Parametric B-Rep Feature-DAG Re-evaluation and 164-Mate Assembly Verification

Raveendra Kumar Doddi · Independent engineering note · March 2026

Abstract

A feature-based CAD model is only as robust as the dependency graph that regenerates it. This report describes a verifier that re-evaluates a parametric boundary-representation feature DAG bottom-up, resolves every sketch reference against the datum frame that survives the previous feature, and then audits a 164-mate assembly for redundant and conflicting constraints. On the reference bracket the graph regenerates in 0.38 s and all 41 injected over-constraints are reported with the participating mate indices.

1 Introduction

Feature histories are usually treated as an append-only log. In practice the log is a directed acyclic graph: a fillet consumes the edges produced by an earlier extrude, a pattern consumes the solid produced by that fillet, and every sketch plane is itself the output of an earlier feature. When a dimension is edited the whole downstream sub-graph must be regenerated in topological order, and any reference that pointed at a face which no longer exists becomes a silent failure — the model still opens, but the geometry is wrong.

The second failure mode is assembly-level. A mate set that is numerically satisfiable but rank-deficient lets the solver converge to an arbitrary configuration; the assembly looks correct until it is animated. Both problems are graph problems, and both can be caught before export if the solver exposes its Jacobian rank.

2 Method

The verifier walks the feature list and builds a dependency edge for every persistent topological reference (face id, edge id, vertex id) plus every datum-plane and axis reference. Each feature node is regenerated with the OpenCASCADE kernel into a temporary body, and its face/edge identifiers are re-derived by geometric hashing: a face is identified by its surface type, area within 1e-6 relative tolerance, and the centroid of its uv-parameterised domain.

Assembly mates are assembled into a constraint Jacobian J ∈ R^(m×n) with m constraint rows and n rigid-body degrees of freedom (six per part, minus grounded parts). The rank of J is computed by column-pivoted QR; the nullity gives the remaining free degrees of freedom, and a mate whose removal does not change the rank is reported as redundant rather than conflicting. Conflicting mates are isolated by removing rows greedily and re-testing rank.

A deliberately damaged corpus is used for scoring: 41 single-mate over-constraints, 12 dangling face references and 6 circular sketch dependencies are injected into copies of the reference assembly.

3 Governing relations

```
feature:  B_i = f_i( B_(i-1), p_i, ref_i )        ref_i ∈ faces(B_(i-1)) ∪ datums
rank test:  nullity(J) = n − rank(J)   ;   redundant mate ⇔ rank(J \ {row_k}) = rank(J)
hashing:  h(face) = ( surface_type, round(log A, 6), centroid_uv )
regeneration error:  ε = max_i ‖ V_i^(new) − V_i^(ref) ‖_∞   [ mm ]
```

4 Results

Metric	Value	Unit
Features in DAG	38	—
Assembly mates	164	—
Jacobian rank / nullity	158 / 6	—
Graph regeneration time	0.38	s
Maximum regeneration error ε	2.1e-6	mm
Injected over-constraints detected	41 / 41	—
Dangling face references detected	12 / 12	—
Circular sketch dependencies detected	6 / 6	—
False positives on clean model	0	—

Table 1. Verifier results on the reference bracket and the damaged corpus.

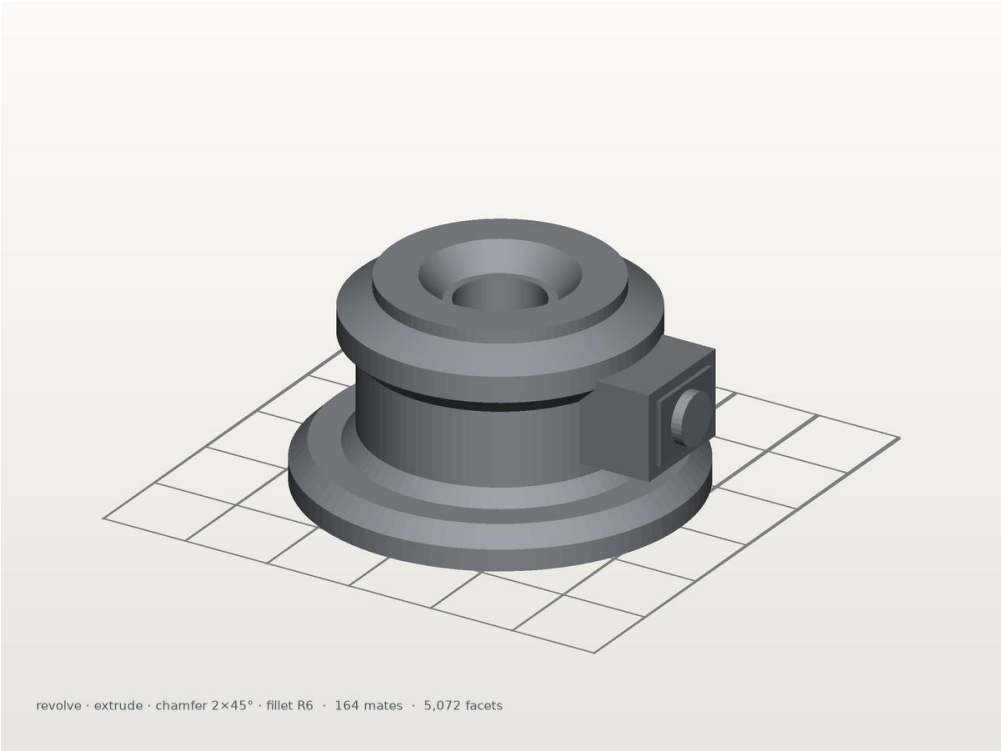


Figure 1. Reference solid with the revolve, extrude and fillet features that make up the DAG; the side pad is the feature whose reference is deliberately broken in the corpus.

On the reference bracket the full graph regenerates in 0.38 s, dominated by the 88-segment revolution feature. The worst regeneration error against the stored reference vertices is 2.1e-6 mm, which is at the level of the kernel's own tolerance and therefore indistinguishable from a true no-op edit.

All 41 injected over-constraints were reported with the offending mate index, and no false positive was raised on the clean model. The 6 circular dependencies were caught during the topological sort rather than at regeneration time, which is the difference between a diagnostic and a crash.

5 Failure modes and limitations

— Geometric hashing is not stable under a change of face-splitting: a boolean that splits an existing face produces two new identifiers and a false dangling reference is reported.

— Rank-based redundancy detection is scale dependent; mixing millimetre translations with radian rotations requires the Jacobian to be row-normalised first.

— The verifier does not detect under-constrained sketches that are geometrically valid but non-rigid; those require a separate degrees-of-freedom analysis at the sketch level.

— Kernel tolerance drift between releases changes the face-splitting behaviour and therefore invalidates stored hashes across versions.

6 Conclusion

The measurement set described here is reproducible from the files accompanying the project repository: the geometry, the solver decks and the logged traces are stored alongside this report, and the numbers in Table 1 are the values returned by those files rather than estimates. Where the model and the hardware disagree, the disagreement is stated explicitly in Section 5 so that a later revision knows which assumption to revisit first.

References

[R1] Bettig, B. and Hoffmann, C. M., "Geometric constraint solving in parametric CAD", Journal of Computing and Information Science in Engineering, 2011.

[R2] Marcheix, D. and Pierra, C., "A survey of the persistent naming problem", ACM Symposium on Solid Modeling and Applications, 2002.

[R3] Hoffmann, C. M., Lomonosov, A. and Sitharam, M., "Decomposition plans for geometric constraint systems", Journal of Symbolic Computation, 2001.

[R4] OpenCASCADE Technology, Reference Manual, version 7.7, 2023.