Do Structured Ideation Methods Still Help Frontier Models? A Blind Patent-Rediscovery Study

Alaric Tabariès

18 November 2026 · Alaric Tabariès' Notebook
altab.fr/en/billets/structured-ideation-blind-patent-rediscovery/

Abstract

Many recent tools wrap a structured ideation method (TRIZ, effect catalogues, cross-domain analogy) around a large language model, on the assumption that guiding the model yields better solutions than simply asking it for ideas. We test this assumption with a blind rediscovery protocol. From 24 European and international patent applications published after the stated knowledge cutoffs of the models under test, we derive problem statements from the prior-art section only, and ask whether each approach produces an idea matching the mechanism of the main claim. We compare plain prompting (7 or 30 ideas) with a contradiction-matrix TRIZ procedure, a forced pass over a 255-entry catalogue of physical effects, and function-based analogy with patent search, using Claude Fable 5.2 and GPT-6.1 Sol as solvers and as cross-model judges. No structured method recovers more inventions than plain prompting in any configuration. With GPT-6.1 Sol, two identical 30-idea runs recover 5 and 8 inventions out of 24, while the catalogue and TRIZ recover 6 and 8: the gap between repeated plain runs is as large as the gap between plain prompting and any method. An apparent complementarity of the effect catalogue with one model does not replicate with the other. We also show that problem statements written with access to the full patent inflate rediscovery rates (17 of 24 versus 8 of 24 for the same model). The study is exploratory, with a small sample and LLM judges. We release the protocol, statements, idea lists and verdicts.

Keywords. large language models · TRIZ · ideation · patents · evaluation

1. Introduction

Structured ideation methods were designed to move human engineers away from habitual solutions. TRIZ maps a technical contradiction to inventive principles; effect catalogues list physical phenomena to consider; design-by-analogy searches distant domains for a mechanism that performs the same function. A growing set of tools now applies these methods to large language models, by prompting the model through the steps of the method.

These tools share an assumption that is rarely tested: that the method improves the ideas the model would have produced when simply asked. Tool papers typically report case studies or fidelity to the method, without a controlled comparison against the same model prompted plainly. Broader comparisons of prompting strategies exist, but they score divergence with embedding distances between generated ideas, without an external reference for usefulness.

We ask a narrower and harder question. Given a real engineering problem, does a model recover the invention that was actually patented for it, and does a structured method make this more likely? Rediscovery of a held-out invention gives a ground truth that does not depend on a judge's taste for novelty. To keep the reference out of the models' knowledge, we use patent applications published after their stated knowledge cutoffs.

Our contributions are:

1. A blind patent-rediscovery protocol for engineering ideation, with randomly drawn patent classes, statements written from prior art only, an explicit leak audit, anonymised lists and two judges.

 CC BY 4.0

2. A comparison of three method families with plain prompting at equal list length, on two frontier models, with each model also judging the other's lists.
3. Evidence that run-to-run variation of plain prompting is as large as any difference between plain prompting and a method, which a single-run baseline would hide.
4. A measurement of how much problem-statement leakage inflates rediscovery rates.

2. Related work

LLM tools built on ideation methods. AutoTRIZ (Jiang et al., 2024) automates the TRIZ contradiction workflow and is evaluated on contradiction detection against textbook cases and on a case study; its stated benefits are accessibility and an interpretable process. TRIZ Agents (Szczepanik & Chudziak, 2025) distributes the workflow over several agents and assesses each step of one engineering case study against the original human study. Neither reports a controlled comparison of solution quality with the same model prompted without the method. A study in molecular materials science (Gusarov et al., 2026) does run a baseline mode without TRIZ constraints beside its TRIZ-guided workflow, with an evaluator model scoring both; across nine local models and eleven chemistry problems, quality and novelty scores for the two modes stay within the variation between models, and the benefit it reports is lower token use. It uses no held-out reference solution. These tools build on a long line of work on TRIZ (Altshuller, 1984) and on computational analogy (Hope et al., 2017; Kang et al., 2022).

Method prompts versus simple prompts. IDEAFix (Carichon et al., 2026) is the closest work. It compares 25 prompts from eight families, including TRIZ, SCAMPER, C-K theory and design thinking, against control prompts on five models and 81 briefs. It reports that simple prompts with explicit divergence cues often match or beat method-based prompts. Its metrics are automated embedding distances (diversity, novelty relative to control outputs, rarity); it uses no human raters and no external ground truth, implements each method as a single prompt, and does not analyse gains by model. Its authors call for usefulness measures and for iterative, multi-step procedures. Our study answers both points with a different measure, and adds retrieval-based analogy and cross-model replication. Meincke et al. (2024) compare 35 prompting strategies for idea diversity.

Diminishing returns of prompting techniques. Studies in software engineering (Wang et al., 2024) and question answering (Meincke et al., 2025) find that techniques developed for earlier models help less, or hurt, with recent ones. We are not aware of an equivalent study for engineering ideation.

Rediscovery as evaluation. Several benchmarks ask models to recover a published research idea or finding under a temporal cutoff, among them Reconstruction (Chen et al., 2026), MOOSE-Chem (Yang et al., 2024), ResearchBench (Liu et al., 2025), AI Idea Bench 2025 (Qiu et al., 2025) and FIRE-Bench (Wang et al., 2026). All target scientific research. We found no equivalent using patented inventions as the reference.

LLM judges of ideas. Novelty judgments by LLMs are sensitive to prompt wording and can diverge from expert judgments (Ji et al., 2026; Si et al., 2024). We therefore avoid novelty ratings: judges decide whether an idea discloses the mechanism of a specific claim, a narrower task, and we check stability by blind re-judging and by a judge from another provider.

3. Method

The protocol was written down before any run; later changes are dated in the protocol file.

3.1 Patent sample

We drew 36 CPC classes at random, after removing nine classes: H10 (semiconductor devices), C06 (explosives), F22 and F23 (steam generation, combustion), C07 and C12 (organic chemistry,

 CC BY 4.0

biochemistry), G06 and H04 (computing, communication) and A61 (medical). For each class, a preparation agent queried the EPO Open Patent Services for EP and WO applications published from 1 August 2026, examined results in the order returned, and kept the first one meeting fixed criteria: claim 1 describes an identifiable physical or chemical mechanism; no family member published before 1 August 2026; claims text available. 24 applications were retained, one per class, from 158 logged candidates; every rejection is logged with its reason. Because the service returns recent publications first, the retained applications (9 EP, 15 WO) were all published between 29 October and 11 November 2026, with priority dates in April and May 2025.

Model	Stated knowledge cutoff	Role
Claude Fable 5.2	end of June 2026	solver, judge, preparation
GPT-6.1 Sol	30 April 2026	solver and judge, second series
GPT-5.6 Sol	16 February 2026	solver and judge, first series

3.2 Problem statements

First series (v1). A preparation agent wrote each statement from the full patent, with the instruction not to reveal the solution.

Second series (v2). One agent extracted only the background and prior-art section of the description, stopping before any sentence announcing the invention. A second agent, which never saw the claims, wrote the statement from that extract alone.

Leak audit. A separate agent compared each statement with its patent and rated leakage: none, orientation (constraints or objectives leave little but the solution family), or clear leak. The same grid was applied to both series.

3.3 Approaches

Each (problem, approach) pair ran in a fresh session with no web access and no access to other outputs.

Code	Approach	Budget
T7	Plain prompting, best 7 ideas	7
T30	Plain prompting, 30 varied ideas; run twice in v2	30
AT	Contradiction-matrix TRIZ, after AutoTRIZ: state a contradiction among the 39 parameters, look up principles in the classical matrix through a script, write solutions for each returned principle; repeated passes	30
CA	Forced pass over a 255-entry catalogue of physical effects, built by an agent with no knowledge of the cases; ideas only from retained entries	30
FA	Function-based analogy with search: abstract the problem into functions, search patents and papers dated before the priority date in distant domains, transpose	30

All outputs use the same format (title, mechanism, implementation) and carry no trace of the method, so that judges cannot tell lists apart. FA was run in v1 with Claude only.

3.4 Judging

Lists were anonymised with random labels per case. Two independent judges scored every idea against the reference: 2 for the same mechanism as claim 1, applied at the same place in the system; 1 for partial kinship; 0 otherwise; ties broken downward. An invention counts as recovered when both judges give at least one idea a score of 2. Controls: lists already judged were mixed blind into later batches; a third judge used a reworded grid on a subset; and each model judged the lists produced by the other.

 CC BY 4.0

3.5 Execution

Claude runs used sub-agents in an isolated workspace. GPT runs used the Codex CLI in non-interactive mode, one fresh session per pair, web search disabled and command network access blocked, from the same statement, instruction, matrix and catalogue files.

4. Results

No structured method recovers more inventions than plain prompting in any configuration, and repeated plain runs differ as much as methods do. Counts are inventions recovered according to both judges.

4.1 Statement leakage

Leak rating	v1 statements	v2 statements
None	3	12
Orientation	19	12
Clear leak	2	0

With the same model and the same 7-idea prompt, Claude recovers 17 of 24 inventions from v1 statements and 8 of 24 from v2 statements. Three v2 statements were rated too vague to be solvable, because the patent background runs to a few sentences.

4.2 First series (v1 statements, 24 cases)

Approach	Claude Fable 5.2	GPT-5.6 Sol
T7	17	13
T30	18	15
AT	16	10
CA	17	12

Paired against T30 with an exact McNemar test (McNemar, 1947), AT wins 1 case and loses 3 with Claude ($p = 0.63$) and wins 2 and loses 7 with GPT-5.6 Sol ($p = 0.18$); pooled over both models, 3 wins and 10 losses ($p = 0.09$). CA wins 1 and loses 2 with Claude, wins 2 and loses 5 with GPT-5.6 Sol. Methods also push the matching idea down the list: it falls within the first seven ideas in 16 of the T30 lists but 8 of the AT lists with Claude.

FA, run with Claude on 23 cases, recovers 14 inventions against 17 for T30 (1 win, 4 losses, $p = 0.38$). Its instruction allowed only ideas traced to a retrieved source, and patent search ran on abstracts under heavy rate limiting.

The two models recover different inventions: across all approaches, 22 of 24 are recovered by at least one of them.

4.3 Second series (v2 statements)
GPT-6.1 Sol, all 24 cases.

Approach	Recovered of 24
T7	5
T30, first run	5
T30, second run	8
CA	6
AT	8

Claude Fable 5.2. T7 recovers 8 of 24. On the 16 cases it misses:

 CC BY 4.0

Approach	Recovered of 16
T30, first run	3
T30, second run	3
AT	3
CA	4

What each addition brings to a first T30 run.

Added to T30	New cases, Claude	New cases, GPT-6.1 Sol
A second T30 run	1	3
AT	1	4
CA	3	2

The catalogue appears complementary with Claude and not with GPT-6.1 Sol; TRIZ appears complementary with GPT-6.1 Sol and not with Claude. Neither pattern holds across models, and both are within the spread between two identical plain runs.

4.4 Judge reliability

Check	Agreement on "recovered"
Two Claude judges, v1 Claude lists	98%, Cohen's kappa 0.95 (Cohen, 1960)
Two Claude judges, v2 Claude lists	64 of 64
Two Claude judges, v2 GPT lists	110 of 120
Blind re-judging of lists by new judges	36 of 36 identical
Third judge with reworded grid	24 of 26
GPT-5.6 Sol versus Claude judges, Claude v1 lists	92 of 99, kappa 0.83
GPT-6.1 Sol versus Claude judges, GPT v1 lists	90 of 96, kappa 0.88

The ranking of approaches is the same under either model's judgments. The size of the TRIZ deficit in the first series depends on the judge: 5 cases below T30 by Claude's judges, 2 by GPT's.

5. Discussion

The assumed gain is not observed. Across two statement versions, three method families, two solver models and two judge models, no method recovered more inventions than plain prompting at equal list length. This is consistent with IDEAFix (Carichon et al., 2026), which finds on divergence metrics that simple prompts often match method prompts, and with reports that prompting techniques lose value on recent models (Meincke et al., 2025; Wang et al., 2024). It extends them to a ground-truth measure and to multi-step procedures.

Variance is the missing baseline. Two identical 30-idea runs of the same model on the same statements differ by three recovered inventions out of 24. A method evaluated against a single plain run can gain or lose that much by chance. The appropriate control is plain prompting repeated with the same number of calls and ideas as the method consumes. Our own data illustrate the risk: the catalogue's three additional cases with Claude looked like complementarity until the other model showed the opposite pattern.

Why methods may not help. A contradiction matrix or an effect catalogue supplies knowledge that a frontier model already holds; the procedure then mainly constrains which part of that knowledge is used. When the imposed principle or entry has no purchase on the problem, the list fills with forced ideas and the matching one moves down. The retrieval-based method replaced the model's own ideas with those it could trace to a source, and lost cases the model

 CC BY 4.0

found unaided. A design that adds retrieved ideas to the model's own, instead of replacing them, was not tested.

Model diversity versus method diversity. In the first series the two models together recovered 22 of 24 inventions, more than any approach added to either model. Changing the model brought more new correct ideas than changing the method.

What this does not show. The methods were built for people. Whether TRIZ or analogy search helps human engineers, alone or working with a model, is outside this study, as is any benefit in traceability or shared process that a structured report provides. Rediscovery also ignores good ideas that differ from the patented one. Finally, absence of an observed advantage in a small sample is not evidence of equivalence.

Where value may lie. The results point away from procedural guidance and toward what a tool supplies that the model lacks: documents after its cutoff, private data, an expert's unstated constraints, and verification of ideas against prior art. These were not evaluated here.

6. Limitations

- **Sample size.** 24 patents and one or two runs per approach. No difference is statistically significant, and only large effects were detectable.
- **Residual leakage.** Half of the v2 statements still orient toward the solution family, since they derive from the patent's own framing of the problem.
- **Prior disclosure.** We checked patent families, not earlier papers, products or talks by the inventors. "Published after the stated cutoff" does not guarantee the idea was unseen. Low v2 recovery rates and the different cases recovered by each model do not suggest memorisation.
- **LLM judges and no human raters.** Agreement is high and stable, but one case shows systematic disagreement between judges, and no expert has yet audited a sample.
- **Method fidelity.** AT follows the published AutoTRIZ workflow with a public copy of the classical matrix, not the authors' code, and is extended to 30 ideas by repeated passes. FA is our own implementation, run under search rate limits.
- **Model versions.** GPT runs used GPT-5.6 Sol in the first series and GPT-6.1 Sol in the second. FA was run with one model and one statement version.
- **Same model family for preparation, solving and judging** in the Claude arm, mitigated by cross-model judging.
- **Sampling frame.** EP and WO applications with available text, all published within a two-week window; nine technical classes excluded.

AI-assistance statement

The study was designed and run with Claude (Anthropic) as research assistant under the author's direction: protocol drafting, statement preparation, solver and judge runs, analysis and a first draft of this text. GPT runs were launched by the author with the Codex CLI. The author is responsible for the content.

References

Altshuller, G. S. (1984). *Creativity as an exact science: The theory of the solution of inventive problems.* Gordon and Breach.

Carichon, F., Sharma, S., Girard, M., Rampa, R., & Farnadi, G. (2026). *IDEAFix: Evaluation framework for creative defixation prompting in LLMs* (arXiv:2606.00875). arXiv. https://arxiv.org/abs/2606.00875

Chen, S., Fei, Y., Liu, N., Yu, X., Li, L., Thapa, R., Ciobanu, M., Singh, N. P., Mao, Q., & Das, R. (2026). *Reconstruction: A blind benchmark for recovering research ideas from pre-publication bibliographies* (arXiv:2608.16645). arXiv. https://arxiv.org/abs/2608.16645

 CC BY 4.0

Cohen, J. (1960). A coefficient of agreement for nominal scales. *Educational and Psychological Measurement, 20*(1), 37–46. https://doi.org/10.1177/001316446002000104

Gusarov, S., Sapelnikova, S., Valdes, J. J., Hu, A., & Stoyanov, S. R. (2026). Integrating the theory of inventive problem solving with large language models: Enhancing reasoning for innovation in materials science at the molecular scale. *ChemEngineering, 10*(4), Article 54. https://doi.org/10.3390/chemengineering10040054

Hope, T., Chan, J., Kittur, A., & Shahaf, D. (2017). Accelerating innovation through analogy mining. In *Proceedings of the 23rd ACM SIGKDD International Conference on Knowledge Discovery and Data Mining* (pp. 235–243). ACM. https://doi.org/10.1145/3097983.3098038

Ji, F., Li, Y., Yang, J., Wu, J., Zhang, F., Cui, Z., Xie, Y., Peng, M., Xie, Q., Chen, X., & Xie, Z. (2026). *Style wins, substance loses: A diagnosis of LLM-as-judge in idea generation* (arXiv:2608.01666). arXiv. https://arxiv.org/abs/2608.01666

Jiang, S., Li, W., Qian, Y., Zhang, Y., & Luo, J. (2024). *AutoTRIZ: Automating engineering innovation with TRIZ and large language models* (arXiv:2403.13002). arXiv. https://arxiv.org/abs/2403.13002

Kang, H. B., Qian, X., Hope, T., Shahaf, D., Chan, J., & Kittur, A. (2022). Augmenting scientific creativity with an analogical search engine. *ACM Transactions on Computer-Human Interaction, 29*(6), Article 57. https://arxiv.org/abs/2205.15476

Liu, Y., Yang, Z., Xie, T., Ni, J., Gao, B., Li, Y., Tang, S., Ouyang, W., Cambria, E., & Zhou, D. (2025). *ResearchBench: Benchmarking LLMs in scientific discovery via inspiration-based task decomposition* (arXiv:2503.21248). arXiv. https://arxiv.org/abs/2503.21248

McNemar, Q. (1947). Note on the sampling error of the difference between correlated proportions or percentages. *Psychometrika, 12*(2), 153–157. https://doi.org/10.1007/BF02295996

Meincke, L., Mollick, E. R., Mollick, L., & Shapiro, D. (2025). *Prompting science report 2: The decreasing value of chain of thought in prompting.* Wharton Generative AI Labs. https://gail.wharton.upenn.edu/research-and-insights/tech-report-chain-of-thought/

Meincke, L., Mollick, E. R., & Terwiesch, C. (2024). *Prompting diverse ideas: Increasing AI idea variance* (arXiv:2402.01727). arXiv. https://arxiv.org/abs/2402.01727

Qiu, Y., Zhang, H., Xu, Z., Li, M., Song, D., Wang, Z., & Zhang, K. (2025). *AI Idea Bench 2025: AI research idea generation benchmark* (arXiv:2504.14191). arXiv. https://arxiv.org/abs/2504.14191

Si, C., Yang, D., & Hashimoto, T. (2024). *Can LLMs generate novel research ideas? A large-scale human study with 100+ NLP researchers* (arXiv:2409.04109). arXiv. https://arxiv.org/abs/2409.04109

Szczepanik, K., & Chudziak, J. A. (2025). TRIZ Agents: A multi-agent LLM approach for TRIZ-based innovation. In *Proceedings of the 17th International Conference on Agents and Artificial Intelligence* (Vol. 1, pp. 196–207). SCITEPRESS. https://doi.org/10.5220/0013321900003890

Wang, G., Sun, Z., Gong, Z., Ye, S., Chen, Y., Zhao, Y., Liang, Q., & Hao, D. (2024). *Do advanced language models eliminate the need for prompt engineering in software engineering?* (arXiv:2411.02093). arXiv. https://arxiv.org/abs/2411.02093

Wang, Z., Bai, F., Luo, Z., Su, J., Sun, K., Yu, X., Liu, J., Zhou, K., Cardie, C., Dredze, M., Hu, Z., & Xing, E. P. (2026). *FIRE-Bench: Evaluating AI agents on the rediscovery of scientific insights* (arXiv:2602.02905). arXiv. https://arxiv.org/abs/2602.02905

Yang, Z., Liu, W., Gao, B., Xie, T., Li, Y., Ouyang, W., Poria, S., Cambria, E., & Zhou, D. (2024). *MOOSE-Chem: Large language models for rediscovering unseen chemistry scientific hypotheses* (arXiv:2410.07076). arXiv. https://arxiv.org/abs/2410.07076

 CC BY 4.0