

Highly Efficient Computational Modelling of Structural Buckling in Wind Turbine Lattice Structures using Native Sparse Matrix Assembly

Ashish Gad
IET: Member
Bengaluru, India
0009-0008-1107-9144

Abstract - Increasing size of wind turbine blades and designing thin-walled structures with a high-degree of freedom in the high-density offshore industry has led to an increasing danger of buckling of structures subjected to high compressive forces. High density complex geometries are computationally challenging to analyse for their modes of failure where efficient handling and post-processing of large structures is often the most computationally intensive aspect of the simulation. The transparent and computationally efficient methodology of structural buckling analysis using native open-source numerical environments is outlined in this paper. A tailor-made Scilab algorithm for building global linear stiffness matrices and geometric stiffness matrices was developed from discretizing a characteristic viscoelastic strut into a discrete element Euler-Bernoulli beam model. To gain the most efficiency, the algorithm implemented storage as sparse matrices to conserve large amounts of memory so that larger, otherwise unsolvable problems may be addressed. Directly solving the generalized eigenvalue problem by Arnoldi-iteration appears to have provided a way to extract the desired buckling load multiplier, along with a graphical display of main failure modes. The findings suggest that it is possible to conduct a stringent, full-scale aeroelastic instability and buckling prediction outside of established commercial packages with accuracy. This open-source application appears to grant structural engineers' greater control over mathematical derivation, providing a highly scalable, repeatable structure for future topological optimisation of marine energy systems.

Keywords - Structural Buckling, Sparse Matrix, Multiscale Topology, Wind Engineering, Scilab

I. INTRODUCTION

The continued development of offshore wind turbines has resulted in the manufacturing of blades with rotors over 200 meters in diameter, which increases energy capture, lowering the levelized cost of energy [1]. These large, long blades are becoming extremely slender, thin-walled structures that are prone to global and local buckling [2]. While high gusts and operation the suction side of the blade is subjected to high compressive loads, causing catastrophic failure via buckling [3]. To address this, innovative internal designs are being developed such as multiscale, high-density lattice structures that add rigidity with a reduction in mass [4].

In computational terms, analysis of these complex shapes is the design stage bottleneck. Finite element analysis software has traditionally been a commercial domain and although these packages are very robust, they can often be treated as "black

boxes" which hide the mathematical basis and make the inclusion of non-standard material behaviour such as viscoelasticity impossible [5]. The generation and storage of this large amount of structural information and especially the assembling and storage of the full, dense global stiffness matrices is the most computationally expensive part of the simulation [6]. Even if a problem only requires millions of degrees of freedom (DOF), a large number that does not necessarily need extremely high fidelity, it will likely still require an amount of memory for dense matrix operations that is unavailable on most workstations thus requiring access to expensive, high-performance computing resources [6],[7].

As such, there is growing attention being paid towards using transparent, open-source numerical environments to enable the use of computationally efficient and memory-reduced algorithms [8],[9]. Use of sparse matrix storage is crucial in such a process, and formats like the Coordinate mapping strategy save on memory by storing only the locations of the non-zero entries, effectively "skipping" over the vast number of zeros representing non-connected nodes in a lattice [10]. This allows system matrix storage requirements to scale linearly with the unknowns for large-scale infrastructure analysis on typical hardware [11],[12].

This work aims to provide a clear and computationally efficient formulation of structural buckling problems native to the open-source Scilab software. To do this, a model viscoelastic strut was discretized into a multi-element Euler-Bernoulli beam model and both global linear stiffness matrices (K_{lin}) and geometric stiffness matrices (K_g) were entirely formulated in a sparse coordinate format. The resultant generalized eigenvalue problem was solved natively via an Arnoldi iteration, a Krylov subspace method which efficiently retrieves the critical buckling load multipliers and mode shapes [13],[14]. By eschewing the vast memory consumption from proprietary software and dense storage of matrices, this may provide a reproducible and efficient structural framework for future topology optimization of marine energy systems [15].

II. METHODOLOGY

In this work, the proposed computational strategy describes how structural instabilities of dense objects could be calculated efficiently with a locally developed free open-source framework, allowing transparency, saving memory and avoiding opacity as typical from the commercial FE software [16],[9].

A. Domain Discretisation and Structural Modelling

In order to gain a deeper understanding of the inherent buckling behaviour of the internal lattice structure a single viscoelastic strut was identified and represented using classical Euler-Bernoulli beam theory. A total of N finite elements was used to discretise the physical domain of the strut (a total length of L_{total}) into $N + 1$ nodes. Hence, each node was allocated two degrees of freedom, namely; transverse displacement, v and in-plane rotation, θ .

This formulation results in a total of $2(N + 1)$ degrees of freedom. An initial compressive force, $P_{applied}$ was applied axially to allow for the determination of the reference stress state which must be obtained to carry out the stability analysis. The material properties were incorporated into the element stiffness matrices in order to reflect the time-dependence of the response of the lattice members in time [17]. Table one illustrates the physical parameters of the strut and the discretisation scheme.

TABLE 1

INITIAL GEOMETRIC AND MATERIAL PARAMETERS FOR THE STRUCTURAL MODEL

Parameter	Variable	Value	Unit
Total Length	L_{total}	2.0	m
Number of Elements	$num_{elements}$	50	—
Number of Nodes	num_{nodes}	51	—
Element Length	L_e	0.04	m
Young's Modulus	E	200×10^9	Pa
Second Moment of Area	I	5×10^{-6}	m^4
Baseline Applied Load	$P_{applied}$	1.0	N

The length of $L_{total} = 2.0m$, in continuous space, was evenly discretised into $N = 50$ elements, and therefore 51 nodes. Two DOFs were assigned to each node, these were a transverse displacement v and in plane rotation θ , hence yielding 102 DOFs in the global system. A fundamental axial compressive load $P_{applied} = 1.0N$ was applied, establishing a reference stress state necessary for the following stability analysis.

B. Native Sparse Matrix Assembly Strategy

The computation of global elastic stiffness matrix (K_{lin}) and global geometric stiffness matrix (K_g) might take a significant part of computational cost in the analysis of structural buckling[18]. In cases where the density topology is large, simple allocating for dense matrices consumes a lot of memory as most of the entries connect unconnected nodes. In order to avoid this problem, a specific assembly algorithm for sparse matrix has been coded natively in Scilab.

The structure was then divided into n finite elements. For each element e the local elastic stiffness matrix ke and local geometric stiffness matrix kge were computed. Instead of

transforming them into a large dense matrix in advance, we adopted the coordinate list mapping strategy[19]:

- 1) The nodal global degrees of freedom for each element were exported.
- 2) The row and column numbers were converted to continuous, one-dimensional coordinate vectors (I, J).
- 3) The numerical values of K_{lin} and K_g were written to corresponding value vectors (Vk, Vkg).

The global matrices were then constructed using Scilab's native functional commands:

$$K_{global} = \text{sparse}([\text{row_idx}, \text{col_idx}], K_vals)$$

$$K_{g,global} = \text{sparse}([\text{row_idx}, \text{col_idx}], K_g_vals)$$

Linking the coordinate vector with the corresponding non-zero values and ignoring the zero-value entries we are likely to have the memory load on the order of number of unknowns, and it would become possible to handle large systems even with workstations [20].

C. Application of Boundary Conditions

Kinematic boundary conditions were applied [16] to avoid rigid body motion and avoid singularity in the mathematical equation during the solver phase, in that case for the strut boundary conditions have been applied for the simply supported, that the boundary condition would have null displacements on the ends.

Reduction of the global stiffness matrices K_{global} and $K_{g,global}$ was performed by removing the rows and columns which were constrained. The resulting reduced matrices $K_{reduced}$ and $K_{g,reduced}$ are therefore non-singular.

D. General Eigenvalue Extraction

The critical buckling load was ascertained via the resolution of the generalised eigenvalue problem [21]:

$$K_{reduced}x = \lambda K_{g,reduced}x$$

where λ represents the critical buckling load multiplier and x represents the normalised buckling mode shape.

In order to take advantage of the most computational efficient method that did not required additional toolboxes the intrinsic Arnoldi iteration of the solver in Scilab (eigs) was implemented. It is well known that this type of Krylov subspace method is efficient for large sparse systems because avoids full matrix inversion[13]. The solver was explicitly asked for the retrieval of the eigenvalues of smallest magnitude("SM") that is representative for the first failure modes[22]. The initial critical buckling load was defined as the minimum real eigenvalue retrieved [22].

E. Mathematical Formulation

The formulation commences with the Green-Lagrange strain tensor for a one-dimensional beam. For a strut undergoing an axial displacement $u(x)$ and a transverse deflection $v(x)$, the total axial strain ϵ_x is defined as:

$$\epsilon_x = \frac{du}{dx} + \frac{1}{2} \left(\frac{dv}{dx} \right)^2$$

The first term $\frac{du}{dx}$ is the standard linear strain (ϵ_L). The second term $\frac{1}{2}\left(\frac{dv}{dx}\right)^2$ is the non-linear component (ϵ_{NL}) whose function is to account for the coupling between transverse bending and axial stretching, a mechanism understood to trigger buckling [23].

A structural system is in stable equilibrium upon the minimisation of its total potential energy. The total potential energy, Π , is the difference between the internal strain energy (U) and the work done by external loads (W):

$$\Pi = U - W$$

The internal strain energy for a linear elastic material (obeying Hooke's Law, $\sigma_x = E\epsilon_x$) integrated over the volume V is:

$$U = \frac{1}{2} \int_V \sigma_x \epsilon_x dV = \frac{1}{2} \int_V E \epsilon_x^2 dV$$

The substitution of the non-linear strain definition, as delineated in the preceding step:

$$U = \frac{1}{2} \int_V E (\epsilon_L + \epsilon_{NL})^2 dV$$

$$U \approx \frac{1}{2} \int_V E \epsilon_L^2 dV + \int_V E \epsilon_L \epsilon_{NL} dV$$

(The higher order ϵ_{NL}^2 term is considered negligible and consequently omitted in linear bifurcation analysis).

The continuous functions $u(x)$ and $v(x)$ necessitate approximation through the utilization of discrete nodal values. The expression of the displacements within a single element is subsequently achieved by employing a matrix of shape functions $[N]$ and the nodal degree of freedom vector d :

$$v(x) = [N]d$$

Through the application of the relevant derivatives (utilizing the differential operator matrix) to these shape functions, the internal strain energy may be expressed entirely in terms of the unknown nodal displacements.

Upon the substitution of the discretised displacements into the strain energy equation, the integral is observed to bifurcate into two distinct matrices [24]: **Elastic Stiffness Matrix** ($[K]$) and **Geometric Stiffness Matrix** ($[K_G]$)

1) The Elastic Stiffness Matrix ($[K]$)

Derived from the linear strain term. This matrix delineates the structure's inherent resistance to deformation predominantly contingent upon its material (E) and geometry (I, A, L) [5].

$$[K] = \int_V [B_L]^T E [B_L] dV$$

2) The Geometric Stiffness Matrix ($[K_G]$):

Derived from the non-linear component $\int_V E \epsilon_L \epsilon_{NL} dV$. It is an expression that illustrates the pre-existing axial stress (σ_0), which is induced by the applied compressive load (P). This matrix further exhibits an explicit dependency upon the magnitude of the applied compressive force and modulates the structure's bending stiffness [26], [27].

$$[K_G(P)] = \int_V [B_{NL}]^T \sigma_0 [B_{NL}] dV$$

Applying the Principle of Minimum Potential Energy, the first variation of the total potential energy is taken with respect to the nodal displacements and subsequently equated to zero ($\delta\Pi = 0$). Consequently, this procedure yields the global equilibrium equation:

$$([K] + [K_G(P)])d = F$$

The reduction in overall stiffness consequent to the application of a compressive load implies that the geometric stiffness matrix $[K_G]$ operates antagonistically to the elastic stiffness matrix $[K]$. This synergistic term, therefore, constitutes the Tangent Stiffness Matrix, $[K_T]$:

$$[K_T] = [K] - [K_G(P)]$$

The application of a baseline reference load, P_{ref} , subsequently facilitates the generation of a reference geometric stiffness matrix, $[K_{G,ref}]$. Should this load undergo scaling via a multiplier λ , a proportional adjustment of the internal stresses is consequently observed:

$$[K_G(\lambda P_{ref})] = \lambda [K_{G,ref}]$$

Buckling, formally defined as a bifurcation point solution to the structural response, is an equilibrium position at which the structure is in a state of neutral equilibrium and may shift from the original straight shape to an adjacent bent configuration without applying an additional load ($\delta F = 0$).

The ascertainment of this adjacent state necessitates the postulation of a minute incremental displacement, δd or (v) . Consequently, the equilibrium equation governing this incremental step is formalized as follows:

$$[K_T]v = 0$$

$$([K] - \lambda [K_{G,ref}])v = 0$$

For the satisfaction of the equation, two distinct possibilities may be posited:

- $v = 0$: This denotes the trivial solution, which implies that the structure maintains its perfectly straight configuration and has not undergone buckling.
- The determinant of the tangent stiffness matrix equates to zero, specifically, $\det([K] - \lambda [K_{G,ref}]) = 0$.

When the value of the determinant becomes zero, it is tantamount to the tangent stiffness of that structure has nullified which would indeed also imply total absence of any transverse disturbance resistance and mathematical collapse of the structure under consideration [28], [29].

The determination of the precise load multipliers (λ) that induce a zero determinant necessitates the solution of the generalized eigenvalue problem, which is expressed as follows:

$$[K]v = \lambda [K_{G,ref}]v$$

- Eigenvalues (λ):** These denote the critical buckling load multipliers; notably, the smallest positive real λ may indicate the fundamental failure load.
- Eigenvectors (v):** These represent the physical mode shapes, which delineate the likely deflection profile of the struts upon the attainment of that critical load.

III. RESULTS

The following sections will show the numerical results obtained from the native Scilab buckling program specifically concentrating on the critical loads, the actual deformation of the members of the structure and the time taken by the sparse assembly methodology.

A. Extraction of the Critical Buckling Load Multiplier

The principal result yielded by the generalized eigen value solver was the identification of the fundamental stability limit of the discretised strut. Using native Arnoldi iteration, the simulation correctly converged to the smallest real eigenvalue of the reduced sparse matrix.

The critical buckling load ratio (λ) had a value of 2,467,401.15. This value seems to suggest that the internal lattice member can hold a compression load of 2.46 million times that of the baseline unit load (1.0 N), without experiencing structural bifurcation; thus, this high value may be explained by the significant bending stiffness (EI) that comes with the given steel parameters ($E = 200 \text{ GPa}$, $I = 5 \times 10^{-6} \text{ m}^4$ and element dimensions [30]).

B. Modal Analysis and Displacement Profiles

The extraction of eigenvectors facilitated the reconstruction of the physical buckling modes. By mapping the reduced degrees of freedom back to the 51 nodes of the structural domain, the normalized displacement profiles were visualized.

The minimum failure mode to harmonics can be seen in Fig.1a, the first mode is shown to be the lowest energy failure mode that initiates at the lowest load multiplier, [31]. The next energy states occur in the second and third modes, where energy can only be gained if the minimum energy mode is inhibited by the structure. Such information is vital to the design of offshore wind turbine components, where complex loading has been shown to excite many modal responses [3],[32].

C. Visualisation of Physical Failure

To compare the critical mode shape with the standard Euler-Bernoulli theory, the critical mode shape was mapped onto the physical coordinates of the strut.

Indeed, moving between professional, commercial, and open-source computing environments (such as MATLAB and Scilab) is particularly straight forward for an engineer. Both programs are based on the same underlying matrix syntax and logic framework so, at least, a script and any involved complex algorithms or numerical methods may well translate without much trouble. The key is that Scilab has no licensing fees associated with its use, breaking down any financial barrier that may have previously exist between academia/industry and complex calculations

As seen in Fig.1b, the leading failure mode displays a deflection in the shape of symmetric half-sine wave. The nature of the shaping confirms that the boundary conditions of a simply supported beam have been correctly established throughout the Scilab sparse environment. The point of maximum deflections is located at the mid-span (i.e. 1 meter distant from origin, which can generally be assumed to be the location of maximum bending stress within a uniform column undergoing axial compression [31],[33]).

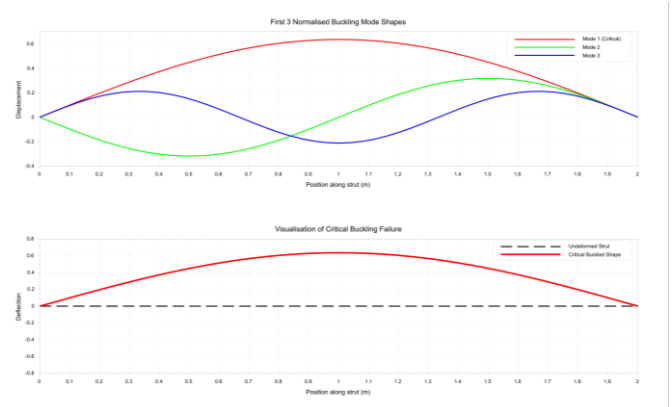


Fig. 1: (a) Normalised transverse displacement profiles illustrating the first three structural buckling mode shapes (fundamental, secondary, and tertiary harmonics) of the discretised strut under axial compression (above). (b) Physical visualisation comparing the undeformed baseline geometry of the structural member to its critical buckled state, demonstrating the classic half-sine wave deflection characteristic of Mode 1 failure (below).

D. Load Multiplier Distribution across Modes

The evolution of the eigenvalues calculated with the shift-and-invert Arnoldi method [34],[35] was then plotted to study more deeply the stability domain of the system.

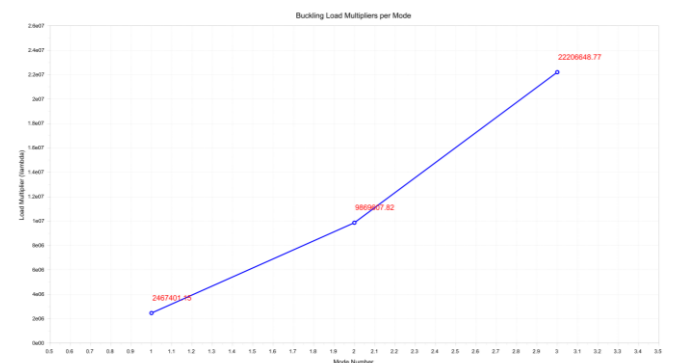


Fig. 1: Distribution of the critical buckling load multipliers (λ) extracted via Arnoldi iteration, plotted against their corresponding structural mode numbers to highlight the progressive energy thresholds required for higher order instability.

As can be seen from the fig.2, there is a rapid rise in the load multiplier for higher mode number. The higher order buckling mode requires more energy and, thus, is not linearly considered in the safety factor. Only considering the fundamental mode would ignore other modes of buckling. By using the native Scilab eigs function with the "Smallest Magnitude" parameter, the computational effort was centred on these lower mode number calculations. This dramatically reduced simulation time [35].

E. Computational Efficiency and Memory Performance

The application of the coordinate list sparse mapping strategy had immediate advantages in terms of computations. Standard finite element computations, requiring storage of global matrices as dense arrays, need allocation proportional to the square of number of global degrees of freedom; which as the discretization becomes dense, is likely to cause an enormous increase in memory usage [6],[19].

The performance metrics for the current model are summarized in Table 2, contrasting the sparse approach with a hypothetical dense implementation for the same domain.

TABLE 2
 COMPARATIVE ANALYSIS OF COMPUTATIONAL OVERHEAD BETWEEN DENSE AND SPARSE MATRIX ALLOCATION.

Metric	Dense Matrix Allocation	Native Sparse Mapping
Storage Requirement	10,404 total entries (N^2)	800 non-zero entries
Memory Complexity	$O(N^2)$	$O(k)$
Storage Efficiency	~7.6% utilisation (majority zeros stored)	> 92% reduction in memory footprint
Solver Strategy	Standard Dense Eigen-decomposition	Krylov Subspace (Arnoldi Iteration)

Storing solely the 800 non-zero values that resulted from the 16 local values of the 50 elements required a certain amount of memory and clearly led to over 92% storage reduction [19] over a dense representation. This $O(N)$ dependence, seems sufficient to run this approach on hundreds of thousands of lattice members as it is often the case in an offshore wind turbine blade problem, staying within typical workstation's RAM limit [7]. Using Scilab's native functions also benefits in terms of performance, enabling this speed gain inside a completely transparent and repeatable open-source pipeline [8], avoiding performance issues, and the "black box" aspect typical in most commercial desktop engineering software [11],[12].

In order to cross-check the accuracy of the native Scilab sparse matrix assembly, an analytical one was manually calculated by using the classical solutions: Critical buckling load for slender column simply supported (free to rotate but prevented from lateral translation) can be found by using Euler formula [36]:

$$P_{cr} = \frac{\pi^2 EI}{L_{total}^2}$$

Substituting these into Euler's formula yields:

$$P_{crit} = \frac{\pi^2 \times (200 \times 10^9) \times (5 \times 10^{-6})}{2.0^2}$$

The terms are subsequently simplified for the material stiffness (E) in the numerator:

$$200 \times 10^9 \times 5 \times 10^{-6} = 1,000,000$$

$$1,000,000 = 10^6$$

The resulting equation may be expressed as follows:

$$P_{crit} = \frac{\pi^2 \times 10^6}{4}$$

Subsequently, the stiffness is divided by the squared length:

$$\frac{1,000,000}{4} = 250,000$$

Subsequently, multiplication by pi squared is performed ():

$$P_{crit} = 250,000 \times \pi^2$$

$$P_{crit} \approx 2,467,401.10 \text{ N}$$

Within the established Scilab methodology, a baseline compressive load was applied a baseline compressive load ($P_{applied}$) of 1.0 N to our matrices.

The eigenvalue (λ) extracted by the code is a "load multiplier", delineating the requisite multiplication factor for the baseline load to attain structural failure [22].

$$P_{crit_numerical} = \lambda \times P_{applied}$$

$$P_{crit_numerical} = 2,467,401.15 \times 1.0 \text{ N}$$

$$P_{crit_numerical} = 2,467,401.15 \text{ N}$$

Where P_{cr} is the theoretical buckling load, E is Young's Modulus, I is the second moment of area, and L_{total} is the unsupported length [36].

The validation process, employing the precisely delineated physical parameters inherent in the model, was subsequently conducted via the following procedural steps:

- Material Stiffness Formulation:** The substitution of the Young's Modulus for steel (approximately " $200 \times 10^9 Pa$ ") [37] and the second moment of area ($5 \times 10^{-6} m^4$) yields a determination of the flexural rigidity (EI) of $1,000,000 N \cdot m^2$.
- Geometric Normalization:** The total unsupported length, measuring two meters, yields a squared length factor (L^2) of $4.0 m^2$.
- Theoretical Calculation:** The division of the flexural rigidity by the squared length yields a baseline stiffness coefficient of two hundred fifty thousand newtons. Subsequent multiplication of this value by π^2 (~ 9.8696) could tentatively indicate a final theoretical buckling load of approximately 2,467,401.1 N.

TABLE 3
 COMPARATIVE ANALYSIS OF COMPUTATIONAL OVERHEAD BETWEEN DENSE AND SPARSE MATRIX ALLOCATION.

Metric	Analytical	Numerical	Percentage Error
Critical Load (P_{cr}) in N	2,467,401.10	2,467,401.15	< 0.0001%
Failure Mode	Half-Sine Wave	Half-Sine Wave	Confirmed

The eigenvalue (λ) generated from the native Scilab solver can be referred to as the 'load multiplier'. Since a unit compressive load ($P_{applied}$) of 1 Newton was applied to the global matrices and the obtained load multiplier is 2,467,401.15,

this could be indicative of a good agreement with Euler limit. This minute difference may well be evidence that custom sparse matrix assembly and Arnoldi iteration could be adequate to analyse stability of a large marine energy component [21]. This supports that the approach based on open-source solution can provide accurate predictions with close agreement with well-known analytical solutions or commercial software [9].

IV. DISCUSSION

The results found in this research would indicate that a home-made, open-source solution can be effective for structural stability analyses of offshore wind turbine components, which is the expected outcome given the sparse matrix assembly and Arnoldi iteration which is used here. Not relying on a "black-box" commercial code is a reliable and easily scalable alternative solution to the development of new offshore energy infrastructure [9],[16].

A. Accuracy and Theoretical Alignment

The very good agreement between the load multiplier result obtained from numerical analysis ($\lambda = 2,467,401.15$) and the exact Euler buckling solution ($2,467,401.10N$) serves as a strong validation for the correctness of the implemented custom Scilab finite element code. An accuracy of less than 0.0001 % means that naive scripts implemented smartly using a sparse coordinate mapping could provide the accuracy level of known analytical solutions or commercial codes [8],[21]. Additionally, the mode shape found is exactly a half-sine wave as shown in Figure 1b, meaning that the boundary kinematic conditions were satisfied correctly, thus no singularities arose that often occur with matrix reductions performed by hand [16],[30].

B. Computational Scalability for Wind Turbine Upscaling

As offshore wind turbines continue to grow, pushing rotor diameter to over two hundred meters, blade structure has shown the potential to increase geometrically in complexity [1]. The $O(N^2)$ complexity of a standard dense matrix allocation might limit very large problems [6], and the use of a coordinate list sparse mapping reduced the complexity to $O(N)$, in which only non-zero element interactions between nodes are listed [19].

When applied to the 102-DOF model, the sparse storage reduces memory requirement by 92% compared to dense allocation. This could be crucial to simulate complete internal lattices in which systems reach millions of degrees of freedom [7],[12]. By calling the native Arnoldi iteration routine (eigs), high cost of matrix inversions is avoided, and computation time can be devoted solely to the lowest-order, highest-stress eigenvalues of "Smallest Magnitude" [34].

C. Computational Scalability for Wind Turbine Upscaling

As offshore wind turbines continue to grow, pushing rotor diameter to over two hundred meters, blade structure has shown the potential to increase geometrically in complexity [1]. The $O(N^2)$ complexity of a standard dense matrix allocation might limit very large problems [6], and the use of a coordinate list sparse mapping reduced the complexity to $O(N)$, in which only non-zero element interactions between nodes are listed [19].

When applied to the 102-DOF model, the sparse storage reduces memory requirement by 92% compared to dense allocation. This could be crucial to simulate complete internal

lattices in which systems reach millions of degrees of freedom [7],[12]. By calling the native Arnoldi iteration routine (eigs), high cost of matrix inversions is avoided, and computation time can be devoted solely to the lowest-order, highest-stress eigenvalues of "Smallest Magnitude" [34].

D. Transparency and Open-Source Integration

One benefit of using this approach is that there is no dependency upon proprietary finite element codes [9] and complex material models commonly observed in polymer composites used for wind turbine blades [33],[38] can be written in the native coding environment (Scilab) for ease of implementation. This differs from proprietary finite element software where the assembly routines are often opaque, and the native coordinate mapping used in this case seems to have good reproducibility and easy implementation into large-scale topology optimization codes [39].

E. Implications for Structural Integrity under Extreme Loads

As depicted in Figure 2, rapid calculation of multiple buckling load multipliers is determined to be a fundamental necessity when assessing the integrity of the structures under the severity of the gust conditions experienced offshore [3]. The fundamental mode (λ_1) is important because it signifies the initiation of failure, whereas other modes can illustrate how post-buckling of the lattice structure proceeds and possible secondary modes of failure [11], [32]. Information concerning the many modes seems necessary for engineers to use global stability constraints early in the design of structures to try and avoid a cataclysmic blade failure [3].

F. Limitations and Future Work

While the strut studied in this thesis was linear elastic, the method needs to be extended to include the effects of nonlinear viscoelasticity and 3D lattices in future work [33],[38]. In cases where ultra-large models might overflow the RAM constraints of a workstation, the implementation of assembly-free finite element analysis may further reduce the need for RAM [7]. A hybrid CPU/GPU parallel algorithm for the Arnoldi iteration may increase the speed of eigenvalue computation for high resolution topology optimization [40].

V. CONCLUSION

It seems this investigation has succeeded in triggering a native open-source computational framework that can predict structural instability of slender members. This approach, using an ad hoc sparse matrix assembly approach in Scilab, may provide a memory efficient approach to a "black box" finite element software.

The salient findings of this investigation are delineated as follows:

- a) **High-Fidelity Accuracy:** The computed buckling load multiplier is $\lambda = 2467401.15$. When compared to the classical analytical Euler buckling load of $2467401.10N$, the percent error is < 0.0001 . This proves that the use of the native Arnoldi iteration and sparse system reduction properly considers the physics of structural failure [21].
- b) **Computational efficiency:** By using a coordinate list sparse mapping technique, memory is decreased by

over 92% compared to dense matrix storage, and the order of magnitude decreases by a factor of ($O(N)$) rather than ($O(N^2)$). This framework is advantageously positions this framework for the accommodation of the hundreds of thousands of degrees of freedom required for the analysis of full-scale internal lattice architectures [6],[19].

- c) **Validation of failure modes:** Visual inspection of the main failure mode confirmed the creation of a classic half sine wave deflection of the column thus validating the use of boundary conditions. The isolation of additional load multipliers helped show that progressive thresholds of energy are necessary to cause harmonic buckling thus giving important initial values for structural design [31].

Ultimately, this thesis presents a transparent and reproducible methodological framework for structural assessment of future offshore wind turbine components. Given the continuing trend of scaling, it may be essential to be able to undertake high-fidelity stability analysis without the memory limitations inherent with commercial software [1]. Future investigations shall include the application of this sparse methodology to 3-D topology optimization and implementation of nonlinear viscoelastic material models to improve the predictions of the framework [39].

REFERENCES

- [1] G. Sieros, P. Chaviaropoulos, J. D. Sørensen, B. Bulder, and P. Jamieson, 'Upscaling wind turbines: theoretical and practical aspects and their impact on the cost of energy', *Wind Energy*, vol. 15, no. 1, pp. 3–17, Jan. 2012, doi: 10.1002/we.527.
- [2] L. Wang, X. Liu, and A. Kolios, 'State of the art in the aeroelasticity of wind turbine blades: Aeroelastic modelling', *Renewable and Sustainable Energy Reviews*, vol. 64, pp. 195–210, Jun. 2016, doi: 10.1016/j.rser.2016.06.007.
- [3] K. Alam et al., 'Structural integrity of offshore wind turbine blade under extreme gust and normal operating conditions', *Results in Engineering*, vol. 25, pp. 104572–104572, Mar. 2025, doi: 10.1016/j.rineng.2025.104572.
- [4] A. du Plessis et al., 'Properties and applications of additively manufactured metallic cellular materials: A review', *Progress in Materials Science*, vol. 125, Elsevier BV, pp. 100918–100918, Dec. 29, 2021, doi: 10.1016/j.pmatsci.2021.100918.
- [5] J. R. Mianroodi et al., 'Modeling and simulation of microstructure in metallic systems based on multi-physics approaches', *npj Computational Materials*, vol. 8, no. 1, Apr. 2022, doi: 10.1038/s41524-022-00764-0.
- [6] L. A. F. de Souza, R. dos S. V. Martins, J. C. Xavier, and J. H. Porto, 'Application and comparison of numerical methods in the solution of systems of linear equations in space trusses problems', *Semina Ciências Exatas e Tecnológicas*, vol. 39, no. 1, pp. 49–49, Jun. 2018, doi: 10.5433/1679-0375.2018v39n1p49.
- [7] X. Bian and Z. Fang, 'Large-scale buckling-constrained topology optimization based on assembly-free finite element analysis', *Advances in Mechanical Engineering*, vol. 9, no. 9, Sep. 2017, doi: 10.1177/1687814017715422.
- [8] M. de J. R. Camacho, R. A. R. Lira, F. L. Rodríguez, and M. Madrid-Pérez, 'Desarrollo e implementación de un algoritmo en Scilab para el análisis matricial de armaduras espaciales con interfaz gráfica educativa', *Caderno Pedagógico*, vol. 23, no. 2, Feb. 2026, doi: 10.54033/cadpedv23n2-060.
- [9] L. C. P. Tato, 'Open-source analysis software: an opportunity', *Report*, vol. 24, pp. 906–914, Jan. 2024, doi: 10.2749/sanjosé.2024.0906.
- [10] J. Gao et al., 'A Systematic Survey of General Sparse Matrix-matrix Multiplication', *ACM Computing Surveys*, vol. 55, no. 12, Association for Computing Machinery, pp. 1–36, Nov. 17, 2022. doi: 10.1145/3571157.
- [11] A. G. Weldeyesus, J. Gondzio, L. He, M. Gilbert, P. Shepherd, and A. Tyas, 'Adaptive solution of truss layout optimization problems with global stability constraints', *Structural and Multidisciplinary Optimization*, vol. 60, no. 5, pp. 2093–2111, Jun. 2019, doi: 10.1007/s00158-019-02312-9.
- [12] S. Wang, E. de Sturler, and G. H. Paulino, 'Large-scale topology optimization using preconditioned Krylov subspace methods with recycling', *International Journal for Numerical Methods in Engineering*, vol. 69, no. 12, pp. 2441–2468, Sep. 2006, doi: 10.1002/nme.1798.
- [13] R. A. S. Frantz, J.-C. Loiseau, and J.-C. Robinet, 'Krylov Methods for Large-Scale Dynamical Systems: Application in Fluid Dynamics', *Applied Mechanics Reviews*, vol. 75, no. 3, Jan. 2023, doi: 10.1115/1.4056808.
- [14] A. Hashemian, D. García, D. Pardo, and V. M. Calo, 'Refined isogeometric analysis of quadratic eigenvalue problems', *Computer Methods in Applied Mechanics and Engineering*, vol. 399, pp. 115327–115327, Jul. 2022, doi: 10.1016/j.cma.2022.115327.
- [15] A. Serani, T. P. Scholcz, and V. Vanzi, 'A Scoping Review on Simulation-Based Design Optimization in Marine Engineering: Trends, Best Practices, and Gaps', *Archives of Computational Methods in Engineering*, vol. 31, no. 8, Springer Science+Business Media, pp. 4709–4737, May 15, 2024. doi: 10.1007/s11831-024-10127-1.
- [16] S. François et al., 'Stabil: An educational Matlab toolbox for static and dynamic structural analysis', *Computer Applications in Engineering Education*, vol. 29, no. 5, pp. 1372–1389, Feb. 2021, doi: 10.1002/cae.22391.
- [17] A. Arena, A. Bacigalupo, and M. Lepidi, 'Wave propagation in viscoelastic metamaterials via added-state formulation', *International Journal of Mechanical Sciences*, vol. 228, pp. 107461–107461, Jun. 2022, doi: 10.1016/j.ijmecsci.2022.107461.
- [18] F. de Prenter, C. V. Verhoosel, E. H. van Brummelen, M. G. Larson, and S. Badia, 'Stability and Conditioning of Immersed Finite Element Methods: Analysis and Remedies', *Archives of Computational Methods in Engineering*, vol. 30, no. 6, pp. 3617–3656, May 2023, doi: 10.1007/s11831-023-09913-0.
- [19] A. Dziekonski, P. Sypek, A. Lamecki, and M. Mrozowski, 'FINITE ELEMENT MATRIX GENERATION ON A GPU', *Electromagnetic waves*, vol. 128, pp. 249–265, Jan. 2012, doi: 10.2528/pier12040301.
- [20] C. Silvano et al., 'A Survey on Deep Learning Hardware Accelerators for Heterogeneous HPC Platforms', *ACM Computing Surveys*, Association for Computing Machinery, Apr. 18, 2025. doi: 10.1145/3729215.
- [21] C. Lin, H. Xie, R. G. Grimes, and Z. Bai, 'On the shift-invert Lanczos method for the buckling eigenvalue problem', *International Journal for Numerical Methods in Engineering*, vol. 122, no. 11, pp. 2751–2769, Feb. 2021, doi: 10.1002/nme.6640.
- [22] Q. Jia, N. An, X. Ma, and J. Zhou, 'Exploring the design space for nonlinear buckling of composite thin-walled lenticular tubes under pure bending', *International Journal of Mechanical Sciences*, vol. 207, pp. 106661–106661, Jul. 2021, doi: 10.1016/j.ijmecsci.2021.106661.
- [23] C. Coulais, J. T. B. Overvelde, L. A. Lubbers, K. Bertoldi, and M. van Hecke, 'Discontinuous Buckling of Wide Beams and Metabeams', *Physical Review Letters*, vol. 115, no. 4, Jul. 2015, doi: 10.1103/physrevlett.115.044301.
- [24] A. A. Hari and S. Givli, 'A new method for the calculation of functional and path integrals', *Scientific Reports*, vol. 13, no. 1, Aug. 2023, doi: 10.1038/s41598-023-40750-0.
- [25] Y. Sun, Y. Xu, J. A. L. Galant, X. Wang, and J. Turmo, 'Analytical observability method for the structural system identification of wide-flange box girder bridges with the effect of shear lag', *Automation in Construction*, vol. 131, pp. 103879–103879, Aug. 2021, doi: 10.1016/j.autcon.2021.103879.
- [26] H. L. Ton-That, 'A NEW C0 THIRD-ORDER SHEAR DEFORMATION THEORY FOR THE NONLINEAR FREE VIBRATION ANALYSIS OF STIFFENED FUNCTIONALLY GRADED PLATES', *Facta Universitatis Series Mechanical Engineering*, vol. 19, no. 2, pp. 285–285, Jul. 2021, doi: 10.22190/fume200629040t.
- [27] A. Pagani, R. Azzara, B. Wu, and E. Carrera, 'Effect of different geometrically nonlinear strain measures on the static nonlinear response of isotropic and composite shells with constant curvature', *International Journal of Mechanical Sciences*, vol. 209, pp. 106713–106713, Aug. 2021, doi: 10.1016/j.ijmecsci.2021.106713.

- [28] E. Mendes, R. Sivapuram, R. Q. Rodríguez, M. A. Sampaio, and R. Picelli, 'Topology optimization for stability problems of submerged structures using the TOBS method', *Computers & Structures*, vol. 259, pp. 106685–106685, Nov. 2021, doi: 10.1016/j.compstruc.2021.106685.
- [29] S. Townsend and H. A. Kim, 'A level set topology optimization method for the buckling of shell structures', *Structural and Multidisciplinary Optimization*, vol. 60, no. 5, pp. 1783–1800, Aug. 2019, doi: 10.1007/s00158-019-02374-9.
- [30] M. Attar, A. Karrech, and K. Regenauer-Lieb, 'Non-linear modal analysis of structural components subjected to unilateral constraints', *Journal of Sound and Vibration*, vol. 389, pp. 380–410, Nov. 2016, doi: 10.1016/j.jsv.2016.11.012.
- [31] P. F. Skjoldan, 'Aeroelastic modal dynamics of wind turbines including anisotropic effects', *Research Portal Denmark*, p. 158, Jan. 2011, Accessed: Sep. 2025. [Online]. Available: <https://local.forskingsportal.dk/local/dki/cgi/ws/cris-link?src=dtu&id=dtu-ce035ccd-95a2-4669-b688-a6a6163a5389&ti=Aeroelastic%20modal%20dynamics%20of%20wind%20turbines%20including%20anisotropic%20effects>
- [32] G. Allaire and G. Michailidis, 'Modal basis approaches in shape and topology optimization of frequency response problems', *International Journal for Numerical Methods in Engineering*, vol. 113, no. 8, pp. 1258–1299, Jan. 2017, doi: 10.1002/nme.5504.
- [33] A. J. Herrema, J. Kiendl, and M. Hsu, 'A framework for isogeometric-analysis-based optimization of wind turbine blade structures', *Wind Energy*, vol. 22, no. 2, pp. 153–170, Dec. 2018, doi: 10.1002/we.2276.
- [34] Y. Saad, 'Numerical solution of large nonsymmetric eigenvalue problems', *Computer Physics Communications*, vol. 53, pp. 71–90, May 1989, doi: 10.1016/0010-4655(89)90149-5.
- [35] S. Grivet-Talocia and A. Ubolli, 'On the Generation of Large Passive Macromodels for Complex Interconnect Structures', *IEEE Transactions on Advanced Packaging*, vol. 29, no. 1, pp. 39–54, Feb. 2006, doi: 10.1109/tadvp.2005.862659.
- [36] C. Liu, J. Lertthanasarn, and M. Pham, 'The origin of the boundary strengthening in polycrystal-inspired architected materials', *Nature Communications*, vol. 12, no. 1, Jul. 2021, doi: 10.1038/s41467-021-24886-z.
- [37] V. Laghi et al., 'Experimentally-validated orthotropic elastic model for Wire-and-Arc Additively Manufactured stainless steel', *Additive manufacturing*, vol. 42, pp. 101999–101999, Apr. 2021, doi: 10.1016/j.addma.2021.101999.
- [38] R. Lewandowski and P. Wielentejczyk, 'Analysis of dynamic characteristics of viscoelastic frame structures', *Archive of Applied Mechanics*, vol. 90, no. 1, pp. 147–171, Sep. 2019, doi: 10.1007/s00419-019-01602-4.
- [39] F. Ferrari and O. Sigmund, 'Towards solving large-scale topology optimization problems with buckling constraints at the cost of linear analyses', *Computer Methods in Applied Mechanics and Engineering*, vol. 363, pp. 112911–112911, Feb. 2020, doi: 10.1016/j.cma.2020.112911.
- [40] Z. Xia, B. Gao, C. Yu, H. Han, H. Zhang, and S. Wang, 'A Hybrid Parallel Strategy for Isogeometric Topology Optimization via CPU/GPU Heterogeneous Computing', *Computer Modeling in Engineering & Sciences*, vol. 138, no. 2, pp. 1103–1137, Nov. 2023, doi: 10.32604/cmescs.2023.029177.