent the resolution limits of the 3D printing process and to ensure accurate detail reproduction.

References

[1] Pailhés, J., Sallaou, M., Nadeau, J.-P., and Fadel, G. M., 2011, "Energy Based Functional Decomposition in Preliminary Design," ASME J. Mech. Des., **133**(5), p. 051011.

[2] Fiorineschi, L., Rotini, F., and Rissone, P., 2016, "A New Conceptual Design Approach for Overcoming the Flaws of Functional Decomposition and Morphology," J. Eng. Des., **27**(7), pp. 438–468.

[3] Weber, R. G., and Condoor, S. S., 1998, "Conceptual Design Using a Synergistically Compatible Morphological Matrix," FIE '98: 28th Annual Frontiers in Education Conference, Tempe, AZ, Nov. 4–7, pp. 171–176.

[4] Chen, Y., Feng, P., He, B., Lin, Z., and Xie, Y, 2005, "Automated Conceptual Design of Mechanisms Using Improved Morphological Matrix," ASME J. Mech. Des., **128**(3), pp. 516–526.

[5] Ma, H., Chu, X., Xue, D., and Chen, D, 2017, "A Systematic Decision Making Approach for Product Conceptual Design Based on Fuzzy Morphological Matrix," Exp. Syst. Appl., **81**, pp. 444–456.

[6] Allaire, D., and Willcox, K., 2014, "A Mathematical and Computational Framework for Multifidelity Design and Analysis With Computer Models," Int. J. Uncertain. Quantif., **4**(1), pp. 1–20.

[7] Sobieszczanski-Sobieski, J., 1995, "Multidisciplinary Design Optimization: An Emerging New Engineering Discipline," *Advances in Structural Optimization*, J. Herskovits, ed., Solid Mechanics and Its Applications, Springer Netherlands, Dordrecht, pp. 483–496.

[8] Cramer, E. J., Dennis, J. E., Lewis, R. M., and Shubin, G. R., 1994, "Problem Formulation for Multidisciplinary Optimization," SIAM J. Optim., **4**(4), pp. 754–776.

[9] Martins, J. R. R. A., and Lambe, A. B., 2013, "Multidisciplinary Design Optimization: A Survey of Architectures," AIAA J., **51**(9), pp. 2049–2075.

[10] Amadori, K., Tarkian, M., Ölvander, J., and Krus, P, 2012, "Flexible and Robust CAD Models for Design Automation," Adv. Eng. Inform., **26**(2), pp. 180–195.

[11] Meibody, M. N. P., Naseh, H., and Ommi, F., 2021, "Developing a Multi-objective Multi-disciplinary Robust Design Optimization Framework," Sci. Iranica, **28**(4), pp. 2150–2163.

[12] Massoudi, S., Picard, C., and Schiffmann, J., 2024, "An Integrated Approach to Designing Robust Gas-Bearing Supported Turbocompressors Through Surrogate Modeling and Constrained All-at-Once Multi-Objective Optimization," ASME J. Mech. Des., **146**(12), pp. 121706.

[13] Odot, A., Haferssas, R., and Cotin, S., 2022, "DeepPhysics: A Physics Aware Deep Learning Framework for Real-Time Simulation," Int. J. Numer. Methods Eng., **123**(10), pp. 2381–2398.

[14] Massoudi, S., and Schiffmann, J., 2022, "ARRID: ANN-based Rotordynamics for Robust and Integrated Design," Presented at the Machine Learning in Computational Design Workshop, 39th International Conference on Machine Learning (ICML), 2022.

[15] Puzyrev, V., Ghommem, M., and Meka, S., 2019, "*pyROM*: A Computational Framework for Reduced Order Modeling," J. Comput. Sci., **30**, pp. 157–173.

[16] Lambourne, J. G., Willis, K. D. D., Jayaraman, P. K., Sanghi, A., Meltzer, P., and Shayani, H., 2021, "BRepNet: A Topological Message Passing System for Solid Models," IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR 2021), Virtual Conference, June 19–25, pp. 12768–12777.

[17] Jayaraman, P. K., Sanghi, A., Lambourne, J. G., Willis, K. D. D., Davies, T., Shayani, H., and Morris, N., 2021, "UV-Net: Learning From Boundary Representations," IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR 2021), Virtual Conference, June 19–25, pp. 11698–11707.

[18] Jayaraman, P. K., Lambourne, J. G., Desai, N., Willis, K. D. D., Sanghi, A., and Morris, N. J. W., 2024, "SolidGen: An Autoregressive Model for Direct B-rep Synthesis," International Conference on Learning Representations (ICLR 2024), Vienna, Austria, May 7–11.

[19] Jones, B. T., Hu, M., Kim, V. G., and Schulz, A, 2023, "Self-Supervised Representation Learning for CAD," IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR 2023), Vancouver, Canada, June 18–22.

[20] Song, B., Yuan, C., Permenter, F., Arechiga, N., and Ahmed, F., 2024, "Data-Driven Car Drag Prediction With Depth and Normal Renderings," ASME J. Mech. Des., **146**(5), p. 051714.

[21] Yoo, S., Lee, S., Kim, S., Hwang, K. H., Park, J. H., and Kang, N., 2021, "Integrating deep learning into CAD/CAE system: generative design and evaluation of 3D conceptual wheel," Struct. Multidisc. Optim., **64**, pp. 2725–2747.

[22] Wu, R., Xiao, C., and Zheng, C., 2021, "DeepCAD: A Deep Generative Network for Computer-Aided Design Models," IEEE/CVF International Conference on Computer Vision (ICCV 2021), Virtual Conference, Oct. 11–17, pp. 6772–6782.

[23] Ganin, Y., Bartunov, S., Li, Y., Keller, E., and Saliceti, S, 2021, "Computer-Aided Design as Language," Advances in Neural Information Processing Systems (NeurIPS 2021), Virtual Conference, Dec. 6–14.

[24] Xu, X., Jayaraman, P. K., Lambourne, J. G., Willis, K. D. D., and Furukawa, Y., 2023, "Hierarchical Neural Coding for Controllable CAD Model Generation," 40th International Conference on Machine Learning (ICML 2023), Hawaii Convention Center, Honolulu, HI, July 23–29, 2023.

[25] Zhou, S., Tang, T., and Zhou, B., 2023, "CADParser: A Learning Approach of Sequence Modeling for B-Rep CAD," Thirty-Second International Joint Conference on Artificial Intelligence (IJCAI 2023), Macao, China, Aug. 19–25, pp. 1804–1812.

[26] Bokeh Development Team, 2023, "Bokeh: Python Library for Interactive Visualization," https://bokeh.pydata.org/en/latest/

[27] Massoudi, S., Picard, C., and Schiffmann, J., 2022, "Robust Design Using Multiobjective Optimisation and Artificial Neural Networks With Application to a Heat Pump Radial Compressor," Des. Sci., **8**, pp. 1041–1050.

[28] Massoudi, S., and Schiffmann, J., 2023, "Robust Design of Herringbone Grooved Journal Bearings Using Multi-objective Optimization With Artificial Neural Networks," ASME J. Turbomach., **145**(11), p. 111010.

[29] Gu, L., Guenat, E., and Schiffmann, J., 2020, "A Review of Grooved Dynamic Gas Bearings," ASME Appl. Mech. Rev., **72**(1), p. 010802.

[30] Wagner, P. H., Wuillemin, Z., Constantin, D., Diethelm, S., Van herle, J., and Schiffmann, J., 2020, "Experimental Characterization of a Solid Oxide Fuel Cell Coupled to a Steam-Driven Micro Anode Off-Gas Recirculation Fan," Appl. Energy, **262**, p. 114219.

[31] Rosset, K., Pajot, O., and Schiffmann, J, 2021, "Experimental Investigation of a Small-Scale Organic Rankine Cycle Turbo-Generator Supported on Gas-Lubricated Bearings," ASME J. Eng. Gas Turbines Power, **143**(5), p. 051015.
[32] Schiffmann, J., and Favrat, D., 2009, "Experimental Investigation of a Direct Driven Radial Compressor for Domestic Heat Pumps," Int. J. Refrig., **32**(8), pp. 1918–1928.
[33] Guenat, E., and Schiffmann, J., 2020, "Dynamic Force Coefficients Identification on Air-Lubricated Herringbone Grooved Journal Bearing," Mech. Syst. Signal Process., **136**, p. 106498.
[34] Papavasileiou, E., Cornelis, J., and Jansen, B., 2021, "A Systematic Literature Review of the Successors of 'NeuroEvolution of Augmenting Topologies'," Evol. Comput., **29**(1), pp. 1–73.
[35] He, K., Zhang, X., Ren, S., and Sun, J., 2015, "Delving Deep Into Rectifiers: Surpassing Human-Level Performance on ImageNet Classification," IEEE International Conference on Computer Vision (ICCV 2015), Santiago, Chile, Dec. 11–18, pp. 1026–1034.
[36] Glorot, X., and Bengio, Y., 2010, "Understanding the Difficulty of Training Deep Feedforward Neural Networks," Thirteenth International Conference on Artificial Intelligence and Statistics (AISTATS 2010), Sardinia, Italy, May 13–15, pp. 249–256.
[37] LeCun, Y. A., Bottou, L., Orr, G. B., and Müller, K.-R., 2012, "Efficient Backprop," *Neural Networks: Tricks of the Trade*, G. Montavon, G. B. Orr, and K.-R. Müller, eds., Springer, Berlin, Germany, pp. 9–48.
[38] Ganaie, M. A., Hu, M., Malik, A. K., Tanveer, M., and Suganthan, P. N., 2022, "Ensemble Deep Learning: A Review," Eng. Appl. Artif. Intell., **115**, p. 105151.
[39] Massoudi, S., Bejjani, J., Horvath, T., Üstün, D., and Schiffmann, J., 2024, "ParaturboCAD: Automated Parametric Geometry Construction for Gas-Bearing Supported Turbocompressor Design With Python," ASME 2024 International Design Engineering Technical Conferences & Computers and Information in Engineering Conference (IDETC-CIE 2024), Washington, DC, Aug. 25–28.
[40] Massoudi, S., 2024, "DARTS-NETGAB Demonstration Video," https://drive.google.com/file/d/1NPLCpDtpBdiC1VKdvwmiecZCQHChjlTo/view?usp=drive_link, Accessed: 2024-03-12.
[41] Massoudi, S., Bush, C., and Schiffmann, J., 2025, "Robust Design Optimization of Gas-Lubricated Herringbone Grooved Journal Bearings: Surrogate Modeling and Experimental Validation," Tribol. Int., **204**, p. 110429.