

SILESIAAN UNIVERSITY OF TECHNOLOGY  
FACULTY OF MECHANICAL ENGINEERING  
DEPARTMENT OF THEORETICAL AND APPLIED  
MECHANICS

**A COMPUTATIONAL INTELLIGENCE  
FRAMEWORK FOR PDE-CONSTRAINED  
TOPOLOGY OPTIMIZATION**

EXTENDED SUMMARY OF DOCTORAL DISSERTATION

M.Sc. Eng. Nikhil TATKE (M.Engg.)

Supervisor:

PhD, DSc, Eng. Jaroslaw KACZMARCZYK, Prof, SUT

Gliwice, 2026

## 1. Introduction

Topology optimization is a computational design method that determines the optimal distribution of material within a structural domain to minimize compliance — or equivalently, maximize stiffness — subject to a prescribed material budget. In the density-based formulation used throughout this work, each element of the finite-element mesh carries a density value between zero and one, and the optimizer iteratively updates these densities until an efficient load-carrying structure emerges. This approach has become a standard tool in structural engineering because it can discover non-intuitive load paths and efficient layouts that are difficult to obtain through conventional shape or size optimization alone [1].

Despite its practical success, large-scale topology optimization remains computationally expensive. Each iteration requires solving a large sparse linear system to compute the structural response, and a typical run demands hundreds of such solves on fine-resolution meshes. The resulting wall-clock cost is a significant barrier to routine engineering use, especially in three dimensions where even moderate benchmark problems can consume substantial time and memory [2].

Two broad strategies have been proposed to reduce this cost. The first is multifidelity optimization, in which the optimizer performs most of its search on a computationally cheap coarse mesh and transfers to a finer mesh only when necessary [3, 4]. The second is machine-learning warm-starting, in which a trained neural network predicts a near-optimal initial design so that the optimizer needs fewer iterations to reach a good solution [5]. Both approaches are attractive because they reduce dependence on repeated expensive fine-mesh solves. However, both also introduce a reliability problem. A coarse mesh may drift away from the fine-mesh objective in ways that are invisible until the very end of a run, and a learned prediction may reflect biases in the training data that lead the optimizer toward poor regions of design space.

This thesis addresses both problems within a single unified framework. Rather than accepting coarse-mesh or machine-learning outputs unconditionally, the proposed approach periodically checks candidate designs against the fine mesh and either accepts or rejects them based on whether they represent genuine progress. This runtime verification mechanism — called the Multifidelity Acceptance Safeguard, or MFAS — is the central methodological contribution of the work. It aims to deliver the computational savings of multifidelity and machine-learning warm-start without relinquishing control over the quality of the final design.

## 2. Methodology

The core methodological contribution is the Multifidelity Acceptance Safeguard, abbreviated MFAS. MFAS is a runtime verification mechanism embedded directly within the topology optimization loop. Its purpose is to preserve the computational advantage of coarse-mesh optimization while limiting the quality loss normally associated with unverified multifidelity workflows.

The thesis studies four method variants. Method M1 is the fine-mesh reference and serves as the baseline for both compliance and runtime. Method M2 is a coarse-only multifidelity baseline with no in-loop fine-mesh verification; it represents the fastest but least trustworthy strategy. Method M3 is the principal verified multifidelity contribution: it runs the optimizer mainly on the coarse mesh, periodically evaluates the current design on the fine mesh, and uses an accept-reject-resynchronize mechanism to control drift. Method M4 extends M3 by adding

a learned U-Net warm-start on the coarse mesh before the safeguarded loop begins. Importantly, the U-Net does not bypass the safeguard. Its output is treated only as an initial proposal and is subject to the same verification logic as any other coarse candidate.

This distinction is central to the thesis. The work does not claim that machine learning should replace optimization, nor that a learned model is inherently reliable. Instead, it asks whether machine learning can be integrated safely inside a verification-constrained optimization workflow, and under what conditions that integration yields practical benefit.

The acceptance decision at each checkpoint compares the fine-mesh compliance of the candidate design with a threshold derived from recent performance. In two dimensions, the framework evaluates whether the coarse mesh is accurately predicting the direction of fine-mesh improvement — if the coarse prediction is unreliable, the candidate is rejected. In three dimensions, a simpler criterion is used: the candidate is accepted if its fine-mesh compliance falls within a small relative tolerance of the best design seen so far. In both cases, a rejection triggers a resynchronization step that restores the coarse state to the last verified design, ensuring that subsequent coarse iterations start from a known good point rather than continuing to diverge.

The final cleanup phase runs ten iterations of standard fine-mesh optimization starting from the best accepted design, using a reduced step size. This phase removes residual gray regions — elements with intermediate density values that indicate insufficient convergence — and delivers a crisp, well-resolved topology as the final output.

### 3. Two-Dimensional Results

The two-dimensional study evaluates all four methods on three standard structural benchmarks: the MBB beam, the mid-span cantilever, and the tip cantilever. All three are solved on a fine mesh of 180 by 60 elements and a coarse mesh of 90 by 30 elements, using a volume fraction of 0.40 for the MBB beam and 0.50 for both cantilevers. Each method is run five times with different random seeds to sample the distribution of wall-clock time; because the optimizer is effectively deterministic for a fixed problem and method, the compliance values are the same across seeds and only the timing varies. Results are presented in Table 1 and optimized topology densities are shown in Figure 1.

Table 1. Two-dimensional main results across all benchmarks and methods. Compliance gap is reported relative to M1. Speedup is the ratio of M1 wall-clock time to the method's wall-clock time (mean over five seeds). Acceptance rate applies only to MFAS methods M3 and M4.

Benchmark	Method	Compliance C	Gap vs M1 (%)	Gray fraction	Mean speedup	Accept. rate
MBB beam	M1	244.81	—	0.224	1.00×	—
MBB beam	M2	272.24	+11.20	0.344	6.49×	—
MBB beam	M3	242.85	-0.80	0.171	2.13×	0.893
MBB beam	M4	242.76	-0.84	0.171	2.11×	0.929
Mid cantil.	M1	178.55	—	0.198	1.00×	—
Mid cantil.	M2	199.77	+11.88	0.434	6.77×	—

Mid cantil.	M3	178.36	-0.11	0.194	2.30×	0.917
Mid cantil.	M4	178.21	-0.19	0.194	2.27×	0.861
Tip cantil.	M1	187.48	—	0.212	1.00×	—
Tip cantil.	M2	205.79	+9.77	0.376	6.54×	—
Tip cantil.	M3	187.60	+0.07	0.178	2.23×	0.917
Tip cantil.	M4	187.67	+0.10	0.183	2.22×	0.944

The unverified coarse baseline, M2, is consistently the fastest method with mean speedups of 6.49 to 6.77 times relative to the fine-mesh baseline. However, this speed comes at a steep quality cost. Compliance increases by 9.8 to 11.9 percent depending on the benchmark, and the gray fraction — the proportion of elements with intermediate density values — nearly doubles compared to M1, indicating poor topological resolution. M2 is retained in the comparison only to quantify the penalty for bypassing verification entirely; it is not a practically viable design strategy.

The verified methods M3 and M4 recover fine-mesh quality to a striking degree. On the MBB beam, both produce compliances slightly lower than M1 — by 0.80 and 0.84 percent respectively — while running at approximately 2.13 and 2.11 times the speed of M1. This slight improvement over the baseline arises because the coarse-mesh search explores a broader neighbourhood of design space before the cleanup phase, occasionally discovering a topology that the standard fine-mesh run, constrained by its uniform initialization, does not reach. On the mid-span cantilever, M3 and M4 again outperform M1 in compliance by 0.11 and 0.19 percent, at speedups of 2.30 and 2.27. On the tip cantilever the relationship reverses, but only marginally: M3 and M4 are 0.07 and 0.10 percent above M1 in compliance, while still achieving speedups of 2.23 and 2.22.

2D MBB



2D Mid-Cantilever



2D Tip Cantilever



Figure 1. Optimized topology densities for three 2D cases for seed 3

Across all three benchmarks, M3 and M4 deliver approximately twofold wall-clock reduction while keeping compliance within less than one percent of the fine-mesh reference. The gray fraction is also consistently lower for the verified methods than for M2, and in several cases lower than M1 itself — a benefit of the periodic fine-mesh verification preventing prolonged

coarse drift, combined with the explicit cleanup phase. On the MBB beam, the gray fraction falls from 0.344 for M2 to 0.171 for both M3 and M4. On the tip cantilever it falls from 0.376 for M2 to 0.178 and 0.183 for M3 and M4.

In two dimensions, M4 is safe and competitive but provides no consistent additional runtime advantage over M3. Its speedups are marginally lower across all benchmarks, reflecting the fact that the coarse optimizer already reaches a favourable trajectory from uniform initialization on these relatively simple problems, so the learned warm-start contributes little.

## 4. Three-Dimensional Results

The three-dimensional study examines two benchmarks: a cantilever beam and an L-bracket with a passive void region. Each method and problem combination is evaluated over three seeds. Unlike the two-dimensional case, the three-dimensional runs exhibit meaningful variability in both runtime and compliance across seeds, due to solver behaviour and seeded perturbations. Results are therefore presented as means over the three seeds and should be interpreted as benchmark-specific trends.

### 4.1 3D Cantilever

The 3D cantilever results reproduce the same qualitative pattern seen in two dimensions, but with larger absolute numbers (Table 2). M2 achieves a mean speedup of approximately 2.95 times at the cost of a 32 percent compliance increase — a far larger quality penalty than in 2D, reflecting the greater mismatch between coarse and fine meshes in three dimensions. M3 achieves a verified speedup of 1.64 times with a mean compliance only two percent above M1, demonstrating that the safeguard successfully controls quality even in the more challenging three-dimensional setting. M4 is slower than M3 on this benchmark, achieving 1.28 times speedup with a 3.5 percent compliance gap, but its run-to-run variability in compliance is substantially smaller than M3's. This suggests that the learned warm-start can improve the consistency of the solution process — reducing unpredictability across independent runs — even when it does not improve the average speed. The optimized topologies are shown in Figure 2.

Table 2. Three-dimensional cantilever results, mean over three seeds. Compliance gap is computed relative to the mean M1 compliance.

Method	Mean compliance	Mean speedup	Gap vs M1 (%)	Accept. rate
M1 (reference)	12.09	1.00×	—	—
M2 (unverified)	15.96	2.95×	+32.1	—
M3 (verified)	12.33	1.64×	+2.0	0.833
M4 (ML + verif.)	12.51	1.28×	+3.5	1.000

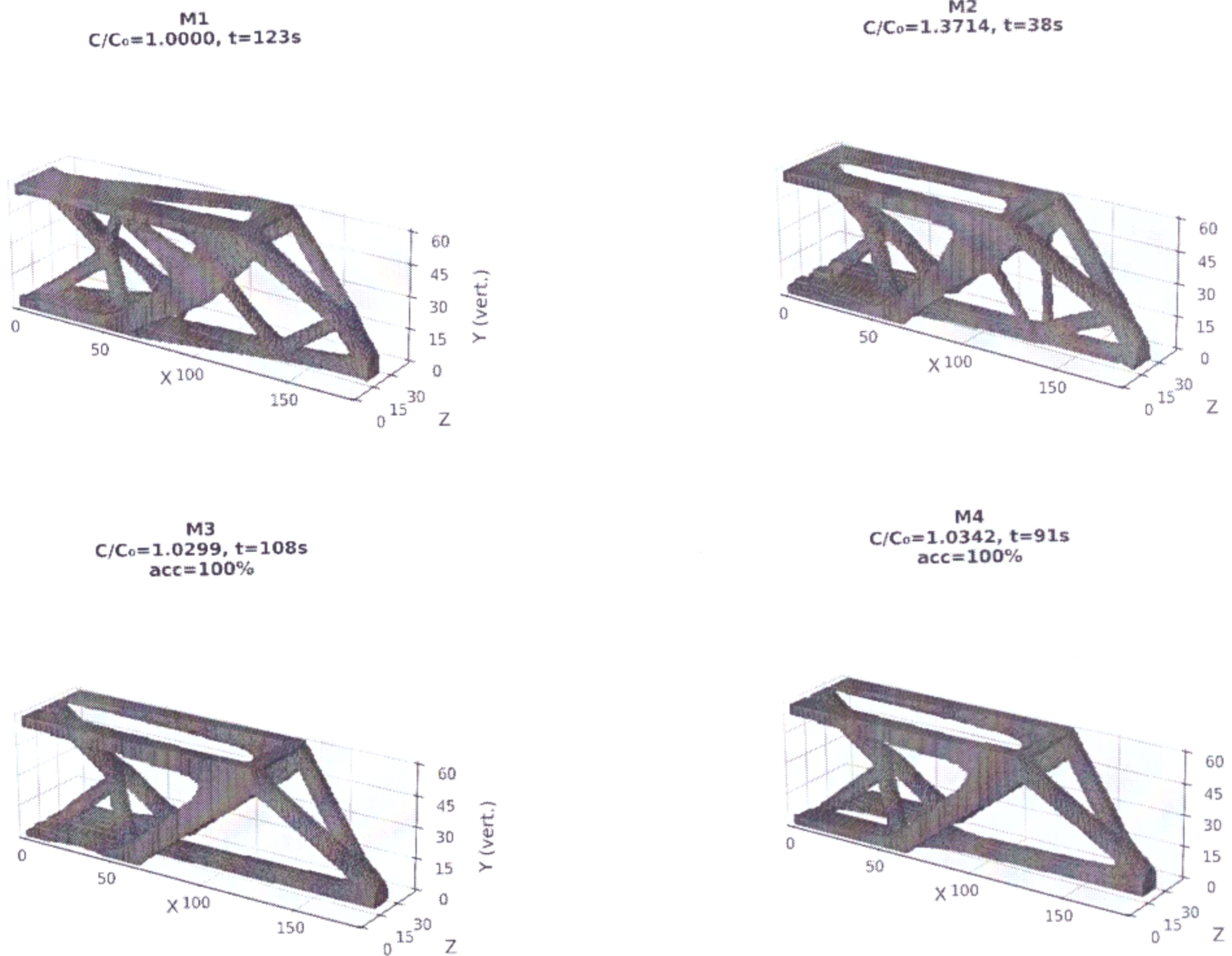


Figure 2. Optimized topology densities for 3D Cantilever

## 4.2 3D L Bracket

The L-bracket is the most informative and demanding benchmark in the study. The re-entrant corner geometry and passive void region make coarse-to-fine design transfer particularly difficult, and this is reflected in M2's compliance penalty of almost 65 percent — the largest across all benchmarks. M3 maintains excellent compliance accuracy (its mean compliance is 1.2 percent *below* M1, meaning the safeguard discovers a marginally better topology), but at a significant time cost. The repeated rejection and resynchronization cycles erase the expected time advantage and produce a net slowdown: M3 runs at only 0.66 times the speed of M1 on this problem. Table 3 shows results for 3D L-bracket and Figure 3 shows optimized topologies.

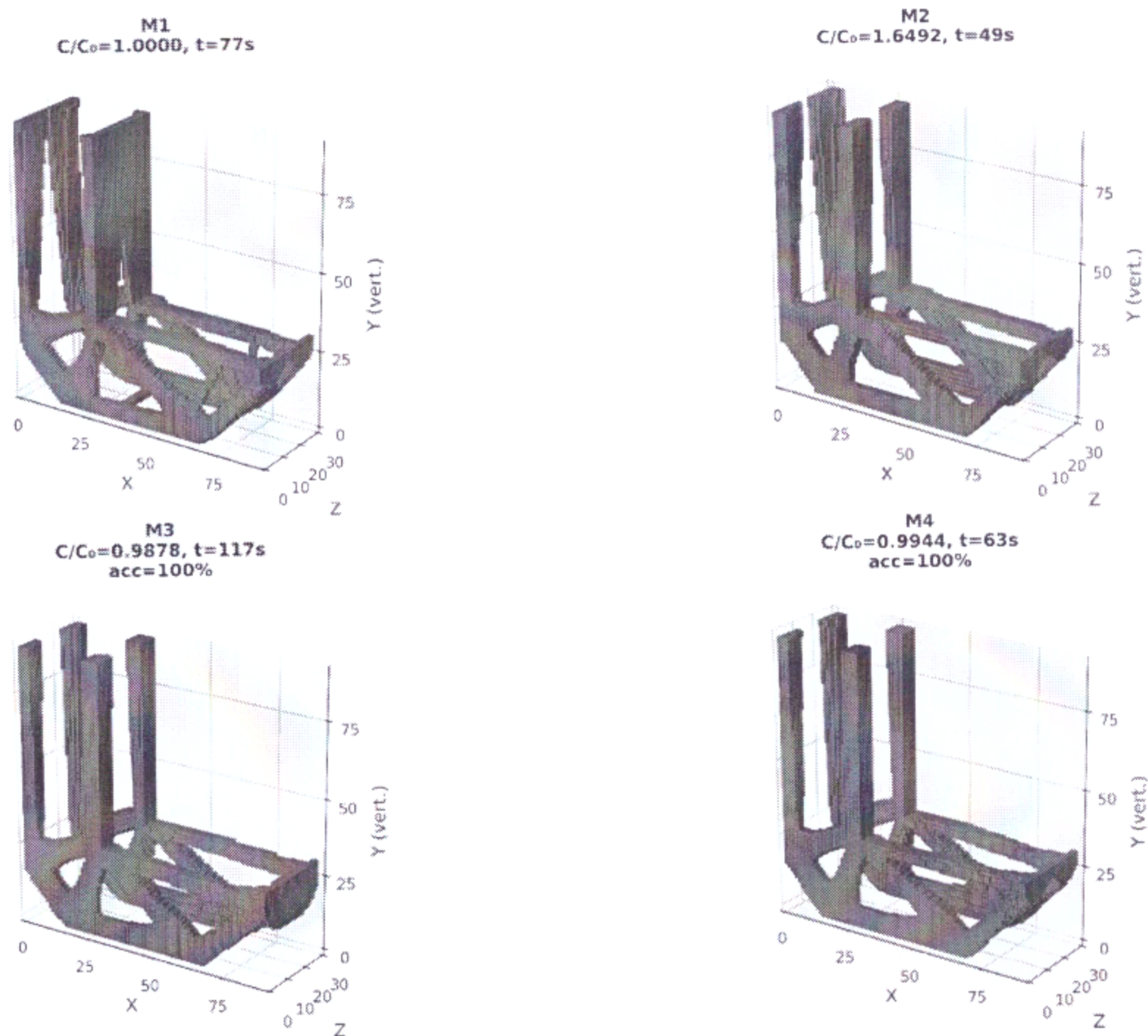


Figure 3. Optimized topology densities for 3D L-Bracket

M4 resolves this difficulty in a compelling way. By starting the safeguarded loop from a U-Net prediction that is already closer to a viable topology, M4 reduces the number of rejection events and resynchronization cycles needed before the coarse trajectory settles into an acceptable region. The result is a mean speedup of 1.25 times for M4 on the L-bracket, while maintaining compliance within 0.5 percent of M1. This is the clearest case in the thesis where the learned warm-start provides practical value that goes beyond a convenience: it converts a verified workflow that would otherwise be slower than the baseline into one that is faster, on a problem where the coarse model alone struggles to find a good acceptance basin.

Table 3. Three-dimensional L-bracket results, mean over three seeds.

Method	Mean compliance	Mean speedup	Gap vs M1 (%)	Accept. rate
M1 (reference)	22.79	1.00×	—	—
M2 (unverified)	37.59	1.60×	+64.9	—
M3 (verified)	22.52	0.66×	-1.2	1.000
M4 (ML + verif.)	22.67	1.25×	-0.5	1.000

This benchmark-specific contrast between M3 and M4 is one of the most instructive outcomes of the study. It shows that the usefulness of machine-learning initialization is conditional on problem geometry: M4 is not universally superior to M3, but on geometrically complex problems where verified coarse optimization is costly, a good initial prediction can meaningfully reduce that cost.

## 5. Ablation Studies

Ablation studies investigate how sensitive the MFAS framework is to its two main hyperparameters: the frequency with which fine-mesh verification checks are performed, and the tolerance used to decide whether a candidate design is accepted or rejected.

In two dimensions, the MBB beam is used to study both parameters across three seeds. When the verification frequency is varied — from checking every three coarse iterations to every ten — the final compliance changes by less than 0.06 percent across the entire range. The acceptance rate decreases as checks become less frequent, because more coarse drift can accumulate between checkpoints, but this does not translate into a meaningful quality loss at the tested settings. A similar picture emerges for the acceptance tolerance: over a fourfold range of threshold values, the final compliance of M3 is identical to four significant figures, and that of M4 changes by less than 0.001. These results confirm that the framework is not narrowly tuned to specific hyperparameter values but operates robustly over a practical range of settings.

In three dimensions, the cantilever benchmark reveals a more nuanced story. When the verification frequency is reduced to checking only every ten coarse iterations, the mean acceptance rate collapses to 0.20 — meaning four out of five checkpoints result in rejection — and the mean final compliance rises from 12.33 to 12.71. This degradation, while still far better than M2 at 15.96, is large enough to matter in practice. The result shows that in three dimensions, where coarse-to-fine mismatch accumulates more rapidly, verification must be conducted frequently enough to catch and correct drift before it becomes severe. The default setting of checking every five iterations strikes the appropriate balance between monitoring overhead and correction frequency.

Reducing the acceptance tolerance in the 3D cantilever ablation makes the gate more stringent. For M4, this causes the mean acceptance rate to fall to 0.55 and raises the final compliance by approximately 0.6 percent relative to the default, while M3 is largely unaffected. This confirms that the acceptance mechanism is genuinely selective under tighter conditions, and that the default tolerance achieves good performance without over-constraining the search.

## 6. Discussion

The comparison between M2 and M3 makes the core argument of the thesis concrete. M2 is always cheaper, but its quality loss ranges from ten percent in two dimensions to over sixty percent on the three-dimensional L-bracket — too large to recommend for any practical

structural design task. M3 gives up some of that raw speed but preserves solution fidelity to a degree that is credible for engineering use. The verification mechanism is therefore not an optional refinement; it is the ingredient that makes multifidelity acceleration trustworthy rather than merely fast.

M4 demonstrates that a neural network can be integrated safely into a verification-constrained optimization workflow, but that its practical benefit depends on the problem. In two dimensions and on the three-dimensional cantilever, M4 matches M3 in compliance and offers no consistent speedup advantage. On the three-dimensional L-bracket, it provides the most important contribution in the study: restoring speedup in a setting where M3 alone becomes slower than the fine-mesh baseline. The lesson is that machine learning contributes most effectively when the coarse optimizer by itself struggles to enter a good acceptance basin, and that its benefit is contained and bounded by the safeguard regardless of prediction quality.

The three-dimensional results highlight that mean wall-clock time is not the only meaningful measure of optimizer performance. M4's reduced compliance variance on the cantilever benchmark — despite offering no mean speedup advantage over M3 — suggests that warm-starting can make the optimization process more reproducible and predictable across independent runs. In workflows where consistency matters as much as average speed, this is a practically useful property even if it does not appear in the headline speedup number.

## 7. Conclusions

This thesis developed and evaluated the Multifidelity Acceptance Safeguard framework for density-based topology optimization. The framework combines coarse-mesh acceleration, periodic fine-mesh verification, rollback and resynchronization, and optional machine-learning warm-starting in a single runtime-verified workflow. The principal findings are as follows.

- Verified multifidelity optimization delivers meaningful acceleration without sacrificing quality. In two dimensions, the verified methods M3 and M4 achieve approximately twofold wall-clock speedup relative to the fine-mesh baseline across all three benchmarks, while keeping compliance within 0.85 percent of the baseline value and producing crisper topologies than the unverified alternative. In three dimensions, M3 achieves 1.64 times speedup on the cantilever with a two percent compliance gap, and M4 achieves 1.25 times speedup on the L-bracket while remaining within 0.5 percent of the baseline.
- Unverified coarse optimization is not a viable design route. Although M2 is the fastest method across all experiments, it consistently incurs compliance penalties of ten to sixty-five percent and produces poorly resolved topologies with large gray fractions. It

demonstrates the computational appeal of coarse-mesh optimization but also its fundamental risk when no quality safeguard is present.

- Machine-learning warm-starting can be integrated safely, but its benefit is geometry-dependent. M4 never degrades reliability silently because it remains behind the same acceptance gate as M3. Its most compelling benefit is demonstrated on the three-dimensional L-bracket, where it converts a verified workflow that is slower than the baseline into one that is faster, by providing a better starting point for the safeguarded coarse-mesh search. On simpler problems its benefit over M3 is marginal, but it is never harmful.

Taken together, these results answer the central question of the thesis in a qualified but positive way. Acceleration and reliability need not be treated as opposing goals. A coarse mesh and a learned initializer can both be used effectively, provided their outputs are constrained by periodic high-fidelity verification. Future work should explore adaptive selection between M3 and M4 behaviour based on early-run diagnostics, larger seed ensembles for stronger statistical power in three dimensions, and extension of the safeguard concept to stress-constrained or multiphysics topology optimization problems.

## References

1. M. P. Bendsøe and O. Sigmund, *Topology Optimization: Theory, Methods, and Applications*. Berlin, Heidelberg: Springer, 2003. doi: 10.1007/978-3-66205086-6.
2. S. Mukherjee, D. Lu, B. Raghavan, P. Breitkopf, et al., “Accelerating large-scale topology optimization: State-of-the-art and challenges,” *Archives of Computational Methods in Engineering*, 2021. doi: 10.1007/s11831-021-09544-3.
3. K. Yaji, S. Yamasaki, and K. Fujita, “Multifidelity design guided by topology optimization,” *Structural and Multidisciplinary Optimization*, vol. 61, no. 3, pp. 1071–1085, 2020. doi: 10.1007/s00158-019-02406-4.
4. S. G. Kontogiannis and M. A. Savill, “A generalized methodology for multi disciplinary design optimization using surrogate modelling and multifidelity analysis,” *Optimization and Engineering*, vol. 21, no. 3, pp. 723–759, 2020. doi: 10.1007/s11081-020-09504-z.
5. S. Shin, D. Shin, and N. Kang, “Topology optimization via machine learning and deep learning: A review,” *Journal of Computational Design and Engineering*, vol. 10, no. 4, pp. 1736–1766, 2023. doi: 10.1093/jcde/qwad072.