The Physicist, the Coder, and the Critic: Agentic Photonic Crystal Fiber Inverse Design via Verbal Gradient Descent

Shengchao Chen[1,2], **Ting Shu**[1]
[1]School of Artificial Intelligence, Shenzhen University, Shenzhen, China
[2]Australian AI Institute, University of Technology Sydney, Sydney, Australia
Correspondence: tingshu@szu.edu.cn

Abstract

Inverse design of photonic crystal fibers (PCFs) is a challenging simulation-driven problem due to highly non-linear physics, black-box solvers, and strict fabrication constraints. While large language models (LLMs) offer new opportunities for iterative design, existing text-only agents lack physical grounding and effective feedback for reliable optimization. To address these challenges, we propose PhysAgent-PCF, an agent framework that reformulates PCF inverse design as an iterative, language-guided interaction with simulators. PhysAgent-PCF decouples high-level physical reasoning from code execution via a dual-brain architecture, and employs a multimodal critic to convert visual and spectral simulation discrepancies into structured natural-language feedback, enabling trajectory-aware, gradient-free optimization. We also construct DeepPCF-1M, the first large-scale multimodal benchmark for PCF inverse design. Experiments show that PhysAgent-PCF consistently outperforms strong baselines across diverse design tasks and robustness settings, demonstrating its effectiveness and superiority for physically grounded inverse design. Code and dataset are available at PhysAgent.

1 Introduction

Photonic crystal fibers (PCFs) are microstructured optical fibers whose periodic claddings enable optical properties beyond those of conventional fibers (Markos et al., 2017). Designing PCFs to meet target behaviors such as dispersion or modal control typically relies on iterative interaction with computationally expensive Maxwell solvers (Zhu et al., 2025), making the process slow and difficult to scale. Consequently, PCF inverse design exemplifies a broader class of complex, simulation-driven scientific optimization problems.

To automate PCF design, prior work has explored inverse methods ranging from population-based optimizers to neural networks. Classical approaches such as genetic algorithms (Yang et al., 2022) and particle swarm optimization (Kaziz et al., 2024) perform black-box parameter search but require extensive solver evaluations and scale poorly in high-dimensional spaces. More recent learning-based methods replace physics solvers with neural surrogates to enable faster forward prediction and gradient-based optimization (Chen et al., 2023; Ren et al., 2024). However, inverse photonic design is inherently ill-posed: multiple geometries can yield identical optical responses, causing discriminative models to collapse to ambiguous or physically meaningless solutions. Generative approaches (Ma et al., 2022) alleviate this one-to-many mapping by sampling diverse designs, yet often lack fabrication awareness and semantic control. As a result, existing methods remain limited in their ability to reason over physical constraints, incorporate natural-language specifications, or provide interpretable design rationales.

Recent advances in large language models (LLMs) and agentic frameworks suggest a promising alternative for simulation-driven inverse design (Hao et al., 2025; Han et al., 2025). In principle, LLM-based agents can iteratively propose designs, execute black-box simulations, and refine hypotheses through closed-loop interaction. However, directly applying existing LLMs or text-only agents to PCF design remains challenging. Such tasks involve subtle, highly non-linear couplings between geometric parameters and optical field responses, where small structural changes can induce qualitatively different modal behaviors. Standard LLMs struggle to simultaneously manage high-level physical reasoning and low-level implementation details, while textual simulation feedback alone is insufficient to diagnose geometric or field-level failures (Chen et al., 2025). Therefore, current agentic workflows are prone to hallucination, oscillatory refinement, and inefficient exploration, limiting their effectiveness in realistic PCF design settings.

Findings of the Association for Computational Linguistics: EMNLP 2026, pages 21837–21855
October 24-29, 2026 ©2026 Association for Computational Linguistics

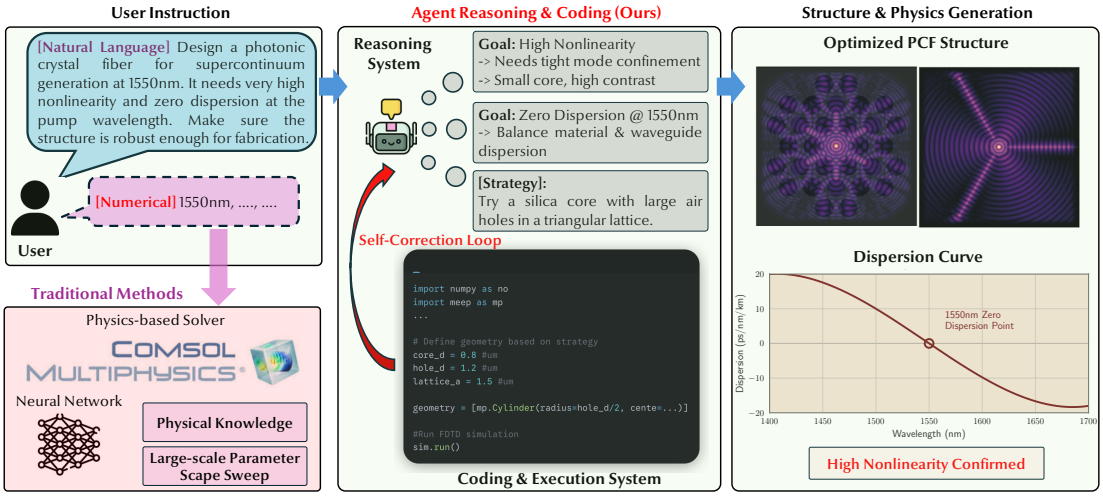


Figure 1: Overview of **traditional** parameter optimization methods and **our** agentic method for PCF inverse design.

To address these challenges, this paper proposes PhysAgent-PCF, the first multimodal agent framework tailored for PCF inverse design beyond traditional methods (refer to **Fig. 1**), which formulates the problem as an iterative, trajectory-aware interaction between language models and black-box simulators. PhysAgent-PCF decouples physical reasoning from code execution via a planner that generates high-level design hypotheses in natural language and a coder that translates them into executable simulation scripts. A multimodal critic further provides visual feedback from simulation outputs, converting discrepancies between observed and target behaviors into structured linguistic guidance that steers subsequent iterations without gradients. Together with episodic memory, this design enables grounded and interpretable exploration of the highly non-linear PCF design space, effectively bridging high-level physical intent and low-level executable code. Our contributions are three-fold:

- We construct the first large-scale, multimodal dataset for PCFs, containing 1 million triplets. The dataset features a stratified test set covering 9 topological tiers, serving as a rigorous testbed for zero-shot scientific reasoning.
- We propose PhysAgent-PCF, a multimodal agent framework for PCF inverse design built on *Verbal Gradients*, a semantic derivative-free mechanism that turns multimodal simulation feedback into directional updates over executable code, bridging physical reasoning and implementation.
- Extensive experiments show that PhysAgent-PCF consistently outperforms reasoning-centric, coding-specialist, and text-only agentic baselines across diverse topologies and physical objectives, confirming the effectiveness and generality of PhysAgent-PCF.

2 Related Work

Learning-based PCF Inverse Design. PCF inverse design is typically formulated as an optimization problem over structural parameters, driven by the highly non-linear coupling between microstructured geometries and optical responses (Chen et al., 2023). To reduce the cost of physics-based solvers, a wide range of machine learning approaches have been explored to approximate structure–property relationships, including regression models (Ma et al., 2022), convolutional (Xiao et al., 2024) and graph-based networks (Liu et al., 2026), as well as generative models (Bhadwal et al., 2025) that synthesize candidate geometries conditioned on target specifications; related surrogate pipelines have also been applied to knowledge-fused reverse design (Ren et al., 2024) and to photonic signal interrogation (Chen et al., 2022). While these methods improve efficiency, they predominantly operate as parameter-level optimizers, producing designs without explicit design feedback or interpretable reasoning. Most require task-specific training, offer limited generalization across objectives, and lack the ability to iteratively diagnose and correct design failures (Chen et al., 2025). Existing PCF inverse design approaches remain disconnected from interactive, feedback-driven design workflows that are essential for unified and extensible design systems.

LLMs and Agentic Scientific Design. Recent advances in LLMs have demonstrated strong ca-

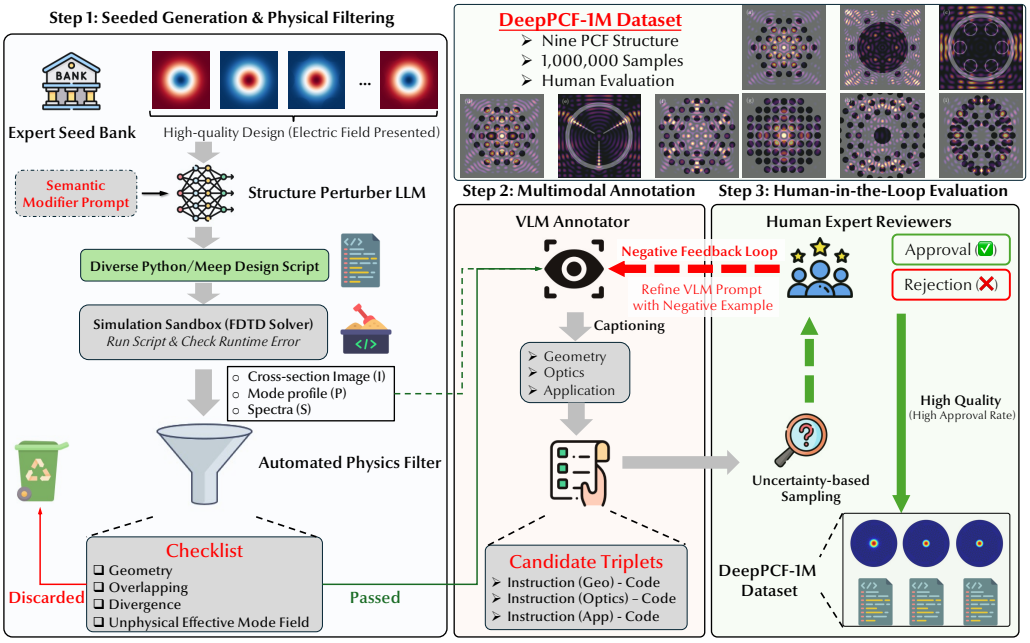


Figure 2: Physically-Grounded Human-in-the-Loop pipeline for DeepPCF-1M construction, the dataset spans 9 different PCF topological tiers, and its statistics are in **Table 11**. The details of human evaluation in **Appendix A**.

pabilities in scientific reasoning (Ma et al., 2025), code generation (Zamfirescu-Pereira et al., 2025), and tool-based interaction (Lu et al., 2025), enabling their use in automated experimentation, scientific programming, and black-box optimization. Building on these capabilities, agentic frameworks allow LLMs to interact with external environments such as simulators and compilers through closed-loop execution and feedback (Wang et al., 2025a; Ma et al., 2024). However, applying existing agentic frameworks to physically constrained inverse design tasks poses additional challenges (Polat et al., 2025). In PCF design, small geometric changes can cause highly non-linear variations in optical fields, requiring agents to jointly reason over geometry, physics, and execution outcomes. Under such conditions, scalar rewards or textual logs offer limited diagnostic guidance, while short-horizon, execution-centric agent frameworks struggle to capture long-term design trajectories and iterative hypothesis refinement. Concurrent efforts explore memory-centric agents that accumulate reusable design skills for photonic inverse design (Chen et al., 2026), yet they remain text-only and do not exploit visual simulation evidence. This paper proposes PhysAgent-PCF, an agent framework tailored for PCF inverse design. Our proposed PhysAgent-PCF systematically integrates multimodal simulation feedback with trajectory-aware, language-guided reasoning, filling a critical gap in physics-driven inverse design workflows.

3 DeepPCF-1M Benchmark

We construct DeepPCF-1M, the first large-scale multimodal dataset pairing natural language instructions with executable simulation scripts for PCF inverse design. Unlike general-purpose code datasets, DeepPCF-1M captures the complex coupling between geometric parameters and optical properties such as dispersion, confinement loss, and nonlinearity. To ensure data validity and structural diversity, we curate the dataset using a Physically-Grounded Human-in-the-Loop (PG-HIL) pipeline (**Fig. 2**), detailed in the following section.

Seeded Generation via Semantic Perturbation. Unlike random parameter sweeping, which often produces physically trivial designs, we adopt a Seeded Perturbation Strategy. We curate an expert bank of 2,000 high-quality seeds spanning nine PCF topological categories, and prompt a code LLM to apply semantic modifiers (e.g., "Transform the cladding into a Kagome lattice for low-loss transmission") (Argyros and Pla, 2007). This process yields 1.6M raw scripts, enabling broad coverage of complex PCF design variations with diverse physical properties.

Automated Physical Grounding. To filter out hallucinated or non-manufacturable designs, we deploy a three-stage Automated Physics Filter, including **(1)** Syntax Check, Verifies if the generated Python script compiles and correctly calls the

FDTD solver APIs (we use Meep in this work). **(2)** Physical Feasibility Check, We rasterize the defined geometry into a high-resolution 2D matrix G and compute the Euclidean distance map. This detects physically impossible overlaps (e.g., clashing air holes) or disjointed waveguides that violate fabrication constraints. **(3)** Solver Convergence, We execute a short-time FDTD simulation. Scripts that cause solver divergence (NaN fields) or yield non-physical effective indices ($n_{\text{eff}} < 1$ or $n_{\text{eff}} > n_{\text{silica}}$) are discarded. This rigorous filtering removes approx. 37.5% of the raw data, resulting in a refined dataset of **1,000,000** valid pairs.

Multimodal Reverse Captioning. We employ a multi perspective annotation strategy to bridge visual geometry and optical performance. For each triplet (I, S, C), where I denotes the fiber cross section image, S contains simulation outputs such as dispersion curves and mode profiles, and C is the executable script, we generate a natural language instruction T that captures the underlying design intent. Specifically, annotations are constructed from complementary views, including **(1)** Geometric Perspective: Describes topological features (e.g., *"A six-ring hexagonal lattice with a high air-filling fraction..."*), **(2)** Optical Perspective: Interprets the mode field and spectral response (e.g., *"The mode is tightly confined, exhibiting anomalous dispersion at 1.55μm..."*), **(3)** Instruction Synthesis: We fuse these descriptions to synthesize diverse instructions T, ranging from precise parameter requests to abstract functional requirements (e.g., *"Design a fiber suitable for supercontinuum generation"*).

Uncertainty-based Active Verification. To improve annotation reliability at scale, we incorporate an active learning (Ren et al., 2021)–based verification loop into the data curation pipeline. Rather than uniform sampling, verification is selectively applied to samples with high VLM perplexity or physical properties near boundary conditions. A statistically representative subset of 5,000 samples is reviewed by domain experts with advanced photonics training, who provide binary judgments on whether the generated instruction T faithfully reflects the physical consistency of the corresponding image I and structure S. This process yields a human acceptance rate of 94.2% (Table 10). The dominant source of rejection arises from geometric hallucinations (42.5%), which we mitigate by explicitly grounding the VLM on the rasterized geometry matrix G. Rejected samples are further reused as negative demonstrations in subsequent annotation batches, enabling progressive correction of systematic errors and continuous improvement of annotation quality as the dataset scales.

4 Methodology: PhysAgent-PCF

Problem Formulation. We formulate PCF inverse design as training-free black-box optimization over executable simulation scripts. Let $\mathcal{C}$ be the code space and $f_{\text{sim}} : \mathcal{C} \to \mathcal{P}$ a non-differentiable solver mapping code to physical responses. A frozen LLM iteratively generates candidate scripts $c_t{}_{t=1}^{T}$ conditioned on the interaction history, and the best-performing script in the trajectory is selected. Since ∇f_{sim} is unavailable, the agent is guided by structured semantic signals extracted from visual discrepancies rather than a numerical objective. Crucially, unlike fixed-template optimization over a set of 3–8 continuous parameters (e.g., pitch Λ and hole diameters $d_1, \ldots, d_n$), operating in code space lets the agent dynamically expand this set, e.g., by adding nested tubes/breaking symmetries.

Overview. PhysAgent-PCF (see **Fig. 3**) performs dual-loop reasoning, with an inner loop ensuring code executability through syntax-level validation and self-correction, and an outer loop performing refinement via iterative updates informed by multimodal feedback. The framework comprises three decoupled modules: a Dual-Brain Generator for hierarchical planning and implementation, a Multimodal Critic that converts simulation feedback into verbal gradients, and an Episodic Memory that enables trajectory-aware hypothesis refinement.

4.1 Decoupled Dual-Brain Reasoning

Standard LLMs often struggle to jointly handle high-level physical reasoning (e.g., dispersion shaping) and low-level implementation details (e.g., Meep API syntax), leading to context dilution where physical constraints are lost amid syntax correction. To mitigate this issue, we decouple generation into two specialized roles. A Physicist (System-2 Planner π_{plan}) operates purely in the semantic space, analyzing the task and producing a natural-language design blueprint z that encodes physical strategies and geometric intent. A Coder (System-1 Actor π_{code}) then grounds this blueprint into an executable simulation script c. The overall generation is factorized hierarchically as

$$P(c|t) = \sum_{z} P_{\text{code}}(c|z) \cdot P_{\text{plan}}(z|t, \mathcal{H}_t) \qquad (1)$$

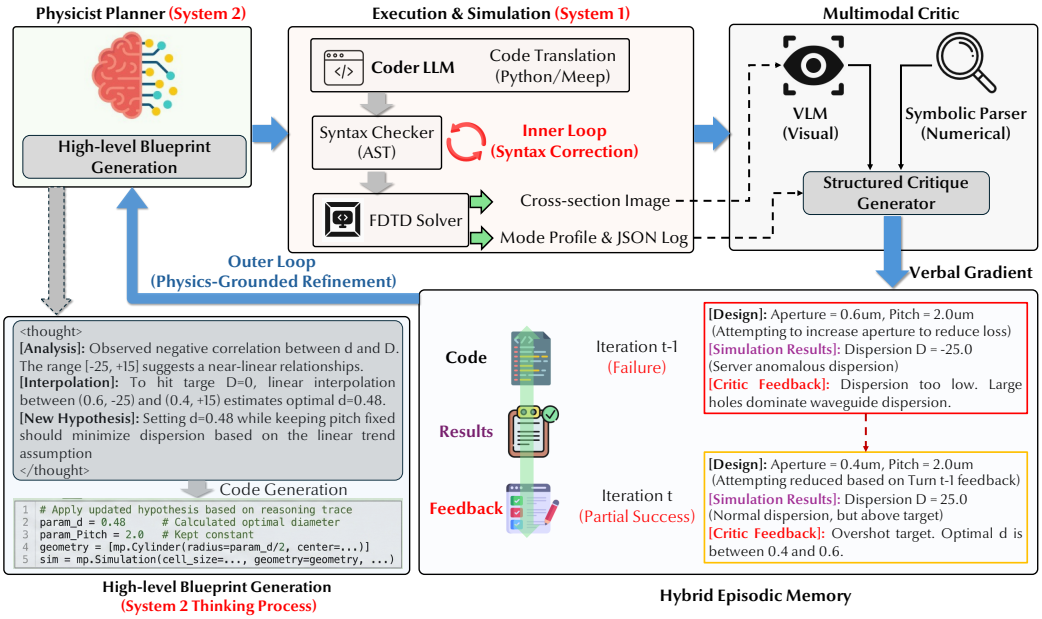


Figure 3: Overview of the proposed PhysAgent-PCF framework.

where $\mathcal{H}_t$ is the interaction history. This allows the planner to focus on why a structure should be designed, while the coder concentrates on how to implement it, reducing reasoning–execution interference and improving physical fidelity.

4.2 Multimodal Critic with Verbal Gradients

In simulation-driven optimization, scalar rewards signal failure but neither explain its cause nor suggest a correction, making black-box exploration inefficient. We therefore introduce verbal gradients (∇_{verbal}), a semantic alternative to numerical derivatives that bridges the *semantic-to-numeric gap*: numerical gradients cannot operate over discrete topology edits in code, while raw critic text is verbose and unstable. We cast them as derivative-free, sign-based directional signals over executable code rather than fixed continuous vectors (Appendix C). A vision–language critic analyzes the multimodal simulation output $O_t = \{I_{\text{mode}}, S_{\text{spec}}\}$ and produces structured feedback that diagnoses failures, grounds them in visual or spectral evidence, and proposes directional adjustments. Formally, the critic C_ϕ maps the observation–target discrepancy to a semantic update signal:

$$\nabla^{(t)}_{\text{verbal}} = C_\phi(O_t, p_{\text{target}}) \approx \frac{\partial \mathcal{L}_{\text{phy}}}{\partial \text{Semantics}}, \quad (2)$$

which acts as a prompt-based pseudo-gradient steering the planner toward physically meaningful refinements (e.g., enlarging cladding holes to reduce mode leakage). A lightweight deterministic parser then converts this diagnosis into a strict schema $\{\text{PARAMETER}, \text{DIRECTION} \in \{-1, 0, +1\}, \text{MAGNITUDE}\}$, giving the planner a clean, math-aligned instruction rather than unstructured prose. The signal is thus conceptual rather than numerical, operating in semantic space (Detailed analysis in Appendix C).

4.3 Hypothesis Update via Episodic Memory

A common failure mode in agentic optimization is parameter oscillation, where the agent repeatedly toggles between suboptimal configurations without convergence due to the lack of long-term trajectory awareness. To address this issue, we introduce an Episodic Memory Replay mechanism that enables trajectory-aware planning, in the spirit of treating accumulated design experience as a reusable policy (Chen et al., 2026). The agent maintains a memory buffer $\mathcal{M} = (z_i, c_i, O_i, \nabla^{(i)}_{\text{verbal}})_{i=0}^{t-1}$, which records past blueprints, executed codes, observed outcomes, and corresponding verbal gradients. Before generating the next blueprint z_{t+1}, the planner explicitly retrieves relevant past experiences and performs a hypothesis update, synthesizing prior failures into a revised physical conjecture that guides subsequent exploration. Formally, the next blueprint is generated by conditioning the planning policy on the retrieved episodic context and the latest verbal feedback. It can be formulated as

$$z_{t+1} = \arg\max_{z} \pi_{\text{plan}}\big(z \mid t,\ \text{Retrieve}(\mathcal{M}, k),\ \nabla^{(t)}_{\text{verbal}}\big), \quad (3)$$

thereby transforming a memoryless Markov process into trajectory-aware optimization procedure that suppresses oscillatory behavior and promotes smooth convergence through logical deduction.

5 Experiments and Results

Baselines. We benchmark our PhysAgent-PCF against advanced baselines, categorizing them into three distinct paradigms, including (1) Reasoning Models (System 2 Native): Gemini-3-Flash (Team et al., 2025), DeepSeek-R1 (Guo et al., 2025), OpenAI o1, Qwen3-235B-A22B-Thinking-2507 (Yang et al., 2025), (2) Coding Specialists (System 1 Native): Qwen3-Coder-480B-A35B-Instruct (Yang et al., 2025), DeepSeek-Coder-V2-Instruct (DeepSeek-AI et al., 2024), Claude Sonnet 4.5, Gemini 3 Pro Preview (Team et al., 2025), GPT-5.1-Codex-Max, (3) Agentic Loops: Standard ReAct (GPT-4o and Claude 3.5) (Yao et al., 2023), SWE-Agent (GPT-4o and Claude 3.5) (Yang et al., 2024), OpenHands (GPT-4o and Claude 3.5) (Wang et al., 2025b).

Implementation Details All models are accessed through the OpenRouter API service, and simulations are conducted locally with Meep v1.28 (Oskooi et al., 2010). For fair comparison, all agentic baselines operate under a strict budget of $T_{\max} = 5$ simulation turns per task, where each turn consists of a full cycle of code generation, simulation execution, and feedback. PhysAgent-PCF is implemented using a modular Dual-Brain configuration. The System-2 Planner employs Qwen3-235B-Thinking with temperature $\tau = 0.6$. The System-1 Coder uses Claude 4.5 Sonnet with $\tau = 0.1$. For multimodal feedback, we adopt GPT-4o-2025-Vision as the visual-language critic. For in-context grounding, PhysAgent-PCF and *all* agentic baselines retrieve the top-k ($k=2$) Meep scripts from the *training corpus only* via dense-vector search (`text-embedding-3-small`, cosine similarity over the target specification); retrieved scripts supply syntax and lattice/boundary templates rather than structural solutions. To preclude leakage, a strict out-of-distribution split places test targets beyond the perturbation radius of any retrieved seed, so reaching them requires genuine topology rewriting, not parameter copying.

Evaluation Metrics We evaluate model performance using multiple complementary metrics. Execution Rate (Exec%) (Chen et al., 2021) measures whether generated scripts run without syntax errors or solver divergence, while Physical Consistency Score (PCS) (Jiang et al., 2019) assesses geometric correctness via IoU with the target topology. A generation is considered successful if it is executable and its simulated optical properties fall within ±5% of the target, yielding the Success Rate (SR) (Peurifoy et al., 2018). We also report Pass@1 (Chen et al., 2021) to measure generation stability. Details for these metrics are provided in **Appendix D**.

5.1 Main Results

As shown in **Table 1**, PhysAgent-PCF achieves the best overall performance on DeepPCF-1M, attaining the highest success rate and execution robustness with fewer interaction turns. These gains arise from two complementary sources rather than architectural decoupling alone: the dual-brain design improves *syntactic stability* (98.8% Pass@1) and enables role specialization, while the *multimodal critic* drives the macroscopic success-rate gains through simulation-grounded feedback (see the ablation in Table 2). Consequently, PhysAgent-PCF consistently outperforms pure reasoning models and text-only agents, which lack both structured multimodal feedback and role specialization.

5.2 Qualitative Analysis

Fig. 4 visualizes representative geometries and their transient E_z fields from FDTD. Existing paradigms show distinct failure modes: reasoning-centric models hallucinate geometry, coding specialists collapse to generic symmetric templates, and text-only agents, lacking geometric awareness, converge to suboptimal solutions. In contrast, PhysAgent-PCF uses multimodal visual feedback to jointly resolve topology and fine geometry, producing valid, task-aligned designs (e.g., symmetry-broken PM fibers, hollow-core ARFs, and ultra-thin suspended cores with well-confined modes). These results highlight visual grounding as key to physical correctness beyond text-only generation.

5.3 Ablation Study

We perform a component ablation using two diagnostic metrics: Spatial Accuracy, measuring topological correctness, and Spectral Accuracy, assessing optical performance given a valid topology. As shown in **Table 2**, removing the VLM-based visual critic leads to a pronounced drop in Spatial Accuracy, while disabling episodic memory increases interaction cost and degrades Spectral Accuracy due

Method / Model	Backbone / Size	Code Quality Pass@1	Code Quality Exec.	Physical Grounding PCS (IoU)	Physical Grounding SR (%)	Efficiency Avg. Turns
Cat 1: Reasoning Models (System 2 Native) - Zero Shot						
Gemini-3-Flash (Team et al., 2025)	-	88.1	94.5	55.2	45.4	1.0
Qwen3-235B-A22B-Thinking-2507 (Yang et al., 2025)	235B (MoE)	91.8	96.8	68.9	60.1	1.0
DeepSeek-R1 (Guo et al., 2025)	671B (MoE)	90.5	95.2	70.1	61.2	1.0
OpenAI o1	-	95.4	98.9	69.8	63.5	1.0
Cat 2: Coding Specialists (System 1 Native) - Zero Shot						
DeepSeek-Coder-V2-Instruct (DeepSeek-AI et al., 2024)	236B	89.2	96.1	56.4	43.8	1.0
Gemini 3 Pro Preview (Team et al., 2025)	-	94.5	98.2	58.1	47.5	1.0
Qwen3-Coder-480B-A35B-Instruct (Yang et al., 2025)	480B	96.8	99.1	61.5	49.8	1.0
Claude Sonnet 4.5	-	97.5	99.4	60.8	50.2	1.0
GPT-5.1-Codex-Max	-	**99.1**	**99.6**	63.4	52.6	1.0
Cat 3: Agentic Loops (Text-Feedback Baselines)						
Standard ReAct (Yao et al., 2023)	GPT-4o	89.4	95.8	75.1	70.4	4.2
	Claude 3.5	90.1	96.5	73.8	67.8	4.6
OpenHands (Wang et al., 2025b)	GPT-4o	93.5	97.4	76.2	71.5	4.8
	Claude 3.5	96.1	98.5	78.1	74.8	4.6
SWE-Agent (Yang et al., 2024)	GPT-4o	94.2	98.1	77.8	74.5	4.4
	Claude 3.5	95.9	98.5	77.1	73.9	4.7
Ours: PhysAgent-PCF (Multimodal-Feedback)						
PhysAgent-PCF (Unified)	GPT-4o	96.8	99.0	88.5	84.9 (+10.4)	3.1
PhysAgent-PCF (Unified)	Claude 3.5	97.2	99.2	90.1	86.8 (+12.0)	2.9
PhysAgent-PCF (Hybrid)*	**Qwen3-Think + Claude 4.5**	98.8	**99.6**	**95.4**	**91.2**	**2.4**

Hybrid Config: Planner = Qwen3-235B-A22B-Thinking, Coder = Claude Sonnet 4.5. Δ compares against the best Cat 3 agent with the same backbone.

Table 1: Main results on DeepPCF-1M benchmark. **Bold**: the best; Underline: the second-best.

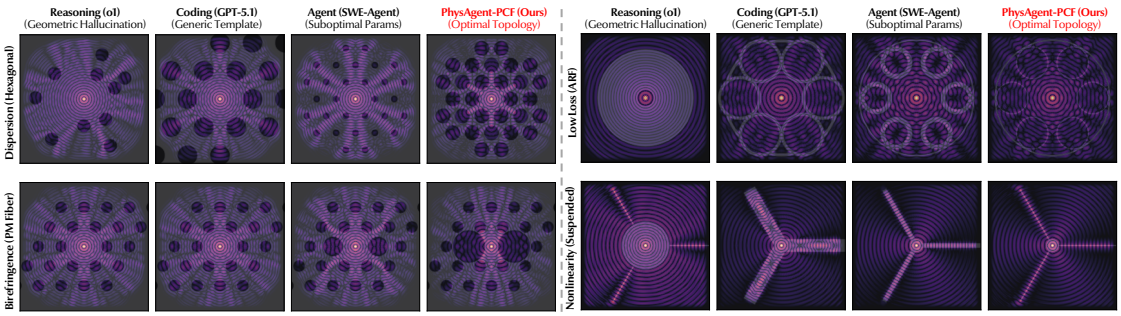


Figure 4: Comparative visualization of generated fiber topologies with transient E_z fields. More case studies on complex topogies are in **Appendix F** and failure modes analysis are provided in **Appendix G**.

to repeated exploration of failed parameters. Replacing the dual-brain hybrid with a unified model further reduces performance. These highlight the complementary roles of visual grounding, memory, and role-specialized reasoning in PhysAgent-PCF.

Configuration	Overall SR (%)	Granular Diagnosis Spatial Acc.	Granular Diagnosis Spectral Acc.	Cost Turns
Full (Hybrid)	**91.2**	**97.8**	**93.4**	**2.4**
w/o Multimodal Critic	77.4 (-13.8)	81.2 (-16.6)	90.1 (-3.3)	4.2
w/o Episodic Memory	83.5 (-7.7)	97.5 (-0.3)	85.6 (-7.8)	4.9
w/o Dual-Brain	88.1 (-3.1)	95.4 (-2.4)	91.2 (-2.2)	3.0
w/o Hypothesis Update	86.8 (-4.4)	96.1 (-1.7)	88.4 (-5.0)	3.5

Table 2: Ablation analysis results. *w/o*: without. These two diagnostic metrics details are in **Appendix D.2**.

5.4 Iterative Optimization Capabilities

Fig. 5 shows the optimization trajectories of different agents measured by Dispersion MSE, where lower values indicate closer alignment with the target dispersion. While standard ReAct agents exhibit oscillatory, random-walk–like behavior, PhysAgent-PCF achieves smooth and monotonic error reduction resembling steepest-descent optimization. As highlighted in the inset, PhysAgent-PCF is the only method that consistently reaches the success threshold (MSE < 5.0) within the 5-turn budget, demonstrating superior stability and sample efficiency in iterative refinement.

5.5 Comparison with Classical Optimizers

Classical derivative-free optimizers (DFO) operate over a *fixed* continuous parameter vector, whereas PhysAgent-PCF optimizes in *executable code space* and can perform topology-changing edits. To isolate optimization efficiency under a fair, fixed-topology setting, we benchmark against Random Search, PSO, GP-BO, CMA-ES, and a surrogate-MLP with gradient descent on dispersion minimization (optimizing d_1, d_2, and Λ of a hexag-

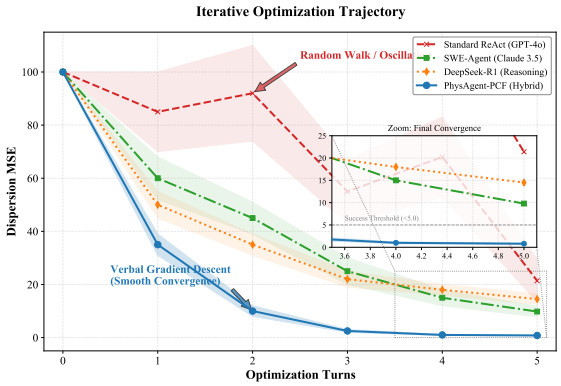


Figure 5: Optimization trajectory across agents.

onal PCF at 1.55 μm).

As shown in **Table 3**, continuous solvers reach slightly lower MSE through dense micro-stepping, but only after *hundreds to thousands* of FDTD evaluations; PhysAgent-PCF attains competitive precision within 1–3% of that budget (5 calls). When the template is physically bounded, continuous solvers stall at local minima, whereas PhysAgent-PCF rewrites the topology to escape them. On a fixed 6-capillary anti-resonant fiber (Ding et al., 2019), GP-BO converges to 0.12 dB/km after 100 FDTD evaluations, while PhysAgent-PCF reaches **0.015 dB/km** in 5 turns by inserting nested features at turn 3. The advantage is thus *sample efficiency and topology discovery*, not single-point precision.

Method	Type	Best MSE (↓)	FDTD Calls (↓)	Topo. Disc.
Random Search	Baseline	3.458	488	No
PSO	Swarm	1.114	370	No
GP-BO	Bayesian	0.921	151	No
CMA-ES	Evolutionary	0.880	799	No
Surrogate-MLP+Grad	Deep Learning	**0.882**	5,000+	No
PhysAgent-PCF (Ours)	LLM Agent	1.027	**5**	**Yes**

Table 3: Fixed-topology dispersion optimization across classical DFO and ML methods. Classical solvers achieve marginally lower MSE but require 30–1000× more solver calls and cannot perform discovery.

5.6 Framework Analysis

Hyperparameter Sensitivity. We analyze the sensitivity of PhysAgent-PCF to key hyperparameters and find the framework to be robust across a wide operating range. As shown in **Fig. 6(a)**, the planner temperature exhibits a clear Goldilocks zone between 0.4 and 0.6, balancing hypothesis diversity and physical validity, with more complex anti-resonant structures benefiting from higher temperatures. **Fig. 6(b)** shows that increasing the memory window size initially accelerates convergence, but saturates around $K = 4$, beyond which token costs increase without performance gains. Finally, **Fig. 6(c)** indicates that a moderate critic confidence threshold ($\delta_{\text{critic}} \approx 0.6$) yields the best performance by filtering noisy visual feedback. These suggest intuitive and stable hyperparameter regimes consistent with the design principles of PhysAgent-PCF.

Robustness to Linguistic Perturbations. We use a Hard Prompt subset with under-specified, conflicting, and noisy instructions. As shown in **Table 4**, PhysAgent-PCF shows substantially smaller performance drops than standard ReAct agents across all noise types. The gain is largest under conflicting prompts with physically impossible constraints: the System-2 planner detects inconsistencies and reconciles them before code generation, whereas ReAct blindly follows the prompt, yielding invalid designs or solver crashes. Thus, explicit reasoning and planning provide stronger semantic robustness under realistic, imperfect user queries.

Noise Type	ReAct (GPT-4o)	PhysAgent (Ours)	Advantage
Clean Prompts	70.4%	**91.2%**	-
Under-specified	35.2% (-50%)	78.4% (-14%)	Planner infers defaults
Conflicting	12.5% (-82%)	65.1% (-28%)	Planner detects error
Noisy / Typos	65.8% (-6%)	89.5% (-2%)	Robust parsing
Average Drop	**-46.0%**	**-14.7%**	**More Robust**

Table 4: Linguistic robustness analysis. Lower drop (Δ) indicates better semantic reasoning capability.

Manufacturing Tolerance Awareness. We evaluate the manufacturing robustness under realistic fabrication uncertainty, where geometric parameters deviate from their nominal values. Specifically, we introduce random geometric perturbations ranging from ±1% to ±5% and re-simulate the resulting designs to measure dispersion degradation. As shown in **Fig. 7**, PhysAgent-PCF designs maintain tightly clustered performance distributions under perturbation, indicating low sensitivity to manufacturing noise, whereas baseline designs exhibit substantially larger variance and rapid degradation. These results show that PhysAgent-PCF optimizes not only for nominal physical targets but also for robustness under practical fabrication tolerances.

Verbal-Gradient Noise Profile. We audit 200 agent steps against ground-truth physical gradients from finite-difference probing of the solver. As shown in **Table 5**, the critic acts as a reliable but noisy sign oracle: the directional sign is correct in 86.5% of steps, flipped in 6.5%, and absent in 7.0%, while 18.0% of correct-sign steps over-estimate the step size and induce near-optimal oscillation. This step-wise noise does not accumulate: episodic-

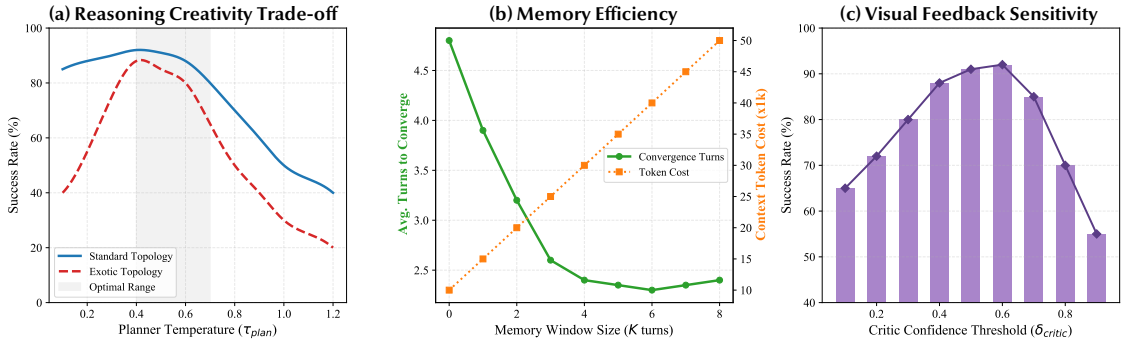


Figure 6: Hyperparameter Sensitivity Analysis. (a) The impact of Planner Temperature τ_{plan} on success rate, (b) The trade-off between Memory Window Size K and efficiency, (c) Sensitivity to VLM Critic Threshold δ_{critic}.

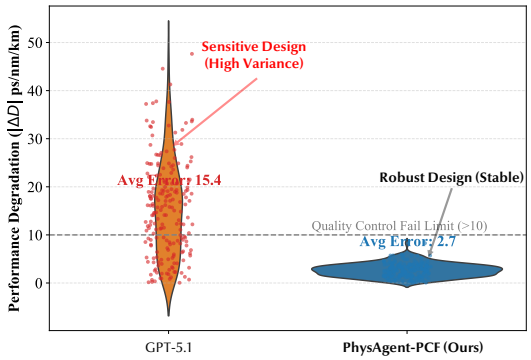


Figure 7: Manufacturing robustness analysis.

memory replay reverts any step that worsens the physical loss and inverts its direction.

Verbal-Gradient Outcome	Frequency
Correct directional sign	**86.5%**
Sign-flip (hallucinated)	6.5%
Vague / no direction	7.0%
(Subset) step-size over-estimation	*18.0%*

Table 5: Empirical noise profile of verbal gradients over 200 audited steps. Sign-flip and over-shoot errors are reversible under episodic-memory replay.

Structured vs. Raw Critic Feedback. We isolate the effect of parsing the critic's diagnosis into a structured verbal gradient versus appending its raw text to the planner prompt (270 cases, 5 runs). Raw text floods the planner and yields an 11.7% parameter-hallucination rate (optimizing variables that do not exist). As shown in **Table 6**, the parsed schema constrains the planner to a single optimization step, cutting hallucinations to 1.5% and raising success from 78.0% to 91.2%.

Computational Cost. We profile the Hybrid configuration over 100 test cases. As summarized in **Table 7**, agent overhead is minor (∼26.5 s and ∼$0.023 per turn); the FDTD solver dominates

Feedback Format	Valid Action Rate (↑)	Param. Halluc. (↓)	SR (↑)
Raw unstructured critic text	78.5%	11.7%	78.0%
Parsed verbal gradient (Ours)	**98.5%**	**1.5%**	**91.2%**

Table 6: Ablation on planner input formatting (averaged over 5 runs, 270 cases).

at ∼245 s/turn, about 10× the agent cost. Since the framework converges in ∼5 simulations rather than the hundreds to thousands required by the classical pipelines in **Table 3**, it reduces 10–50 solver-hours to under 15 minutes, shifting cost from physics simulation to token inference.

Component	Avg. Time (s)	Tokens (In/Out)	API Cost
System-2 Planner	12.5	2,150 / 680	$0.004
System-1 Coder	8.2	1,820 / 450	$0.011
Multimodal Critic	5.8	1,450 / 120	$0.008
Physical Solver	245.0	N/A	Local CPU
Total / Turn	**∼271.5**	**∼5,420 / 1,250**	**∼$0.023**

Table 7: Average computational cost breakdown per interaction turn (Hybrid, 100 cases).

Additional Analyses. We further provide additional analyses including human evaluation of designs (App. E.1), prompt sensitivity (App. E.2), and coarse-to-fine refinement (App. E.3).

6 Conclusion

This paper proposes PhysAgent-PCF, the first multimodal agent framework that enables trajectory-aware, language-guided optimization for photonic crystal fiber inverse design by tightly integrating physical reasoning, executable simulation, and multimodal feedback. Through extensive evaluation on the DeepPCF-1M benchmark, our results demonstrate that multimodal, role-specialized agents provide a robust paradigm for physically grounded inverse design in complex scientific domains.

Limitations

This work targets two-dimensional cross-sectional PCF designs and electromagnetic simulations, without modeling longitudinal variations or multi-physics effects. Adapting the framework to other scientific domains requires domain-specific simulators and validation rules. Language-based updates give reliable direction but coarse step sizes, so the agent is not suited to high-precision floating-point tuning; once the topology is fixed, a lightweight local solver completes last-mile refinement.

Acknowledgments

This work was supported by the Guangdong Basic and Applied Basic Research Foundation (No. 2024A1515510031). This work is supported by the Intelligent Computing Center of Shenzhen University. We used LLMs solely to polish the writing of this paper, such as grammar and phrasing. They were not used for method design, implementation, experimentation, or result analysis.

References

Alexander Argyros and Jarryd Pla. 2007. Hollow-core polymer fibres with a kagome lattice: potential for transmission in the infrared. *Optics express*, 15(12):7713–7719.

Arun Singh Bhadwal, Monika Kumari, and Anil Kumar. 2025. Pcf-vae: posterior collapse free variational autoencoder for de novo drug design. *Scientific Reports*, 15(1):34152.

Mark Chen, Jerry Tworek, Heewoo Jun, Qiming Yuan, Henrique Ponde de Oliveira Pinto, Jared Kaplan, Harri Edwards, Yuri Burda, Nicholas Joseph, Greg Brockman, Alex Ray, Raul Puri, Gretchen Krueger, Michael Petrov, Heidy Khlaaf, Girish Sastry, Pamela Mishkin, Brooke Chan, Scott Gray, and 39 others. 2021. Evaluating large language models trained on code. *Preprint*, arXiv:2107.03374.

Shengchao Chen, Geyao Hu, Sufen Ren, and Ting Shu. 2025. Pcf-llm: Scaling llms for multimodal understanding of structured scientific data in photonic crystal fiber sensors. *Information Fusion*, page 104022.

Shengchao Chen, Ting Shu, and Sufen Ren. 2026. Learning design skills as memory policies for agentic photonic inverse design. *Knowledge-Based Systems*, page 116726.

Shengchao Chen, Xinchen Wang, Sufen Ren, Jianli Yang, Yaqian Zhang, and Guanjun Wang. 2023. Collaborative photonic crystal fiber property optimization: A new paradigm for reverse design. *IEEE Photonics Technology Letters*, 35(19):1035–1038.

Shengchao Chen, Feifan Yao, Sufen Ren, Guanjun Wang, and Mengxing Huang. 2022. Cost-effective improvement of the performance of awg-based fbg wavelength interrogation via a cascaded neural network. *Optics Express*, 30(5):7647–7663.

DeepSeek-AI, Qihao Zhu, Daya Guo, Zhihong Shao, Dejian Yang, Peiyi Wang, Runxin Xu, Y. Wu, Yukun Li, Huazuo Gao, Shirong Ma, Wangding Zeng, Xiao Bi, Zihui Gu, Hanwei Xu, Damai Dai, Kai Dong, Liyue Zhang, Yishi Piao, and 21 others. 2024. Deepseek-coder-v2: Breaking the barrier of closed-source models in code intelligence. *Preprint*, arXiv:2406.11931.

Wei Ding, Ying-Ying Wang, Shou-Fei Gao, Meng-Ling Wang, and Pu Wang. 2019. Recent progress in low-loss hollow-core anti-resonant fibers and their applications. *IEEE Journal of Selected Topics in Quantum Electronics*, 26(4):1–12.

Daya Guo, Dejian Yang, Haowei Zhang, Jun-Mei Song, Ruoyu Zhang, Runxin Xu, Qihao Zhu, Shirong Ma, Peiyi Wang, Xiaoling Bi, Xiaokang Zhang, Xingkai Yu, Yu Wu, Z. F. Wu, Zhibin Gou, Zhihong Shao, Zhuoshu Li, Ziyi Gao, Aixin Liu, and 178 others. 2025. Deepseek-r1 incentivizes reasoning in llms through reinforcement learning. *Nature*, 645:633 – 638.

Xiao-Qi Han, Xin-De Wang, Meng-Yuan Xu, Zhen Feng, Bo-Wen Yao, Peng-Jie Guo, Ze-Feng Gao, and Zhong-Yi Lu. 2025. Ai-driven inverse design of materials: Past, present, and future. *Chinese Physics Letters*, 42(2):027403.

Yun Hao, Che Fan, Beilin Ye, Wenhao Lu, Zhen Lu, Peilin Zhao, Zhifeng Gao, Qingyao Wu, Yanhui Liu, and Tongqi Wen. 2025. Inverse materials design by large language model-assisted generative framework. *arXiv preprint arXiv:2502.18127*.

Jiaqi Jiang, David Sell, Stephan Hoyer, Jason Hickey, Jianji Yang, and Jonathan A Fan. 2019. Free-form diffractive metagrating design based on generative adversarial networks. *ACS nano*, 13(8):8872–8878.

Sameh Kaziz, Fraj Echouchene, and Mohamed Hichem Gazzah. 2024. Optimizing pcf-spr sensor design through taguchi approach, machine learning, and genetic algorithms. *Scientific reports*, 14(1):7837.

Yupeng Liu, Weijia Wang, Yan Wang, Xiaojie Sun, Tonglei Cheng, and Jun Fu. 2026. Fibergnn: A graph neural network framework for efficient photonic crystal fiber parameters prediction. *Expert Systems with Applications*, 309:131223.

Jiarui Lu, Thomas Holleis, Yizhe Zhang, Bernhard Aumayer, Feng Nan, Haoping Bai, Shuang Ma, Shen Ma, Mengyu Li, Guoli Yin, Zirui Wang, and Ruoming Pang. 2025. ToolSandbox: A stateful, conversational, interactive evaluation benchmark for LLM tool use capabilities. In *Findings of the Association for Computational Linguistics: NAACL 2025*, pages 1160–1183, Albuquerque, New Mexico. Association for Computational Linguistics.

Pingchuan Ma, Tsun-Hsuan Wang, Minghao Guo, Zhiqing Sun, Joshua B. Tenenbaum, Daniela Rus, Chuang Gan, and Wojciech Matusik. 2024. LLM and simulation as bilevel optimizers: A new paradigm to advance physical scientific discovery. In *Proceedings of the 41st International Conference on Machine Learning*, volume 235 of *Proceedings of Machine Learning Research*, pages 33940–33962. PMLR.

Taigao Ma, Mustafa Tobah, Haozhu Wang, and L. Jay Guo. 2022. Benchmarking deep learning-based models on nanophotonic inverse design problems. *Opto-Electronic Science*, 1(1):210012–1–210012–15.

Zhiyuan Ma, Zhenya Huang, Jiayu Liu, Minmao Wang, Hongke Zhao, and Xin Li. 2025. Automated creation of reusable and diverse toolsets for enhancing llm reasoning. In *Proceedings of the AAAI Conference on Artificial Intelligence*, volume 39, pages 24821–24830.

Christos Markos, John C Travers, Amir Abdolvand, Benjamin J Eggleton, and Ole Bang. 2017. Hybrid photonic-crystal fiber. *Reviews of Modern Physics*, 89(4):045003.

Ardavan F Oskooi, David Roundy, Mihai Ibanescu, Peter Bermel, John D Joannopoulos, and Steven G Johnson. 2010. Meep: A flexible free-software package for electromagnetic simulations by the fdtd method. *Computer Physics Communications*, 181(3):687–702.

John Peurifoy, Yichen Shen, Li Jing, Yi Yang, Fidel Cano-Renteria, Brendan G DeLacy, John D Joannopoulos, Max Tegmark, and Marin Soljačić. 2018. Nanophotonic particle simulation and inverse design using artificial neural networks. *Science advances*, 4(6):eaar4206.

Can Polat, Mehmet Tuncel, Mustafa Kurban, Erchin Serpedin, and Hasan Kurban. 2025. xchemagents: Agentic ai for explainable quantum chemistry. *arXiv preprint arXiv:2505.20574*.

Pengzhen Ren, Yun Xiao, Xiaojun Chang, Po-Yao Huang, Zhihui Li, Brij B Gupta, Xiaojiang Chen, and Xin Wang. 2021. A survey of deep active learning. *ACM computing surveys (CSUR)*, 54(9):1–40.

Sufen Ren, Shengchao Chen, Jiahao Wang, Haoyang Xu, Xuan Hou, Mengxing Huang, Jianxun Liu, and Guanjun Wang. 2024. A distributed photonic crystal fiber reverse design framework based on multi-source knowledge fusion. *Optical Fiber Technology*, 84:103718.

Gemini Team, Rohan Anil, Sebastian Borgeaud, Jean-Baptiste Alayrac, Jiahui Yu, Radu Soricut, Johan Schalkwyk, Andrew M. Dai, Anja Hauth, Katie Millican, David Silver, Melvin Johnson, Ioannis Antonoglou, Julian Schrittwieser, Amelia Glaese, Jilin Chen, Emily Pitler, Timothy Lillicrap, Angeliki Lazaridou, and 1332 others. 2025. Gemini: A family of highly capable multimodal models. *Preprint*, arXiv:2312.11805.

Kun Wang, Guibin Zhang, Zhenhong Zhou, Jiahao Wu, Miao Yu, Shiqian Zhao, Chenlong Yin, Jinhu Fu, Yibo Yan, Hanjun Luo, Liang Lin, Zhihao Xu, Haolang Lu, Xinye Cao, Xinyun Zhou, Weifei Jin, Fanci Meng, Shicheng Xu, Junyuan Mao, and 84 others. 2025a. A comprehensive survey in llm(-agent) full stack safety: Data, training and deployment. *Preprint*, arXiv:2504.15585.

Xingyao Wang, Boxuan Li, Yufan Song, Frank F Xu, Xiangru Tang, Mingchen Zhuge, Jiayi Pan, Yueqi Song, Bowen Li, Jaskirat Singh, Hoang Tran, Fuqiang Li, Ren Ma, Mingzhang Zheng, Bill Qian, Daniel Shao, Niklas Muennighoff, Yizhe Zhang, Binyuan Hui, and 5 others. 2025b. Openhands: An open platform for ai software developers as generalist agents. In *International Conference on Learning Representations*, volume 2025, pages 65882–65919.

Fangxin Xiao, Wei Huang, Haomiao Yu, and Binbin Song. 2024. Predicting optical properties of different photonic crystal fibers from 2d structural images using convolutional neural network and transfer learning. *Optics Communications*, 558:130363.

An Yang, Anfeng Li, Baosong Yang, Beichen Zhang, Binyuan Hui, Bo Zheng, Bowen Yu, Chang Gao, Chengen Huang, Chenxu Lv, Chujie Zheng, Dayiheng Liu, Fan Zhou, Fei Huang, Feng Hu, Hao Ge, Haoran Wei, Huan Lin, Jialong Tang, and 41 others. 2025. Qwen3 technical report. *Preprint*, arXiv:2505.09388.

Dan Yang, Zhulin Wei, Huobin Qin, Bin Xu, and Tonglei Cheng. 2022. Optimization of structural parameters of pcf polarization filter by a genetic algorithm. *IEEE Photonics Journal*, 14(6):1–9.

John Yang, Carlos Jimenez, Alexander Wettig, Kilian Lieret, Shunyu Yao, Karthik Narasimhan, and Ofir Press. 2024. Swe-agent: Agent-computer interfaces enable automated software engineering. *Advances in Neural Information Processing Systems*, 37:50528–50652.

Shunyu Yao, Jeffrey Zhao, Dian Yu, Nan Du, Izhak Shafran, Karthik Narasimhan, and Yuan Cao. 2023. ReAct: Synergizing reasoning and acting in language models. In *International Conference on Learning Representations (ICLR)*.

JD Zamfirescu-Pereira, Eunice Jun, Michael Terry, Qian Yang, and Bjoern Hartmann. 2025. Beyond code generation: Llm-supported exploration of the program design space. In *Proceedings of the 2025 CHI Conference on Human Factors in Computing Systems*, pages 1–17.

Bofeng Zhu, Kevin Hean, Stephan Wong, Yuxi Wang, Rimi Banerjee, Haoran Xue, Qiang Wang, Alexander Cerjan, Qi Jie Wang, Wonkeun Chang, and Yidong Chong. 2025. Topological photonic crystal fiber. *Science Advances*, 11(46):eady1476.

Appendix

A DeepPCF-1M **12**
- A.1 Human-in-the-Loop Validation . . 12
- A.2 Dataset Statistics 13

B Prompt Definitions **13**

C Verbal Gradient Descent **14**
- C.1 Verbal Feedback as Directional Signals 14
- C.2 Step Size Selection 14
- C.3 Verbal Gradient Update Rule . . . 15

D Evaluation Metrics **15**
- D.1 Metrics in Main Results 15
- D.2 Diagnostic Metrics for Ablation Analysis 15

E Additional Experiments **16**
- E.1 Human Evaluation of Designs . . 16
- E.2 Prompt Sensitivity 16
- E.3 Coarse-to-Fine Refinement 17

F Case Studies on Complex Topologies **17**

G Failure Analysis **18**

H Potential Risks and Broader Impact **19**

A DeepPCF-1M

A.1 Human-in-the-Loop Validation

Constructing a million-scale benchmark requires rigorous quality control to ensure that the (Instruction, Code, Property) triplets are physically consistent and error-free. While the bulk of DeepPCF-1M is generated via our automated physics engine, we employed a strict Human-in-the-Loop (HITL) validation protocol to certify the dataset's integrity.

Dimension	Checklist question	Pass criteria
Executability	Does the Meep script run without syntax errors or runtime crashes?	Code must execute end-to-end and produce an electromagnetic field output.
Semantic Alignment	Does the code implement the geometry described in the text?	If text says "7-core", the geometry must define exactly 7 cores. If text says "Hollow-core", refractive index must be 1.0.
Physical Validity	Are the simulated properties within realistic bounds?	Dispersion $\in [-500, 500]$ ps/nm/km; Loss > 0. Outliers (e.g., Dispersion $> 10^5$) indicate simulation divergence and are rejected.
Visual Integrity	Is the generated structure visually coherent?	No overlapping holes (unless intended), no fragmented pixels. The simulation grid resolution must be sufficient to resolve thin features.

Table 8: Quality assurance rubric for DeepPCF-1M. A sample is discarded if it fails any single criterion.

Setup. Since manual verification of all one million samples is infeasible, we adopt a stratified random sampling protocol for dataset validation. In Phase 1, we manually verify all 500+ seed templates (the parametric logic generators underlying the 2,000-seed expert bank) to ensure the underlying code logic is physically sound. In Phase 2, after full dataset generation, we construct a Golden Set of 5,000 instances (approximately 555 per topological category), which provides a 95% confidence level with a margin of error within $\pm 2\%$. These sampled triplets are reviewed by five domain experts (PhD in photonics) using a strict validation rubric (as shown in **Table 8**), with the primary criterion being alignment whether the natural language description faithfully reflects the physical structure and behavior defined by the code.

Filtering Stage	Discarded Samples	Reason
Raw Generation	1,600,000	(Initial Pool)
Syntax Check	-180,000	Compilation Errors
Geometry Check	-220,000	Overlapping / Collisions
Physics Sanity Check	-200,000	Non-converged or Non-physical Fields
Final	**1,000,000**	**High-Quality Triplets**

Table 9: Statistics of the automated filtering pipeline.

Iterative Refinement Pipeline Dataset construction followed an iterative human-in-the-loop refinement process. We conducted three rounds of expert review. In the pilot round, an initial pass rate of 86% revealed implementation issues in the anti-resonant generator (e.g., invalid tube thickness), which were corrected at the template level. In the scaling round, the pass rate increased to 94%, with feedback indicating that some fractal-related instructions were overly vague; this was addressed by enriching prompt templates with explicit geometric constraints. In the final round, a spot check on the released 1M dataset achieved a pass rate of 99.2%, with the remaining noise (<1%) considered acceptable for large-scale pretraining. Note that this post-refinement pass rate differs from the single-pass human acceptance rate of 94.2% reported in Table 10, which is measured before template-level corrections.

Data Filtering Statistics. Prior to human review, we applied aggressive automated filtering to remove syntactically invalid, physically infeasible, or non-convergent samples. The resulting filtering statistics are in **Table 9**, ensuring a high signal-to-noise ratio in the final DeepPCF-1M benchmark.

Error Category	Description of Failure Mode	Freq. (%)	Impact
Geometric Hallucination	The text describes structural features (e.g., "elliptical holes") absent in the image/code, or vice versa.	42.5%	High
Physical Inconsistency	The described optical property (e.g., "zero dispersion") contradicts the simulation log S.	31.0%	High
Vague / Generic	The instruction is too broad (e.g., "Design a fiber") without specific constraints, failing the diversity check.	15.5%	Low
Spatial Misalignment	Incorrect spatial references (e.g., confusing "core" with "cladding" or "inner ring" with "outer ring").	8.0%	Medium
Other / Unclear	Subjective disagreement or ambiguous phrasing.	3.0%	Low
Total Acceptance	**94.2%**	-	-

Table 10: Human evaluation and error analysis on DeepPCF-1M. The "Impact" column indicates whether the error affects code executability or semantic consistency.

Topology Category	Structural Description	Count	Ratio	Target Application
Solid-Core Hexagonal	Standard triangular lattice with solid silica core.	300,000	30.0%	Supercontinuum / Solitons
Hollow-Core PBG	Honeycomb lattice creating photonic bandgaps.	150,000	15.0%	High-Power Delivery
Anti-Resonant (ARF)	Negative curvature tubes (e.g., Kagome, Nested).	120,000	12.0%	Ultra-Low Loss / Gas Sensing
High-Birefringence (PM)	Asymmetric holes (Elliptical/Panda) for polarization.	100,000	10.0%	Fiber Gyroscopes
Suspended / Cobweb	Small core supported by thin nano-struts.	100,000	10.0%	Mid-IR Sensing
Multi-Core PCF	Multiple isolated cores in a single cladding.	80,000	8.0%	Space Division Multiplexing
Square / Rectangular	Non-hexagonal symmetry lattices.	80,000	8.0%	Mode Shaping
Vortex / Ring-Core	Annular core supporting OAM modes.	50,000	5.0%	Orbital Angular Momentum
Fractal / Quasi-Crystal	Penrose tiling or Sierpinski patterns.	20,000	2.0%	Topological Photonics
Total Dataset	9 Categories of Complex Microstructures	1,000,000	100%	General Purpose Design

Table 11: Statistical breakdown of DeepPCF-1M across nine topological categories.

Final Human Review. After filtering, domain experts manually verified all 5,000 selected samples to ensure that the underlying code logic and physical interpretations were sound. The evaluation was conducted in a blinded manner: model identities and generation sources were anonymized, and samples were randomly shuffled prior to review to minimize potential bias. Each sample was independently assessed by experts with advanced training in photonics, following a strict rubric that focused on physical alignment. Specifically, whether the natural language instruction faithfully reflected the geometry and optical behavior defined by the executable code and simulation outputs. **Table 10** summarizes the human evaluation results .

A.2 Dataset Statistics

Table 11 provides a granular statistical breakdown of the DeepPCF-1M. The dataset is constructed to reflect the real-world complexity landscape of photonic design, exhibiting a natural long-tailed distribution: while standard industrial designs (e.g., Solid-Core Hexagonal, 30%) constitute the majority, the benchmark intentionally preserves a diverse tail of exotic and experimental topologies (e.g., Fractal/Quasi-Crystals, 2%) to evaluate the agent's out-of-distribution generalization capabilities. Each of samples is curated as a multimodal triplet

$$\langle \text{Natural Language Instruction}, \text{Executable Meep Script}, \text{Physical Property} \rangle \quad (4)$$

ensuring that the evaluation covers the full spectrum from linguistic understanding to code generation and physical reasoning.

B Prompt Definitions

This section provides the full system instructions used in the PhysAgent-PCF framework. We utilize a modular prompting strategy where the Planner, Coder, and Critic possess distinct personas.

System Prompt: Physicist Planner

Role: You are a senior Computational Physicist specializing in Photonic Crystal Fibers (PCF).
Task: detailed design plan based on the user's optical requirements. You have access to a retrieval database of successful designs.
Input Context:

- **User Query:** `{{user_query}}`
- **Retrieved Exemplars:** `{{retrieved_examples}}` (3 successful Meep scripts similar to the query)
- **History:** `{{conversation_history}}` (Previous attempts and Critic feedback)

Instructions:

1. **Reasoning:** First, analyze the physical constraints. Determine the optimal topology (e.g., Hexagonal vs. Anti-Resonant) based on the target property (e.g., Dispersion vs. Loss).

2. **Parameter Estimation:** Estimate initial geometric parameters (pitch Λ, hole diameter d, ring count N). Explicitly state your physical intuition (e.g., "Increasing d will flatter the dispersion slope").

3. **Refinement Strategy:** If this is a re-try step, analyze the *Verbal Gradient* provided by the Critic. Explain *why* the previous parameter failed and calculate a specific update (e.g., "Decrease d by 5%").

Output Format: Return a structured "Thought Trace" followed by a specific instruction for the Coder.

System Prompt: Simulation Engineer

Role: You are an expert Meep (Python) developer. You translate physical specifications into error-free simulation scripts.
Input:

- **Physics Plan:** `{{planner_output}}`
- **Syntactic Constraints:** Use `meep`. Ensure PML layers are sufficiently thick. Set resolution ≥ 20.

Instructions:

1. Implement the geometry defined in the Plan exactly.
2. Define the source and monitors correctly for the target wavelength (e.g., 1.55 μm).
3. **Self-Correction:** Before outputting, check for common errors:
   - Ensure geometry does not overlap with PML.
   - Ensure materials are defined (SiO2 vs Air).
   - Ensure the computational cell size covers the evanescent field.

Output: A single, executable Python code block wrapped in "'`python ...` "'.

System Prompt: Multimodal Critic

Role: You are a Quality Assurance Physicist. You verify if the simulation results match the design target.
Input:

- **Target:** `{{user_goal}}`
- **Simulation Log:** `{{sim_log}}` (Numerical, errors)
- **Visual Field:** `{{visual_field}}`
- **Structure:** `{{structure}}` (Meep script)

Instructions:

1. **Visual Inspection:** Look at the heatmap. Is the light confined in the core? Is there leakage into the cladding? Is the mode shape correct (e.g., fundamental vs. multi-mode)?
2. **Numerical Check:** Compare the simulated value (e.g., Dispersion = 5.2) with the target (e.g., 0.0).
3. **Verbal Gradient:** If the design failed, generate a specific, directional feedback.
   - *Bad Example:* "Try again."
   - *Good Example:* "Mode is leaking. The air-filling fraction is too low. Suggest increasing hole diameter d."

Output: JSON format { "success": bool, "feedback": str }.

C Verbal Gradient Descent

We introduced verbal gradients as a high-level abstraction for guiding iterative refinement without numerical derivatives. This section provides an algorithmic formalization that connects language-based feedback to parameter updates, framing our method within the broader family of derivative-free optimization (DFO) techniques.

C.1 Verbal Feedback as Directional Signals

Let $\mathbf{x} \in \mathbb{R}^d$ denote the vector of geometric parameters governing a simulation $f_{\text{sim}}(\mathbf{x})$, and let $\mathcal{L}(\mathbf{x}) = |f_{\text{sim}}(\mathbf{x}) - \mathbf{y}_{\text{target}}|$ be the physical discrepancy to minimize. Since $f$sim is non-differentiable, the true gradient $\nabla\mathcal{L}$ is unavailable. Instead of estimating numerical gradients via finite differences, PhysAgent-PCF relies on semantic feedback. A multimodal critic observes the mismatch between the simulated outcome and the target, expressed through visual and spectral cues, and generates a natural language feedback message w_t that qualitatively describes how the current design deviates from the target. We interpret this feedback as providing a directional signal rather than a numeric gradient. Formally, a lightweight, deterministic semantic parser $\Phi(\cdot)$—a JSON/regex extractor that enforces a fixed schema—maps feedback text to a signed update direction for each relevant parameter:

$$\text{dir}_t = \Phi(w_t) \in \{-1, 0, +1\}, \tag{5}$$

where the sign encodes whether the parameter should be decreased, preserved, or increased. Importantly, Φ is not designed to recover exact gradients, but to extract coarse directional guidance from language, analogous to sign-based gradient methods.

C.2 Step Size Selection

While the critic provides directional information, the magnitude of the update is determined by the System 2 Planner. The planner interprets linguistic intensity modifiers in the feedback, such as slightly or significantly, together with the optimization context and prior trajectory stored in episodic memory.

Algorithm 1 Verbal Gradient Update Step

Input: Current parameters $\mathbf{x}_t$, simulation output y_t, target y^*, base step size η_{base}

Output: Updated parameters $\mathbf{x}_{t+1}$

1: Observe multimodal discrepancy between y_t and y^*.
2: Critic generates natural language feedback w_t.
3: Extract directional signal $\text{dir}_t \leftarrow \Phi(w_t)$.
4: Infer step size multiplier $\eta_t \leftarrow \eta_{\text{base}} \cdot \Psi(w_t)$.
5: Update parameters: $\mathbf{x}_{t+1} \leftarrow \mathbf{x}_t + \eta_t \cdot \text{dir}_t$.

We model this process as an adaptive step size:

$$\eta_t = \eta_{\text{base}} \cdot \Psi(w_t), \tag{6}$$

where $\Psi(\cdot)$ maps qualitative intensity cues to scalar multipliers. This step size selection is heuristic and context-dependent, reflecting deliberate reasoning rather than numerical estimation of $|\nabla \mathcal{L}|$.

C.3 Verbal Gradient Update Rule

Combining the directional signal from the critic and the step size inferred by the planner, PhysAgent-PCF updates parameters as:

$$\mathbf{x}_{t+1} = \mathbf{x}_t + \eta_t \cdot \text{dir}_t. \tag{7}$$

This update resembles a stochastic sign-gradient descent step, with the crucial distinction that the gradient sign is inferred semantically from multimodal feedback rather than computed numerically. As such, verbal gradient descent should be understood as a conceptual analogy to sign-based optimization, not a strict estimator of numerical derivatives.

D Evaluation Metrics

We evaluate all methods using a comprehensive set of execution, geometric, and physical metrics designed to assess both syntactic validity and physical correctness.

D.1 Metrics in Main Results

Execution Rate (Exec%) Execution Rate (Chen et al., 2021) measures the proportion of generated simulation scripts that can be successfully executed without syntax errors or solver divergence. Formally,

$$\text{Exec\%} = \frac{1}{N} \sum_{i=1}^{N} \mathbb{I}(f_{\text{sim}}(c_i) \text{ terminates normally}), \tag{8}$$

where c_i denotes the generated simulation script for sample i, f_{sim} is the electromagnetic solver, and $\mathbb{I}(\cdot)$ is the indicator function.

Physical Consistency Score (PCS) PCS (Jiang et al., 2019) evaluates whether the generated geometry conforms to the target structural topology implied by the prompt. We compute PCS as the intersection-over-union (IoU) between the binary mask of the generated structure G_i and the reference topology mask R_i:

$$\text{PCS}_i = 100 \times \frac{|G_i \cap R_i|}{|G_i \cup R_i|}, \tag{9}$$

where the score ranges from 0 to 100. Higher PCS indicates better geometric consistency and strongly correlates with successful physical simulation.

Success Rate (SR) A generation is considered a success if it satisfies both syntactic and physical criteria (Peurifoy et al., 2018). Specifically, a sample i is counted as successful if:

$$\mathbb{I}_{\text{success}}(i) = \mathbb{I}\Bigg(\text{Exec}_i = 1 \wedge \frac{|p_i - p_{\text{target}}|}{p_{\text{target}}} \leq 0.05\Bigg), \tag{10}$$

where p_i denotes the simulated optical property (e.g., dispersion) and p_{target} is the user-specified target. The Success Rate is then defined as

$$\text{SR} = \frac{1}{N} \sum_{i=1}^{N} \mathbb{I}_{\text{success}}(i). \tag{11}$$

Pass@k Pass@k measures generation stability under stochastic decoding (Chen et al., 2021). For each prompt, we sample k independent generations and report whether at least one satisfies the success criterion:

$$\text{Pass@}k = \frac{1}{N} \sum_{i=1}^{N} \mathbb{I}(\exists j \in \{1, \ldots, k\} : \mathbb{I}_{\text{success}}(i, j) = 1). \tag{12}$$

In this work, we report Pass@1, which directly reflects single-run reliability without retries.

D.2 Diagnostic Metrics for Ablation Analysis

To precisely diagnose different failure modes, we define two complementary metrics used in our ablation study: Spatial Accuracy, which evaluates geometric validity, and Spectral Accuracy, which evaluates physical performance conditioned on valid geometry. Let $\mathcal{D} = (t_i, p_i)_{i=1}^{N}$ denote the evaluation

set, where t_i is an instruction and p_i is the target physical property. Given the generated simulation code c_i, we obtain the geometry $g_i = \text{Render}(c_i)$ and the simulated property $\hat{p}_i = f_{\text{sim}}(c_i)$.

Spatial Accuracy Spatial Accuracy measures whether the generated geometry g_i satisfies the topological and geometric constraints implied by the instruction t_i, independent of spectral performance. It penalizes structural hallucinations (e.g., solid core instead of hollow core) and fabrication-invalid geometries (e.g., overlapping components). We define a geometric validity function $V(g_i, t_i) \in 0, 1$, which returns 1 if and only if (1) the generated structure belongs to the correct topological class specified by t_i, and (2) the geometry is physically feasible, satisfying non-overlap and minimum thickness constraints. Spatial Accuracy is then defined as

$$\text{Acc}_{\text{spatial}} = \frac{1}{N} \sum_{i=1}^{N} \mathbb{1}[V(g_i, t_i) = 1] \,. \quad (13)$$

In practice, $V(\cdot)$ is implemented using a rule-based geometry checker complemented by VLM-based visual verification.

Spectral Accuracy Spectral Accuracy evaluates whether a spatially valid design achieves the target physical specification. We define the relative physical error $\mathcal{E}(i) = \frac{|\hat{p}_i - p_i^*|}{|p_i^*| + \epsilon_{\text{stab}}}$, where ϵ_{stab} is a small constant for numerical stability. A design is considered spectrally accurate if $\mathcal{E}(i) < \tau$, with $\tau = 0.05$ in our experiments. Spectral Accuracy is defined conditionally as

$$\text{Acc}_{\text{spectral}} = \frac{\sum_{i=1}^{N} \mathbb{1}[V(g_i, t_i) = 1 \wedge \mathcal{E}(i) < \tau]}{\sum_{i=1}^{N} \mathbb{1}[V(g_i, t_i) = 1]}. \quad (14)$$

This conditional formulation isolates parameter optimization ability from structural generation ability: models with high Spatial but low Spectral Accuracy can generate correct topologies but fail to fine-tune physical parameters.

E Additional Experiments

This section collects supplementary experiments and analyses that complement the main results.

E.1 Human Evaluation of Designs

To complement automatic metrics, three domain experts (PhDs in photonics) blindly rated 270 designs (30 per topology) for every method in Table 1, on a 1–5 scale for **Topological Compliance** (alignment with the prompt's structural intent) and **Fabricability** (manufacturability via stack-and-draw, penalizing floating or intersecting elements). The two failure patterns separate cleanly (Table 12): coding-oriented models inherit a rigid template bias, while reasoning-oriented models produce unmanufacturable geometries. PhysAgent-PCF scores high on both axes, and the Hybrid variant ranks first (4.78 / 4.65).

Method	Topo. Compliance	Fab.	Overall Pref.
Gemini-3-Flash	2.25	2.50	2.38
Qwen3-235B-Thinking	2.85	2.35	2.60
DeepSeek-R1	2.92	2.42	2.68
OpenAI o1	3.12	2.45	2.80
DeepSeek-Coder-V2	2.55	3.25	2.85
Gemini 3 Pro Preview	2.82	3.32	3.05
Qwen3-Coder-480B	3.05	3.55	3.25
Claude Sonnet 4.5	3.25	3.65	3.38
GPT-5.1-Codex-Max	3.65	3.78	3.42
ReAct (GPT-4o)	3.75	3.25	3.52
ReAct (Claude 3.5)	3.58	3.35	3.45
OpenHands (GPT-4o)	3.85	3.45	3.65
OpenHands (Claude 3.5)	3.92	3.55	3.72
SWE-Agent (GPT-4o)	4.05	3.52	3.85
SWE-Agent (Claude 3.5)	4.12	3.65	3.92
PhysAgent-PCF (GPT-4o)	4.52	4.35	4.45
PhysAgent-PCF (Claude 3.5)	4.65	4.42	4.58
PhysAgent-PCF (Hybrid)	**4.78**	**4.65**	**4.75**

Table 12: Blinded human evaluation of generated PCF designs (1–5 scale, higher is better; 3 experts, 270 cases).

E.2 Prompt Sensitivity

Prompt Sensitivity. To separate architectural gains from prompt tuning, we test three prompting styles for the Planner/Coder/Critic: *Minimal* (bare task instructions), *Standard* (our default structured roles), and *Verbose* (heavy domain context). Minimal prompting already reaches 88.4% SR, 2.8 points below Standard yet 13.9 points above the strongest text-only baseline (SWE-Agent, 74.5%; Table 13). The gains therefore trace to the dual-brain and multimodal-critic architecture, with the simulation-grounded critic correcting prompt-induced errors within the loop.

Method / Prompt	SR	Gap to Standard
SWE-Agent (GPT-4o)	74.5%	−16.7
Ours: Minimal	88.4%	−2.8
Ours: Standard	**91.2%**	–
Ours: Verbose	90.5%	−0.7

Table 13: Prompt-sensitivity analysis (success rate).

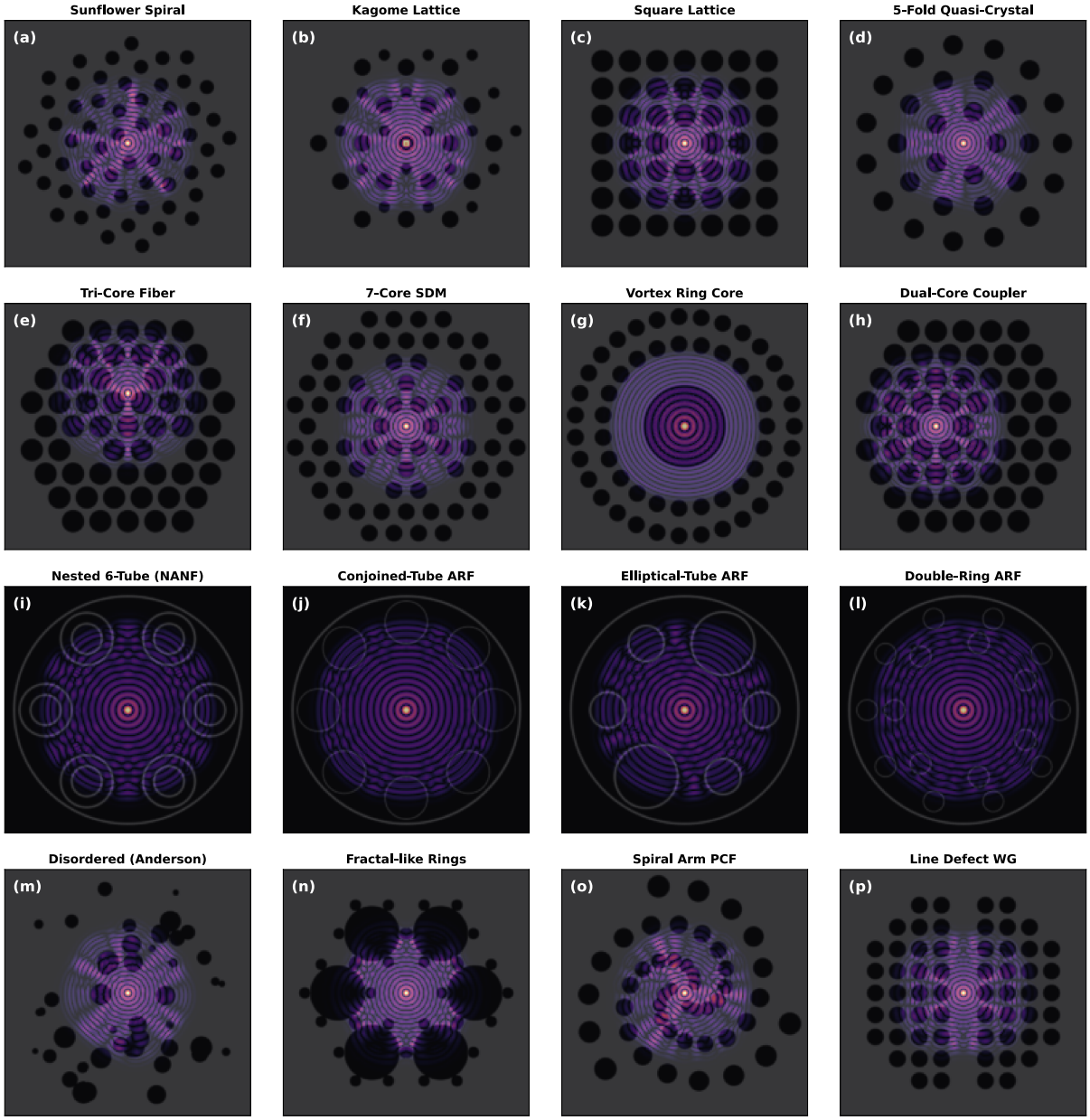


Figure 8: More case studies on complex topologies.

E.3 Coarse-to-Fine Refinement

The oscillation and last-mile limitations from our failure analysis (Appendix G) mark a deliberate design boundary: PhysAgent-PCF targets topological-basin discovery in code space, not high-precision floating-point tuning. Once the topology is fixed, a local solver finalizes the continuous parameters. On a zero-dispersion target (**Table 14**), the agent locates the correct topology (MSE 1.027 in 5 calls), and a Nelder–Mead solver warm-started from its blueprint drives MSE to 0.049 in 15 further simulations. Oscillation is therefore a near-convergence artifact, not a structural blocker.

Optimization	Best MSE (↓)	Sim. Calls	Status
PhysAgent-PCF (orig.)	1.027	5	Discovers topology
+ Nelder–Mead	**0.049**	5+15	Warm-started refine

Table 14: Coarse-to-fine handoff on a zero-dispersion target: the agent locates the topological basin, and a local solver finalizes the continuous parameters.

F Case Studies on Complex Topologies

To demonstrate the diversity and difficulty of the DeepPCF-1M benchmark, **Fig. 8** presents a gallery of 16 representative complex structures generated by PhysAgent-PCF in a zero-shot setting. These designs extend well beyond conventional hexagonal lattices and span multiple advanced topolog-

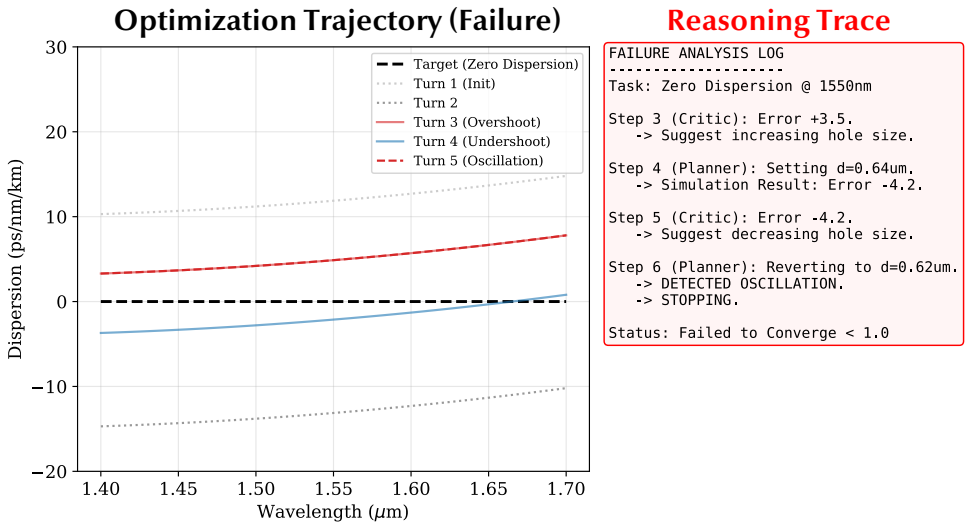


Figure 9: Failure analysis on a precision dispersion task.

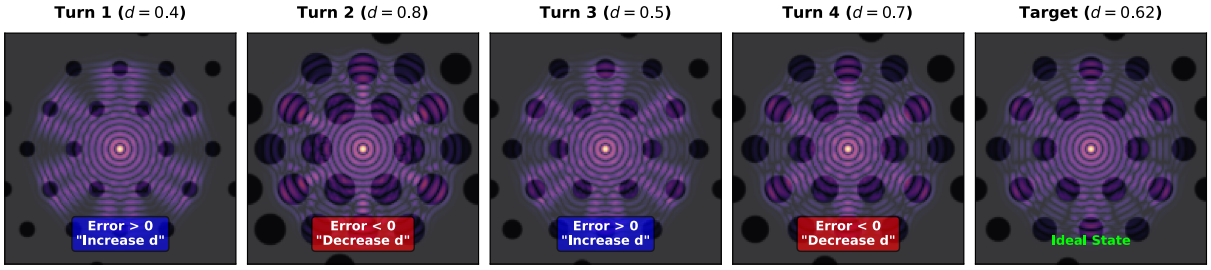


Figure 10: Failure modes: parameter oscillation near convergence, where over-correction causes the mode field to alternately expand and contract, preventing the agent from reaching the target state within the interaction budget.

ical regimes, including mathematical and aperiodic lattices (e.g., sunflower spirals, Kagome lattices, and quasicrystals), next-generation communication fibers for space-division multiplexing and orbital angular momentum transmission (e.g., multicore and ring-core fibers), state-of-the-art antiresonant hollow-core designs with nested or conjoined tubes, and highly irregular structures such as disordered lattices for Anderson localization and fractal geometries. The successful generation of these structurally complex and nontrivial topologies highlights both the structural completeness of the DeepPCF-1M benchmark and the strong zero-shot generalization capability of the PhysAgent-PCF framework across a wide spectrum of photonic design challenges.

G Failure Analysis

Precision Dispersion Task. **Fig. 9** presents a representative failure case in which PhysAgent-PCF does not converge within the interaction budget on an ultra-precision dispersion task. The agent successfully narrows the search space and approaches the target region, but ultimately oscillates between two nearby suboptimal parameters (Turns 3 and 4), with Turn 5 repeating a previous state rather than locating the precise root. The reasoning trace reveals that the verbal gradient correctly identifies the direction of correction, yet the System-1 coder lacks the numerical resolution to apply the infinitesimal update (e.g., $d \approx 0.635$) required to reach exact zero dispersion. This case highlights an inherent limitation of discrete, language-based updates for continuous ultra-fine parameter tuning, despite effective high-level reasoning and error diagnosis.

Failure modes. While PhysAgent-PCF achieves strong overall performance, it occasionally fails near convergence, with the dominant failure mode (65% of failed cases) being parameter oscillation. **Fig. 10** illustrates a representative trajectory in which the agent optimizes a hexagonal PCF for zero dispersion: starting from an under-confined design, the multimodal critic correctly identifies the need to increase the hole diameter, but the subsequent update overshoots, leading to an over-confined structure. Lacking precise numerical cali-

bration, the agent then alternates between parameters that are slightly too large or too small, exhausting the interaction budget without settling on the optimal intermediate value. This behavior reflects a broader limitation of current LLM-based agents when addressing continuous optimization problems: while the planner reliably infers the direction of improvement, it struggles to estimate the appropriate step size in a zero-shot setting.

H Potential Risks and Broader Impact

This work is scoped to the physical inverse design of photonic crystal fibers. The artifacts it produces are executable simulation scripts and geometric parameters verified by an electromagnetic solver, rather than natural-language content intended for human consumption. We therefore see no realistic pathway from this system to the harms typically associated with generative language technology: the pipeline consumes and emits physical design specifications, involves no personal or human-subject data, and confers no capability beyond conventional photonics engineering.

DeepPCF-1M is generated programmatically from parametric geometry generators and FDTD simulation, so it cannot contain personally identifying information or offensive content. The residual risks are technical rather than societal. The agent may fail to converge within its interaction budget, or may propose geometries that are fabrication-invalid; both behaviors are characterized in the Limitations section and in Appendix G, and both are screened by the geometric feasibility and solver-convergence checks described in Section 3. We nonetheless stress that any design produced by this system must pass full physical simulation and fabrication-tolerance analysis before it is manufactured.

Finally, PhysAgent-PCF reaches its design targets in roughly five solver calls where classical derivative-free pipelines require hundreds to thousands (**Table 3**), which lowers rather than raises the computational and energy cost of PCF design.