# AlphaEvolve: A Paradigm Shift in Algorithmic Discovery and Its Profound Implications for Computer Science and Mathematics

**I. Executive Summary**
Google DeepMind's AlphaEvolve marks a pivotal advancement in the realm of AI-driven scientific discovery, leveraging sophisticated Gemini models within an evolutionary framework to autonomously generate, test, and refine complex algorithms. A landmark achievement of this system is the discovery of an algorithm capable of multiplying two 4x4 complex-valued matrices using only 48 scalar multiplications. This breakthrough surpasses the 56-year-old benchmark set by the Strassen algorithm, which, when recursively applied to 4x4 matrices, requires 49 multiplications. Crucially, AlphaEvolve's solution maintains recursive applicability over general fields and corresponds to a novel rank-48 decomposition of the matrix multiplication tensor, thereby addressing a long-standing theoretical challenge in algebraic complexity theory. Beyond this significant mathematical contribution, AlphaEvolve has already yielded tangible efficiency improvements across Google's extensive computing infrastructure, including enhancements in data center scheduling, hardware design, and the training processes for AI models. These practical applications underscore a new era where AI functions as an autonomous co-creator in both fundamental scientific inquiry and applied engineering domains.

**II. Introduction: The Evolving Landscape of AI-Driven Scientific Discovery**
Google DeepMind's recent innovations signal a strategic shift towards harnessing artificial intelligence not merely for conventional applications, but for driving fundamental breakthroughs in science and engineering. This strategic direction is exemplified by a suite of initiatives, including AlphaFold for advancements in biology, GNoME for discoveries in materials science, and now AlphaEvolve, which is pushing the frontiers in mathematics and computer science.
AlphaEvolve is introduced as a groundbreaking evolutionary coding agent, fundamentally redefining the paradigm of automated problem-solving. It transcends the capabilities of a mere code assistant, operating as an agentic system that can autonomously discover and refine algorithms with a defined purpose. This report will meticulously explore AlphaEvolve's innovative methodology, detail its specific breakthrough in 4x4 matrix multiplication, examine its broader real-world impacts within Google's operational infrastructure, and discuss the profound implications for the future of human-AI collaboration in scientific and technological research.

**III. AlphaEvolve's Core Methodology: An Evolutionary AI Agent**
AlphaEvolve operates as an agentic system, demonstrating purposeful autonomy rather than merely responding passively to prompts. It integrates advanced problem-solving capabilities with automated evaluators to ensure verification of its outputs.
The system leverages an ensemble of state-of-the-art Gemini Large Language Models (LLMs). Gemini Flash is employed to maximize the breadth of ideas explored, facilitating the rapid generation of diverse code variations and enabling fast idea generation. In contrast, Gemini Pro provides critical depth through insightful suggestions, focusing on the meticulous refinement and improvement of the quality of the generated code.

The core of AlphaEvolve's operation is its evolutionary loop. This system maintains a dynamic database of candidate programs, which serves as a fertile ground for an evolutionary algorithm. This algorithm intelligently selects which programs will be utilized as context and inspiration for subsequent prompts directed to the Gemini models. This iterative process establishes a continuous evolutionary cycle encompassing the generation of novel code variations, their rigorous automated evaluation, and subsequent refinement. This allows AlphaEvolve to systematically explore and optimize solutions by iterating through millions of code variations. The process involves making direct, often highly non-trivial, modifications to the codebase. For example, the discovery of the 4x4 matrix multiplication algorithm necessitated 15 distinct mutations during its evolutionary journey, illustrating the profound depth of its exploratory capabilities.

A cornerstone of AlphaEvolve's reliability is its dependence on automated evaluators. These evaluators are responsible for verifying, executing, and scoring the proposed programs using predefined, objective evaluation metrics. These metrics provide a quantifiable assessment of each solution's accuracy and quality. This rigorous scoring mechanism is indispensable for ensuring the verifiable correctness of the discovered algorithms and for actively mitigating common LLM-related errors or imprecise solutions. The profound success of AlphaEvolve fundamentally relies on the ability to define precise, reliable, and automatable evaluation functions. This suggests that the discipline of "evaluator engineering" is rapidly emerging as a new critical bottleneck and a strategic advantage for organizations seeking to harness such advanced AI agents. This paradigm shift redefines the human role in the discovery process, moving from direct problem-solving to meticulously defining objective success criteria and quantifiable performance metrics for AI-driven exploration. If AI can autonomously generate solutions, the primary human challenge shifts to precisely and objectively measuring and validating those solutions. This implies a significant reorientation of research and development efforts, where expertise in crafting robust, automatable evaluation frameworks becomes paramount for effectively leveraging AI's generative power.

## IV. The Strassen Algorithm: A Cornerstone of Computational Efficiency

Volker Strassen, a distinguished German mathematician born in 1936, is widely acknowledged as the foundational figure of algebraic complexity theory, a field dedicated to studying the computational complexity of algebraic problems. In 1969, Strassen published his seminal algorithm for matrix multiplication, a discovery that was considered a "real surprise in algorithm analysis" at the time.

His work was transformative because it definitively demonstrated that the standard matrix multiplication algorithm, which had been in use for centuries, was not optimal. This revelation fundamentally challenged long-held assumptions in computer science and mathematics, thereby initiating an intensive, global search for asymptotically superior matrix multiplication algorithms. Strassen's algorithm was not merely an incremental improvement; it fundamentally altered the approach to algorithmic optimization by proving that deeply ingrained, seemingly optimal algorithms could indeed be surpassed. This historical precedent is crucial for understanding the significance of AlphaEvolve's current breakthrough, as it reinforces the concept that even long-established "best" algorithms are dynamic targets, continuously open to re-evaluation and improvement through novel approaches. It showed that fundamental operations could be re-thought and improved upon, a conceptual revolution that AlphaEvolve directly continues.

The core innovation of Strassen's algorithm lies in its ability to multiply two 2x2 matrices using only 7 scalar multiplications, a reduction from the traditional 8 multiplications required by the naive method. This reduction is achieved through a clever rearrangement of terms, which, while increasing the number of additions and subtractions of sub-blocks, are computationally less expensive than multiplications, especially for large matrices. The algorithm is inherently recursive: larger matrices, ideally with dimensions as powers of two, are partitioned into smaller submatrices. The Strassen method is then applied repeatedly to these submatrices until they degenerate into individual scalars. This recursive application leads to a significantly better asymptotic complexity of $O(N^{\log_2(7)})$, which approximates $O(N^{2.808})$, making it asymptotically faster than the standard $O(N^3)$ algorithm for large matrices. In practical implementations, the algorithm typically transitions to standard matrix multiplication methods for sufficiently small submatrices. This is done to mitigate the overhead introduced by the increased number of additions and subtractions, as well as the recursive function call overhead, which can outweigh the benefits for very small matrix sizes.

When Strassen's algorithm is recursively applied to multiply two 4x4 matrices, it conceptually treats each 4x4 matrix as a 2x2 matrix whose elements are themselves 2x2 block matrices. Applying Strassen's 7-multiplication scheme at this top level results in 7 multiplications of these 2x2 block matrices. Each of these 2x2 block multiplications then requires another 7 scalar multiplications, by recursively applying Strassen's algorithm again to the 2x2 blocks. Therefore, the total number of scalar multiplications required becomes $7 * 7 = 49$. This recursive applicability is a crucial property of Strassen's algorithm and requires that the underlying operations (i.e., the multiplication of matrix blocks) do not rely on commutativity, as matrix multiplication is generally non-commutative (meaning $Y_2Y_1 \neq Y_1Y_2$ for general matrices).

**V. AlphaEvolve's Breakthrough: Redefining 4x4 Matrix Multiplication**

AlphaEvolve has achieved a significant milestone by being the first method to discover an algorithm that multiplies two 4x4 complex-valued matrices using only 48 scalar multiplications. This achievement directly improves upon Strassen's 1969 algorithm, which, when recursively applied to 4x4 matrices, required 49 scalar multiplications. This reduction of a single multiplication represents the surpassing of a 56-year-old human research and development record in a foundational mathematical problem, highlighting AI's burgeoning capacity for genuine, novel breakthroughs.

The true significance of AlphaEvolve's 48-multiplication algorithm lies not merely in the numerical reduction from 49 to 48, but in its critical property of being recursively applicable to larger matrices over *any field with characteristic 0*. This directly addresses a fundamental limitation of previous 48-multiplication schemes, such as the Winograd scheme, which were restricted to commutative rings and thus unsuitable for recursive application to general matrices or for achieving asymptotic improvements for larger matrix sizes. An author of the AlphaEvolve paper explicitly clarified that the Winograd scheme "only works over commutative rings, which means that it's not applicable recursively to larger matrices (and doesn't correspond to a rank 48 factorization of the <4,4,4> matrix multiplication tensor)". In stark contrast, AlphaEvolve's algorithm "works over any field with characteristic 0" and *is* recursively applicable. This is the precise, crucial technical detail that elevates AlphaEvolve's achievement from a minor numerical improvement to a genuine, significant breakthrough for asymptotic complexity, as it overcomes a long-standing theoretical and practical limitation.

The problem of finding efficient matrix multiplication algorithms is fundamentally equivalent to finding low-rank decompositions of matrix multiplication tensors. This algebraic framework provides a formal way to represent and analyze the complexity of matrix multiplication. A tensor decomposition for matrix multiplication involves finding a list of matrix triplets (A, B, C) that define the linear combinations and products of sub-blocks. The number of such triplets, R, is known as the rank of the tensor. The objective of fast matrix multiplication is to find the smallest possible rank. Strassen's algorithm for 2x2 matrices corresponds to a rank-7 decomposition of the <2,2,2> matrix multiplication tensor, meaning it requires 7 scalar multiplications. AlphaEvolve's discovery of a 48-multiplication algorithm for 4x4 matrices corresponds to a novel rank-48 decomposition of the <4,4,4> matrix multiplication tensor. This represents a significant theoretical advancement in algebraic complexity theory, pushing the known bounds for this specific tensor. It is important to note that the focus on "scalar multiplication" in this context refers to operations within the field over which the matrices are defined (e.g., complex scalar multiplication for complex-valued matrices). These are the fundamental operations that are recursed upon and directly determine the asymptotic complexity of the algorithm. AlphaEvolve's breakthrough is not merely an empirical engineering feat but a direct, provable contribution to fundamental algebraic complexity theory. By finding a novel, lower-rank tensor decomposition for a specific matrix size (<4,4,4> tensor), it demonstrates AI's capacity to advance abstract mathematical theory, not just optimize practical code. This highlights the potential for AI to contribute to foundational knowledge in mathematics, mirroring Strassen's own contributions to tensor rank theory. The AI is not just finding a numerical trick but is discovering a new, provably correct algebraic structure (a lower-rank tensor decomposition), which is a core problem in abstract mathematics. This elevates the achievement beyond a mere engineering optimization to a significant scientific discovery in its own right.

**Table 1: Comparative Analysis of 4x4 Matrix Multiplication Algorithms**

| Algorithm | Multiplications for 4x4 | Recursive Applicability | Field/Ring Compatibility | Correspondence to Tensor Decomposition | Key Limitation/Advantage |
|---|---|---|---|---|---|
| **Strassen's Algorithm (recursively applied)** | 49 | Yes | Any field (characteristic 0) | Rank-49 (derived from 7x7 recursive application) | Established benchmark for recursively applicable algorithms, but not the theoretical minimum for 4x4. |
| **Winograd Scheme (prior 48-multiplications)** | 48 | No | Commutative rings only | Does not correspond to a rank-48 decomposition of the <4,4,4> matrix multiplication tensor | Not recursively applicable to general matrices due to the non-commutative nature of block matrix multiplication; relies on specific algebraic properties not present in all fields. |
| **AlphaEvolve's Algorithm** | 48 | Yes | Any field with characteristic 0 | Novel Rank-48 decomposition | First recursively applicable |

| Algorithm | Multiplications for 4x4 | Recursive Applicability | Field/Ring Compatibility | Correspondence to Tensor Decomposition | Key Limitation/Advantage |
| --- | --- | --- | --- | --- | --- |
| | | | | of <4,4,4> tensor | algorithm with 48 multiplications for 4x4 matrices over general fields, pushing theoretical bounds. |

**VI. Broader Impact and Real-World Applications of AlphaEvolve**

AlphaEvolve's influence extends significantly beyond its mathematical breakthroughs, demonstrating profound practical utility across Google's vast computing ecosystem.

In the domain of **data center scheduling**, AlphaEvolve discovered a "simple yet remarkably effective heuristic" that has been actively deployed in Google's Borg orchestrator for over a year. This solution continuously recovers, on average, 0.7% of Google's worldwide compute resources, an efficiency gain equivalent to optimizing the utilization of approximately 14,000 servers. This sustained efficiency allows for more tasks to be completed on the same computational footprint and offers substantial operational advantages due to its human-readable code, which enhances interpretability, debuggability, predictability, and ease of deployment.

For **hardware design**, AlphaEvolve proposed a Verilog rewrite that successfully eliminated unnecessary bits in a crucial, highly optimized arithmetic circuit specifically tailored for matrix multiplication. This proposal underwent rigorous verification to confirm its functional correctness and was subsequently integrated into an upcoming Tensor Processing Unit (TPU), Google's custom AI accelerator. This exemplifies AlphaEvolve's capacity to foster a collaborative approach between AI and human hardware engineers, thereby accelerating the design of future specialized chips.

AlphaEvolve is also significantly enhancing **AI training and inference** performance and research velocity by identifying more efficient ways to optimize critical computational kernels. It accelerated a vital matrix multiplication kernel within Gemini's architecture by 23%, contributing to a 1% reduction in Gemini's overall training time. Furthermore, AlphaEvolve achieved an impressive speedup of up to 32.5% for the FlashAttention kernel implementation in Transformer-based AI models. This capability dramatically reduces the engineering time required for kernel optimization, transforming a process that previously took weeks of expert effort into mere days of automated experimentation. AlphaEvolve's ability to optimize the very AI models (Gemini) and specialized hardware (TPUs) that power it creates a powerful and potentially exponential recursive self-improvement loop. This suggests an accelerating pace of AI advancement, where AI itself becomes a primary driver of its own evolution, leading to compounding efficiency gains and faster development cycles for future AI systems. This fulfills a long-discussed concept in futurology of AIs improving themselves; the AI improves the underlying infrastructure and models that enable AI, which in turn makes the next iteration of AI development faster and more resource-efficient, implying a non-linear acceleration in AI capabilities.

Beyond its breakthrough in matrix multiplication, AlphaEvolve has been applied to a large number (over 50) of **open mathematical problems** across diverse domains, including

mathematical analysis, geometry, combinatorics, and number theory. The system successfully rediscovered state-of-the-art solutions for approximately 75% of these problems and, more remarkably, found new, provably better solutions for 20% (specifically, 13 problems) that were previously unknown. These achievements include significant progress in problems such as the Erdős Minimum Overlap, improving the known bounds for the Kissing Number in 11 dimensions (finding a solution with 593 spheres compared to the previous best of 592), and refining solutions for uncertainty inequalities. This demonstrates AlphaEvolve's profound capacity to propose novel approaches to complex mathematical problems and actively advance the frontiers of algorithm discovery, pushing beyond the limits of human-derived solutions.
AlphaEvolve's success across diverse mathematical problems suggests a transformative shift in the role of human experts. In this new paradigm, human scientists may increasingly act as "problem framers," focusing on defining precise evaluation criteria and problem spaces, while AI autonomously handles the iterative exploration and solution generation. Furthermore, AlphaEvolve's general-purpose nature and its ability to "bypass human cognition" could enable it to identify unexpected connections or novel solutions across traditionally siloed scientific disciplines that human experts might overlook, thereby accelerating cross-disciplinary research and discovery. This capability, combined with the "problem framer" role, suggests that AI could become a powerful catalyst for interdisciplinary breakthroughs by finding unexpected solutions or connections across different fields, as it is not constrained by human disciplinary silos or cognitive biases.
AlphaEvolve's **cross-domain versatility** is a testament to its general nature, implying its applicability to any problem whose solution can be described as an algorithm and can be automatically verified. Anticipated use cases span a wide array of scientific and engineering disciplines, including materials science, drug discovery, sustainability, and quantum computing, where it could significantly accelerate scientific discovery and optimize resource utilization. Its demonstrated capacity to discover "fundamentally new algorithmic pathways" and "pioneer new computational pathways" positions AlphaEvolve as a versatile and transformative engine for computational breakthroughs across a wide array of diverse fields, promising efficiency gains and novel solutions previously unattainable.

**Table 2: AlphaEvolve's Diverse Impact and Achievements**

| Area of Impact | Previous State/Benchmark | AlphaEvolve's Achievement | Significance/Impact |
|---|---|---|---|
| **4x4 Complex Matrix Multiplication** | Strassen's Algorithm (49 scalar multiplications) | Reduced to 48 scalar multiplications | Surpassed a 56-year-old human R&D record in a foundational mathematical problem, demonstrating AI's capacity for novel breakthroughs. |
| **Google Borg Data Center Efficiency** | Internal Google metric (previous heuristic, sometimes deep reinforcement learning solutions) | Discovered a new heuristic recovering 0.7% of Google's worldwide compute resources (equivalent to ~14,000 servers) | Significant real-world energy savings and optimized resource allocation worldwide, with human-readable code for operational advantages. |

| Area of Impact | Previous State/Benchmark | AlphaEvolve's Achievement | Significance/Impact |
|---|---|---|---|
| **Gemini Model Training (Kernel Optimization)** | Standard matrix operations performance | Achieved a 23% speedup in key operations, leading to a 1% reduction in Gemini's overall training time | Demonstrates recursive self-improvement and enables faster AI development cycles. |
| **FlashAttention Kernel Optimization** | Existing compiler optimizations | Delivered up to a 32.5% performance boost for the FlashAttention kernel implementation in Transformer-based AI models | Accelerates AI inference and training, significantly reducing engineering time for kernel optimization from weeks to days. |
| **Open Mathematical Problems (e.g., Kissing Number)** | State-of-the-art solutions for 50+ problems (Kissing Number in 11D: 592 spheres) | Rediscovered SOTA for ~75% and improved for ~20% (Kissing Number in 11D: 593 spheres) | Advances fundamental mathematical frontiers, demonstrating AI's capacity for novel mathematical conjecture and proof refinement. |
| **Hardware Design (TPU Circuit Simplification)** | Highly optimized arithmetic circuits | Proposed a Verilog rewrite that removed unnecessary bits in a key circuit | Integrated into an upcoming Tensor Processing Unit (TPU), promoting human-AI collaboration in accelerating chip design. |

**VII. Challenges, Community Reception, and Future Outlook**

While AlphaEvolve is lauded for discovering "provably correct algorithms" and producing "verifiable code outputs" , notable concerns persist within the scientific community regarding DeepMind's historical practices of not fully open-sourcing its code. Prominent critics, such as Oxford AI researcher Simon Frieder, have publicly highlighted past instances with other DeepMind projects, including AlphaFold2 being released without training scripts and AlphaGeometry reportedly containing bugs. These observations raise legitimate questions about trust, independent verification, and the reproducibility of results. This issue is further contextualized by broader controversies in AI-driven scientific discovery, such as MIT's recent request to arXiv to remove a preprint paper due to concerns about the "validity of this research" without disclosing specific flaws. Such incidents underscore the ongoing challenges of trust, transparency, and the integrity of peer review in the rapidly evolving field of AI-generated scientific output. The tension between proprietary development and the fundamental tenets of open scientific inquiry poses a significant challenge for the broader adoption and trust in AI-driven scientific discoveries. While AlphaEvolve delivers impressive and provable results, the lack of full open-sourcing can impede independent validation, hinder reproducibility, and foster skepticism within the global scientific community, potentially slowing down the collective advancement of scientific knowledge. This highlights a critical need for greater transparency and collaboration in the development and dissemination of powerful AI research. For AI-driven scientific discovery to be fully embraced and trusted by the global scientific community, it must adhere to principles of transparency and reproducibility. Without open access to code and methodologies, even "provably correct" results may face skepticism, hindering the collaborative nature of scientific progress.

AlphaEvolve's approach also prompts a profound re-evaluation of what constitutes "discovery" in mathematics. While it excels at finding optimal solutions through exhaustive iterative search, the potential absence of human-like intuition, theoretical elegance, or generalizable proof structures in its process raises fundamental questions about the future role of human mathematicians and the very evolution of mathematical knowledge. While AlphaEvolve achieves demonstrable numerical improvements in open mathematical problems (e.g., finding 593 spheres for the Kissing Number in 11 dimensions compared to the previous best of 592), some mathematicians may perceive the AI's evolutionary search method as less "mathematically interesting" or insightful. This perception often contrasts with human-derived theoretical analyses that yield elegant proofs, reveal underlying mathematical structures, or provide deeper conceptual understanding, which are highly valued in pure mathematics. This highlights a philosophical debate about the nature of mathematical discovery and the value placed on the "how" of a solution (the process of discovery and its inherent insights) versus just the "what" (the numerical result or optimal configuration).

**VIII. Conclusion**

AlphaEvolve represents a significant inflection point in the trajectory of AI, demonstrating its capacity to transcend mere assistance and become an autonomous agent for scientific and algorithmic discovery. Its breakthrough in matrix multiplication, specifically the discovery of a recursively applicable 48-multiplication algorithm for 4x4 complex-valued matrices, is not only a numerical improvement over the 56-year-old Strassen algorithm but a profound contribution to algebraic complexity theory through a novel tensor decomposition. This achievement, coupled with its tangible impact on Google's operational efficiency and its success in solving numerous open mathematical problems, underscores a future where AI actively drives fundamental scientific progress.

The system's reliance on robust automated evaluators highlights a new imperative for human experts: the meticulous design of quantifiable success metrics, shifting the focus from direct problem-solving to precise problem framing. Furthermore, AlphaEvolve's ability to optimize its own underlying AI models and hardware suggests a powerful recursive self-improvement loop, potentially accelerating the pace of AI advancement itself. However, for AI-driven scientific discovery to fully realize its global potential, challenges related to open-sourcing, independent verification, and the philosophical integration of AI-generated results into established scientific paradigms must be addressed transparently. As AI continues to evolve, the collaborative dynamic between human ingenuity and artificial intelligence will increasingly define the frontiers of scientific and technological innovation.

## Works cited

1. AlphaEvolve: A Gemini-powered coding agent for designing advanced algorithms, https://deepmind.google/discover/blog/alphaevolve-a-gemini-powered-coding-agent-for-designing-advanced-algorithms/ 2. From the paper, "Notably, for multiplying two 4 × 4 matrices, applying ..., https://news.ycombinator.com/item?id=43986909 3. AlphaEvolve: AI Outsmarts 56-Year Math Record? - Latenode, https://latenode.com/blog/ai-smashes-math-record 4. Powering scientific discovery with Google Cloud,

https://cloud.google.com/blog/topics/hpc/powering-scientific-discovery-with-google-cloud 5. Science - Google DeepMind, https://deepmind.google/science/ 6. DeepMind's AlphaEvolve Redefines Algorithm Discovery - Turing, https://www.turing.com/blog/exploring-alphaevolve 7. AlphaEvolve: DeepMind's Evolutionary Leap in Scientific Algorithmic Discovery, https://hackernoon.com/alphaevolve-deepminds-evolutionary-leap-in-scientific-algorithmic-discovery 8. Google DeepMind Unveils AI Coding Agent AlphaEvolve - InfoQ, https://www.infoq.com/news/2025/05/google-alpha-evolve/ 9. How AlphaEvolve Transforms Algorithm Discovery - Latenode, https://latenode.com/blog/gemini-has-been-integrated-with-github 10. AlphaEvolve: A coding agent for scientific and algorithmic discovery - Googleapis.com, https://storage.googleapis.com/deepmind-media/DeepMind.com/Blog/alphaevolve-a-gemini-powered-coding-agent-for-designing-advanced-algorithms/AlphaEvolve.pdf 11. Volker Strassen (1936 - ) - Biography - MacTutor History of ..., https://mathshistory.st-andrews.ac.uk/Biographies/Strassen/ 12. Strassen algorithm - Wikipedia, https://en.wikipedia.org/wiki/Strassen_algorithm 13. math.mit.edu, https://math.mit.edu/~djk/18.310/18.310F04/Matrix_%20Multiplication.html#:~:text=The%20consequence%20we%20draw%20from,a%20total%20of%2049%20multiplications. 14. 25. Strassen's Fast Multiplication of Matrices Algorithm, and Spreadsheet Matrix Multiplications, https://math.mit.edu/~djk/18.310/18.310F04/Matrix_%20Multiplication.html 15. Why Google DeepMind's AlphaEvolve math and server wins could signal future R&D payoffs, https://www.rdworldonline.com/why-google-deepminds-alphaevolve-incremental-math-and-server-wins-could-signal-future-rd-payoffs/ 16. Ruling Out Low-rank Matrix Multiplication Tensor Decompositions with Symmetries via SAT, https://arxiv.org/html/2402.01011v1 17. AlphaEvolve: A Gemini-powered coding agent for designing advanced algorithms - Reddit, https://www.reddit.com/r/Futurology/comments/1koqm5t/alphaevolve_a_geminipowered_coding_agent_for/ 18. Some comments on AlphaEvolve - NYU Computer Science, https://cs.nyu.edu/~davise/papers/AlphaEvolveNotes.pdf 19. MIT asks arXiv to remove preprint paper on AI and scientific discovery - Reddit, https://www.reddit.com/r/ArtificialInteligence/comments/1koa5lj/mit_asks_arxiv_to_remove_preprint_paper_on_ai_and/