UDC 629.7.023.2 : 004.89 **doi: 10.32620/aktt.2025.4sup1.18**

Vadym PASKO, Sviatoslav YUTSKEVYCH

State University "Kyiv Aviation Institute", Ukraine

APPLICATION OF MESH-FREE METHODS IN THE WING RIGIDITY ANALYSIS TO SUPPORT AUTOMATION OF UAV DESIGN

Modern aircraft design, including both manned aircraft and unmanned aerial vehicles (UAVs), faces computational challenges balancing aerodynamic efficiency, structural integrity, and weight optimization within practical timeframes. Conventional high-fidelity methods create bottlenecks that limit the design space exploration essential for UAV development. This paper presents a computational framework that integrates mesh-free structural analysis with generative knowledge-based engineering (KBE) and surrogate modelling for the optimization of rapid automated UAV wing design.
The methodology combines the formalization of classical aerodynamic and structural mechanics knowledge with programmable CAD integration using the open-source Python package CadQuery. The developed framework automatically generates parametric wing geometries, extracts geometric properties, including cross-sectional moments of inertia and volumes, and performs structural analysis without mesh generation or finite element preprocessing. Aerodynamic loads are estimated using reusable meta-models from CFD studies stored as B-spline approximations in SplineCloud, enabling decoupled workflows and rapid evaluation.
The mesh-free algorithm implements the numerical integration of beam bending equations, incorporating distributed aerodynamic and gravitational loads with variable cross-sectional properties. This eliminates the computational overhead of mesh generation while maintaining sufficient accuracy for preliminary design. The workflow is embedded in a KBE wing model, automating geometry generation and structural evaluation for swept wings with variable materials and geometries. The validation studies used three NACA airfoil families (2410, 2412, 2415) across aspect ratios (6-9), sweep angles (12°-18°), and spans (500-2500 mm). Individual evaluations completed in ~20 seconds versus hours/days for FEM simulations, achieving 2-3 orders of magnitude efficiency improvement. Generated 2nd-order meta-models enable sub-millisecond response evaluations suitable for iterative optimization requiring thousands of evaluations. This research advances automated design methodologies, providing computationally efficient alternatives to high-fidelity approaches while maintaining engineering accuracy for preliminary optimization. Open-source implementation ensures accessibility for the UAV design community. Future work will focus on FEM validation, aeroelastic coupling, and extensions to complex configurations.

Keywords: *aircraft; UAV; design automation; mesh-free methods; numerical analysis; stress; strength; MDO; programmable CAD; knowledge-based engineering.*

Introduction

The modern field of aircraft design, encompassing both manned aircraft and unmanned aerial vehicles (UAVs) design is increasingly shaped by complex multidisciplinary challenges that require advanced computational methods to achieve optimal system performance. The integration of aerodynamic efficiency, structural integrity, weight reduction, radar signature minimization, and mission-specific constraints calls for design methodologies capable of managing strong inter-dependencies across disciplines while remaining computationally feasible for extensive design space exploration.

This paper presents a methodology to address the problem of design automation in the context of analyzing and optimizing a tapered wing console with variable sweep angle, span, and aspect ratio. The core of the proposed approach lies in the integration of surrogate models for computational fluid dynamics (CFD) analysis with a custom mesh-free method for evaluating wing stiffness and strength, all embedded within a generative knowledge-based engineering (KBE) model of the wing console. The method leverages an industry-grade CAD kernel (OpenCASCADE) through its Python interface, CadQuery, to generate 3D geometry based on parametric rules and to extract geometric properties (e.g., cross-sectional moments of inertia and volumes) required to support structural analysis.

While the implementation details and software architecture of the generative KBE model are beyond the scope of this paper, we focus here more on the underlying numerical methods used to estimate wing bending under operational loads. The structure of this paper is as follows: Section 1 provides an overview of the principles of Multidisciplinary Design Optimization (MDO) and Knowledge-Based Engineering (KBE), highlighting the state-of-the-art and current challenges in these fields. Section 2 introduces the proposed approach for enabling multidisciplinary analysis and optimization of the wing console using generative KBE modeling, with emphasis on model structure and the application of surrogate models for aerodynamic load estimation. Section 3 details the custom numerical method developed for wing strength analysis. Section 4 presents a case study involving a swept wing console designed for UAV applications and discusses the results.

1. Computational Challenges in UAV Design Optimization

Aircraft design presents unique challenges that amplify the computational limitations of traditional MDO approaches. The relatively small scale of UAV systems demands high precision in aerodynamic and structural analysis, while the diverse mission requirements necessitate extensive parametric studies to identify optimal configurations. In [1] comprehensive studies on UAV design optimization were conducted, demonstrating that effective design space exploration requires thousands of design evaluations, each involving tightly coupled aerodynamic and structural analyses.

The computational intensity of high-fidelity methods creates a fundamental bottleneck in UAV design automation. Computational Fluid Dynamics (CFD) analysis, essential for accurate aerodynamic performance prediction, typically requires significant computational resources and time. Similarly, Finite Element Method (FEM) structural analysis, while providing high accuracy, demands substantial preprocessing effort for mesh generation and extensive computational resources for solution convergence.

Hwang and Martins [2] addressed these challenges through the development of gradient-based optimization techniques for UAV design, achieving efficiency improvements via analytical sensitivity analysis. However, their approach remained dependent on high-fidelity simulation tools, limiting its applicability in cases requiring broad design space exploration or rapid iteration cycles.

To address these limitations, alternative formulations and tool architectures have been studied and incorporated into modern MDO workflows. These approaches aim to reduce computational cost while maintaining adequate model fidelity for design decision-making.

1.1. Multidisciplinary Design Optimization in Aerospace Applications

MDO has emerged as a critical methodology for addressing the complexity inherent in modern aerospace system design. The term MDO was first formalized and widely disseminated in the late 1980s and early 1990s. Work [3] defined MDO as a methodology that systematically integrates and optimizes multiple interacting disciplines in the design of complex systems, such as aerospace vehicles. MDO aims to improve overall system performance by coordinating disciplinary analyses and optimizing across the full system, rather than treating each discipline in isolation.

MDO and related fields - such as knowledge-based engineering [4], generative design [5], and computational engineering [6] are actively evolving, integrating diverse methods, tools, and models into comprehensive computational pipelines. These developments target two primary goals: reducing computational time and improving the fidelity and automation of simulation-based design workflows.

The Multidisciplinary Feasible (MDF) method is one of the earliest and most widely used MDO architectures [7]. In MDF, the optimization is performed over a fully coupled multidisciplinary analysis (MDA), ensuring that each design point satisfies all interdisciplinary consistency constraints. While this approach is conceptually straightforward, it becomes computationally expensive when applied to large-scale problems or systems with weakly coupled disciplines.

To overcome these challenges, alternative architectures were proposed, including Individual Discipline Feasible (IDF), Collaborative Optimization (CO), and Concurrent Subspace Optimization (CSSO) [7]. These methods introduce different levels of problem decomposition and coordination to improve scalability and flexibility. IDF relaxes the requirement of full consistency during optimization. CO employs a hierarchical structure that distributes the optimization problem across disciplines, coordinating via shared variables. CSSO introduces surrogate models within decomposed subspaces, reducing the number of high-fidelity evaluations required and enabling concurrent optimization across disciplines.

While these architectures address limitations of MDF, they also highlight a broader dependency shared by all MDO frameworks: the quality and flexibility of geometry representation and numerical analysis methods. MDO performance is influenced not only by solver accuracy but also by the level of automation in geometry modeling, the ability to extract and reuse geometric properties, and the ease of coupling between design and simulation environments.

This diversity is reflected in available MDO frameworks. Commercial tools such as iSight and ModelCenter provide visual interfaces and broad integration capabilities with proprietary solvers. Open-source frameworks like OpenMDAO and OpenAeroStruct offer Python-based environments that emphasize extensibility and gradient-based optimization. Tools such as pyMDO focus on lightweight, research-oriented implementations.

Geometry representation remains a key challenge. Open-source tools typically employ simplified parametric models to support gradient-based optimization, but may sacrifice geometric fidelity or require manual setup for mesh consistency. Integration with external CAD systems via APIs introduces further complexity, especially when derivative information is needed.

The same principle applies to numerical analysis methods. Proprietary frameworks often rely on tightly coupled, high-fidelity solvers, while open-source frameworks favor modular, customizable, and lower-fidelity models. This confirms the search for the trade-off between accuracy and computational efficiency.

The integration of KBE with optimization techniques has shown particular promise for managing MDO complexity. In work [8] author developed knowledge-based design environments for aircraft conceptual design, integrating automated geometry generation with multidisciplinary analysis capabilities. Their approach enabled rapid evaluation of design alternatives while maintaining the flexibility required for preliminary design exploration.

1.2. Knowledge-Based Engineering

Knowledge-Based Engineering has evolved as a complementary and increasingly essential methodology for managing the growing complexity and automation demands of modern engineering design. KBE integrates engineering knowledge, design rules, parametric modeling, and reasoning logic into a unified computational framework to support decision-making and reduce design cycle times. As defined in [9], KBE systems aim to capture expert knowledge and embed it within design automation processes to improve design consistency, reduce manual effort, and enable the reuse of validated engineering solutions.

A foundational application of KBE to aerospace was demonstrated in [10], who developed a KBE system for aircraft design that supported parametric modeling, design rules, and multi-disciplinary analysis integration. Their work showed that KBE enables rapid and consistent generation of design variants, significantly reducing development time in preliminary design phases without compromising engineering accuracy.

Modern developments in KBE have increasingly focused on enhancing integration with parametric CAD systems and simulation tools. Recent reviews and case studies [9, 11] emphasize that current KBE systems are being extended with support for knowledge graphs, semantic modeling, and model-based systems engineering (MBSE) principles. These developments reflect a broader push toward data-driven and model-centric engineering workflows that tightly couple geometry, simulation, and rule-based logic.

Work [12] previously laid the groundwork for knowledge-based design environments by combining geometric modeling with physics-based analysis tools, allowing design alternatives to be evaluated automatically.

The current trajectory of KBE emphasizes openness, modularity, and integration with multidisciplinary optimization workflows. As discussed in recent surveys [13], modern KBE systems not only automate routine design tasks but also facilitate the reuse of domain-specific knowledge in collaborative engineering contexts, making them ideal companions to MDO frameworks. Their ability to explicitly represent design knowledge and structure geometry generation processes positions them as enabling technologies for scalable, surrogate-enhanced, and decoupled MDO approaches.

Current work represents an initial effort of the authors in exploring the advantages of integrating surrogate modelling with KBE to develop a framework for generative UAV design and MDO using open-source tools (including programmable CAD) and open knowledge management platform SplineCloud.

2. Generative KBE model of the UAV wing console

A Generative Knowledge-Based Engineering model is a computational model that encodes engineering knowledge, rules, and constraints to automatically generate, modify, and evaluate design configurations. It combines the declarative structure of traditional KBE systems (i.e., rules, parameters, logic) with procedural capabilities to dynamically create geometry, simulations, and associated metadata based on high-level design inputs or goals.

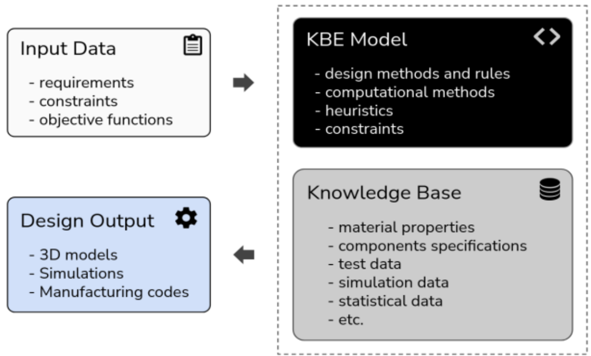


Fig. 1. A general KBE system schema

The important aspect of any KBE model is the efficient reuse of formalized knowledge, either stored in the external database (like SplineCloud) or encoded in the algorithms that generate geometry and perform engineering analysis.

2.1. Reuse of meta models from parametric CFD studies

In the proposed approach, reusable knowledge is represented in the form of meta-models and datasets stored in the SplineCloud repositories and accessed in computer code over the API. These meta models are represented in the form of B-splines that approximate dependencies between aerodynamic coefficients and other parameters (angle of attack, wing span coordinate). Such an approach allows for decoupling aerodynamics from the stress analysis. This enables the reuse of aerodynamics data without the necessity to build and run CFD simulations.

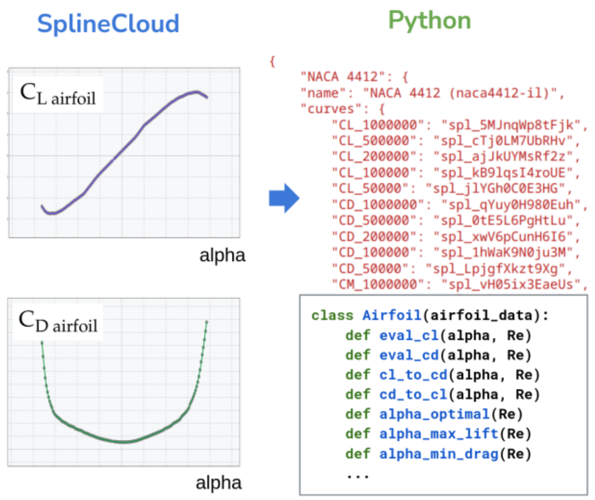


Fig. 2. Ruse of airfoil performance curves in Python

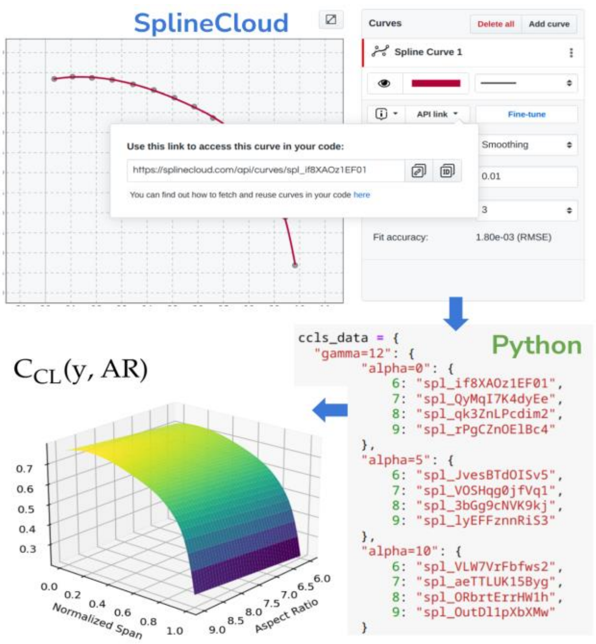


Fig. 3. Reuse of lift efficiency curves in Python code and a lift efficiency response surface model

Particularly, the following models are used to evaluate aerodynamic loads:

– Dependencies of airfoils' lift and drag coefficients on the angle of attack (Fig. 2);

– Lift efficiency (lift coefficient magnitude) as a function of normalized wing span (Fig. 3);

– Span efficiency factor as a function of aspect ratio and taper ratio (Fig. 5).

Airfoil's performance data is obtained from the XFOIL simulations for various Reynolds numbers. Simulation data is provided by the popular resource AirfoilTools.

Lift efficiency distribution curves are calculated using a parametric swept wing model in OpenVSP software (Fig. 4), using the VLM method.

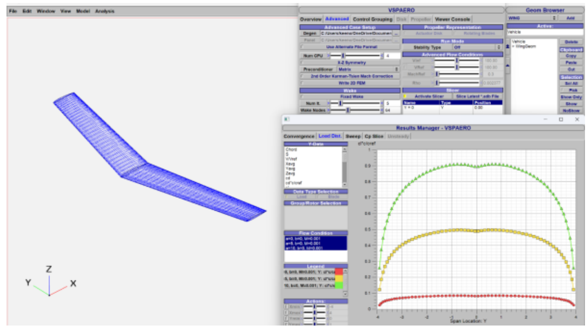


Fig. 4. Parametric wing model in OpenVSP and calculated lift distribution data

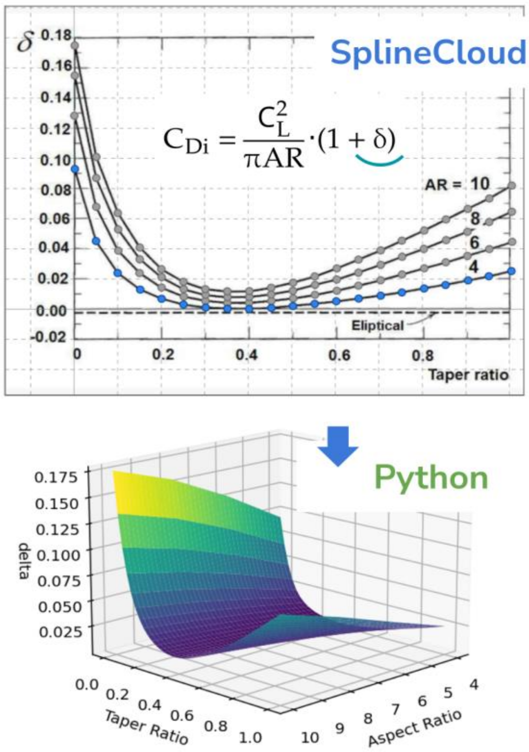


Fig. 5. Span efficiency factor response surface model

Induced drag is calculated using empirical relations [14], digitized on SplineCloud and converted into the response curves, loaded into Python code over the API, and converted into a response surface model (Fig.5).

The detailed description of the approach towards performing CFD analysis and converting results into dimensionless response surface models goes beyond the scope of the current work. However, it is important to mention that such an approach allows for to prediction of the values of aerodynamic coefficients for a large number of airfoils and wing geometries by evaluating approximating models, avoiding the need to perform simulations on each step of the wing performance analysis.

2.2. Wing console structure and generative model

The wing's internal structure is represented by the three structural components: an internal rigid box, an airfoil-shaped body, and an outer shell (Fig. 6). The rigid box is a load-bearing component and can be made of aluminum alloys or composite materials. The shaper body can be made out of lightweight materials: polystyrene foam or honeycomb. The outer shell can be made of fiberglass or another material that would protect the shaper body from external factors.

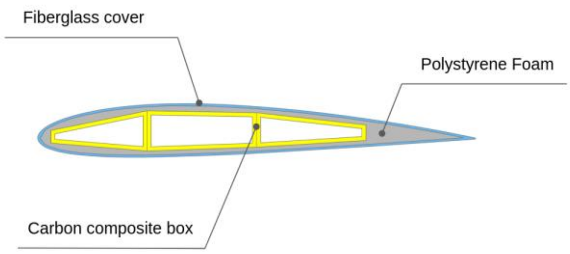
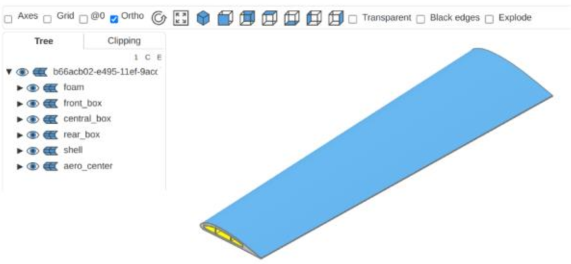


Fig. 6. Wing console section and internal structure




```
class SweptWingConsole(box_section):
    def build_geometry(...)
    def assign_materials(...)
    def compute_drag_force(...)
    def compute_lift_force(...)
    def fit_chord_to_required_lift_force(...)
        # Find proper chord length to fit required lift force
        # and ensure tip deflection is not greater than 'tip_delta_max'
    def fit_length_to_required_tip_deflection(...)
        # Find console length that ensures tip deflection equal to 'tip_delta_max'
    def get_box_mass(...)
    def get_foam_mass(...)
    def get_shell_mass(...)
    def get_component_section_inertia_moments(...)
    ...
```

Fig. 7. Swept wing console mathematical model

The wing geometry generation algorithm is based on the open-source Python library CadQuery, which uses the OpenCASCADE kernel and allows building 3D models programmatically. The usage of CadQuery allows encoding design and engineering rules to:

– automatically rebuild geometry for the selected airfoil;

– evaluate volumes of elements of the wing structure (and masses, given the densities of materials);

– evaluate areas and moments of inertia of the wing console sections to support the calculation of the bending moment and wing tip displacement in a meshless way.

A SweptWingConsole class structure and the generated output are given in Fig. 7.

3. Wing bending analysis

Wing load scheme is given in Fig. 8. A cantilever beam is studied with distributed loading from aerodynamic forces and gravity forces.

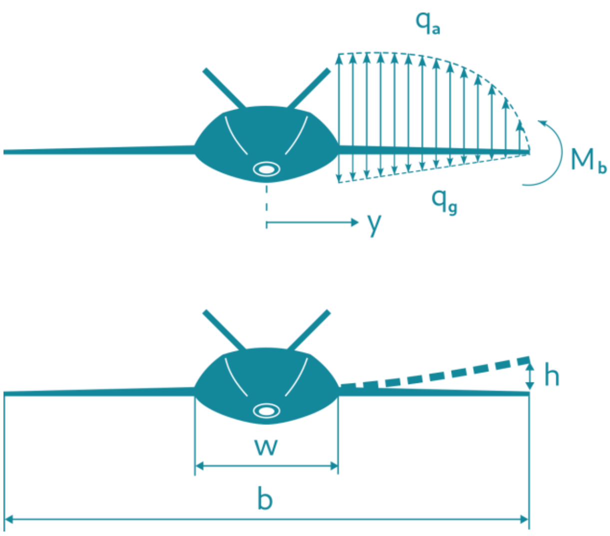


Fig. 8. Wing console loading scheme

In the given problem setup, the following simplifications and assumptions are taken to simplify the engineering analysis process:

– pure bending is studied - no torque and shear are analyzed.

– isotropic material properties are considered (for simplicity);

– no aeroelasticity effects are considered (wing load scheme is taken for the undeformed wing, small tip displacements are expected);

– all components act in the bending loading, proportional to their rigidity.

The bending moment and tip deflection are calculated numerically, integrating the following equations:

ISSN 1814-4225 (print)
ISSN 2663-2012 (online)

$$h = \int_{y=b/2}^{y=w/2} \theta(y)dy, \quad (1)$$

$$\theta(y) = \frac{M_b(y)}{\sum_{i=1}^{k} E_i I_i(y)}, \quad (2)$$

$$M_b(y) = \int_{y=b/2}^{y=w/2} \left(q_a(y) - q_g(y)\right) y dy, \quad (3)$$

where h – the absolute wing tip displacement in the vertical direction, θ – the local bending angle, y – the coordinate from the aircraft longitudinal axis to the wing tip, w – the fuselage width, E_i - is the tensile modulus of the structural component material, I_i – the i-th component section moment of inertia at the distance y from the aircraft longitudinal axis (evaluated using CadQuery), b – the wing span (doubled wing console length), M_b – is the bending moment, that is calculated by numerically integrating the difference between the distributed aerodynamic load q_a, and the distributed gravity load q_g:

$$q_a(\alpha, y) = (C_L(\alpha, y)\cos\alpha + + C_D(\alpha, y) \sin\alpha)\frac{\rho v^2}{2} c_0, \quad (4)$$

$$q_g(y) = \frac{dm_c(y)}{dy} g, \quad (5)$$

where c_0 is the – wing root chord; ρ is – air density; α – angle of attack; $C_L(\alpha, y)$ – local lift coefficient, obtained by multiplying lift coefficient of the airfoil by the lift efficiency evaluated from the response surface model (Fig. 3), that already includes local chord to root chord ratio; $C_D(\alpha, y)$ – local drag coefficient, evaluated as a sum of the zero drag and induced drag coefficients; and v is the airspeed.

Numerical integration is implemented inside the wing KBE model as a Python method of the SweptWingConsole class. The important part here is the ability to inform the algorithm with data from the CAD kernel: for each finite slice of the wing (from the tip to the chord), the slice mass and the component sections' inertia moments are computed. This allows to avoid the need for building mesh and solving FEM equations, which significantly simplifies the workflow.

4. Case study - wing rigidity analysis for different airfoil types

To demonstrate the computational benefits of the proposed knowledge-based approach, several wing console models with different airfoils (Fig. 9) and design parameters (sweep angle and aspect ratio) were generated and analyzed. No comparison with FEM approach was performed on this stage but this work will be conduced in the future.

For each of the selected airfoils (NACA 2410, NACA 2412 and NACA 2415) a range of wing models with different wing spans, varying from 500 mm to 2500 mm were generated for three values of sweep angle: 12, 15 and 18 degrees. A fixed value of the taper ratio (TR=0.75) was selected for all cases to reduce variability in the study. All models have the same values of the box thickness (0.5 mm) and shell thickness (0.25 mm) for the same considerations. The aspect ratio was changing from 6 to 9 for all cases.

For all cases same flight conditions were analyzed: airspeed 70 m/s, angle of attack 1 degree. All initial parameters a grouped in Table 1.

Table 1

Solution cases

Airfoil	NACA 2410	NACA 2412	NACA 2415
Taper ratio	0.75		
Aspect ratio	6, 7, 8, 9		
Sweep angle ◦	12, 15, 18		
Wing Span, mm	500..2500 with step 100		
Average time for a single model analysis, s	~20		

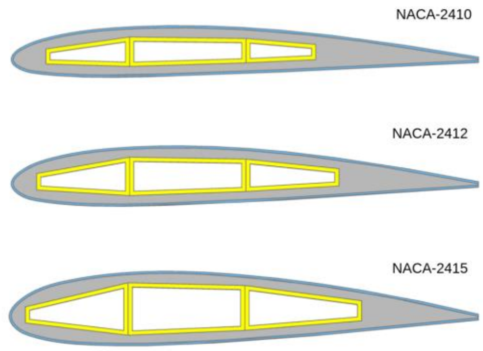


Fig. 9. Generated airfoil sections for selected airfoils

For all wing models, the same materials for the wing structural components were considered (Table 2).

Table 2

Material properties

Component	Box	Shaper	Shell
Material	Carbon composite	XPS foam	Fiberglass coat
Density, kg/m^3	1500	30	1800
Tensile strength, MPa	450	0.001	290
Tensile modulus, GPa	35	0.025	12.4

4.1. Results and Discussion

Results of each case study were uploaded to SplineCloud open repository [15], and the following relations were constructed on the platform as a function of the wing console excess lift force (a pure lift force that includes the wing weight):

– Aerodynamic quality;
– Bend rigidity;
– Wing span.

Such relations allow for further analysis and can be reused in MDO applications to quickly find the optimal solution (with sufficient rigidity and high aerodynamic quality). Response curve evaluation time takes under a millisecond on the modern computers, comparing to hours or even days for a high fidelity FEM or CFD simulation execution.

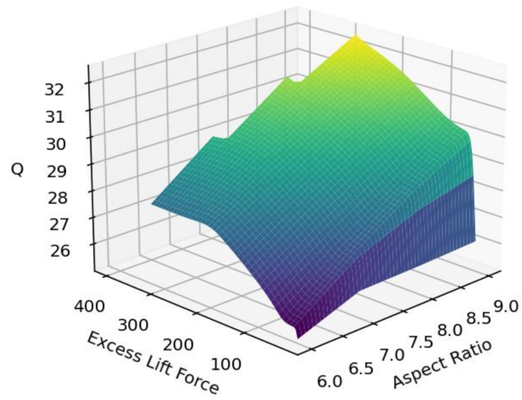


Fig. 10. Aerodynamic quality of the generated NACA 2410 wing console families with sweep angle 12◦

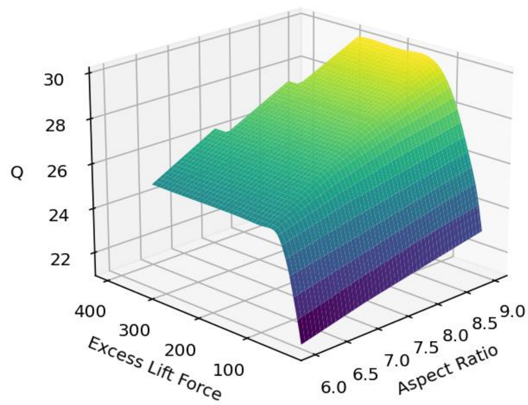


Fig. 11. Aerodynamic quality of the generated NACA 2412 wing console families with sweep angle 12◦

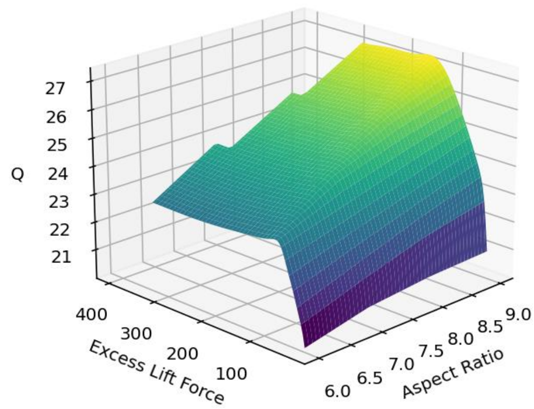


Fig. 12. Aerodynamic quality of the generated NACA 2415 wing console families with sweep angle 12◦

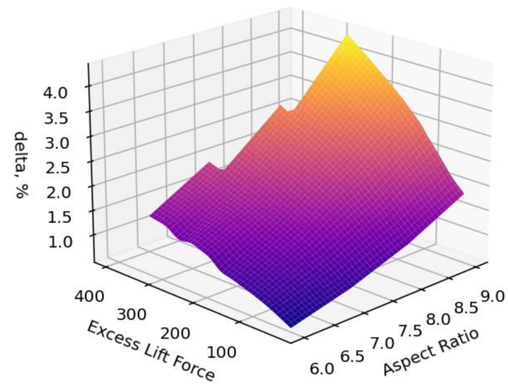


Fig. 13. Relative wing tip bend deflection of the generated NACA 2410 wing console families with sweep angle 12◦

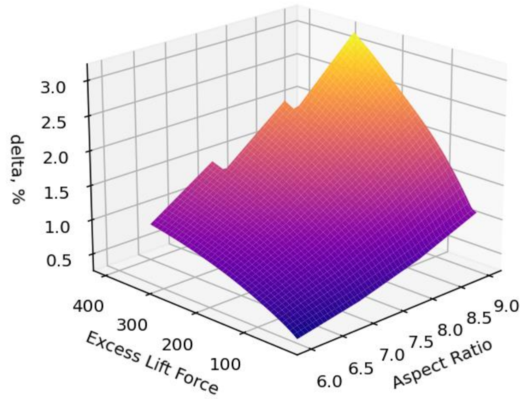


Fig. 14. Relative wing tip bend deflection of the generated NACA 2412 wing console families with sweep angle 12◦

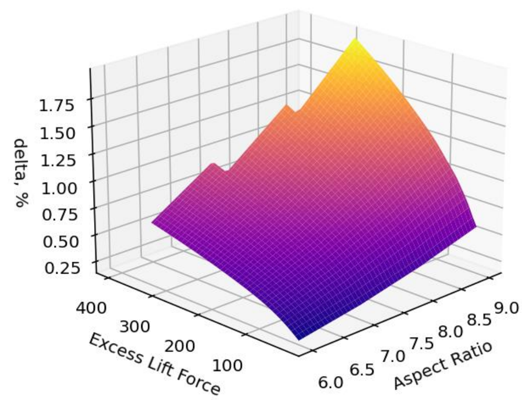


Fig. 15. Relative wing tip bend deflection of the generated NACA 2415 wing console families with sweep angle 12◦

The response curves were loaded into Python code for analysis and converted into response surface models using linear interpolation by the wing aspect ratio. The images of these surfaces are presented in Fig. 10-15.

Results show that, as expected, wings with thinner airfoils have lower rigidity, but at the same time have higher quality. The peculiar observation from the results is the steep drop in the aerodynamic quality for small wings (those that produce lower lift). This can be explained by lower numbers of Reynolds numbers, resulting in a lower airfoil performance.

Response surfaces for other sweep angles (15° and 18°) are not presented in the current paper due to the discovered small difference in both aerodynamic quality and relative tip deflection with the families of wing console models with a 12° sweep angle.

The construed response surfaces of the parametric studies can be considered as 2nd-order meta models (1st-order meta-models were used to evaluate aerodynamics inside the wing console KBE model). These meta models can significantly speed up multidisciplinary design optimization processes and inform decision making in the early stages of the design with formalized knowledge about the approximate (due to accepted assumptions and simplifications) wing performance.

One example of the reuse of the obtained meta-models is the comparison of the wing consoles with different airfoils (Table 3).

Table 3

Comparison of wing models with the same rigidity

Airfoil	NACA 2410	NACA 2412	NACA 2415
Excess lift force, N	200	200	200
Relative wing tip bending deflection, %	1.5	1.5	1.5
Wing aspect ratio	6.36	7.24	8.6
Aerodynamic quality	28.4	27.5	26.7

The analysis shows that, having the same excess lift force and relative rigidity, wings with thinner airfoils (NASA 2410) require a lower aspect ratio and at the same time have slightly higher aerodynamic quality. Such results can inform decision-making in the early stages and prove wrong initial guesses based on the general recommendations, which say that "increasing the aspect ratio generally improves aerodynamic quality". Such discrepancy is caused by deep modeling and multidisciplinary analysis, which includes wing mass and rigidity estimation, and not only geometrical and aerodynamic properties.

Another example of the possible reuse of the obtained meta-model is the iterative optimization process for wing size selection. In a hypothetical situation, when designing a UAV, the size and mass of the fuselage depend on the volume of the payload and systems, including a battery or tank with fuel. The volume of these energy storage systems depends in turn on the UAV's drag and aerodynamic quality. So, only by running an iterative matching process is it possible (in a computationally feasible way) to find the wing shape and size that satisfies the required technical requirements. In this process, the computational speed of the single iteration can play a critical role in selecting a truly optimal combination of design parameters. With 2nd-order meta models of the wings (evaluation of which takes under a millisecond on modern machines), the design optimization problem becomes computationally feasible, given that other calculation processes do not take significantly longer time.

Conclusions

The proposed approach proves effective in addressing the computational challenges inherent in early-stage UAV wing design, particularly in the context of multidisciplinary design optimization and automation. By integrating a custom mesh-free method for wing rigidity analysis within a generative knowledge-based engineering (KBE) framework, the methodology enables rapid evaluation of structural responses without the need for computationally intensive mesh generation or finite element pre-processing.

The use of programmable CAD tools and automated geometric property extraction facilitates a seamless link between parametric geometry generation and structural analysis, significantly improving workflow efficiency. The incorporation of reusable aerodynamic meta-models further enhances the framework by decoupling aerodynamic load estimation from structural analysis, enabling rapid evaluation of aerodynamic loads. However, this puts a limit on the applicability of such an approach to the rigid wings, where aeroelastic effects can be neglected.

Preliminary results demonstrate that the mesh-free method provides notable reductions in computational cost. This makes the approach particularly suitable for iterative design processes where a large number of configurations must be assessed within limited time frames. Nevertheless, the accuracy of the implemented mesh-free method for wing rigidity analysis has to be validated by comparing results with FEM-based methods.

Future work will focus on conducting the validation of the proposed method through comparison with high-fidelity FEM simulations, incorporating aeroelastic coupling effects, and extending the framework to support more complex UAV configurations and load scenarios. The presented methodology lays the groundwork for scalable, automated, and knowledge-driven UAV design processes compatible with modern open-source design ecosystems.

Contributions of authors: conceptualization, methodology – **Vadym Pasko**; formulation of tasks, analysis – **Vadym Pasko, Sviatoslav Yutskevych**; development of mathematical model for wing rigidity analysis – **Sviatoslav Yutskevych**; development of computational model, analysis of results, visualization – **Vadym Pasko**; writing – original draft preparation – **Vadym Pasko**, writing – review and editing – **Sviatoslav Yutskevych.**

Conflict of Interest

The authors declare that they have no conflict of interest regarding this research, whether financial, personal, authorship, or otherwise, that could affect the research and its results presented in this paper.

Financing

This study was conducted without financial support.

Data Availability

The work has associated data in the data repository [15]

Use of Artificial Intelligence

The authors have used artificial intelligence technologies within acceptable limits to provide their own verified data, as described in the research methodology section.

Acknowledgments

The conducted work was enabled by the following open-source software: Python, NumPy, SciPy, CadQuery, Jupyter Notebook, Matplotlib, and the open knowledge management platform SplineCloud.

All the authors have read and agreed to the published version of this manuscript.

References (GOST 7.1:2006)

1. *OpenMDAO: an open-source framework for multidisciplinary design, analysis, and optimization [Text] / J. S. Gray, J. T. Hwang, J. R. R. A. Martins, & et al. // Structural and Multidisciplinary Optimization. – 2019. – Vol. 59, № 4. – P. 1075–1104.*

2. *Hwang, J. T. A computational architecture for coupling heterogeneous numerical models and computing coupled derivatives [Text] / J. T. Hwang, & J. R. R. A. Martins // ACM Transactions on Mathematical Software (TOMS). – 2018. – Vol. 44, iss. 4. – P. 1–39.*

3. *Sobieszczanski-Sobieski, J. Multidisciplinary aerospace design optimization: Survey of recent developments [Text] / J. Sobieszczanski-Sobieski, & R. T. Haftka // Structural Optimization. – 1997. – Vol. 14, iss. 1. – P. 1–23.*

4. *La Rocca, G. Knowledge-based engineering: Between AI and CAD. Review of a language-based technology to support engineering design [Text] / G. La Rocca // Advanced Engineering Informatics. – 2012. – Vol. 26, iss 2. – P. 159–179.*

5. *Chen, W. Deep generative model for efficient 3D airfoil parameterization and generation [Text] / W. Chen, & A. Ramamurthy // AIAA Scitech 2021 Forum. – 2021. – P. 1–12.*

6. *Simultaneous shape and topology optimization of wings [Text] / L. Høghøj, C. Conlan-Smith, O. Sigmund, & C. S. Andreasen // Structural and Multidisciplinary Optimization. – 2023. – Vol. 66. – P. 1–22.*

7. *Martins, J. R. R. A. Multidisciplinary design optimization: A survey of architectures [Text] / J. R. R. A. Martins, & A. B. Lambe // AIAA Journal. – 2013. – Vol. 51, iss. 9. – P. 2049–2075.*

8. *Flexible and robust CAD models for design automation [Text] / K. Amadori, M. Tarkian, J. Ölvander, & P. Krus // Advanced Engineering Informatics. – 2012. – Vol. 26, iss. 2. – P. 180–195.*

9. *A critical review of knowledge-based engineering: An identification of research challenges [Text] / W. J. C. Verhagen, P. Bermell-Garcia, & R. E. C. van Dijk, R. Curran // Advanced Engineering Informatics. – 2012. – Vol. 26, iss. 1. – P. 5–15.*

10. *La Rocca, G. Knowledge-based engineering to support aircraft multidisciplinary design and optimization [Text] / G. La Rocca, & M. J. L. van Tooren // Proceedings of the Institution of Mechanical Engineers, Part G: Journal of Aerospace Engineering. – 2007. – Vol. 221, iss. 6. – P. 977–991.*

11. *Pedersen, A. Y. The evolution of knowledge-based engineering from a design research perspective : Literature review 2012–2021 [Text]/ A. Y. Pedersen, A. Clausen, & N. H. Mortensen // Advanced Engineering Informatics. – 2023. – Vol. 55. – 500 p.*

12. *Chapman, C. B. The application of a knowledge-based engineering approach to the rapid design and analysis of an automotive structure [Text] / C. B. Chapman, & M. Pinfold // Advances in Engineering Software. – 2001. – Vol. 32, iss. 12. – P. 903–912.*

13. *Knowledge-Based Aerospace Engineering — A Systematic Literature Review [Text] / T. Wittenborg, I. Baimuratov, L. Knöös Franzén, & et al. // arXiv preprint, arXiv:2505.10142. – 2025. – P. 1–36.*

14. *McCormick, B. W. Aerodynamics, Aeronautics, and Flight Mechanics [Text] / B. W. McCormick. – 2nd ed. – Chichester: John Wiley & Sons, 1995. – 652 p.*

15. *Swept wing generated results [Electronic resource]. – Available at: https://splinecloud.com/repository/nomad-vagabond/Swept_wing_generated_results/. – Date of access: 15.06.2025.*

References (BSI)

1. Gray, J. S., Hwang, J. T., Martins, J. R. R. A., Moore, K. T., Naylor, B. A., Brelje, B. J., & et al. OpenMDAO: an open-source framework for multidisciplinary design, analysis, and optimization. *Structural and Multidisciplinary Optimization*, 2019, vol. 59, no. 4, pp. 1075–1104.

2. Hwang, J. T., & Martins, J. R. R. A. A computational architecture for coupling heterogeneous numerical models and computing coupled derivatives. *ACM Transactions on Mathematical Software (TOMS)*, 2018, vol. 44, no. 4, pp. 1–39.

3. Sobieszczanski-Sobieski, J., & Haftka, R. T. Multidisciplinary aerospace design optimization: Survey of recent developments. *Structural Optimization*, 1997, vol. 14, no. 1, pp. 1–23.

4. La Rocca, G. Knowledge-based engineering: Between AI and CAD. Review of a language-based technology to support engineering design. *Advanced Engineering Informatics*, 2012, vol. 26, no. 2, pp. 159–179.

5. Chen, W., & Ramamurthy, A. Deep generative model for efficient 3D airfoil parameterization and generation. *AIAA Scitech 2021 Forum,* 2021, pp. 1–12.

6. Høghøj, L., Conlan-Smith, C., Sigmund, O., & Andreasen, C. S. Simultaneous shape and topology optimization of wings. *Structural and Multidisciplinary Optimization,* 2023, vol. 66, pp. 1–22.

7. Martins, J. R. R. A., & Lambe, A. B. Multidisciplinary design optimization: A survey of architectures. *AIAA Journal,* 2013, vol. 51, no. 9, pp. 2049–2075.

8. Amadori, K., Tarkian, M., Ölvander, J., & Krus, P. Flexible and robust CAD models for design automation. *Advanced Engineering Informatics*, 2012, vol. 26, no. 2, pp. 180–195.

9. Verhagen, W. J. C., Bermell-Garcia, P., van Dijk, R. E. C., & Curran, R. A critical review of knowledge-based engineering: An identification of research challenges. *Advanced Engineering Informatics,* 2012, vol. 26, no. 1, pp. 5–15.

10. La Rocca, G., & van Tooren, M. J. L. Knowledge-based engineering to support aircraft multidisciplinary design and optimization. *Proceedings of the Institution of Mechanical Engineers, Part G: Journal of Aerospace Engineering,* 2007, vol. 221, no. 6, pp. 977–991.

11. Pedersen, A. Y., Clausen, A., & Mortensen, N. H. The evolution of knowledge-based engineering from a design research perspective: Literature review 2012–2021. *Advanced Engineering Informatics,* 2023, vol. 55. 500 p.

12. Chapman, C. B., & Pinfold, M. The application of a knowledge-based engineering approach to the rapid design and analysis of an automotive structure. *Advances in Engineering Software,* 2001, vol. 32, no. 12, pp. 903–912.

13. Wittenborg, T., Baimuratov, I., Knöös Franzén, L., Staack, I., Römer, U., & Auer, S. Knowledge-Based Aerospace Engineering – A Systematic Literature Review. *arXiv preprint, arXiv:2505.10142,* 2025, pp. 1–36.

14. McCormick, B. W. Aerodynamics, *Aeronautics, and Flight Mechanics.* 2nd ed. Chichester, John Wiley & Sons, 1995. 652 p.

15. *Swept wing generated results (online "SplineCloud").* Available at: https://splinecloud.com/repository/nomad-vagabond/Swept_wing_generated_results/ (accessed 15 June 2025).

Received 15.06.2025, Accepted 18.08.2025

ЗАСТОСУВАННЯ БЕЗСІТКОВИХ МЕТОДІВ В АНАЛІЗІ ЖОРСТКОСТІ КРИЛА ДЛЯ ПІДТРИМКИ АВТОМАТИЗАЦІЇ ПРОЕКТУВАННЯ БПЛА

В. І. Пасько, С. С. Юцкевич

Сучасне проектування літальних апаратів, включаючи як пілотовані літаки, так і безпілотні літальні апарати (БПЛА), стикається з обчислювальними проблемами балансування аеродинамічної ефективності, структурної цілісності та оптимізації ваги в практичні терміни. Традиційні високоточні методи створюють вузькі місця, що обмежують дослідження простору проєктування, необхідне для розробки БПЛА. У цій статті представлено обчислювальну базу, що інтегрує безсітковий структурний аналіз із генеративною інженерією на основі знань (KBE) та сурогатним моделюванням для швидкої автоматизованої оптимізації конструкції крила БПЛА.

Методологія інтегрує формалізацію класичних знань з аеродинаміки та механіки конструкцій із програмованим CAD, використовуючи відкритий Python-пакет CadQuery. Розроблена платформа автоматично генерує параметричні геометрії крил, визначає геометричні характеристики, зокрема моменти інерції поперечного перерізу та об’єми, й виконує структурний аналіз без створення сітки чи попередньої обробки для методу скінченних елементів. Аеродинамічні навантаження оцінюються за допомогою повторно використовуваних мета-моделей із CFD-досліджень, збережених як B-сплайнові апроксимації в SplineCloud, що забезпечує розподілені робочі процеси та швидке оцінювання.

Безсітковий алгоритм реалізує числове інтегрування рівнянь згину балок, враховуючи розподілені аеродинамічні сили та навантаження від ваги зі змінними властивостями поперечного перерізу. Це усуває обчислювальні витрати на створення сітки, зберігаючи достатню точність для попереднього проєктування. Робочий процес інтегровано в KBE-модель крила, яка автоматизує генерацію геометрії та структурне оцінювання для крил зі стрілоподібною формою, різними матеріалами та геометріями, включно з різними типами профілів.

Валідаційні дослідження проводилися на трьох сімействах профілів NACA (2410, 2412, 2415) із подовженнями крила (6–9), кутами стрілоподібності (12°–18°) та розмахами (500–2500 мм). Окремі оцінки виконуються приблизно за 20 секунд порівняно з годинами чи днями для FEM-розрахунків, що забезпечує підвищення ефективності на 2–3 порядки. Створені мета-моделі другого порядку дозволяють проводити оцінки з субмілісекундною швидкістю, що підходить для ітераційної оптимізації, яка потребує тисяч оцінок.

Дослідження сприяє розвитку методологій автоматизованого проєктування, пропонуючи обчислювально ефективну альтернативу високоточним підходам зі збереженням інженерної точності для попередньої оптимізації. Реалізація з відкритим кодом забезпечує доступність для спільноти розробників БПЛА. Подальші дослідження зосереджені на валідації методом **скінченних елементів, аероеластичному зв'язку та розширенні на складніші конфігурації.**

Ключові слова: літак; БпЛА; автоматизація проєктування; безсіткові методи; чисельний аналіз; напруження; міцність; міждисциплінарна оптимізація; програмоване CAD; інженерія заснована на знаннях.

Пасько Вадим Ігорович – асп., Державний університет «Київський авіаційний інститут», Київ, Україна.

Юцкевич Святослав Сергійович – канд. техн. наук, декан аерокосмічного факультету, Державний університет «Київський авіаційний інститут», Київ, Україна.

Vadym Pasko – MSc in Aerospace Engineering, PhD Student, State University "Kyiv Aviation Institute", Kyiv, Ukraine,
e-mail: keenon3d@gmail.com, ORCID: 0009-0001-7739-2204.

Sviatoslav Yutskevych – Candidate of Technical Sciences, Dean of the Aerospace Faculty, State University "Kyiv Aviation Institute", Kyiv, Ukraine,
e-mail: sviatoslav.yutskevych@npp.kai.edu.ua, ORCID: 0000-0001-6650-4453, Scopus Author ID: 59710426900.