X3D v2.0: A Parallel 3D Finite Element Multibody Dynamic Analysis for Rotorcraft

Mrinalgouda Patil*,†
Post-Doctoral Research Associate
University of Maryland, College Park, MD

Ravi Lumba‡
U.S. Army Combat Capabilities Command Aviation & Missile Center
Moffett Field, CA

Anubhav Datta
Associate Professor
University of Maryland, College Park, MD

This paper presents X3D version 2.0, a parallel three-dimensional (3D) dynamic solver designed for rotor aeromechanics. Unlike current-generation rotor comprehensive codes, the solver's core structure is built upon the foundational principles of 3D finite element modeling unified with multibody joints. Notably, it distinguishes itself from commercially available finite element codes by incorporating an inherent internal aerodynamics module to facilitate the 3D computational fluid dynamics/computational structural dynamics coupling. The enhancements in version 2.0 are detailed in this paper, encompassing the integration of new parallel solution procedures, the introduction of capabilities such as coaxial rotor analysis, the exploration of novel 3D models, and the addition of new tools essential for generating 3D rotor models from computer-aided design geometry. The implementation of parallel algorithms significantly enhances the speed of this version, while the newly added capabilities make it conducive for exploring advanced rotor configurations.

Nomenclature

C_Q	torque÷ $\rho A(\Omega R)^2 R$
C_T	thrust÷ $\rho A(\Omega R)^2$
F_X	hub longitudinal shear force, lb, positive rearwards
F_Y	hub lateral shear force, lb, positive starboard
F_Z	hub vertical shear force, lb, positive up
LO	lift offset, $\%R$
M_X	hub roll moment, in-lb
M_Y	hub pitch moment, in-lb
M_{tip}	tip Mach number
R	rotor radius, ft
T	rotor thrust normal to hub plane, lb
V_∞	wind tunnel speed or freestream speed, ft/s
α_s	shaft tilt angle, deg, positive forward
θ_{1c}	lateral cyclic, deg
θ_{1s}	longitudinal cyclic, deg
μ	advance ratio÷ tip speed, ΩR
σ	rotor solidity
ψ	blade azimuth angle, deg
Ω	rotor rotation speed, rad/s

*Corresponding author; email: mpcsdspa@gmail.com.
Pre-revised version of this paper presented at the Vertical Flight Society's Sixth Decennial Aeromechanics SpecialistsâĽ™ Conference, Santa Clara, CA, February 6–8, 2024 (Ref. 1). Manuscript received March 2024; accepted March 2026.
†Currently Loads & Dynamics Engineer, Joby Aviation.
‡Currently Rotor Dynamics Engineer, Archer Aviation.

Background

Current rotorcraft structural dynamics analyses model the rotor blade as a slender one-dimensional (1D) composite beam, with two-dimensional (2D) finite element models used to calculate the structural and inertial cross-sectional properties of the blade (Ref. 2). This approach is well documented and provides adequate results for conventional rotorcraft problems due to the long slender shape of rotor blades. However, this method falls short in several key areas. Modern blades have material and geometric discontinuities due to design or ballistic damage that cannot be captured using beams. Modern hub components are three-dimensional (3D) structures that provide kinematic couplings and absorb maximum stresses and hence are crucial for both stability and weight. Today, modifications and enhancements are made to beam boundary conditions to accommodate these structures in a gross manner, but require fabrication and testing to measure their properties. The use of 3D models has the potential to overcome these barriers. The flexible parts of a rotor, the parts that produce strains, can be modeled from first principles with 3D finite elements and bearings that provide constraints modeled with multibody joints. Dynamic stresses and strains can be predicted in the blade and the hub components. This was noted by NASA, which identified 3D structures as an essential tool for next-generation rotorcraft

DOI: 10.4050/JAHS.71.042001

design (Ref. 3). This led to the development of X3D, an experimental (X) three-dimensional (3D) aeroelastic solver for rotors.

The development of X3D can be split into three phases: development of the initial solver, extension to real-world problems, and the parallelization of the solver. X3D development began in 2008, with the first prototype solver presented by Datta and Johnson in 2009 (Ref. 4). Multibody dynamics were soon added (Refs. 4, 5), followed by the first instance of integrated-3D (I3D) analysis, defined as 3D computational fluid dynamics (CFD) coupled with 3D computational structural dynamics (CSD), for an idealized UH-60A-like rotor (Ref. 6). This work represented the first validation of X3D against UH-60A flight-test data. Phase I concluded in 2016 with the release of X3D version 1.0 (Ref. 7). Two significant barriers were identified: lack of geometry and meshing tools and high computational cost. These made the routine use of 3D analysis impractical.

Phase 2 of development spanned 2014–2018 and focused on the first barrier identified in Phase 1. The University of Maryland developed 3D geometry and meshing tools for generic rotor blades (Ref. 8), supported by the U.S. Army's computer-aided design (CAD)-based Modeling of Rotary-Wing Structures (CMARS) program. This enabled the application of X3D to realistic problems, highlighted by the demonstration of I3D analysis for the NASA Tiltrotor Aeroacoustic Model (TRAM) (Refs. 9, 10), specially released by NASA for this task. This work also represented a validation of X3D with wind-tunnel test data of TRAM. However, during this phase, artificially coarsened meshes had to be used due to high computational run times.

Phase 3 of development began in 2018 and has focused on parallelization and scalability. This approach was two pronged in space (Ref. 11) and time (Ref. 12). The parallelization of the X3D solver, combined with the advancements in geometry and meshing made earlier during Phase 2, has allowed for 3D structures to be applied to important problems of current interest. These include Mars Helicopters (Refs. 13, 14), advanced tip geometry blades (Refs. 15–17), and lift offset coaxial rotors (Refs. 18, 19). X3D version 2.0 represents the culmination of Phases 2 and 3 of X3D development, further realizing the vision for next-generation rotorcraft analysis.

Organization of Paper

After detailing the development of the X3D solver, the paper starts by outlining the scope of version 2.0, listing the new features. Next, the 3D workflow is described for X3D, detailing the steps from CAD to analysis. Next, the significant updates introduced in X3D version 2.0 are detailed, including the parallel algorithms, preprocessing tools, and coaxial rotor analysis. The subsequent section lists the principal tasks of X3D version 2.0, highlighting its core functionalities. Following this, the paper provides a list of the available 3D models for utilization, along with sample analysis results for each of the models. Moving forward, the parallel performance exhibited by this new version is studied, shedding light on its efficiency and scalability. Finally, the paper ends with concluding observations.

X3D v2.0: Scope

The principal task of X3D version 1.0 was to model an isolated rotor in forward flight trim and prescribed maneuvers to predict dynamic 3D stresses/strains using CFD/CSD coupling. Version 2.0 expands the solver scope to include modeling coaxial rotors, including CFD/CSD coupling of coaxial rotors with the U.S. Army/CREATE^{TM}-AV Helios framework (Refs. 20, 21). Various pre- and post-processing tools are also introduced to enhance user accessibility. The main addition to X3D version 2.0 is the use of parallel and scalable algorithms that take advantage of modern supercomputer architecture to significantly reduce computational time. Much of the underlying theory, such as the material model, joint formulation, and 3D finite element derivation, is unchanged since version 1.0 (Ref. 7) and such is not included in this work.

X3D version 2.0 runs efficiently on a supercomputer architecture with hundreds to thousands of processors for problem sizes greater than one million degrees of freedom. This is a significant improvement over version 1.0, which only ran on a single processor for maximum problem sizes of 100 K degrees of freedom. This advancement is made possible with two special solvers: one parallel in space and the other parallel in time. Both are independent solvers. The parallel in space solver requires a special mesh partitioner to partition generic meshes from Cubit software (Ref. 22) for efficient performance. Cubit is a full-featured software toolkit for robust generation of 2D and 3D finite element meshes (grids) and geometry preparation. The parallel in time solver requires a parallel shared memory skyline solver for scalability. The integration of these two solvers is currently in work and will be part of the future release.

Review of X3D v1.0

This section provides a review of X3D version 1.0, as documented in Ref. 7. The purpose of this section is to summarize the established 3D workflow, enabling the updates introduced in the new version to be more clearly identified and understood.

3D workflow

3D modeling of rotors is a significant departure from 50 years of convention with beam-based models in aeromechanics. Consequently, the analysis setup demands an entirely new workflow. The primary steps involved in developing the model, from CAD to X3D inputs, were initially outlined in Ref. 7, presenting the first version of X3D. For the sake of completeness and incorporating updates, the same steps are underscored here. An example of the 3D workflow is demonstrated in Refs. 1 and 19 for the Metaltail-II rotor. Metailtail-II is a generic coaxial rotor inspired by the Sikorsky S-97 Raider that is intended to be used as a test case for integration into the U.S. Army/Department of Defense rotorcraft simulation software Computational Research and Engineering Acquisition Tools and Environments (CREATE^{TM})–Air Vehicles (AV) Helicopter Overset Simulations (Helios) framework.

The 3D workflow in X3D is shown in Fig. 1. The starting point of 3D modeling is the 3D CAD obtained from CATIA, Solidworks, or any equivalent software. A detailed CAD is not required; simple conceptual models suffice at early stages of design. The next step is to create the structural analysis representation (SAR). While creating the SAR, the user will divide the parts that are to be modeled into flexible parts and joint parts. Typically, any part which experiences strains is modeled as a flexible part, meaning that it will be meshed with 3D finite elements. Joint parts are parts that experience negligible strains but typically have large deformations, such as bearings, bolts, mechanisms, and devices.

Next, the flexible parts are meshed with 27-node isoparametric hexahedral brick finite elements. Currently, all X3D meshing is performed using the Cubit software (Ref. 22) but can also be done with an equivalent hexahedral meshing tool that outputs the mesh in UNV (universal IDEAS) file format. The 3D meshing is performed part by part, as individual meshes are generated for each flexible part. Each part mesh is assigned three features: blocks, sidesets, and nodesets. A block is a group of elements, where each block has a single set of material properties. A sideset is a group of element faces and is used to identify aerodynamic

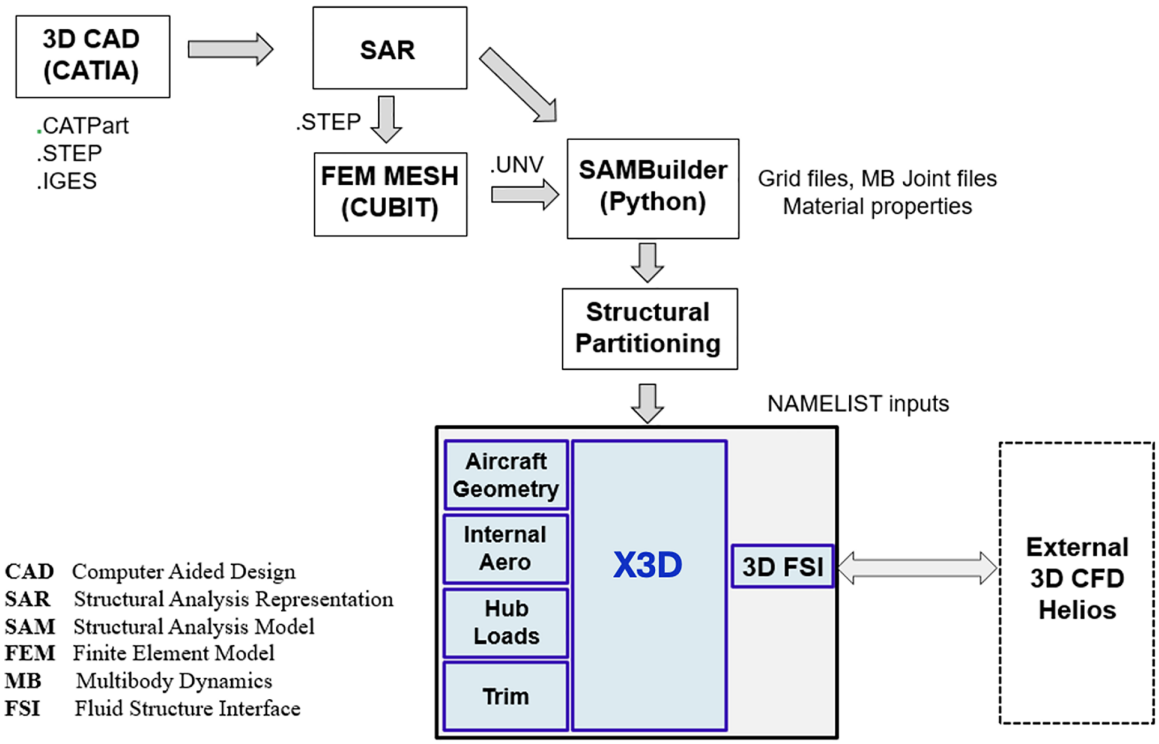


Fig. 1. Integrated 3D workflow; roadmap of steps needed to get from CAD to analysis.

surface nodes. A nodeset is a group of nodes and is used to determine joint connections. Additional details on the meshing of rotor blades and hub components can be found in Ref. 23.

The meshes and joints are then assembled to form the structural analysis model (SAM), which is an input to X3D (*SAM.input*). The SAM is based on the SAR created earlier. The SAM model orders all parts, defines material properties for meshes, assigns joint properties for joints, and creates the final model ready for X3D. A special tool called SAMBuilder is used to create the *SAM.input* file for X3D. This preprocessing tool is a new addition to the X3D version 2.0 package.

The next step is to physically partition the structure into separate subdomains for parallel execution. The SAM model is transferred to the structural mesh partitioner, developed specially for this work, where the structure is divided into multiple subdomains depending on the available computational resources. However, this step is only necessary if the parallel-in-space X3D solver is to be used. This is another addition to X3D version 2.0 and will be discussed in more detail in a later section.

Finally, the X3D inputs are generated in FORTRAN namelist-based text format. There are four input files:

1) *SDN.input:* System definition. It describes environment, air speeds, and top-level geometry of system, subsystems, and components.

2) *SAM.input:* Structural analysis model. It describes details of the structural model, including mesh files and joint files. The material properties and joint properties are defined here.

3) *AER.input:* Aerodynamic analysis model. It describes details of the (internal) aerodynamic model, including lifting line definition, airfoil tables, and fluid–structure interface. The CFD interface required for CFD/CSD coupling is defined here.

4) *SOL.input:* Solution procedure model. It describes details of the solution process, including various restart files for rotor solution, modes, trim Jacobian, and CFD delta files.

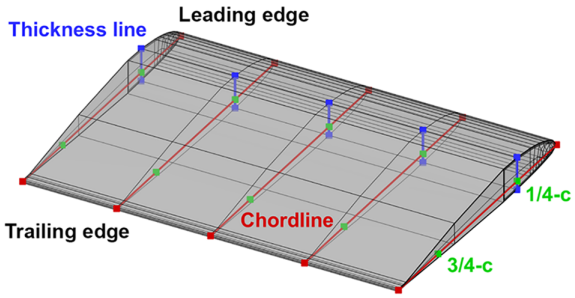
SDN.input and *SOL.input* are needed for all tasks. An aerodynamic-only analysis requires only *SDN*, *AER*, and *SOL* inputs. A structures-only analysis requires only *SDN*, *SAM*, and *SOL* inputs. A full-up coupled analysis requires all input files.

External CFD coupling

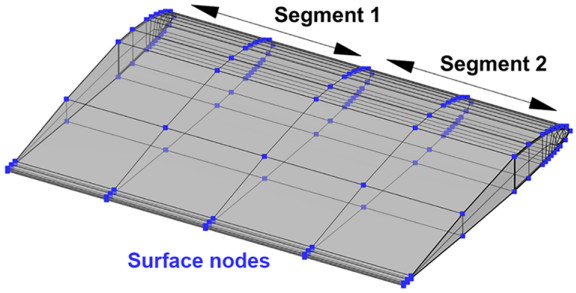
X3D v1.0 provides an interface for coupling with external CFD software. This capability will be summarized here for completeness to help understand the 3D CFD/CSD coupled results later. The interface is available through two methods: a file-based input/output (I/O) capability and a Python-based module. The file I/O coupling works with any generic CFD software, while the Python-based coupling is available with the U.S. Army/CREATE™-AV Helios framework (Refs. 20, 21).

As with X3D version 1.0, a Level-I interface is used for CFD/CSD coupling. A Level-I interface is beam-like, with sectional translations and rotations being passed to the CFD solver and sectional airloads being passed to X3D. All current rotorcraft CFD software use a 1D beam-like interface to exchange deformations and airloads, so this approach allows for existing CFD mesh motion schemes to be used. The development of an exact Level-II interface using 3D patch forces is underway at the University of Maryland.

Deformation interface. As part of the Level-I interface, the 1D deformations must be transferred to CFD. While these deformations come naturally with conventional beam-based analysis, they must be extracted from the 3D deformation field. For this purpose, cross-sectional planes must be defined as part of the model input. A single cross section is defined with four nodes, located at the leading-edge, trailing-edge, top surface, and bottom surface, as shown in Fig. 2(a). These are used to define a chord line and a thickness line. The chord line is uniquely defined, whereas the thickness line is not unique but is constrained by the airfoil geometry. From the chord line, the quarter- and three quarter-chord locations are obtained. The thickness line should be located close to quarter-chord,



(a) Level-I deformation interface; chord and thickness lines used to define cross-sectional plan



(b) Level-I deformation interface; sectional 2D airloads distributed over surface nodes

Fig. 2. Level-I fluid–structure interaction interface showing deformation definition and airload mapping.

but need not be perpendicular to the chord line. Each cross section is associated with a cross-sectional frame located at quarter-chord.

The beam-like deformations are extracted using the deformed chord and thickness lines. The three deflections (u, v, z) are obtained at a user-defined chord location (with a default of quarter-chord). The three rotations (θ_X, θ_Y, θ_Z) are defined by Euler angles extracted from the direction cosines of the deformed cross-sectional frame in any user-defined order, with the default order as lead–lag, flap, and pitch. The extraction of beam-like deformations excludes cross-sectional deformations outside the four nodes used to define the cross section but provides an adequate description of the 3D structure. A 3D description of the deflection field is ideal, but this would require modifications to CFD mesh motion routines, so it is not a part of this release.

Airloads interface. A Level-I interface uses classical 1D airloads that work with both lifting-line and CFD. The CFD solver integrates the pressure and shear to segmental airloads that are passed to X3D at the lifting line stations, the same procedure that is used for conventional beam-based analysis. These airloads are then distributed over the surface nodes of the 3D structural mesh.

The use of 1D segmental airloads implies that the 3D structural mesh must have clear sets of surface elements separated into spanwise segments. Within a segment, elements can be unstructured but the boundaries must line up in the chordwise direction. As is seen throughout the paper, 3D elements are almost always naturally aligned in the radial direction, making the segments easy to define.

Figure 2(b) shows an example of two spanwise segments. Each segment will receive the segmental airloads from the lifting line or CFD, which will be distributed over the surface nodes of that segment. For a given node, the nodal forces are assumed to be

$$f_N = \alpha_0 + \alpha_1 \epsilon; \quad f_C = \alpha_2 + \alpha_1 \eta^2 \tag{1}$$

where f_N and f_C are the normal and in-plane forces, respectively, ϵ and η are the chordwise and thicknesswise position of the node, and $\alpha_0, \alpha_1, \alpha_2$ are three coefficients to be solved for all nodes. Thus, the normal force is assumed to have a linear variation with chord and the chord force a quadratic variation with thickness. The segmental airloads are related to the nodal forces through Eq. (2), where the sum is over all surface nodes in a segment. The axial nodal force is ignored unless axial airloads are supplied, but these normally are negligible.

$$F_N = \sum f_N; \quad F_C = \sum f_C; \quad M_{25} = \sum (f_N \epsilon - f_C \eta) \tag{2}$$

For a total of n surface nodes, the total segmental airloads are equal to the assumed distribution given by Eq. (3).

$$\begin{bmatrix} F_N \\ F_C \\ M_{25} \end{bmatrix} = \begin{bmatrix} n & \sum \epsilon & 0 \\ 0 & \sum \eta^2 & n \\ \sum \epsilon & \sum (\epsilon^2 - \eta^3) & -\sum \eta \end{bmatrix} \begin{bmatrix} \alpha_0 \\ \alpha_1 \\ \alpha_2 \end{bmatrix} \tag{3}$$

The coefficients for each segment are obtained by inverting the matrix, and each segment will have a different set of coefficients. This ensures that, despite the assumed nodal distribution, the segmental airloads, including the pitching moment, are conserved within each segment. Only three constants can be evaluated from three airloads, which is why only three are used to express nodal forces. The internal stresses with a Level-I interface are dependent on the assumed distribution, but since the segmental airloads are conserved, they are expected to follow the general characteristics of external loading.

X3D version 1.0 was integrated into the U.S. Army/CREATE™-AV Helios framework through a Python-based module. The process was similar to beam-based comprehensive codes such as Rotorcraft Comprehensive Analysis System (RCAS) (Ref. 24) and Comprehensive Analytical Model for Rotorcraft Aerodynamics and Dynamics (CAMRADII) (Ref. 25). The Helios environment is equipped to execute the comprehensive analysis software on a single processor based on the current generation of CSD solvers. The Helios fluid–structure interface is equipped to receive beam-like deformations and pass segmental airloads, so a Level-I interface was the most straightforward approach. X3D version 2.0 is built upon the same framework as version 1.0 described above, but it has been extended to support applications involving multi-rotor systems. Results demonstrating these capabilities are presented in later sections, while the underlying methodology remains unchanged.

The following sections describe the updates introduced in version 2.0, including algorithmic enhancements, new analysis modules, and additional tools developed for the 3D framework.

Parallel Algorithms

The major development in version 2.0 is the inclusion of parallel algorithms. The solution procedure in X3D version 1.0 featured a time integration method with options to choose from standard numerical schemes: Newmark, three-point Euler backward, and generalized-alpha. The system of linear equations was solved using a serial skyline solver, executed on a single processor. In X3D version 2.0, the parallel-in-space and parallel-in-time approaches were added to take advantage of the modern supercomputer architecture. The former requires a specialized mesh partitioner and is driven by the finite element tearing and interconnecting dual primal (FETI-DP) algorithm (Refs. 26, 27), and the latter uses a frequency domain based modified harmonic balance (MHB) algorithm (Ref. 12). The serial skyline is now updated to a parallel version built for speed that is implemented using shared memory architecture (Ref. 12). The algorithm forms a kernel for the parallel in space and time approaches described below.

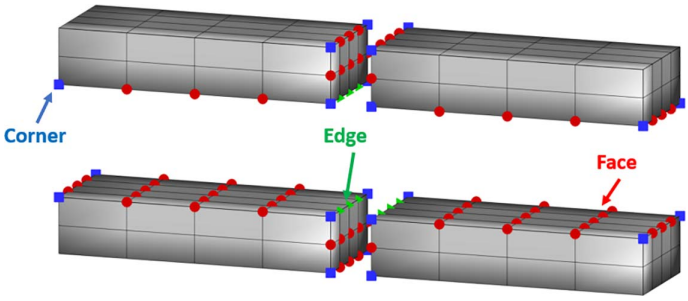


Fig. 3. Description of interface nodes; the red circles indicate face nodes, the green triangles edge nodes, and the blue squares mark the corner nodes.

FETI-DP solver

The parallel-in-space approach uses the FETI-DP algorithm with conjugate gradient updates. This approach still uses the time integration methods introduced in X3D version 1.0, with the iterative FETI-DP solver replacing the serial skyline solver. FETI-DP is a state-of-the-art iterative substructuring algorithm introduced by Farhat and co-workers (Refs. 26, 27). X3D version 1.0 contained an iterative solver equipped with the FETI-DP algorithm (Refs. 4, 28) but lacked a robust partitioner to handle complex rotorcraft structures. X3D version 2.0 includes such a partitioner (Ref. 11), which will be discussed in a later section.

In the FETI-DP algorithm, the structure is partitioned into non-overlapping subdomains. The subdomain nodes are divided into internal and interface nodes, where the interface nodes are shared by multiple subdomains and internal nodes are only present in a single subdomain. The interface nodes are further divided into corner, edge, and face nodes, as shown in Fig. 3. The corner nodes are used to create a coarse representation of the full structure. This is somewhat equivalent to treating each subdomain as a pseudo-element, whose nodes are the corner nodes. The better the coarse problem approximates the full structure, the better the convergence. At the same time, corner nodes must be selected such that locking the corner nodes for a subdomain eliminates its rigid body modes. This is required for all subdomains.

The corner nodes formulate a primal interface problem, where the primal variables remain the original degrees of freedom. The non-corner interface (edge and face) nodes form the dual interface problem. The dual variables are a set of Lagrangian multipliers used to enforce continuity across the interface between subdomains. A preconditioned conjugate gradient (PCG) method is used to solve for the multiplier's value. The continuity of displacements at the corner nodes is enforced at each PCG iteration.

The FETI-DP algorithm is implemented on a distributed memory architecture. Each subdomain is allocated to a single processor. The internal subdomain problem is solved with a direct skyline solver independent of other subdomains. The interface problem is solved iteratively with the PCG method, and this requires that subdomains communicate with neighbors to exchange the residual at the interface. This communication is handled using MPI (message passing interface). However, the preconditioner and preconditioned residual can be constructed in a parallel manner as each subdomain can calculate its own contribution with minimal communication overhead. During each PCG iteration, the coarse problem must be assembled and solved. Each subdomain calculates its contribution, and then the problem is assembled and solved by a single processor before being distributed out to all subdomains. The solution of the coarse problem is a serial step, so minimizing its computational cost will improve parallel scalability.

One alteration made to the FETI-DP algorithm from previous work with rotorcraft (Refs. 4, 28) is the use of a skyline solver for the coarse problem. Typically, the coarse problem is very sparse, so the use of a skyline solver significantly reduces the memory requirements. This not only decreases the coarse problem solution time, but also the communication overhead required for assembling the coarse problem. The authors showed that the use of the skyline storage practically eliminated the barrier of the coarse problem to scalability (Ref. 11).

If the structure has multibody joints, these must remain internal to a subdomain. This leaves the interface unaffected by the inclusion of multibody dynamics and leaves the numerical scalability of the FETI-DP algorithm unchanged. Therefore, the introduction of multibody joints does not have much effect on the solver, but instead places additional demands on the partitioner. Flexible parts cannot be partitioned on their own and connected via joints; the assembled structure must be attacked by the partitioner. The mesh partitioner will be covered in detail in a later section, starting with the requirements from the FETI-DP algorithm.

Modified harmonic balance

The parallel in time approach uses the MHB algorithm. The solution in time for every degree of freedom is decomposed into harmonics as a Fourier series. The governing equation in time is transformed into the frequency domain with coefficients of harmonics as the new variables to solve. These coefficients are solved in parallel on a distributed memory architecture (MPI). Each harmonic solve internally uses a skyline solver that is parallelized on shared memory (OpenMP). The details of the MHB algorithm and the parallelization of the skyline solver can be found in Ref. 12.

The overall algorithm is implemented on a combined shared and distributed memory architecture that is typical of a modern supercomputer. Figure 4 shows the schematic representation of the employment of shared and distributed memory processors in implementing MHB. Each harmonic solve is assigned to an individual MPI task, which internally unleashes multiple OpenMP threads for the parallel skyline solver. A total of $N + 1$ MPI tasks are assigned for obtaining a solution with N harmonics $(0, 1, 2, \ldots, N)$. Each MPI task is assigned with 19 shared memory processors (OpenMP programming). Overall, hybrid MPI–OpenMP parallelism involving a total of $19 \times (N + 1)$ processors is used. Note that for large-scale finite element structures with multibody joints, the construction of matrices and forcing vectors at all azimuths significantly contributes to the total solution time. This workload of construction is distributed across MPI tasks and is internally parallelized using OpenMP. This hybrid parallelism improves performance dramatically while reducing memory requirements for each processor.

In addition to the parallel algorithm, the solution procedure for helicopter rotors calls for a modification. An artificial damping with a relaxation factor is added and removed through iterations for faster convergence.

The MHB algorithm was implemented in X3D to build a new, refined, parallel version 2.0. In addition to the solution procedure, all other parts of X3D were parallelized using the shared memory architecture. Only the inputs and outputs were left serial. The details of its parallel performance are discussed in the Parallel Performance section.

Coaxial Rotor Analysis

X3D version 2.0 expands the scope of modeling rotors to multi-rotor configurations, primarily the coaxial rotor—a high-speed future vertical lift configuration. Unless otherwise noted, coaxial rotor refers to both articulated and lift offset configurations, with emphasis on the lift offset case. Lift offset coaxial rotors absorb high loads with stiff hingeless hubs.

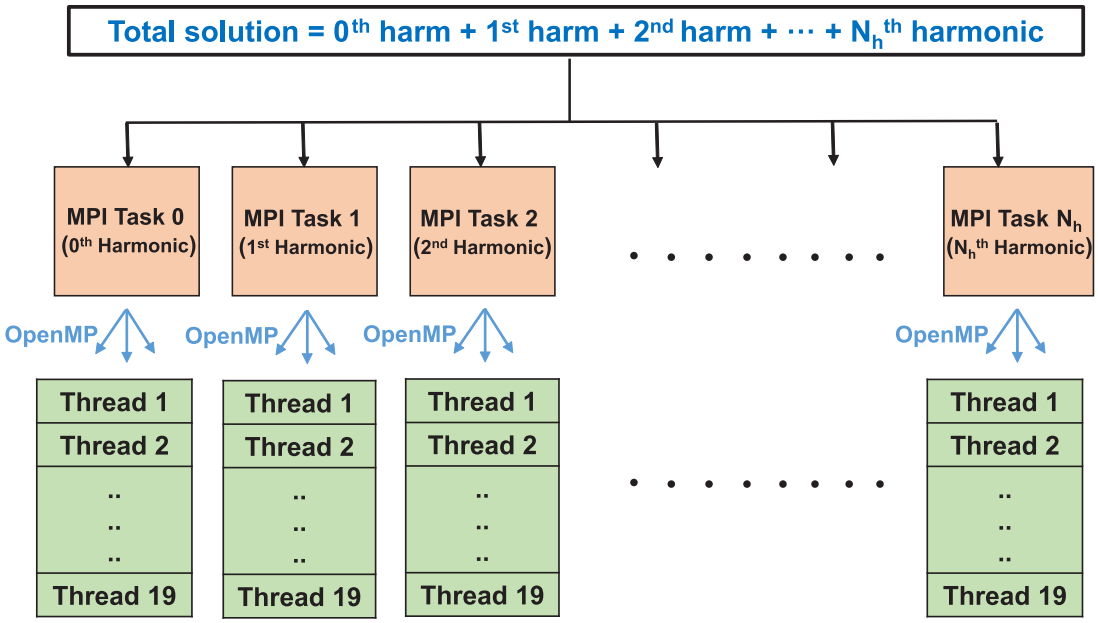


Fig. 4. Implementation of modified harmonic balance using shared and distributed memory architecture.

However, absorbing high loads within allowable stresses requires a heavy hub, which impacts the entire aircraft. To maximize the full potential of the lift offset coaxial, hub stresses must be accurately predicted, better understood, and ultimately mitigated with minimum weight. This makes X3D a very useful tool to study modern coaxial rotors.

The X3D extension to coaxial rotors is made possible through the following refinements.

Structural dynamics. The structural dynamics module of X3D is modified to allow modeling of an arbitrary number of rotors spinning in counterclockwise (CCW) or clockwise (CW) directions. The nominal analysis in X3D is carried out in a right-handed CCW frame. To model CW rotation, the calculations are performed as a CCW rotor, with the conditions, motions, and loads changing signs appropriately. For example, the wake analysis is in the fixed frame so blade motions into the wake and inflow out of the wake must be adjusted appropriately. The hub forces and moments for CW rotation can be obtained from the nominal analysis by changing the signs according to Eq. (4), where $F_X^{CCW}, F_Y^{CCW}, F_Z^{CCW}, M_X^{CCW}, M_Y^{CCW}$, and M_Z^{CCW} are the calculated values with CCW rotation.

$$\begin{aligned} F_X^{CW} &= F_X^{CCW}; \quad F_Y^{CW} = -F_Y^{CCW}; \quad F_Z^{CW} = F_Z^{CCW} \\ M_X^{CW} &= -M_X^{CCW}; \quad M_Y^{CW} = M_Y^{CCW}; \quad M_Z^{CW} = -M_Z^{CCW} \end{aligned} \tag{4}$$

For a coaxial rotor, X3D takes in only the input of the CCW rotor, and the analysis for both rotors is carried out in the right-handed CCW frame with appropriate sign changes for the CW rotor as mentioned above.

Aerodynamics. The aerodynamics of the coaxial rotors are modeled using quasi-steady lifting line theory coupled with time accurate free wake. X3D version 2.0 contains an updated version of the Maryland free wake model (Ref. 29), which uses a combination of vortex-based near-wake and far-wake that can analyze multi-rotors with variable and transient RPM (including stopped rotor). This new wake model brings in the inter-rotor interactions of a coaxial rotor.

Trim. The trim module is now appended with multiple options for coaxial trim. The upper and lower rotor collectives provide total thrust and torque balance. The cyclics can be solved with two options. In the first option, the lateral ($\theta_{1c}^U, \theta_{1c}^L$) and longitudinal ($\theta_{1s}^U, \theta_{1s}^L$) cyclics can be solved for specified roll and pitch moments on each rotor. In the second option, the lateral cyclic is kept the same for both upper and lower rotors ($\theta_{1c}^U = \theta_{1c}^L = \theta_{1c}$) and this value along with the two longitudinal ($\theta_{1s}^U, \theta_{1s}^L$) cyclics is obtained for a specified lift offset and total zero roll and pitch moments. This option is more similar to how real aircraft are trimmed.

X3D v2.0: List of Tasks

The tasks that X3D version 2.0 can perform are summarized in this section, along with their respective outputs. Each task executes a different solution procedure, and the solution flag for each task is included. These tasks can be performed for both single and coaxial rotor configurations. Note that all these tasks can be executed in parallel mode using the new X3D .

(1) *Structural response:* It calculates the linear and nonlinear static response of the model to prescribed forcing.

(2) *Frequency response:* It calculates the natural frequencies and mode shapes of a rotating or nonrotating structure. For a rotating blade, the converged blade response is obtained in vacuum at a steady rotational speed and the natural frequencies are taken about this point.

(3) *Aerodynamic response:* The aerodynamic response task is a special task involving only aerodynamics. It calculates the rotor response (airloads, inflow) for a given flight condition. The aerodynamic analysis can be performed with no deformations (rigid rotor) or prescribed deformations.

(4) *Hover response:* The hover task is a full-up capability including structures, aerodynamics, and controls for ideal hover (axisymmetric flow with collective control only). This task invokes X3D's internal lifting-line aerodynamics. One option iterates the inflow at a fixed collective to calculate the converged response, while another performs a full trim in which both collective and inflow are iterated to achieve the targeted thrust.

(5) *Forward flight response:* The forward flight task is a full-up capability (structures, aerodynamics, and controls) for a vehicle in edgewise flight. One option iterates the inflow at fixed control angles to calculate

Table 1. X3D models. Models with (*) are restricted

Models	Configuration Type	Validated for	Description of Model
Articulated rotor	SMR	Frequencies	Representative geometry Representative materials
UH-60A-like	SMR	Blade airloads	Representative geometry Representative materials
UMD DARB	SMR	Strains, frequencies	Exact geometry Exact materials
NASA ROAMX blade (*)	SMR	Hover strains	Exact geometry Exact materials
NASA TRAM (*)	Tiltrotor	Airloads, blade loads	Exact geometry Representative materials
Maryland Tiltrotor Rig (MTR) blades	Tiltrotor	Strains, frequencies	Exact geometry Exact materials
Metaltail-II	Coaxial rotor	Blade airloads	Notional geometry Notional materials
Mars Ingenuity blade (*)	Coaxial rotor	Frequencies	Exact geometry Representative materials

the converged response, while another performs full-up trim where control angles are iterated, along with inflow, to reach the targeted thrust and moments.

(6) *Trim with CFD/CSD:* The forward flight trim is performed with CFD/CSD coupling. The CFD coupling can be via file I/O or a Python driver if using Helios. After a full-up trim solution is obtained, the deformations are sent to CFD. The next trim solution is obtained with the CFD delta airloads. The CFD/CSD iterations are repeated until the control angles are converged.

X3D v2.0: List of Models

The structural modeling in X3D is a departure from conventional rotorcraft aeromechanics, therefore 3D models are provided as examples. The examples provided here cover different rotor configurations and are summarized in Table 1. Some models are exact, while others are representative of the actual aircraft. Depending on the availability of test data, these models will serve as test cases for validation and verification.

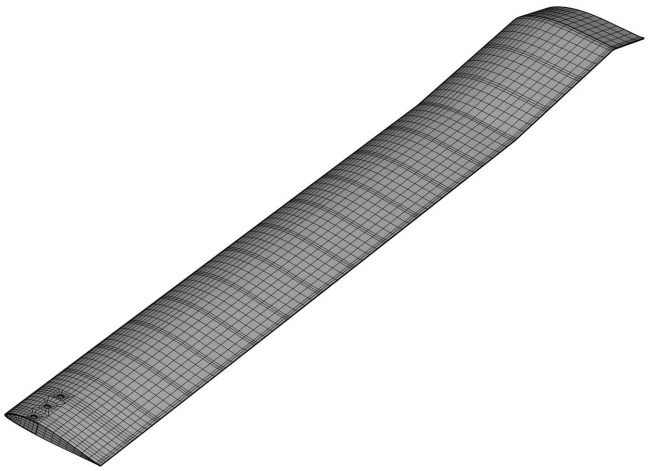


Fig. 5. Three-dimensional brick mesh of the UMD-Boeing dihedral anhedral rotor blade.

Single rotor

There are three single rotor models: the UH-60A-like rotor, the UMD-Boeing Dihedral-Anhedral Rotor Blade (DARB) and the NASA Rotor Optimization for the Advancement of Mars eXploration (ROAMX) blade.

UH-60A-like. The UH-60A model is the same as that used in version 1.0 (Ref. 7). The internal structure is an idealization, reverse-engineered to reproduce a similar fan plot as the real rotor. Although this model is not an exact representation, it serves as an excellent test case for validation of blade airloads and vibratory harmonics obtained from the UH-60A Airloads Program (Refs. 30, 31). This model was studied in a low-speed transition flight: counter 8513 of the UH-60A Black Hawk Airloads Program. Reference 12 shows the blade airloads using the parallel in time solver compared with the time-marching solution (version 1) and measured data. The performance improvements with the new version are highlighted in a later section.

UMD-Boeing DARB. The UMD-Boeing DARB is inspired by the advanced Chinook rotor blade (Ref. 17). The DARB was designed in consultation with Boeing to ensure the geometry is realistic enough to be representative of a modern rotor but still remains open source. The model is Mach-scaled with a radius of 0.853 m (2.8 ft), and the nominal rotational speed is 2282 revolutions per minute (RPM). The cross section is made of a VR-7 airfoil with D-spar, skin, leading-edge weight, and foam. There is a linear built-in twist of -16° per span. The blade has three sections and two transition regions. The first section extends from the root cutout (16.4%R) to 80%R, connecting to a 5° dihedral portion over 80–95%R. The last portion is an anhedral of 15° over the last 5%R. Figure 5 shows the 3D model. There are 4556 27-noded hexahedral bricks for a total of 122,000 degrees of freedom. All three bolt holes at the root are connected to a single joint, which is used to command the pitch commands.

The DARB was fabricated and tested at the University of Maryland. The material properties were obtained by coupon testing the composite materials, minimizing the disparity between X3D and the fabricated blades. The blades were tested in a vacuum, and the rotating frequencies and strains were measured. Figure 6 shows the rotor fan plot. The

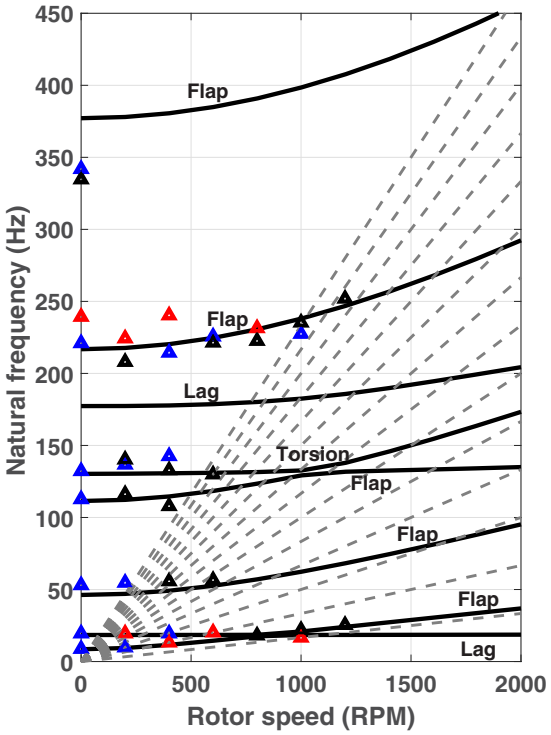


Fig. 6. Measured versus predicted rotor fan plot for the UMD DARB.

triangles are measured frequencies; the different colors correspond to different blades. The solid lines are the predictions from X3D v2.0. Up to seven nonrotating frequencies and six rotating frequencies were measured. The corresponding modes were analyzed in X3D and are identified in Fig. 6. The predictions match well with the test data except for the highest mode (fifth flap mode).

NASA ROAMX. The ROAMX project (Ref. 32) was recently launched by NASA to fabricate and test a set of future rotor blades. The Martian atmosphere necessitates unique airfoils, with sharp leading- and trailing-edges and extremely low thickness-to-chord ratios on the order of 1%, such as those designed by Koning in Ref. (Ref. 33). The ROAMX project is intended to experimentally demonstrate the performance benefits of these unique airfoils. The University of Maryland (UMD) partnered with NASA to use X3D to design the internal structure for the blades (Ref. 14), as the use of 3D FEA was beneficial for modeling the unconventional blades. The new parallel X3D was used for the analysis, which resulted in a significant reduction in time. Over 40 different individual designs were studied with almost 20,000 X3D runs, all made possible by parallel X3D.

The final blade design is shown in Fig. 7. The internal structure varies with radius; the foam core and spar end at 35%R and 40%R, respectively. Outboard of midspan, the airfoil thickness falls to 1% chord, and the cross section is all skin. Additional details can be found in Ref. 14. Figure 8 shows the 3D model. The blade has 2,202 27-noded hexahedral brick elements and 20,147 nodes.

The blade was designed for hover testing, so it was primarily studied in hover with a focus on airloads, deformations, and stresses. Figure 9 shows blade axial stress on the bottom surface during hover at a collective of 20° and a blade loading of $C_T/\sigma = 0.32$. The stress patterns near the root are highly localized due to the high curvature of the root transition. One area of interest was the significant amount of predicted elastic twist deformation, with some cases having up to 10 deg. Further examination

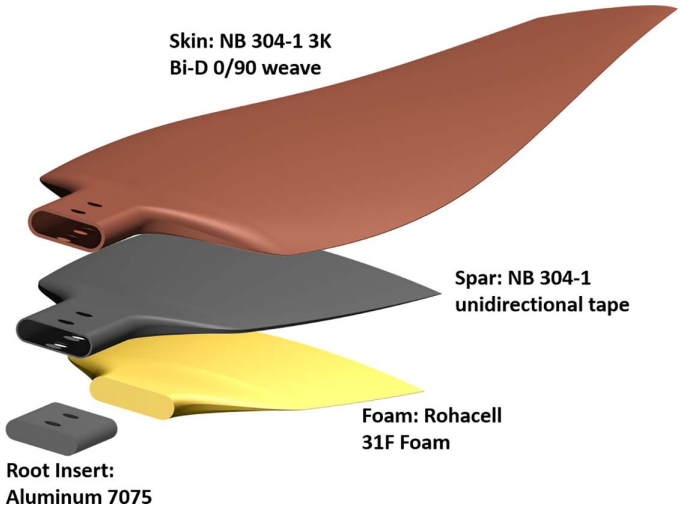


Fig. 7. Materials and internal structure of the NASA ROAMX blade design.

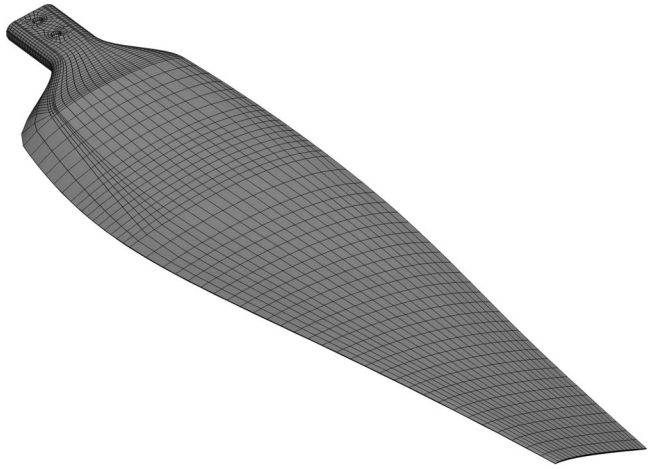


Fig. 8. Three-dimensional brick mesh of the NASA ROAMX blade.

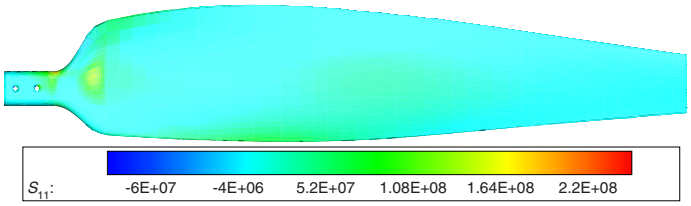


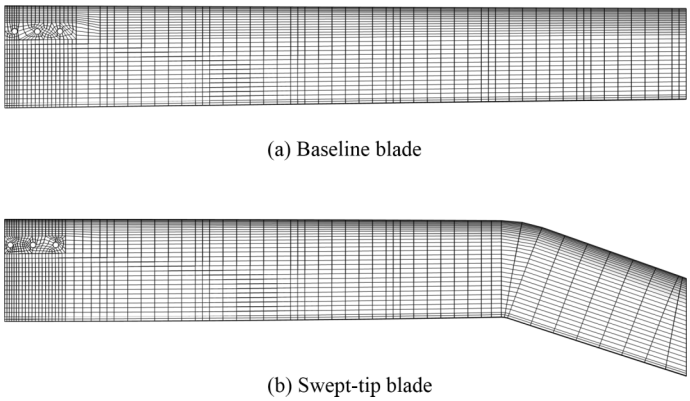
Fig. 9. Axial stress (σ_{11}) on the bottom surface of the NASA ROAMX blade in hover with $C_T/\sigma = 0.32$.

revealed this was a combination of propeller moment and trapeze effect that was highly dependent on the blade collective.

Tilt rotor

The tiltrotor category consists of two models: one, the NASA TRAM, and, two, the Maryland tiltrotor rig (MTR) model.

NASA TRAM. The TRAM is a Mach-scaled test article developed by NASA. It is a 1/4-scale model of the V-22 tiltrotor dynamically scaled to



(a) Baseline blade

(b) Swept-tip blade

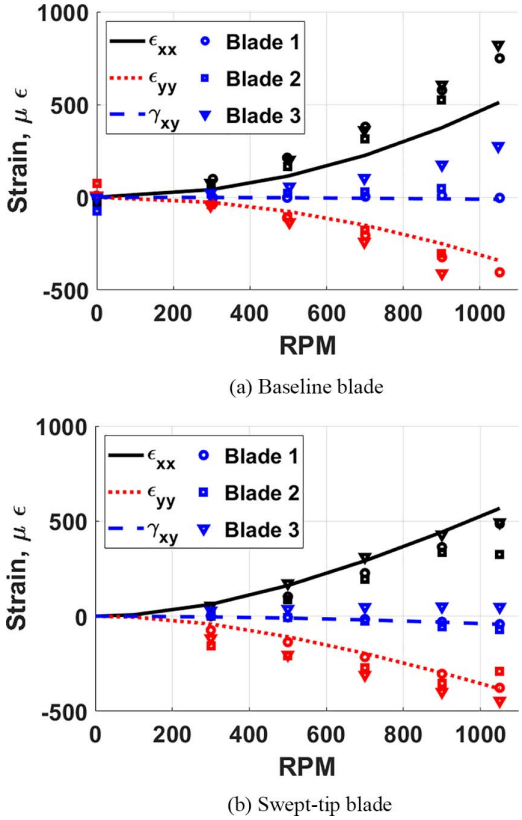
Fig. 10. 3D structural meshes for two MTR blades.

match rotor frequencies in first flap, lag, and torsion. It is the same model used in version 1.0 (Ref. 7). Reference 9 documents the 3D CFD/CA (CFD/comprehensive analysis) coupled analysis of TRAM using version 1.0. The same model was later studied using the parallel in space and time solvers with better performance (Refs. 11, 34).

MTR. The MTR is a new tiltrotor test facility developed at the University of Maryland (Refs. 35,36). The MTR was developed to provide a test bed for research examining the aeromechanics of high-speed tiltrotors, with the main objective to understand and ultimately eliminate whirl-flutter.

There are two blades that have been tested on the MTR. Both have similar properties with a rotor radius of 0.724 m (2.375 ft), chord of 8 cm (3.15 inches), and linear twist of -37° over the span. The blades have a uniform VR-7 cross section made up of composite skin and D-spar, foam, and tungsten Leading Edge weights. The main difference is in the tip region—the baseline blade is straight, while the swept-tip blade has a 20° tip sweep starting at 80% blade radius. The 3D structural models are shown in Fig. 10. Both meshes have similar resolution: the baseline blade has 2938 brick elements and 26,442 nodes, while the swept-tip blade has 2812 brick elements and 25,290 nodes.

Both blades were fabricated and tested at the University of Maryland vacuum chamber, where rotating frequencies and strains were measured (Ref. 16). Figure 11 shows the rotating strains on the top surfaces of the blades at 40%R. Different markers are used to indicate data from three different blades to confirm repeatability. Axial ϵ_{XX}, chordwise ϵ_{YY}, and in-plane shear γ_{XY} strains were measured. The lines indicate the X3D predictions. There is a strong correlation between the predictions and measurements for both blades.



(a) Baseline blade

(b) Swept-tip blade

Fig. 11. Comparison of rotating strain between vacuum chamber tests and X3D.

Coaxial rotor

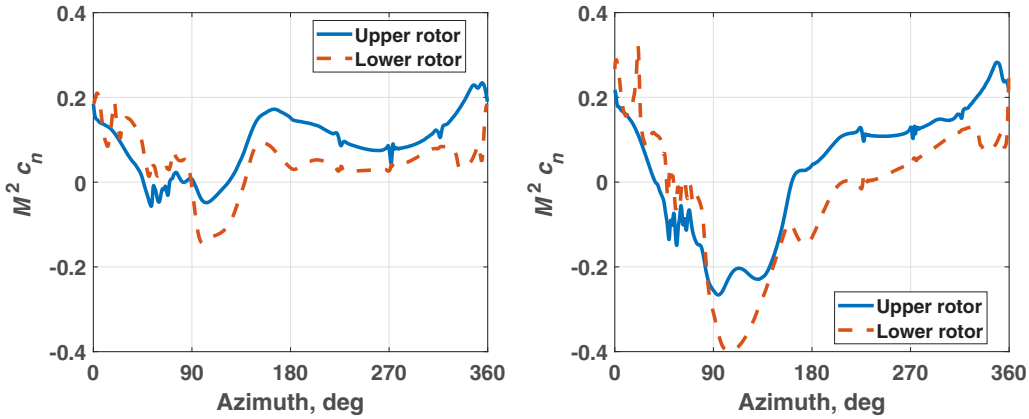
The coaxial rotor category consists of two models: Metaltail-II and the Mars Ingenuity helicopter.

Metaltail-II. Metaltail-II is an open-source four-bladed hingeless model inspired by the gross dimensions of the Sikorsky S-97 Raider. The model shares superficial similarities with the S-97 Raider, such as the hingeless hub, four blades per rotor, same rotor radius, similar planform, and similar inter-rotor spacing. The hub is notional and consists of the following parts: pitch link, pitch horn, inner cuff, outer cuff, thrust bearing, and journal bearing. The cross section is a VR-7 airfoil with a composite D-spar and skin to be representative of a modern rotor. More details can be found in Ref. 19.

The structural model was coupled with lifting-line aerodynamics (Ref. 18) and 3D Reynolds-averaged Navier–Stokes (RANS) CFD (Ref. 19). The 3D CFD/CSD coupling was performed with the Helios software using the I3D coupling methodology described earlier. This coupling was carried out at three different flight speeds: 37 kt ($\mu = 0.1$), 150 kt ($\mu = 0.35$), and 220 kt ($\mu = 0.5$). A forward shaft tilt of $\alpha = -2^\circ$ and a blade loading of $C_T/\sigma = 0.08$ was assumed. Reference 19 discusses each of these cases in detail. The low-speed transition flight was qualitatively compared to results published recently by Sikorsky (Refs. 37, 38) to verify similarity. The highest speed case corresponded to the maximum cruise speed of the S-97 Raider and was studied to identify the maximum stresses on a lift offset coaxial rotor.

At 220 kt flight speed, the rotor could only be trimmed with a nonzero lift offset; hence, a 10%R lift offset was used. The predictions of normal force ($\approx$ lift) distribution are shown in Fig. 12. There is a substantial negative lift on both rotors to produce an individual roll moment to achieve the targeted lift offset. Figures 13(a) and 13(b) show the CFD flow field. The wake is milder at this speed but with more pronounced dual vortices on the advancing side, consistent with the greater negative lift. Two vortices are released from each blade, one from the tip, and the other near where the blade lift passes to zero. The second vortex again hits the following blade, resulting in the first quadrant impulse in airloads.

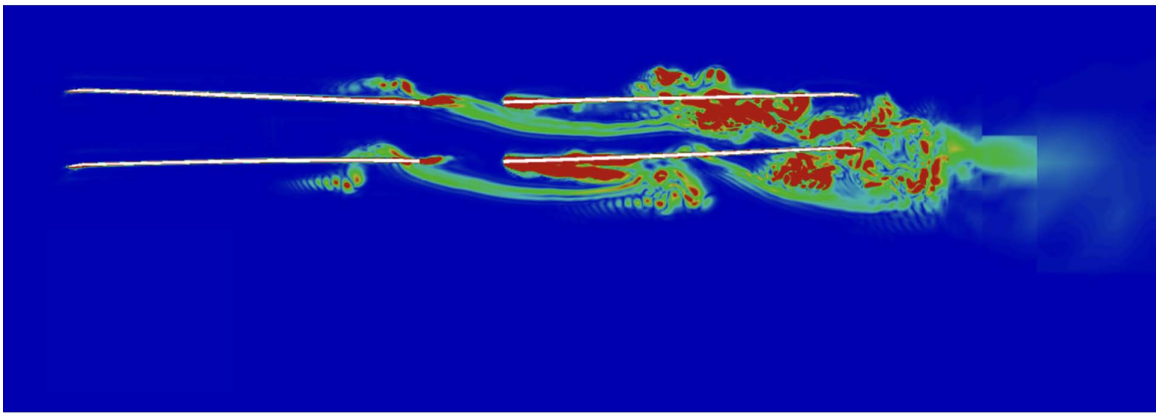
All the interactions and complications observed in airloads are passed on to the blade and hub stresses. Figure 14 shows the axial (bending) stresses (σ_{11}). The blade skin is peeled off to visualize the internal stresses. The maximum bending stress is carried by the spar as expected. The stresses vary in span, chord, and thickness. Reference 19 from the authors studies the I3D stresses at all three speeds (37, 150, and 220 kt). The most important observation is that the maximum stresses at 220 kt are two to three times higher than at the lower speeds, which indicates the severity of this regime.



(a) 82.5% $\boldsymbol{R}$ (b) 93.5% $\boldsymbol{R}$

Fig. 12. Normal force distribution at $\mu = 0.5, C_T/\sigma = 0.08$ for two radial stations.

(a) Isometric view



(b) Side view - with aircraft flying to the left

Fig. 13. Converged 3D CFD flow field in isometric and side view at very high-speed flight, $\mu = 0.5, C_T/\sigma = 0.08$; isosurface of Q-criterion, 0.001, colored by vorticity.

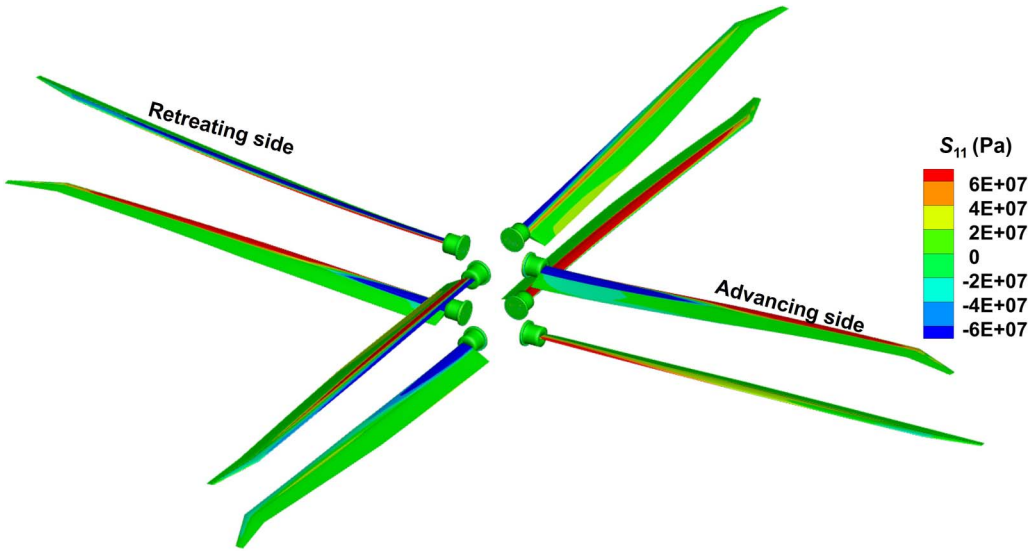
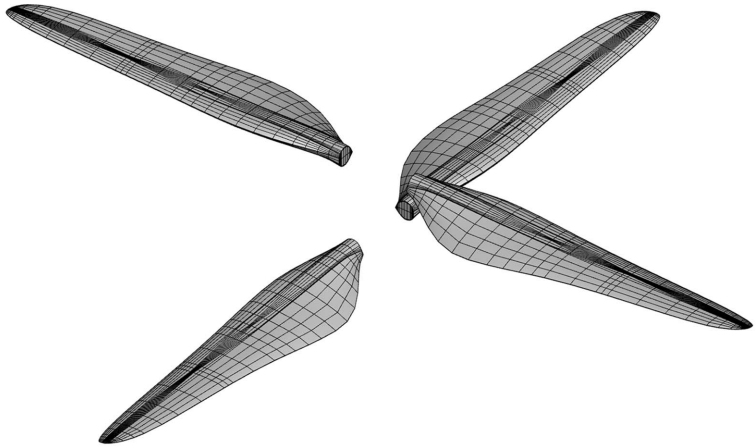
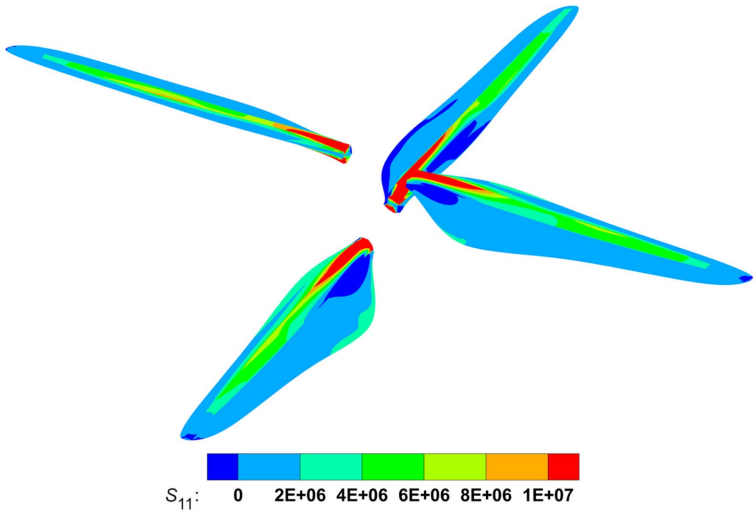


Fig. 14. Axial/bending stress distribution at very high-speed flight, $\mu = 0.5$, $C_T/\sigma = 0.08$.



(a) The 3D structural mesh of the Ingenuity rotor



(b) Predicted 3D axial stress field in forward flight ($\mu = 0.09$, $C_T/\sigma = 0.12$, 2630 RPM)

Fig. 15. Structural mesh and predicted axial stress distribution for the Ingenuity rotor.

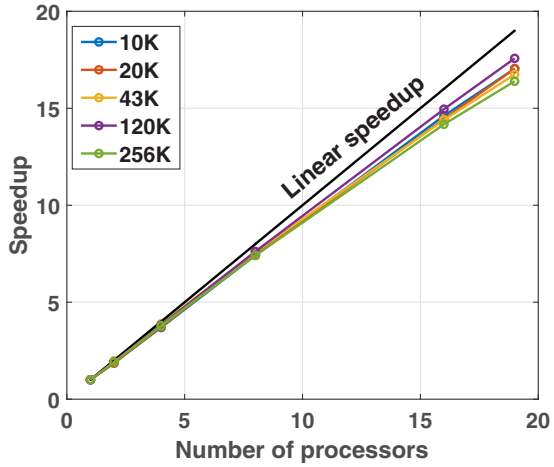


Fig. 16. Speedup of the parallel skyline solver on a single node of 20 processors.

Mars Ingenuity. The Mars Ingenuity is a coaxial helicopter that made history as the first aircraft flown on Mars. The design and development of the Ingenuity involved systematic 3D analysis using X3D, but was unpublished until 2024. Reference 39 describes the detailed development and analysis of the model. The Ingenuity rotor geometry was provided by NASA JPL. The model was meshed in Cubit, and the full coaxial structural model is shown in Fig. 15a. Each blade mesh consisted of 24,525 finite elements, comprising 630,312 degrees of freedom.

The coaxial Ingenuity was studied using the new parallel X3D to obtain rotating frequencies, stresses in hover and forward flight. The rotor was tested in forward flight with an advance ratio of $\mu = 0.09$, shaft tilt of $\alpha_s = 0^\circ$, and trimmed to total lift ($C_T/\sigma = 0.12$), zero net torque, and zero pitch and roll moments on individual rotors. Figure 15(b) shows the axial stress distribution on Ingenuity blades in forward flight. Reference 39 documents all the test cases studied for Ingenuity using X3D in detail.

Parallel Performance

This section discusses the parallel performance of the new algorithms implemented in X3D version 2.0 and compares them with the version 1.0. Note that both the serial and parallel versions were run on Deepthought2, the University of Maryland's High-Performance Computing cluster. It consists of 480 nodes with dual sockets on each node. Each socket has 10 Intel Ivy Bridge E5-2680v2 processors running at 2.8 GHz, for a total of 20 processors available on each node. Each of these processors has a separate L1 cache (64 KB), a separate L2 cache (256 KB), a shared L3 cache (25 MB), and a total shared memory of 128 GB. Some of the models introduced in the previous section are used as test cases to obtain parallel performance.

Parallel skyline

The performance study of the parallel skyline solver uses an elementary test case: a 3D cantilevered beam with a static tip force. This elementary problem makes mesh refinement simple, enabling easy adjustment to the number of degrees of freedom. Figure 16 shows the speedup on a node for various mesh sizes ranging from 10 K to 250 K degrees of freedom. Regardless of size, a speedup of up to 17 is obtained over 19 processors (90% scalability). Of 20 processors on each node, one is always dedicated to system management. Thus, only 19 were used for computation.

Table 2. Solution times for serial and parallel time-marching and modified harmonic balance for UH-60A-like rotor test case; parallel execution on a combined shared and distributed memory architecture

Solution Procedure	Mode	Fixed Controls	Trim Solution	Speedup
Time-marching	Serial	12 min	100 min	1
	Parallel	2 min	16 min	6
Modified harmonic balance	Serial	2.5 min	10 min	10
	Parallel	20 s	2 min	50

Parallel in time X3D

The solution times and speedup for fixed controls and level flight trim solution cases for the UH-60A single rotor model are summarized in Table 2. The time-marching solution was obtained with $\Delta\psi = 5^\circ$. The time-marching solution on a single processor for a trim solution takes 100 min. The parallel skyline, but still with time-marching, reduces it to 16 min, a speedup of around 6. Replacing time-marching with the MHB on a single processor with no parallelization produces a speedup of 10. When implemented in parallel (nine MPI tasks, 0–8 harmonics) with each using 10 shared-memory processors internally, the time drops to 2 min, achieving a total speedup of 50. This massive speedup is a result of the combined effect of the parallel skyline and the inherent parallel nature of the MHB algorithm.

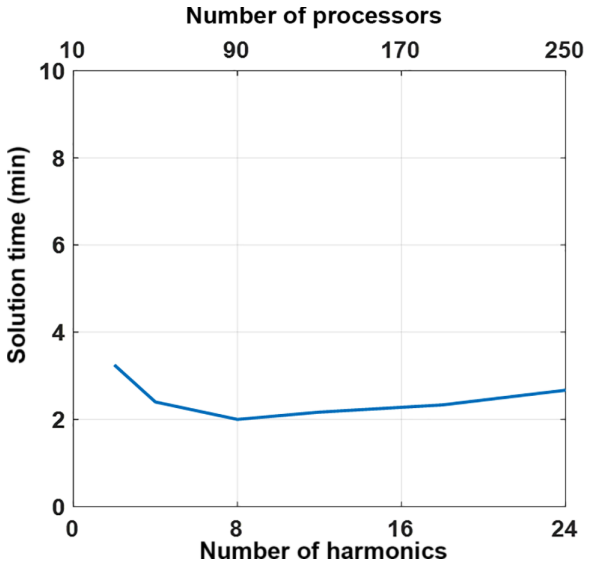
A total of $N + 1$ MPI tasks are launched for obtaining the solution with N harmonics $(0, 1, 2, \ldots, N)$. With each MPI task assigned with 10 shared memory processors, a total of $10 \times (N + 1)$ processors (both MPI and OpenMP) are employed. The solution time for trim solution remains almost constant with an increase in the number of harmonics. Thus, a finer harmonic resolution is possible with no increase in solution time if more processors are available. Figure 17(a) shows the trim solution time versus the number of harmonics and processors. For example, a total of 90 processors are used to solve for $0, 1, 2, \ldots, 8$ harmonics. Figures 16 and 17(a) prove, respectively, the strong and weak scalability of the algorithm, respectively.

Figure 17(b) shows the variation of floating point operations per second (FLOPS) with the number of degrees of freedom for time-marching and MHB. A maximum of only 10 gigaFLOPS can be achieved using time-marching. On the other hand, the MHB reaches higher than 1 teraFLOPS. The key conclusion is that the MHB will produce a gain of at least two orders of magnitude in FLOPS compared to time-marching.

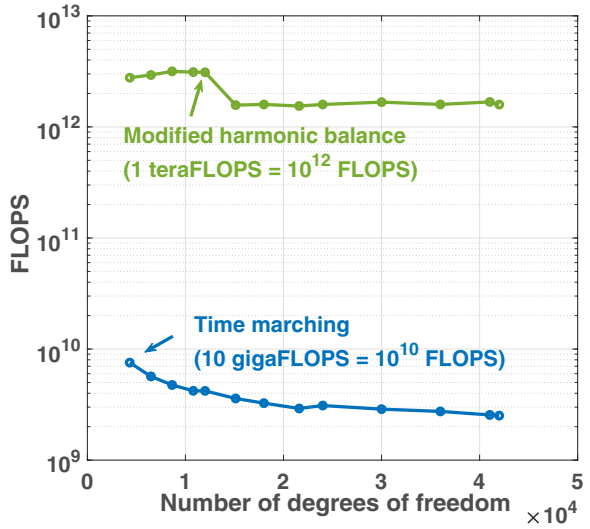
Parallel in space X3D

The performance of the Parallel in Space X3D will be demonstrated for two cases: (1) a 3D cantilevered beam and (2) TRAM. This will showcase the performance and scalability of the solver for two different problems, highlighting the potential of the solver while presenting the complications of realistic problems.

Idealized test case. The idealized test case is a cantilevered beam with 6.6 million Degree of Freedom (DOF). It has a structured mesh, uniform properties, a single joint for boundary conditions, and no control inputs. Load balance is trivial to achieve, and corner node selection is straightforward—none of the special partitioner features are needed. This will determine the performance and limitations of the algorithm for an elementary problem. The solver performance is studied for a single matrix solve (a linear static solution). The full solution with aerodynamics involves repeated iterations of the same solve, so one suffices for algorithmic purposes.



(a) Trim solution time versus number of harmonics and processors; a total of $10 \times (N+1)$ processors are used to solve for $0, 1, 2, \ldots, N$ harmonics. Solution time remains almost constant with increase in number of harmonics and processors



(b) Variation of FLOPS versus the number of degrees of freedom for time-marching and modified harmonic balance

Fig. 17. Scalability and computational performance characteristics of the trim solution and harmonic balance formulations.

Figure 18 shows the solution time on a single processor versus the number of subdomains. For this work, a single processor solution still uses a partitioned structure; however, the subdomains are solved sequentially on the single processor. The computational time decreases significantly with additional partitions, even without parallel implementation. This continues until 1024 subdomains, also known as the optimal number of subdomains, or the number of subdomains which yields the fastest solution, after which the computational time increases. Figure 19(a) shows the computational time when each subdomain is solved on a separate processor. A similar trend is seen, with the computational time falling until 1024 processors before increasing.

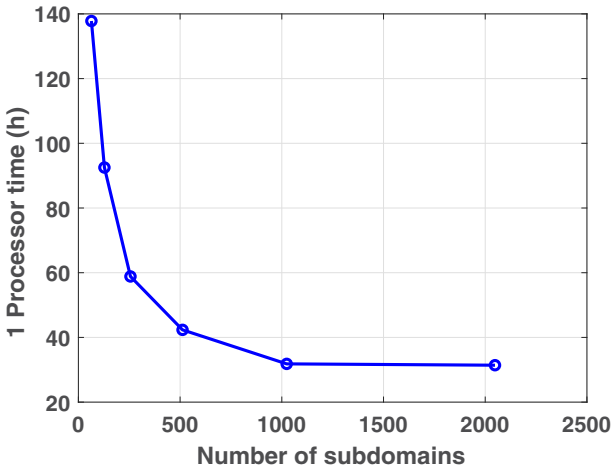


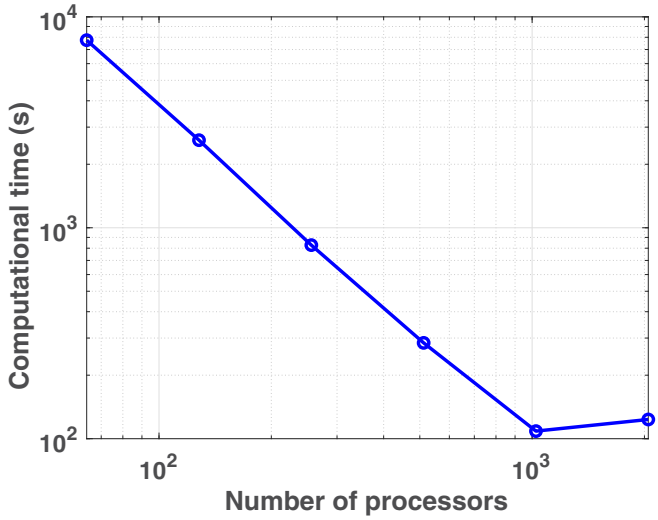
Fig. 18. Computational time on a single processor for the cantilevered beam.

Figure 19(b) shows the parallel speedup, defined as single processor time divided by parallel time for the same number of subdomains. This ensures the speedup is the true parallel speedup, un-contaminated by the inherent benefits of partitioning. Linear scalability is achieved up to 1024 subdomains, before falling off by 2048 subdomains. This high scalability is mainly due to the use of a skyline solver for the coarse problem. For 1024 subdomains, the coarse problem has 60 K DOF, but only 0.5% of the matrix needs to be stored, drastically reducing the time of the serial step and the communication overhead.

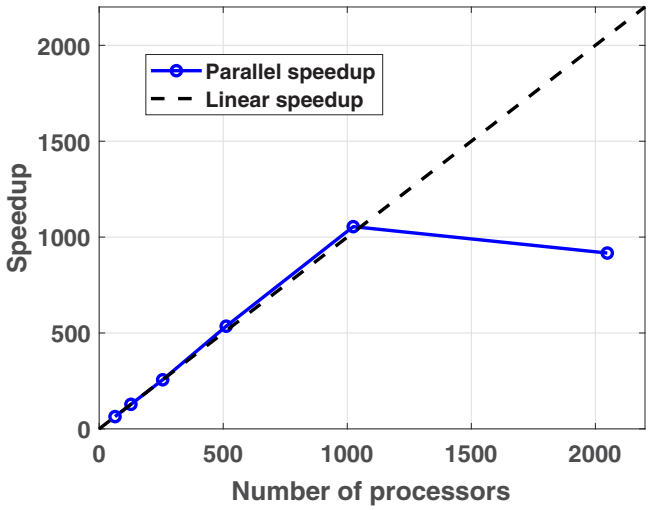
TRAM. Studying the performance of the solver on TRAM introduces all the complexities of an advanced rotor blade and hub—multiple unstructured meshes, each with multiple materials, connected via multibody joints. All of the special partitioner features are needed for this case. The solver performance will be demonstrated both for a single matrix solve as well as for different solution procedures.

Figure 19(c) shows the parallel computational time for a single TRAM solve. Similar to the idealized beam, the computational time drops with partitioning until it reaches the optimal number of subdomains, in this case 48, before increasing again. The optimal number of subdomains is smaller than for the idealized beam for two reasons. First, the problem size is smaller by a factor of 25 (250 K vs. 6.6 million DOF). Second, TRAM is more complex, and the presence of joints can limit the performance gains made due to partitioning. However, the solver time for a single solve drops from almost an hour (serial with no partitioning) to nearly 30 s, providing significant performance benefits.

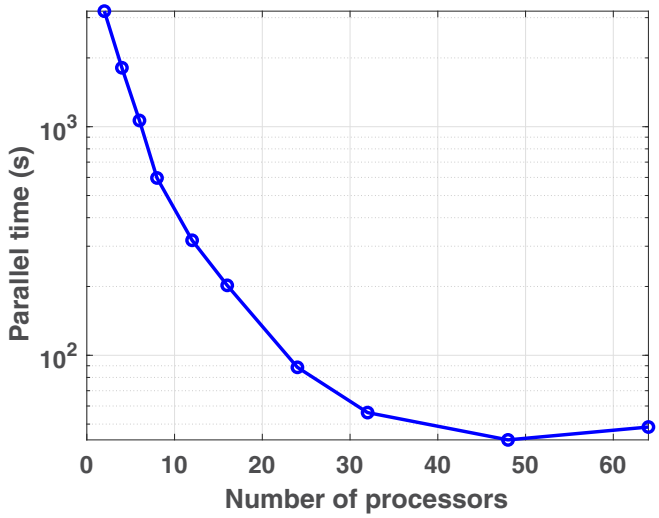
The parallel speedup for TRAM is shown in Fig. 19(d). The speedup is shown for the single matrix solve (static case), as well as three other solution procedures: nonrotating frequencies, rotating frequencies, and hover. Rotating frequencies differ from nonrotating frequencies due to iterations needed to converge the centrifugal force. It is seen that all cases have linear speedup up to 48 subdomains, with varying degrees of superlinear speedup (speedup higher than the number of processors) at lower numbers of subdomains due to cache savings. The performance falls off after 48 subdomains due to a large multibody joint. Greater than 48 subdomains, this joint becomes larger than all other subdomains, causing poor load balance that gets worse with further partitioning. This shows that joints currently impose on parallel performance. Future work will investigate using the OpenMP skyline solver to attack this problem by allocating threads based on the subdomain size, reducing the effect of joints. However, even with this large multibody joint, the key



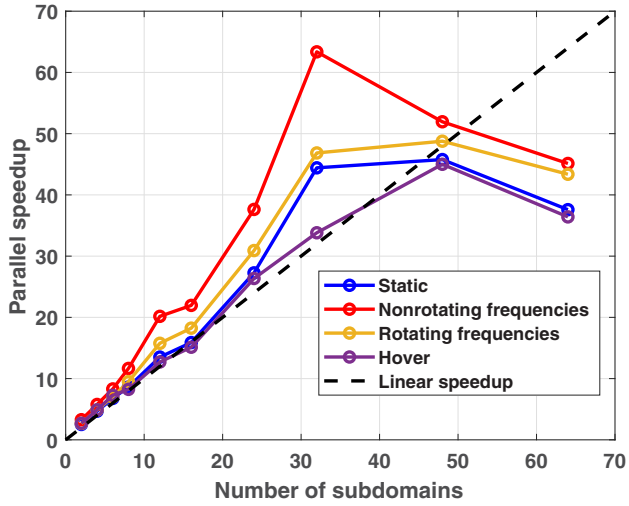
(a) Parallel computational time for the cantilevered beam



(b) Parallel speedup for the cantilevered beam



(c) Parallel computational time for TRAM



(d) Parallel speedup for TRAM for different solution procedures

Fig. 19. Parallel performance results for the cantilevered beam and TRAM simulations.

conclusion is that good scalability is achieved with the parallel in space solver independent of the solution procedure.

Summary

Version 2.0 of X3D—an experimental (X) three-dimensional dynamic (3D) solver for rotor aeromechanics was introduced. It is a marked departure from current generation rotor aeromechanics software as well as commercially available structural dynamics software. The paper described the enhancements in version 2.0 over its predecessor, which involved new parallel solution procedures, new capabilities allowing for the study of new 3D models, including coaxial rotors, and new tools streamlining the workflow to generate 3D rotor models from CAD. This paper also discussed the application of integrated 3D CFD/CSD analysis, defined as a coupling of 3D finite element–based structural dynamics with 3D RANS fluid dynamics, for a modern lift offset coaxial rotor. Finally, the performance characteristics of this new parallel version were presented to demonstrate the significant improvements over version 1.0.

Future Work

Future development of X3D will focus on integrating the parallel-in-space and parallel-in-time solvers into a unified framework to improve scalability and reduce computational cost. Scalability limitations from large multibody joints will be addressed through improved load balancing and enhanced use of the OpenMP skyline solver.

The X3D database will be expanded to include additional rotor, propeller, and tiltrotor configurations, with emphasis on advanced rotorcraft and eVTOL applications. Validation efforts will continue through comparisons with experimental data and high-fidelity simulations to quantify accuracy and uncertainty, supporting broader adoption of X3D in the rotorcraft community.

Acknowledgments

The development of this software was carried out at the Alfred Gessow Rotorcraft Center, University of Maryland at College Park, under the Army/Navy/NASA Vertical Lift Research Center of Excellence

(VLRCOE) grant (number W911W61120012) in collaboration with the U.S. Army Technology Development Directorate. The technical monitor of the VLRCOE program is Dr. Mahendra Bhagwat. His support was instrumental to the progress of this endeavor. We wish to thank the technical points of contact: Dr. Wayne Johnson (NASA), Dr. Andrew Wissink (Army), and Dr. Buvana Jayaraman (Army). Dr. Jayaraman of the Helios team provided CFD support during Helios integration and testing. We also thank Dr. William Staruk (Joby) and Dr. Elizabeth Ward (Bell) for their prior work on CAD-based 3D modeling that has been incorporated into the X3D workflow.

References

[1]Patil, M., Lumba, R., and Datta, A., "X3D v2.0—A Parallel 3D Finite Element Multibody Dynamic Analysis for Rotorcraft," Proceedings of the Vertical Flight Society's 6th Decennial Aeromechanics Specialists' Conference, Santa Clara, CA, Feb. 6–8, 2024.

[2]Wayne Johnson, "A History of Rotorcraft Comprehensive Analyses," NASA/TP–2012–216012, April 2012.

[3]Johnson, W., and Datta, A., "Requirements for Next Generation Comprehensive Analysis of Rotorcraft," Proceedings of the American Helicopter Society Specialists' Conference on Aeromechanics, San Francisco, CA, January 23–25, 2008. https://ntrs.nasa.gov/citations/20080047714.

[4]Datta, A., and Johnson, W., "Three-Dimensional Finite Element Formulation and Scalable Domain Decomposition for High Fidelity Rotor Dynamic Analysis," *Journal of the American Helicopter Society*, **56**, 22003 (2011).

[5]Datta, A., and Johnson, W., "A Multibody Formulation for Three-Dimensional Brick Finite Element Based Parallel and Scalable Rotor Dynamic Analysis," Proceedings of the 66th Annual Forum of the American Helicopter Society, Phoenix, AZ, May 11–13, 2010.

[6]Datta, A., and Johnson, W., "Integrated Aeromechanics with Three-Dimensional Solid-Multibody Structures," Proceedings of the 70th Annual Forum of the Vertical Flight Society, Montreal, Quebec, Canada, May 20–22, 2014.

[7]Datta, A., "X3D–A 3D Solid Finite Element Multibody Dynamic Analysis for Rotorcraft," Proceedings of the American Helicopter Society Technical Meeting on Aeromechanics Design for Vertical Lift, San Francisco, CA, January 20–22, 2016.

[8]Staruk, W., Chopra, I., and Datta, A., "Three-Dimensional CAD-Based Structural Modeling for Next Generation Rotor Dynamic Analysis," Proceedings of the 70th Annual Forum of the American Helicopter Society, Québec, Canada, May 20–22, 2014.

[9]Staruk, W., Datta, A., Chopra, I., and Jayaraman, B., "An Integrated Three-Dimensional Aeromechanics Analysis of the NASA Tilt Rotor Aeroacoustic Model," *Journal of the American Helicopter Society*, Vol. 63, (3), July 2018, pp. 1–12.

[10]Staruk, W., and Datta, A., "Gimbaled Tiltrotor Conversion Flight Loads Prediction Using Three-Dimensional Structural Analysis," *Journal of Aircraft*, Vol. 56, (2), March–April 2019, pp. 758–770, DOI: 10.2514/1.C035075.

[11]Lumba, R., and Datta, A., "Scalable Mesh Partitioning for Multibody–3D Finite Element Based Rotary-Wing Structures," *Computers & Structures*, **281**, 107015 (2023), DOI: 10.1016/j.compstruc.2023.107015.

[12]Patil, M., and Datta, A., "A Scalable Time-Parallel Solution of Periodic Rotor Dynamics in X3D," *Journal of the American Helicopter Society*, **66**, 042007 (2021), DOI: 10.4050/JAHS.66.042007.

[13]Chi, C., Lumba, R., Jung, Y. S., and Datta, A., "Aeromechanical Analysis of a Next-Generation Mars Hexacopter Rotor," *Journal of Aircraft*, Vol. 59, (6), 2022, pp. 1463–1477, DOI: 10.2514/1.C036739.

[14]Lumba, R., Chi, C., Datta, A., Koning, W., Perez, N. P., and Cummings, H., "Structural Design and Aeromechanical Analysis of Unconventional Blades for Future Mars Rotorcraft," *Journal of the American Helicopter Society*, **68**, 42003 (2023), DOI: 10.4050/JAHS.68.042003.

[15]Chi, C., Datta, A., Chopra, I., and Chen, R., "Three-Dimensional Strains on Twisted and Swept Composite Rotor Blades in Vacuum," *Journal of Aircraft*, Vol. 58, (1), January 2021, pp. 1–16, DOI: 10.2514/1.C035746.

[16]Sutherland, J., and Datta, A., "Fabrication, Testing, and 3D Comprehensive Analysis of Swept-Tip Tiltrotor Blades," *Journal of the American Helicopter Society*, **68**, 012002 (2023), DOI: 10.4050/JAHS.68.012002.

[17]Chi, C., Datta, A., and Panda, B., "Vacuum and Hover Tests of a Dihedral–Anhedral Tip Composite Rotor," *Journal of Aircraft*, Vol. 60, (5), 2023, pp. 1700-1711, DOI: 10.2514/1.C037321.

[18]Patil, M., Datta, A., and Jayaraman, B., "Three-Dimensional Blade and Hub Stresses of Coaxial Rotors in High-Speed Forward Flight," AIAA 2023-1892, Proceedings of the AIAA SciTech 2023 Forum, National Harbor, MD & Online. January 23–27, 2023.

[19]Patil, M., Lumba, R., Jayaraman, B., and Datta, A., "An Integrated Three-Dimensional Aeromechanical Analysis for the Prediction of Stresses on Lift Offset Coaxial Rotors," Proceedings of the 49th European Rotorcraft Forum, Bückeburg, Germany, September 5–7, 2023.

[20]Strawn, R., "High-Performance Computing for Rotorcraft Modeling and Simulation," *Computing in Science and Engineering*, Vol. 12, (5), September–October 2010, pp. 27–35.

[21]Sitaraman, J., Postdam, M., Wissink, A., Jayaraman, B., Datta, A., Mavriplis, D., and Saberi, H., "Rotor Loads Prediction Using Helios: A Multisolver Framework for Rotorcraft Aeromechanics Analysis," *Journal of Aircraft*, Vol. 50, (2), March 2013, pp. 478–492.

[22]Blacker, T., Owen, S., Staten, M., Quadros, R., Hanks, B., Meyers, R., Ernst, C., Merkley, K., Morris, R., McBride, C., Stimpson, C., Plooster, M., and Showman, S., "CUBIT Geometry and Mesh Generation Toolkit 15.2 User Documentation," Sandia National Laboratories, SAND2016–1649 R, May 2016.

[23]Staruk, W., "CAD-Based Modeling of Advanced Rotary Wing Structures for Integrated 3-D Aeromechanics Analysis," Ph.D. Dissertation, Department of Aerospace Engineering, University of Maryland, College Park, MD, 2017.

[24]Saberi, H., Hasbun, M., Hong, J. Y., Yeo, H., and Ormiston, R. A., "Overview of RCAS Capabilities, Validations, and Rotorcraft Applications," Proceedings of the 71st Annual Forum of the American Helicopter Society, Virginia Beach, VA, May 5–7, 2015.

[25]Johnson, W., "Technology Drivers in the Development of CAMRAD II," Proceedings of the AHS Aeromechanics Specialists Conference, San Francisco, CA, January 19–21, 1994.

[26]Farhat, C., Lesoinne, M., and Pierson, K., "A Scalable Dual–Primal Domain Decomposition Method," *Numerical Linear Algebra with Applications*, Vol. 7, (7–8), October 2000, pp. 687–714, DOI: 10.1002/1099-1506(200010/12)7:7/8<687::AID-NLA219>3.0.CO;2-S.

[27]Farhat, C., Lesoinne, M., LeTallec, P., Pierson, K., and Rixen, D., "FETI-DP: A Dual-Primal Unified FETI Method — Part I: A Faster Alternative to the Two-Level FETI Method," *International Journal of Numerical Methods in Engineering*, Vol. 50, (7), January 2001, pp. 1523–1544, DOI: 10.1002/nme.76.

[28]Datta, A., and Johnson, W., "Large-Scale Domain Decomposition for a Scalable, Three-Dimensional Brick Finite Element Based Rotor Dynamic Analysis," Proceedings of the American Helicopter Society Specialists' Conference on Aeromechanics, San Francisco, CA, January 20–22, 2010, available at https://apps.dtic.mil/sti/citations/ADA532063.

[29]Shastry, A., and Datta, A., "Predicting Wake and Structural Loads in RPM Controlled Multirotor Aircraft," Proceedings of the Vertical Flight

Society Transformative Vertical Flight Technical Meeting, San Jose, CA, January 21–23, 2020.

[30]Bousman, W. G., and Kufeld, R. M., "UH-60A Airloads Catalog," Technical Report TM-2005-212827, NASA, August 2005.

[31]Datta, A., Nixon, M., and Chopra, I., "Review of Rotor Loads Prediction with the Emergence of Rotorcraft CFD," *Journal of the American Helicopter Society*, Vol. 52, (4), October 2007, pp. 287–317, DOI: 10.4050/JAHS.52.287.

[32]Cummings, H., Perez Perez, N., Koning, W., Johnson, W., Young, L., Haddad, F., Romander, E., Balaram, J., Tzanetos, T., Bowman, J., Wagner, L., Withrow-Maser, S., Isaacs, E., Toney, S., Shirazi, D., Conley, S., Pipenberg, B., Datta, A., Lumba, R., Chi, C., Smith, K., Cornelison, C., Perez, A., Nonomura, T., and Asai, K., "Overview and Introduction of the Rotor Optimization for the Advancement of Mars eXploration (ROAMX) Project," Proceedings of the Vertical Flight Society Aeromechanics for Advanced Vertical Flight Technical Meeting, San Jose, CA, January 25–27, 2022.

[33]Koning, W. J. F., Romander, E. A., and Johnson, W., "Optimization of Low Reynolds Number Airfoils for Martian Rotor Applications Using an Evolutionary Algorithm," Proceedings of the AIAA SciTech 2020 Forum, Orlando, FL, January 6–10, 2020.

[34]Patil, M., and Datta, A., "Three-Dimensional Aeromechanical Analysis of Lift Offset Coaxial Rotors," AIAA 2022-0928, AIAA SciTech 2022 Forum, San Diego, CA & Virtual, January 3–7, 2022, DOI: 10.2514/6.2022-0928.

[35]Datta, A., Tsai, F., and Sutherland-Foggio, J., "Design of a New Tilt Rotor Test Facility at the University of Maryland," Proceedings of the AIAA SciTech 2019 Forum, San Diego, CA, January 7–11, 2020.

[36]Tsai, F., Sutherland-Foggio, J., Akinwale, A., Morin, A., Gul, S., and Datta, A., "Development and Whirl Flutter Test of the Maryland Tiltrotor Rig," *Journal of the American Helicopter Society*, **69**, 012009 (2024), DOI: 10.4050/JAHS.69.012009.

[37]Zhao, J., Brigley, M., and Modarres, R., "S-97 RAIDER Rotor Low Speed Vibratory Loads Analysis using CFD-CSD," AIAA 2019-0860, Proceedings of the AIAA SciTech 2019 Forum, San Diego, CA, January 7–11, 2019, DOI: 10.2514/6.2019-0860.

[38]Zhao, J., Xin, H., Modarres, R., and Brigley, M., "Methodology Correlation for Coaxial Rotor and Blade Load Prediction," Proceedings of the Vertical Flight Society 76th Annual Forum, October 6–8, 2020.

[39]Datta, A., Patil, M., and Staruk, W., "Loads and Stresses on the Mars Helicopter Ingenuity Rotor," *Journal of the American Helicopter Society*, **71**, 032002 (2026), DOI: 10.4050/JAHS.71.032002.