Advances in Computer, Signals and Systems (2025)
Clausius Scientific Press, Canada

DOI: 10.23977/acss.2025.090406
ISSN 2371-8838 Vol. 9 Num. 4

A Study on the Recursive Decomposition and Optimization Strategies of N-Face Solids Based on CAD/CAE Integration

Mengxiang Li

University of Shanghai for Science and Technology, Shanghai, 200000, China

Keywords: CAD/CAE Integration; Recursive Decomposition; Mesh Quality Optimization; Parallel Computing

Abstract: Considering the recurring roadblocks in recursive decomposition of complex solids for CAD/CAE, i.e., the unexpected geometric deviations in initial parameter setting, the fluctuating convergence behavior and the dilemma to seek meshing schemes fulfilling both mesh quality requirement and acceptable computational performance, this paper aims to develop a holistic optimization framework integrating geometric preprocessing, adaptive recursion control strategy, quality-constrained meshing approach and parallelized computational implementation into the decomposition process. Such approach is expected to greatly improve the robustness and computational efficiency of the decomposition procedure and to deliver mesh outputs satisfying both geometric fidelity and engineering simulation requirements as well, which presents a technical solution to achieve high-quality and high-performance mesh generation.

1. Introduction

As the nature of high-end equipment systems becomes more structured, the marriage of CAD and CAE plays an important role in the digital development of products. The major problem in the integration of CAD and CAE is to convert CAD geometry into appropriate meshes for CAE analysis. The problem becomes more intractable when the considered solids present complex topologies, singular geometric features, or large computation domains[1]. The aim of this study is to apply recursive decomposition methods for N-face solids to solve the above problems. By identifying the underlying problems that lead to robust decomposition failure, we propose several optimization methods to enhance the automation of the CAE analysis while maintaining the numerical reliability of simulation outputs. The aim of these contributions is to address more complex engineering applications where previous meshing strategies cannot scale.

2. Theoretical Foundation and Methodological Framework of Recursive Decomposition for N-Face Solids

2.1 B-Rep-Based CAD Modeling and Mesh Adaptability Analysis

Boundary Representation (B-Rep), commonly used in contemporary CAD systems, provides a rigorous means of specifying the geometry and topology of solid models using a hierarchy of faces, loops, edges and vertices[2]. Although this is a powerful way of representing design intent, it also renders the target mesh generation context more complex, particularly in terms of accommodating diverse surface features and possible geometric anomalies. Recursive decomposition methods applied to N-face solids have no choice but to confront B-Rep surfaces. This implies a willingness to address issues related to surface discretisation, the fidelity of preservation of small geometric features, and the possible problems associated with gaps or overlaps[3]. The analytical stage that determines the topological validity and geometric complexity of a B-Rep model provides a pre-requisite for successful decomposition. This control mechanism will clearly limit the intensity of preprocessing required for a given CAD model and will impact on the algorithm's ability to handle difficult configurations, such as high-curvature surfaces, sharp edges or thin-walled regions. Consequently, a knowledge of the B-Rep structure is critical to ensuring that the subsequent meshing stages provide both geometrically faithful and numerically well-behaved outputs.

2.2 Principles and Workflow of Classical Recursive Decomposition Algorithms

Classical recursive decomposition algorithms, e.g. Octree paradigm and its non-uniform extensions, are driven by the idea of sequential spatial subdivision guided by geometry-aware heuristics. Starting from a bounding volume containing the target model, the method recursively subdivides the space into smaller and smaller subregions, repeating this process until some convergence criteria are met[4]. The cycle of splitting, classification and termination is unfolded in a series of iterations, during which spatial cells are checked against the CAD boundary conditions and further subdivision is decided. All cells fully included and lying entirely outside the geometry are processed separately, while those located in critical boundary regions are subject to further refinement. This design enables fine-grained computational focus to be placed on geometrically involved areas; however, the performance of the algorithm still critically depends on the degree of success with which space partitioning strategies and stopping conditions are formulated. Therefore, an accurate tuning of these parameters is also essential for an efficient trade-off between mesh resolution and computational cost.

2.3 Decomposition Objectives and Quality Evaluation Metrics for CAE Analysis

The ultimate goal of recursive decomposition in CAD/CAE workflows is to enable the production of mesh structures that can survive the numerical simulation process, ranging from simple smoothing filters used in visualization to accurate and numerically well-posed finite-element analyses[5]. While achieving some degree of geometric fidelity in approximating the original CAD boundaries is certainly a basic requirement, it is equally important that the produced mesh satisfies stringent quality constraints to avoid instabilities in the numerical solution. Evaluative criteria commonly relate element shape regularity, smoothness in dimension changes between adjoining cells, and spatial resolution in regions of high variation in physical quantities. These quality dimensions serve both as validation measures with respect to the produced output mesh, as well as guiding cues to inform the recursive decomposition path. By aligning behavior of the proposed framework with simulation-oriented quality goals, we aim to bridge the gap between geometric fidelity and computational pragmatism, yielding meshes that are numerically well-posed and performance-aware.

3. Critical Issues in the Recursive Decomposition of N-Face Solids

3.1 Abnormal Initial Conditions Induced by Complex Topologies and Geometric Imperfections

In real-world engineering meshes, CAD models often contain subtle non-watertight topological features (such as multi-body contact, narrow slots or holes in the interior, or deeply nested part nesting) as well as unavoidable geometric irregularities (such as micro-gaps between surface patches, over or under hanging, or non-manifold edges). All of these contribute to the initial conditions failing to be valid for a stable recursive decomposition.

Although the algorithm can run successfully on the ideal "watertight" model, these run on the "non-ideal" geometry will show significant performance degradation. This is most pronounced in the initial spatial partitioning and boundary classification behavior. In particular, a cell that should be classified as internal may be incorrectly classified as external because of subtle surface discontinuities (at the μ m μ m-scale) that cause holes or voids to appear in the resulting mesh. Cells in regions of sharp angle or thin walls may also fail to reflect the underlying geometric intent if the initial cell partitioning is too coarse and thus introduce errors. In both cases, the resulting initial "anomalies" can significantly increase the amount of preprocessing work, but may also lead the algorithm to fail in the recursive procedure with decomposition errors and/or invalid mesh topology.

3.2 Convergence and Stability Breakdown in the Presence of Singular Geometries

When recursive algorithms are used on models with singular geometrical features (i.e., ultra-thin surfaces, sharply curled transition zones, and long tapering corners), their convergence behavior and stability can be severely impacted. For, in this context, algorithmic stability refers to the ability of a classification algorithm to resist oscillations and possible breakdowns due to numerical round-off errors or borderline classification decisions, and convergence refers to the ability to terminate in a finite number of steps. Traditional recursive strategies based on uniform subdivision or standard rules are highly vulnerable to breakdown in the presence of these singular geometrical features. In sharp corners, recursive subdivisions may repeat, repeat, and repeat again without end because individual cells cannot satisfy competing requirements for geometric conformity and quality assurance. Here, we essentially have an infinite recursion. In thin-walled regions, the algorithm may repeatedly oscillate, oscillate, and oscillate again between two conflicting interpretations of "penetration" and "non-penetration," and thus consume computational resources with little promise of convergence. These local breakdowns can greatly increase the cost of a decomposition. Moreover, they may even compromise the integrity of the overall decomposition process. We thus expose a fundamental weakness in the geometric adaptivity and robustness of conventional recursive decomposition methods.

3.3 Conflict between Mesh Quality Preservation and Geometric Fidelity

The classic tradeoff at play in recursive decomposition is between mesh element quality and boundary accuracy. As the algorithm strives to preserve an accurate representation of a CAD boundary, it will typically induce a "cloud" of dense, small, generally non-regular elements near the boundary-typically in places of high curvature or proximity to fine-scale features. While these elements accurately match the geometry, they often possess poor quality measures such as negative aspect ratios or near-degenerate internal angles that reduce the convergence characteristics and solution accuracy of subsequent finite element analyses. Alternatively, if one attempts to improve element quality through smoothing or local merging, the geometry fidelity is compromised. This dilemma is especially pronounced in cases where surface curvature is large or varying, and fine features are spatially proximate to one another. Addressing this classic limitation presents a key challenge for recursive decomposition algorithms. This is accomplished not by mechanically subdividing mechanical spatial subdivision, but by applying intelligent heuristics that adaptively balance geometric and numerical performance criteria at each stage of the recursion.

3.4 Computational Bottlenecks and Memory Overhead in Large-Scale Model Decomposition

In engineering contexts commonly found in aerospace and automotive products, recursive decomposition algorithms are beginning to be limited by their temporal characteristics and memory consumption profiles. Time to solution for the algorithm can be expected to increase not only with model geometry complexity, but also exponentially with recursive depth and breadth. Each recursive layer incurs further sub-elements that must be produced, and all must be tracked and stored with similar information relating to their geometry and topology, and certain states. Element counts in the millions and tens of millions can lead to memory consumption that quickly climbs to prohibitively high levels, and further demands place substantial stresses on hardware infrastructure.

Additionally, traditional recursive algorithms are fundamentally sequential, and therefore struggle to extract value from parallelism available on modern multi-core machines. Recursive tasks are not easily parallelized due to synchronization requirements and data contexts that exist between parent and child tasks (including issues such as parent-child relationships and data sharing that lead to potential synchronization points and race conditions). These factors hinder simple parallel acceleration. These two factors—extended preprocessing and solution times, and prohibitive memory overhead—have conspired to preclude successful scalability and practical implementation of recursive decomposition technology in commercial high performance CAE environments.

4. Optimization Strategies for Efficient and Robust Recursive Decomposition

4.1 Optimization of Initial Conditions through Geometric Repair and Preprocessing

The rationale behind this strategy is to automatize the preprocessing step needed to turn a failed CAD model into a “clean” model that is more likely to pass mesh generation. The key to this strategy is the development of a geometry-tolerance-based repair logic that automatically repairs microscopic discontinuities that are present along the model surface, divides overlapping geometric entities, removes features that cause non-manifold behavior, thereby guaranteeing that the model is watertight and satisfies manifold conditions (prerequisites for a stable decomposition), and, finally, suppress geometrical details such as chamfers, threaded features, etc. that do not add much mechanical behavior to the part, thus decreasing the overall geometry at the source that has to be processed by the recursive algorithm and improving the robustness of the initial conditions.

In practice, this approach is highly successful provided that one is interested in highly intricate engineering parts such as engine blocks, where features such as threaded bolt holes may correspond to apparent gaps of a few tens of micrometres at the micrometer level. If appropriate preprocessing is inadequate, these gaps can induce misleading boundary classifications during decomposition, leading to discontinuous meshing and part separation. However, by forcing a user-defined geometric tolerance, the preprocessing module allows all minute gaps to be smoothed together, thereby recovering the part`s global watertight integrity, while at the same time, the remaining model is also classified such that features below the desired mesh resolution are removed, thereby eliminating the need for resolving unnecessary geometric irregularities at the desired model resolution. By this means, the topological and geometric complexity of the model is greatly diminished, allowing the recursive decomposition to proceed, starting with greater robustness, and continuing towards the generation of analytical meshes.

4.2 Enhancement of Algorithmic Robustness via Adaptive Recursion Rules and Termination Criteria

This strategy stems from a guiding principle of moving away from fixed rules-based recursive recursive mechanisms towards an intelligent, adaptive framework aimed at algorithm stability, especially in the presence of geometrically singular situations. The novel contribution is the design of a self-adaptive rule library allowing the decomposition process to apply partitioning methods and convergence criteria for cells in a manner that is localized to a geometric context, i.e., they can vary the direction of partitioning around sharp corners to avoid pathological hanging edges, turn on heuristics for penetration-detection in thin regions of the part to reduce oscillations in computation, and vary quality and dimensional convergence criteria as a function of geometry. In this way the strategy follows a differentiated approach in maintaining a coherent algorithmic approach in the presence of poor geometry and achieving stable convergence.

An illustrative application of this approach is found in the T-joint regions of the aircraft skin-to-stringer interfaces, which represent a typical scenario of combined appearance of thin-walls and high angles. The recursive approaches with fixed rules usually would fail in these regions. On one hand, they would excessively subdivide in the planar surfaces; on the other hand, they would prematurely terminate recursions in the T-joint areas due to the dimension constraints, and lose the important geometric information in these regions. In contrast, an adaptive rule set would allow the algorithm to first passively perceive the characteristic of thin-walls in these regions and then actively activate the designed “thin-wall optimized partitioning” mode and ensure the mesh continuity around the structural connections. While this is achieved, the algorithm also would continuously actively monitor the element quality around the acute angles and when the predefined tolerance for element shape degradation would be reached, the recursion would be forced to terminate and invoke the post-processing smoothing mode. Through this feedback mechanism, the oscillation and infinite recursions would be prevented and the important features would be preserved. Finally, the overall success rate and the computational efficiency of the decomposition would be improved in these applications that would usually make the recursive approaches become numerically instable.

4.3 Mesh Generation and Smoothing Optimization Guided by Integrated Quality Metrics and Geometric Constraints

The starting point of our work is a conceptual position that treats the quality evaluation of elements as an inherent constraint during the recursive decomposition process and not as a quality check after mesh generation. This can be achieved if an appropriate multi-objective optimization function can be defined that couples classical geometric quality measures (determinants, aspect ratios, and skewness) with spatial quality measures that describe how close an mesh element is located with regards to the original CAD surface. This function then provides a guiding principle for a local optimization during each iteration of the recursive subdivision and node placement adjustment.

In practice, when highly curved aerodynamic components, such as engine blades with prominent suction and pressure sides, are involved, the well-known recursive decomposition may waste many low-quality elements around the leading and trailing edges, where the gradients of curvature are large. In contrast, if the proposed approach is adopted, in addition to making geometric approximation decisions like the recursive decomposition, the algorithm will also make a prediction of the expected quality of elements that will be generated in the future. If the expected quality of elements that will be generated by a candidate subdivision is poor (i.e., the elements will be very highly distorted), the optimization module will either vary the subdivision point or insert auxiliary Steiner points to alleviate this poor quality problem. The subdivision and insertion are carried out such that surface discretization errors remain within specified tolerances. By embedding this "generate–evaluate–optimize" control loop into the meshing pipeline, the proposed algorithm successfully generates boundary-layer meshes that not only adapt to complex CADs but also display numerically favorable element shapes. This new generation of hybrid meshes is well suited for high-fidelity CFD.

4.4 Acceleration of Decomposition Efficiency via Parallel Computing and Spatial Indexing

The theoretical basis of this approach can be found in redeploying the execution model and data management architecture of the recursive decomposition algorithm to alleviate its scalability challenges when applied to large engineering models. Specifically, executing the decomposition as a single, serially dependent, monolithic processing task is replaced by an arrangement of spatially decomposed, parallelizable subproblems. Spatial indexing structures, e.g. Octrees and KD-trees, are used to dynamically partition the computational domain into a collection of quasi-independent spatial subregions, and parallel processing framework, based on either message passing or multithreaded environments, is used to distribute the localized recursive decomposition tasks across multiple cores. This "decentralized" approach can reduce both computational latency and memory pressure, offering step-change in performance scalability.

One typical application of this approach is full-vehicle crash simulation. Due to model complexity and part density, decomposition is often impossible to run in a reasonable amount of time. However, by utilizing this strategy, the application initially invokes spatial indexing algorithms to divide the vehicle model into dozens of logical subdomains (e.g., body-in-white, engine bay, passenger compartment, and body structure behind the trunk) and then assigns each one to a separate computing node. Each node then runs its own meshing task in parallel and communicates only boundary mesh information at shared interfaces to and from other nodes via MPI through a lightweight communication protocol. This approach divides what would have taken hundreds of hours of serial processing into computations that finish in a few minutes, while also limiting peak memory per node. This enables the strategy to recursively decompose ultra-large engineering models in cases of limited computational hardware.

5. Conclusion

This study has formulated an integrated research framework with theoretical basis, problem diagnosis, and suitable optimization for the recursive decomposition of N-face solids in CAD/CAE integration environment. The proposed optimization flow— including geometric pre-processing, adaptive recursion control, adaptive mesh generation, and parallelized numerical execution—overcomes the main issues of abnormal initial condition, convergence instability, mesh quality degradation, and scalability limitation, respectively. The above results validate a large step improvement in both automation and robustness for the CAD/CAE coupled analysis workflow, which provides basic technology support for the digital design of complex engineered systems. In the future, we will further explore the integrated application of intelligent algorithm, especially the focus is paid on the self-adaptive parameter optimization and machine-learning-based decision-making.

References

[1] Zhang, R., Yu, J., & Zhao, Y. (2025). Parametric CAD/CAE integration for contact stress analysis of spur gears. Journal of Mechanical Design and Manufacturing, (05), 236–241.
[2] Wu, Q., Liu, F., Hu, R., & others. (2021). Research on performance optimization method of mechanical parts based on CAD/CAE integration. Journal of Mechanical Strength, 43(06), 1504–1509.
[3] Tang, B. (2020). Analysis on integrated application of mold CAD/CAE/CAM system. Southern Agricultural Machinery, 51(14), 187–188.
[4] Xu, H., & Feng, Q. (2021). Multi-objective optimization of gating system for injection molded parts based on CAD/CAE/MDO integration. Die & Mould Technology, (05), 13–21.
[5] Kirkwood, R., & Sherwood, J. (2020). Sustained CAD/CAE integration: Supporting simplified models. Journal of Computing and Information Science in Engineering, 20(3), 1-34.