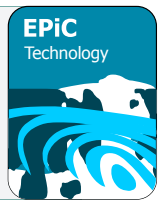
EPiC Series in Technology

Volume 3, 2026, Pages 153–176

Proceedings of AI for Accelerated Research Symposium



LARC: Towards Human-level Constrained Retrosynthesis Planning through an Agentic Framework

Frazier N. Baker[1], Daniel Adu-Ampratwum[2], Reza Averly[1], Botao Yu[1], Huan Sun[1], and Xia Ning[1,2,3,4]

[1] Department of Computer Science and Engineering
[2] Division of Medicinal Chemistry & Pharmacognosy
[3] Department of Biomedical Informatics
[4] Translational Data Analytics Institute
The Ohio State University, Columbus, OH, USA
{baker.3239,adu-ampratwum.1,averly.1,yu.3737,sun.397,ning.104}@osu.edu

Abstract

Large language model (LLM) agent evaluators leverage specialized tools to ground the rational decision-making of LLMs, making them well-suited to aid in scientific discoveries, such as constrained retrosynthesis planning. Constrained retrosynthesis planning is an essential, yet challenging, process within chemistry for identifying synthetic routes from commercially available starting materials to desired target molecules, subject to practical constraints. Here, we present LARC, the first LLM-based Agentic framework for Retrosynthesis planning under Constraints. LARC incorporates agentic constraint evaluation, through an Agent-as-a-Judge, directly into the retrosynthesis planning process, using agentic feedback grounded in tool-based reasoning to guide and constrain route generation. We rigorously evaluate LARC on a carefully curated set of 48 constrained retrosynthesis planning tasks across 3 constraint types: avoiding carcinogens, pyrophoric substances, or a user-specified substance. LARC achieves a 72.9% success rate on these tasks, vastly outperforming LLM baselines and approaching human expert-level success in substantially less time. The LARC framework is extensible, and serves as a first step towards an effective agentic tool or a co-scientist to human experts for constrained retrosynthesis.

1 Introduction

Large language model (LLM) agents have recently shown great promise as evaluators [50]. An LLM agent evaluator, also called an Agent-as-a-Judge, provides grounded evaluations in complex settings, eliciting the strengths of general-purpose LLMs and domain-specific tools. These evaluators are poised to make substantial impacts in scientific discoveries, such as in chemistry, where complex evaluation settings abound [8, 4], accurate evaluations are paramount, and many domain-specific tools exist to ground evaluations [6, 2]. One such application is *constrained retrosynthesis planning* [12], an essential process in chemistry for identifying synthetic routes from commercially available starting materials to desired target molecules (products), subject to practical constraints (e.g., avoiding hazardous molecules).

J. Masnec, H.R. Karimian, P. Kordjamshidi and Y. Li (eds.), AIAS 2025 (EPiC Series in Technology, vol. 3), pp. 153–176

LLM agents for constrained retrosynthesis planning are not explored. Current artificial intelligence (AI) methods are primarily focused on unconstrained retrosynthesis planning [10, 49, 27, 17, 11, 3, 14], aiming to generate synthetic routes that are feasible. Only a few AI methods have attempted to address constrained retrosynthesis planning [47, 20]. However, they only support a very simplistic constraint – including a user-specified molecule in the synthetic routes, and cannot be applied to more general and practical constraints, such as avoiding broad classes of hazardous molecules in the synthetic routes. These constraints are substantially more challenging to evaluate and enforce, requiring specialized knowledge of hazardous materials. Rather than guiding synthesis planning towards a single, clearly-specified goal, these constraints require guiding it away from many diverse hazards. LLM agents are well-suited for such a challenge, as they can leverage specialized chemistry tools to ground their evaluations and make rational decisions to guide the planning. Furthermore, LLM agents could support a variety of constraints, choosing the appropriate tools for each type of constraint. They have the potential to mimic the typical behaviors of human chemists, such as using reference materials [41, 24] to assess safety constraints during retrosynthesis planning [8], and thus, automate, accelerate, and optimize the reliability of outcomes in the constrained retrosynthesis planning process.

Here, we present LARC, an **L**LM-based **A**gentic framework for **R**etrosynthesis planning under **C**onstraints. Figure 1 presents an overview of LARC. LARC uses an Agent-as-a-Judge, equipped with chemistry tools, to evaluate constraints during retrosynthesis planning. This agentic feedback is incorporated back into the retrosynthesis planning process, dynamically guiding and constraining route generation. It addresses key safety constraints in retrosynthesis planning, such as avoiding carcinogens, pyrophoric substances, or a user-specified substance. LARC is extensible by design, allowing it to improve or expand as future capabilities emerge. To the best of our knowledge, LARC is the first agentic framework for constrained retrosynthesis planning, representing an innovative paradigm for this complex scientific problem.

We rigorously evaluate LARC on a carefully curated set of 48 constrained retrosynthesis planning tasks across 3 constraint types. LARC achieves an impressive 72.9% success rate on these tasks, indicating that LARC is very effective at constrained retrosynthesis planning. We compare LARC against general-purpose LLMs and a human expert. The experiments show that LARC vastly outperforms the LLMs and approaches expert-level success in substantially less time. Case studies indicate that LARC can mimic human expert's retrosynthesis planning logic and even produce better synthetic routes on some tasks. An ablation study reveals the key impact of agentic tooling in LARC, enabling high success rate and efficiency through deliberate and grounded evaluations. With further extension to cover comprehensive practical constraints, the LARC framework can serve as an effective agentic tool or a co-scientist to human experts for constrained retrosynthesis. The code and data for LARC are publicly available at https://github.com/ninglab/LARC.

2 Related work.

Constrained retrosynthesis planning. Recently, AI methods have emerged for constrained retrosynthesis planning. Tango* [20] and DESP [47] both constrain retrosynthesis planning to include a user-specified molecule in the synthetic routes. Tango* adapts an unconstrained retrosynthesis planner to constrained retrosynthesis planning, incorporating molecule similarity to the user-specified molecule as constraint guidance. DESP performs a double-ended search, expanding the synthetic route from the target molecule and the user-specified molecule until the route is connected and complete. While Tango* and DESP show that AI can perform constrained retrosynthesis planning, they do not support more practical constraints (e.g., avoiding hazardous molecules). Recently, Bran et al. [7] introduced LLM-based re-ranking of synthetic routes generated by an unconstrained planner to identify those satisfying the constraints. This

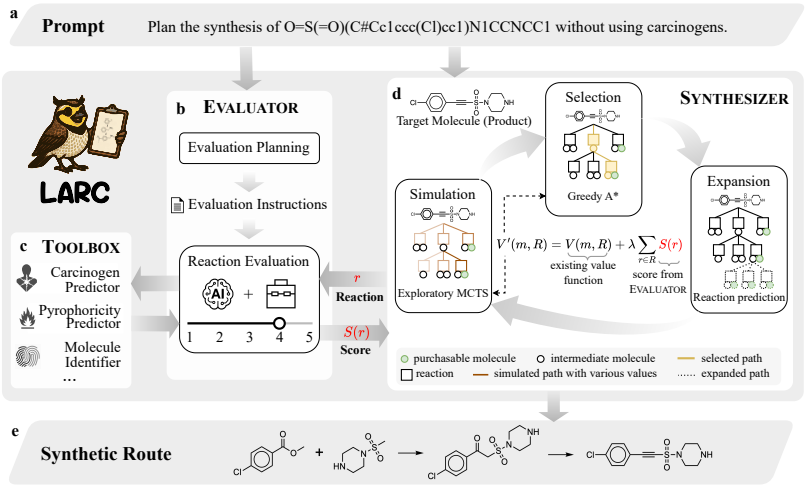


Figure 1: Overview of LARC. **a,** The user prompt specifies the target molecule (product) and the constraint for synthesis. **b,** EVALUATOR acts as a judge and evaluates each individual reaction involved in the retrosynthesis planning with respect to the constraint. **c,** The toolbox contains external tools to ground EVALUATOR's decision-making **d,** SYNTHESIZER explores and constructs synthetic routes, incorporating feedback from EVALUATOR. **e,** LARC outputs a synthetic route that satisfies the constraint.

approach is computationally expensive and assumes that some generated synthetic routes will satisfy the constraint, which may not always occur. Moreover, it relies on an LLM's intrinsic knowledge alone to evaluate routes, which may be insufficient for some constraints. Thus, there remains a need for a framework that directly incorporates agentic constraint evaluation into the planning process, enabling synthetic route generation under practical constraints. Additional related work on LLMs for chemistry is discussed in Appendix C.

3 LARC: An Agentic Framework

LARC is an agentic framework for constrained retrosynthesis planning. For a given target molecule (i.e., product), it effectively plans its synthetic routes that satisfy constraints specified by user prompts. LARC incorporates constraint evaluation directly into its planning process, leveraging agentic feedback grounded in tool-based reasoning to dynamically guide and constrain route generation. LARC features two key components: **(1)** EVALUATOR, which acts as a judge and evaluates each individual reaction involved in the retrosynthesis planning with respect to the constraint, and **(2)** SYNTHESIZER, which explores and constructs synthetic routes, incorporating feedback from EVALUATOR. Through these components, LARC integrates and elicits the strengths of LLMs, cheminformatics tools, and search and planning algorithms, representing an innovative, extensible, agentic paradigm, with Agent-as-a-Judge in the loop, for computer-aided constrained retrosynthesis planning.

3.1 Evaluator

EVALUATOR, an LLM-based Agent-as-a-Judge [50], assesses whether reactions satisfy the constraints specified in the user prompts. EVALUATOR approaches this task in two phases: **(1)** evaluation planning, where it generates evaluation instructions specific to user constraints, and **(2)** reaction evaluation, where, following the evaluation instructions, it evaluates reactions and provides feedback to SYNTHESIZER.

Evaluation planning. Evaluation planning occurs once for each user prompt at the beginning, during which EVALUATOR details its plan for reaction evaluation, including tools it plans to use

and its scoring strategy, into concrete and structured evaluation instructions. These instructions will be consistently followed by Evaluator throughout the entire retrosynthesis planning process. The system instructions for evaluation planning are provided in Appendix B. They were designed to encourage strategic, step-by-step evaluation while providing key information on tool syntax and semantics, including input/output formats.

Reaction evaluation. Following the evaluation instructions, Evaluator evaluates reactions, assessing how well they satisfy the constraint. In doing so, Evaluator adaptively switches between two action modes: **(1)** leveraging its own intrinsic knowledge, acting as an AI expert itself, or **(2)** consulting external tools in the toolbox, inspired by the typical behaviors of synthetic chemists using reference materials [41, 16] during retrosynthesis planning. Evaluator calibrates the evaluation outputs into scores, which will guide Synthesizer. These scores, denoted as $S(r)$, where r is the evaluated reaction, quantify the degree of constraint satisfaction on a discrete scale from 1 to 5, with 1 indicating complete violation and 5 indicating full satisfaction. Evaluator is extensible by design to incorporate new tools or adapt to tool updates, allowing it to improve and expand as future capabilities emerge.

3.2 Synthesizer

LARC uses Synthesizer to generate synthetic routes for target molecules. Synthesizer is built upon existing unconstrained retrosynthesis planners, adapting them to constrained retrosynthesis planning. Unconstrained planners typically leverage a search algorithm, such as A* [21] or Monte Carlo Tree Search (MCTS) [13], to iterate backwards from the target molecule and search for intermediates. The search is expanded through single-step retrosynthesis planning on the current initial intermediates until a full synthetic route is found, starting from commercially available materials. The search expansion is guided by some value function, $V(m, R)$, which estimates the utility of expanding the search along route R upstream from its initial intermediate m. $V(m, R)$ is independently pre-trained with respect to general objectives in unconstrained retrosynthesis planning, such as preference for short routes with feasible and chemically plausible reactions.

To constrain the search, Synthesizer combines $V(m, R)$ with Evaluator's score $S(r)$, thus producing a new, *constraint-aware* value function $V'(m, R)$ to guide the search:

$$V'(m, R) = V(m, R) + \lambda \underbrace{\sum\nolimits_{r \in R} S(r)}_{\text{constraint evaluation by Evaluator over all reactions along the route } R}, \tag{1}$$

where m is the intermediate molecule, R is a synthetic route starting from m, r is a reaction in R, $S(r)$ is the score from Evaluator, and $\lambda > 0$ is a trade-off hyperparameter. Thus, V' incorporates the constraint evaluation by Evaluator on all the reactions along R to guide retrosynthesis planning, ensuring the entire route is maximally subject to the constraint. This represents an innovation, differentiating Synthesizer from existing work on constrained retrosynthesis [47, 20]. Note that Synthesizer can easily adapt any unconstrained retrosynthesis planner without retraining the original $V(m, R)$, allowing LARC to accommodate future advancements in retrosynthesis planning.

4 LARC Instantiation

The current implementation of LARC adapts MEEA* [49], a state-of-the-art method for unconstrained retrosynthesis planning, as Synthesizer. MEEA* uses a two-step process to determine how to expand its search. The first step uses MCTS to simulate route planning over the expanded partial routes, selecting at most K candidate routes for further expansion. In this simulation, LARC uses V'_{MCTS} as its contraint-aware value function, that is, $V'_{\text{MCTS}} =$

$V_{\text{MCTS}} + \lambda \sum_{r \in R} S(r)$ (Equation 1), where V_{MCTS} includes an upper confidence bound (UCB) term [28] to encourage exploration. In this case, $S(r)$ is calculated as follows: For the reactions that have not been evaluated by Evaluator in previous expansions, an optimistic default score ($S(r) = 5$) is used to further encourage exploration; for those that have been evaluated, their actual evaluation score $S(r)$ is used. In the second step of MEEA*, A* search is used to select a single route for expansion from the K candidate routes. In this step, Evaluator first evaluates all the reactions in the K candidate routes. Then, LARC uses V'_{A^*}, that is, $V'_{\text{A}^*} = V_{\text{A}^*} + \lambda \sum_{r \in R} S(r)$ (Equation 1), to select a single route for expansion. Implementation details are presented in Appendix D.

Benchmark dataset for constrained retrosynthesis. We carefully curated a benchmark set of constrained retrosynthesis tasks for a set of target molecules (products) from the USPTO-190 [10], each with a single constraint to satisfy. In this instantiation, we considered constraints of avoiding hazardous substances in the synthetic routes, which can pose serious safety risks to chemists, equipment, and the environment. Three types of hazardous substances are included: **(1)** carcinogens, which are capable of causing cancer based on the classification by the International Agency for Research on Cancer (IARC) [24]; **(2)** pyrophoric substances, which can ignite spontaneously upon exposure to air, according to the U.S. Navy report on air- and water-reactive materials [18]. and **(3)** a user-specified hazardous substance (e.g., phosgene). To ensure these tasks are non-trivial, we selected the tasks such that the state-of-the-art unconstrained retrosynthesis planning methods could generate a valid route but violate the constraint, and the target molecule was not used in $V(m, R)$ pre-training. In the end, 48 tasks were constructed for the benchmark set. Figure 2**b** presents the distribution of the three types of constraints. Please note, these 48 tasks represent diverse constraint violations from the unconstrained planner, with 28 tasks covering 16 distinct carcinogens, 12 tasks covering 9 pyrophoric substances, and 8 tasks covering 8 user-specified molecules. The benchmark tasks are available in Appendix A.

Tools for retrosynthesis constraints. Three specific tools are supplied in the toolbox: **(1)** a carcinogen predictor, which predicts whether a given molecule is a carcinogen using the state-of-the-art ADMET-AI [39] model, **(2)** a pyrophoricity predictor, which predicts the pyrophoricity of molecules by comparing them with known pyrophoric substances [18], with higher molecule similarities indicating higher likelihood of pyrophoricity, and **(3)** a molecule identifier, which identifies specific hazardous molecules using their fingerprints [33, 36]. Outputs from the tools will be calibrated by Evaluator into scores ($S(r)$ in Equation 1).

5 Experimental Settings

Base LLMs and baselines. We select Mistral Nemo [32] and Claude 3.5 Sonnet [1] as the base LLMs for Evaluator, resulting in LARC variations denoted as $\text{LARC}_{\text{Mistral}}$ and $\text{LARC}_{\text{Claude}}$, respectively. Mistral Nemo is selected as a representative small, open-source LLM for its strong instruction-following capabilities. It allows for the evaluation of how LARC performs with a cost-effective, openly available model, thereby demonstrating its practicality and accessibility. Claude 3.5 Sonnet represents a state-of-the-art, closed-source LLM that has demonstrated strong performance in chemistry tasks and tool use [23, 46]. Note that LARC is not limited to these two base LLMs, as its design is model-agnostic and can incorporate other LLMs as the field advances.

We use general-purpose LLMs as the baselines for constrained retrosynthesis planning, including Claude 3.5 Sonnet [1], GPT-4o [34], DeepSeek R1 [15], and Mistral Nemo [32]. We also compare LARC against a human expert in retrosynthesis planning, denoted as Expert. Expert is an experienced synthetic chemist with a doctoral degree and over 17 years of experience in retrosynthesis planning. We detail our rigorous human retrosynthesis planning protocol

below and our LLM baselines in Appendix E. Please note, existing constrained retrosynthesis methods cannot be fairly compared as baselines due to the novel and challenging nature of this problem. DESP and TANGO* do not support the constraints studied here, and the re-ranking approach [7] relies on multiple route generations.

Constrained retrosynthesis planning by human experts. We also compare LARC against a human expert in retrosynthesis planning, denoted as EXPERT. EXPERT is an experienced synthetic chemist with a doctoral degree and over 17 years of experience in retrosynthesis planning. Such human experts are a scarce resource, limiting our experiments to a single EXPERT. For each task, EXPERT was provided with clear instructions on the constraint and the desired chemical product. EXPERT had full access to necessary resources and reference materials, including lists of purchasable materials [10], carcinogenic chemicals, and pyrophoric chemicals, as well as external and online references such as Reaxys [16] and SciFinder [9]. However, EXPERT was not permitted to use computer-aided multi-step retrosynthesis planning tools, such as AIZynthFinder [17] or MEEA* [49]. To ensure both the quality and efficiency of human planning, EXPERT was instructed to complete each task with careful attention to detail while minimizing planning time. To support sustained performance and maintain high-quality planning, EXPERT was encouraged to take a 15-minute break between tasks.

Evaluation metrics. The generated synthetic routes are evaluated according to the following criteria: **(1)** Route presence: the routes are not empty – they contain some molecules; **(2)** Route connectivity: the routes are fully connected – all intermediate molecules are synthesized from preceding precursors; **(3)** Target reachability: the routes lead to the target molecule; **(4)** Commercial availability: the starting materials of the routes are directly purchasable from eMolecules dataset [10]; **(5)** Molecule validity: the molecules involved in the routes are chemically correct; and **(6)** Constraint satisfaction: the routes meet the specific constraints (avoid certain substances). Most of these criteria can be evaluated using automatic, in silico methods; however, molecule validity may require manual verification to account for valid non-SMILES representations from LLM baselines are considered valid. Based on these criteria, the following metrics are used to evaluate synthetic routes, as illustrated in Figure 2**a**: **(1) Success rate**: the percentage (%) of routes that satisfy all the six criteria, that is, successful routes; **(2) Validity rate**: the percentage (%) of routes that satisfy route presence, route connectivity, target reachability, commercial availability, and molecule validity, that is, valid routes per se but they do not necessarily satisfy the constraint; **(3) Presence rate**: the percentage (%) of routes that satisfy route presence, that is, non-empty routes.

6 Results

We focus primarily on carcinogenicity-constrained tasks here, with results of the pyrophoricity-constrained and user-specified-constrained tasks in Appendix F.

6.1 Carcinogenicity-constrained retrosynthesis planning

Figure 2**d** shows that $\text{LARC}_{\text{Mistral}}$ achieves a 82.1% success rate and $\text{LARC}_{\text{Claude}}$ achieves 64.3% in planning synthetic routes that avoid carcinogenic substances, both vastly outperforming the best LLM baseline CLAUDE 3.5 SONNET (success rate only 25.0%). CLAUDE 3.5 SONNET, as a general-purpose LLM, lacks specialized training on retrosynthesis planning [1, 46], and fails in the vast majority of the cases. Unlike general-purpose LLMs, $\text{LARC}_{\text{Mistral}}$ and $\text{LARC}_{\text{Claude}}$ are agentic and intentionally designed for this constrained retrosynthesis, eliciting the strengths of LLM reasoning, domain-specific tools, and efficient search algorithms. The results demonstrate that LARC is very effective in generating synthetic routes avoiding carcinogenic substances.

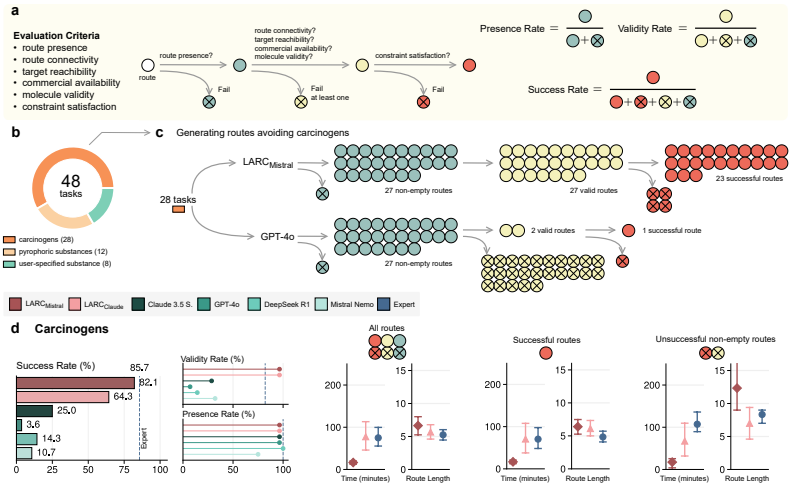


Figure 2: Evaluation. **a,** Evaluation criteria defining success, validity, and presence rates; **b,** Benchmark dataset of 48 curated constrained retrosynthesis tasks spanning 3 constraint types; **c,** Example evaluation showing calculation of presence, validity, and success rates; **d,** success, validity, and presence rates, planning time, and route length for benchmark tasks for avoiding carcinogens.

Expert generates 28 routes with a success rate of 85.7%, only slightly outperforming $\text{LARC}_{\text{Mistral}}$. The successful routes by Expert have an average length of 4.83, which is shorter than that of $\text{LARC}_{\text{Mistral}}$ (6.39). This indicates that Expert may rely on domain-specific heuristics, intuitive shortcut strategies, or tacit knowledge not yet fully captured by $\text{LARC}_{\text{Mistral}}$. However, Expert requires on average 70.42 minutes to generate each successful route, substantially slower than $\text{LARC}_{\text{Mistral}}$ (16.45 minutes) (p=2.8e-4, two-sided two-sample t-test). This highlights the potential for $\text{LARC}_{\text{Mistral}}$ to accelerate constrained retrosynthesis planning while maintaining near-human-level quality.

LARC and Expert exhibit different behaviors in their unsuccessful constrained retrosynthesis planning. LARC is diligent about planning valid routes; all of its non-empty routes are valid. However, LARC's routes can sometimes violate the constraint. This is often the result of errors in the tools outputs and their interpretation, which is discussed further in Appendix F.4. In contrast, Expert is conscientious about constraint satisfaction, but this seems to distract from route validity. Specifically, Expert's synthetic routes may not reach the target molecule, or it may require starting materials that are not commercially available. This highlights a key challenge of constrained retrosynthesis planning for both LARC and Expert: balancing constraint satisfaction with validity criteria.

Interestingly, $\text{LARC}_{\text{Mistral}}$ outperforms $\text{LARC}_{\text{Claude}}$ when planning routes that avoid carcinogenic substances. $\text{LARC}_{\text{Mistral}}$ uses the small, open-source Mistral Nemo with only 12 billion parameters as its base model [32], whereas $\text{LARC}_{\text{Claude}}$ uses the much larger, proprietary Claude 3.5 Sonnet with over 175 billion parameters [1]. $\text{LARC}_{\text{Claude}}$ tends to be slower – 77.56 minutes on average to generate each route, than $\text{LARC}_{\text{Mistral}}$ (16.45 minutes), and evaluate more reactions per route (59.54) on average than $\text{LARC}_{\text{Mistral}}$ (30.8) during the planning process. While it could be a general expectation that larger models perform better due to scaling laws [26], it is observed that LARC enables the use of smaller and thus cheaper models, such as Mistral Nemo, without sacrificing performance by grounding LLM reasoning with specialized chemistry tools. Therefore, LARC provides a fast, accurate, and inexpensive solution to constrained retrosynthesis planning with low costs and a high success rate.

Aside from Claude 3.5 Sonnet, the other LLM baselines also perform poorly, with the

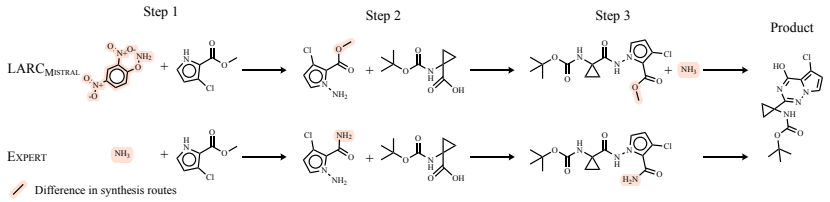


Figure 3: Synthesis route comparison of LARC$_{\text{Mistral}}$ against EXPERT for tert-butyl (1-(5-chloro-4-hydroxypyrrolo[2,1-f][1,2,4]-triazin-2-yl)cyclopropyl)carbamate.

best success rate of only 14.3%. For example, GPT-4o manages to generate only 1 successful route (success rate 3.6%) among only 2 valid routes. This further highlights the necessity of specialized tools for constrained retrosynthesis planning – a specific, very challenging yet important task, and LARC fills this gap.

6.2 Case Studies

Figure 3 shows two plans for the synthesis of the product tert-butyl(1-(5-chloro-4-hydroxypyrrolo [2,1-f][1,2,4]-triazin-2-yl)cyclopropyl)carbamate: one from LARC$_{\text{Mistral}}$ and one from EXPERT. Both LARC$_{\text{Mistral}}$ and EXPERT proposed very similar three-step routes to the final product. In both routes, Step 1 involves the amination of commercially available methyl 3-chloro-1H-pyrrole-2-carboxylate, followed by an amidation reaction in Step 2. Finally, Step 3 shows a condensation/cyclization sequence to provide the product.

The main difference between the two proposed routes lies in the source of the aminating agent in Step 1. LARC$_{\text{Mistral}}$ proposes O-(2,4-Dinitrophenyl)hydroxylamine, to prepare the hydrazine (methyl 1-amino-3-chloro-1H-pyrrole-2-carboxylate), which is known for being a very mild and selective aminating agent for introducing an NH_2 group into a molecule, making it a reagent of choice, especially when a metal-free process is desired. In contrast, EXPERT opted for ammonia (NH_3) in Step 1. NH_3 is often a preferred reagent for large-scale amination reactions due to its availability and low cost [42]. However, using NH_3 can present some challenges in activation and selectivity, often necessitating sophisticated metal catalyst systems.

As shown in Figure 3, LARC$_{\text{Mistral}}$'s use of the mild aminating agent in Step 1 will selectively introduce the NH_2 group on the pyrazole nitrogen without affecting the ester. Conversely, the use of non-selective NH_3 by EXPERT will convert the ester functional group to an amide. Both the ester and the amide can be respectively converted to the product in Step 3, completing the routes. Overall, both proposed routes have similar intermediates and transformations. Note that LARC$_{\text{Mistral}}$'s use of O-(2,4-Dinitrophenyl)hydroxylamine is not an anomaly— 58 patented reactions in the USPTO reaction dataset [30] also use this reagent. Overall, LARC$_{\text{Mistral}}$ can mimic human retrosynthesis planning logic, even in this challenging constrained retrosynthesis setting. An additional case study, showing LARC can even generate better routes than human experts, is presented in Appendix F.3. Furthermore, an ablation study examining the impact of tooling on LARC is presented in Appendix F.4.

7 Discussions and Conclusions

LARC addresses only a very simplified version of practical constrained retrosynthesis planning, with the primary goal of demonstrating the potential of agentic AI in solving such complex scientific problems. Our experiments clearly show that, when equipped with appropriate tools for verifying constraint satisfaction, agentic AI can approach human expert-level performance while being more autonomous, effective, and scalable. Such traits are highly attractive in scientific workflows, as they reduce reliance on time-consuming and potentially inconsistent, and

error-prone manual efforts, and allow for the exploration beyond existing, potentially outdated knowledge of human experts.

Meanwhile, rigorous, systematic, and scalable evaluation and validation of results from agentic AI, including LARC, still fall far short. For retrosynthesis, there is not always a definitive and consensus "ground truth" for reaction feasibility. Compounded with the inherent biases of the search space in unconstrained retrosynthesis planners used in LARC, it is still likely that LARC generates synthetic routes that appear valid but are actually infeasible. Even worse, these routes may be difficult to detect and filter using either tools or human expertise. A more reliable option is to incorporate physics-based models, such as molecular dynamics simulations, to assess reaction feasibility in agentic AI models. Unfortunately, this approach requires a lot of customization (e.g., specific force fields) and is not readily scalable, undermining the advantages of agentic AI in being both autonomous and scalable. Ultimately, testing of the AI-generated synthetic routes in a laboratory will be needed to validate the results from agentic AI and truly translate its advantages into real impacts. This will require a selection of the most promising routes, which will eventually still rely on computational tools or human expertise, suffering from the same issues as in silico evaluation. Meanwhile, large-scale in vitro validation of in silico-generated reactions is still challenging. Thus, though very promising, this research on agentic AI for constrained retrosynthesis calls for in-depth and systemic investigation on its autonomous and scalable evaluation and validation strategies. Additional discussion is available in Appendix G.

Conclusions. LARC is the first agentic framework for constrained retrosynthesis planning, which uses an LLM-based Agent-as-a-Judge to evaluate reactions and guide the constrained planning. LARC achieves a 72.9% success rate over a carefully curated benchmark of 48 constrained retrosynthesis planning tasks spanning 3 constraint types. It vastly outperforms general-purpose LLM baselines and approaches human expert-level performance. LARC can mimic the retrosynthesis planning logic of human experts and can even produce synthetic routes better than human experts. These results establish LARC as a compelling proof-of-concept for leveraging agentic AI to advance next-generation scientific discovery in synthetic chemistry. More broadly, it illustrates the transformative potential of agentic AI to accelerate progress across the sciences.

Acknowledgments

This project was made possible, in part, by support from the National Science Foundation grant nos. IIS-2133650 and IIS-2435819, and the National Library of Medicine grant no. 1R01LM014385. Any opinions, findings, conclusions or recommendations expressed in this paper are those of the authors and do not necessarily reflect the views of the funding agency. We would like to thank our colleague, Xinyi Ling, for assisting with some of the figures.

Ethics Statement

LARC is an agentic framework for constrained retrosynthesis planning, generating synthetic routes for target molecules under practical constraints. We acknowledge that not all target molecules or synthetic routes are safe, and LARC could generate harmful synthetic routes. Thus, we strongly recommend conscious oversight and intervention from human experts when using LARC. Synthetic chemists should confirm the safety of LARC's generated synthetic routes before using them in laboratory experiments. We recommend using both automated and manual safety checks against external sources (e.g. local safety policies, GHS [41], etc.), and following standard procedures for laboratory safety. We advise all users of LARC to exercise their professional discretion and follow all applicable safety guidelines, laws, regulations, and ethical standards.

References

[1] Anthropic. Introducing Claude 3.5 Sonnet, 2024.

[2] Reza Averly, Frazier N. Baker, and Xia Ning. LIDDIA: Language-based Intelligent Drug Discovery Agent. In *Proceedings of the 2025 Conference on Empirical Methods in Natural Language Processing (EMNLP)*, 2025.

[3] Frazier N. Baker, Ziqi Chen, Daniel Adu-Ampratwum, and Xia Ning. RLSynC: Offline–Online Reinforcement Learning for Synthon Completion. *Journal of Chemical Information and Modeling*, 64(17):6723–6735, September 2024. Publisher: American Chemical Society.

[4] Benjamin E. Blass. *Basic principles of drug discovery and development*. Academic Press, London, second edition edition, 2021.

[5] Daniil A. Boiko, Robert MacKnight, Ben Kline, and Gabe Gomes. Autonomous chemical research with large language models. *Nature*, 624(7992):570–578, December 2023. Publisher: Nature Publishing Group.

[6] Andres M. Bran, Sam Cox, Oliver Schilter, Carlo Baldassari, Andrew D. White, and Philippe Schwaller. Augmenting large language models with chemistry tools. *Nature Machine Intelligence*, 6(5):525–535, May 2024. Publisher: Nature Publishing Group.

[7] Andres M. Bran, Theo A. Neukomm, Daniel P. Armstrong, Zlatko Jončev, and Philippe Schwaller. Chemical reasoning in LLMs unlocks steerable synthesis planning and reaction mechanism elucidation, March 2025. arXiv:2503.08537 [cs].

[8] Mike Butters. Route Design and Selection. In *Pharmaceutical Process Development - Current Chemical and Engineering Challenges*, pages 90–116. Royal Society of Chemistry (RSC), 2011.

[9] CAS. Scifinder.

[10] Binghong Chen, Chengtao Li, Hanjun Dai, and Le Song. Retro*: Learning retrosynthetic planning with neural guided a* search. In *The 37th International Conference on Machine Learning (ICML 2020)*, 2020.

[11] Ziqi Chen, Oluwatosin R. Ayinde, James R. Fuchs, Huan Sun, and Xia Ning. G2Retro as a two-step graph generative models for retrosynthesis prediction. *Communications Chemistry*, 6(1):102, May 2023.

[12] E.J. Corey and Xue-Min Cheng. *The Logic of Chemical Synthesis*. Wiley, 1989.

[13] Rémi Coulom. Efficient selectivity and backup operators in Monte-Carlo tree search. In *Proceedings of the 5th international conference on Computers and games*, CG'06, pages 72–83, Berlin, Heidelberg, May 2006. Springer-Verlag.

[14] Sean Current, Ziqi Chen, Daniel Adu-Ampratwum, Xia Ning, and Srinivasan Parthasarathy. DiffER: categorical diffusion ensembles for single-step chemical retrosynthesis. *Journal of Cheminformatics*, 17(1):112, July 2025.

[15] DeepSeek-AI, Daya Guo, Dejian Yang, Haowei Zhang, Junxiao Song, Ruoyu Zhang, et al. DeepSeek-R1: Incentivizing Reasoning Capability in LLMs via Reinforcement Learning, January 2025. arXiv:2501.12948 [cs].

[16] Elsevier Limited. Reaxys.

[17] Samuel Genheden, Amol Thakkar, Veronika Chadimová, Jean-Louis Reymond, Ola Engkvist, and Esben Bjerrum. AiZynthFinder: a fast, robust and flexible open-source software for retrosynthetic planning. *Journal of Cheminformatics*, 12(1):70, November 2020.

[18] Gibson, Jack R. and Weber, Joanne D. Handbook of Selected Properties of Air- and Water-Reactive Materials. Technical Report 144, Defense Technical Information Center, Crane, Indiana, March 1969.

[19] Juraj Gottweis, Wei-Hung Weng, Alexander Daryin, Tao Tu, Anil Palepu, Petar Sirkovic, et al. Towards an AI co-scientist, February 2025. arXiv:2502.18864 [cs].

[20] Jeff Guo and Philippe Schwaller. It Takes Two to Tango: Directly Optimizing for Constrained

Synthesizability in Generative Molecular Design, October 2024. arXiv:2410.11527 [q-bio].

[21] Peter E. Hart, Nils J. Nilsson, and Bertram Raphael. A Formal Basis for the Heuristic Determination of Minimum Cost Paths. *IEEE Transactions on Systems Science and Cybernetics*, 4(2):100–107, July 1968.

[22] Kexin Huang, Serena Zhang, Hanchen Wang, Yuanhao Qu, Yingzhou Lu, Yusuf Roohani, and others. Biomni: A General-Purpose Biomedical AI Agent. *bioRxiv*, pages 2025–05, 2025. Publisher: Cold Spring Harbor Laboratory.

[23] Yuqing Huang, Rongyang Zhang, Xuesong He, Xuyang Zhi, Hao Wang, Xin Li, et al. ChemEval: A Comprehensive Multi-Level Chemical Evaluation for Large Language Models, September 2024. arXiv:2409.13989 [cs].

[24] IARC. IARC monographs on the identification of carcinogenic hazards to humans. Technical report, IARC, 2025.

[25] Ruofan Jin, Zaixi Zhang, Mengdi Wang, and Le Cong. STELLA: Self-Evolving LLM Agent for Biomedical Research, July 2025. arXiv:2507.02004 [cs].

[26] Jared Kaplan, Sam McCandlish, Tom Henighan, Tom B. Brown, Benjamin Chess, Rewon Child, Scott Gray, Alec Radford, Jeffrey Wu, and Dario Amodei. Scaling Laws for Neural Language Models, January 2020. arXiv:2001.08361 [cs].

[27] Junsu Kim, Sungsoo Ahn, Hankook Lee, and Jinwoo Shin. Self-Improved Retrosynthetic Planning. In *Proceedings of the 38th International Conference on Machine Learning*, pages 5486–5495. PMLR, July 2021. ISSN: 2640-3498.

[28] T. L Lai and Herbert Robbins. Asymptotically efficient adaptive allocation rules. *Advances in Applied Mathematics*, 6(1):4–22, March 1985.

[29] Michael S. Lajiness, Gerald M. Maggiora, and Veerabahu Shanmugasundaram. Assessment of the Consistency of Medicinal Chemists in Reviewing Sets of Compounds. *Journal of Medicinal Chemistry*, 47(20):4891–4896, September 2004.

[30] Daniel Lowe. Chemical reactions from US patents (1976-Sep2016), 2017. Artwork Size: 1494665893 Bytes Pages: 1494665893 Bytes.

[31] Andrew D. McNaughton, Gautham Krishna Sankar Ramalaxmi, Agustin Kruel, Carter R. Knutson, Rohith A. Varikoti, and Neeraj Kumar. CACTUS: Chemistry Agent Connecting Tool Usage to Science. *ACS Omega*, 9(46):46563–46573, November 2024. Publisher: American Chemical Society.

[32] Mistral AI. Mistral NeMo, July 2024.

[33] H. L. Morgan. The Generation of a Unique Machine Description for Chemical Structures-A Technique Developed at Chemical Abstracts Service. *Journal of Chemical Documentation*, 5(2):107–113, May 1965.

[34] OpenAI, Aaron Hurst, Adam Lerer, Adam P. Goucher, Adam Perelman, Aditya Ramesh, et al. GPT-4o System Card, October 2024. arXiv:2410.21276 [cs].

[35] Robert J. Ouellette and J. David Rawn. Aldehydes and Ketones. In *Organic Chemistry*, pages 629–657. Elsevier, 2014.

[36] RDKit: Open-source cheminformatics.

[37] David Rein, Betty Li Hou, Asa Cooper Stickland, Jackson Petty, Richard Yuanzhe Pang, Julien Dirani, Julian Michael, and Samuel R. Bowman. GPQA: A Graduate-Level Google-Proof Q&A Benchmark. In *Proceedings of the Conference on Language Modeling (COLM)*, August 2024.

[38] Marwin H. S. Segler, Mike Preuss, and Mark P. Waller. Planning chemical syntheses with deep neural networks and symbolic AI. *Nature*, 555(7698):604–610, March 2018. Publisher: Nature Publishing Group.

[39] Kyle Swanson, Parker Walther, Jeremy Leitz, Souhrid Mukherjee, Joseph C Wu, Rabindra V Shivnaraine, et al. ADMET-AI: a machine learning ADMET platform for evaluation of large-scale chemical libraries. *Bioinformatics*, 40(7):btae416, July 2024.

[40] Yuji Takaoka, Yutaka Endo, Susumu Yamanobe, Hiroyuki Kakinuma, Taketoshi Okubo, Youichi

Shimazaki, et al. Development of a Method for Evaluating Drug-Likeness and Ease of Synthesis Using a Data Set in Which Compounds Are Assigned Scores Based on Chemists' Intuition. *Journal of Chemical Information and Computer Sciences*, 43(4):1269–1275, July 2003.

[41] UNECE. Globally Harmonized System of Classification and Labelling of Chemicals (GHS Rev. 10, 2023), 2023.

[42] Rahul Verma, Yaru Jing, Honghu Liu, Varun Aggarwal, Harish Kumar Goswami, Ekta Bala, et al. Employing Ammonia for Diverse Amination Reactions: Recent Developments of Abundantly Available and Challenging Nitrogen Sources. *European Journal of Organic Chemistry*, 2022(25), July 2022. Publisher: Wiley.

[43] Yubo Wang, Xueguang Ma, Ge Zhang, Yuansheng Ni, Abhranil Chandra, Shiguang Guo, et al. MMLU-Pro: A More Robust and Challenging Multi-Task Language Understanding Benchmark. *Advances in Neural Information Processing Systems*, 37:95266–95290, December 2024.

[44] Jason Wei, Xuezhi Wang, Dale Schuurmans, Maarten Bosma, brian ichter, Fei Xia, Ed H. Chi, Quoc V Le, and Denny Zhou. Chain of thought prompting elicits reasoning in large language models. In Alice H. Oh, Alekh Agarwal, Danielle Belgrave, and Kyunghyun Cho, editors, *Advances in Neural Information Processing Systems*, 2022.

[45] Botao Yu, Frazier N. Baker, Ziqi Chen, Xia Ning, and Huan Sun. LlaSMol: Advancing Large Language Models for Chemistry with a Large-Scale, Comprehensive, High-Quality Instruction Tuning Dataset. In *Proceedings of the Conference on Language Modeling (COLM)*, August 2024.

[46] Botao Yu, Frazier N. Baker, Ziru Chen, Garrett Herb, Boyu Gou, Daniel Adu-Ampratwum, Xia Ning, and Huan Sun. ChemToolAgent: The Impact of Tools on Language Agents for Chemistry Problem Solving. In *Findings of the Association for Computational Linguistics: NAACL 2025*, 2025.

[47] Kevin Yu, Jihye Roh, Ziang Li, Wenhao Gao, Runzhong Wang, and Connor W. Coley. Double-Ended Synthesis Planning with Goal-Constrained Bidirectional Search. *Advances in Neural Information Processing Systems*, 37:112919–112949, December 2024.

[48] Mert Yuksekgonul, Federico Bianchi, Joseph Boen, Sheng Liu, Pan Lu, Zhi Huang, et al. Optimizing generative AI by backpropagating language model feedback. *Nature*, 639(8055):609–616, March 2025. Publisher: Nature Publishing Group.

[49] Dengwei Zhao, Shikui Tu, and Lei Xu. Efficient retrosynthetic planning with MCTS exploration enhanced A* search. *Commun Chem*, 7(1):1–12, March 2024. Publisher: Nature Publishing Group.

[50] Mingchen Zhuge, Changsheng Zhao, Dylan Ashley, Wenyi Wang, Dmitrii Khizbullin, Yunyang Xiong, et al. Agent-as-a-Judge: Evaluate Agents with Agents, October 2024. arXiv:2410.10934 [cs].

A Dataset

Table A1 shows the benchmark dataset curated for our experiments. Tasks are provided as natural language prompts, organized by constraint type.

Table A1: Constrained Retrosynthesis Planning Dataset

ID	Constraint Type	Task
C1	Carcinogen	Find the shortest synthesis route for `O=S(=O)(C#Cc1ccc(Cl)cc1)N1CCNCC1` that avoids all carcinogenic substances.

Continued on next page

ID	Constraint Type	Task
C2	Carcinogen	Identify a synthesis route for CC(C)(C)OC(=O)NC1(c2nc(O)c3c(Cl)ccn3n2)CC1 that avoids known or suspected carcinogens.
C3	Carcinogen	Plan the synthesis of CC(=O)OCc1nc2cnc(Br)cc2n1C(C)(C)COC(C)=O, avoiding any carcinogenic substances.
C4	Carcinogen	Plan the synthesis of FC(F)(F)Cn1ncnc1-c1cc2n(n1)-c1cc(C3CCNCC3)ccc1OCC2, avoiding any known or suspected carcinogens.
C5	Carcinogen	Perform synthesis planning for C=C(C[C@@H](Cc1ccc(-c2ccccc2)cc1)NC(=O)OC(C)(C)C)C(=O)O without using any carcinogenic substances.
C6	Carcinogen	Provide a synthesis route for COc1cc2c(Oc3cc(C)c(C)nc3-c3cccc(C)n3)ccnc2cc1OCCNCCO that avoids carcinogens.
C7	Carcinogen	Find the shortest synthesis route possible for C#CC1(O)C(C)=CC2(CC1(C)C(F)(F)F)OC(C)C(C)O2 without using carcinogens.
C8	Carcinogen	Identify the shortest possible synthetic route for ClCc1ccc2c(c1)Nc1nccnc1S2 that avoids carcinogens.
C9	Carcinogen	Identify the best synthesis route for COC(=O)c1ccc2c(c1)C=CC(=C(Cl)Cl)CO2 that avoids carcinogenic materials.
C10	Carcinogen	Design a synthesis plan for C[C@@H](O)c1nc2cnc3ccsc3c2n1[C@H]1CC[C@H](CO)CC1 without using any carcinogens.
C11	Carcinogen	Provide the shortest synthesis route for Cn1oc(=O)nc1/C(=N\\OCc1cccc(N)n1)c1ccccc1 that does not use any carcinogens.
C12	Carcinogen	Plan the synthesis of CC(C)(CO)n1c(CO)nc2cnc(Br)cc21 without using carcinogens.
C13	Carcinogen	Plan the synthesis of CCCC[Sn](/C=C/C1(O)C(C)=CC2(CC1(C)C(F)(F)F)OC(C)C(C)O2)(CCCC)CCCC, but do not use any carcinogens in your synthesis route.
C14	Carcinogen	Find the shortest synthesis path for COC(=O)CCc1cc2cc(-c3noc(-c4ccc(OC(C)C)c(Cl)c4)n3)ccc2n1C that does not use any carcinogenic substances.
C15	Carcinogen	Plan the synthesis of CC(C)(C)OC(=O)N1CC=C(c2ccc3c(c2)-n2nc(-c4ncnn4CC(F)(F)F)cc2CCO3)CC1 without using any carcinogens.
C16	Carcinogen	Find the shortest synthesis route for C[Si](C)(C)CCOCn1cc(C2CCc3c(C(=O)O)nn(COCC[Si](C)(C)C)c3C2)cn1 that doesn't use any carcinogenic substances.
C17	Carcinogen	Plan the synthesis of O=C(OCc1ccccc1)N1CC[C@H]2CCCN(CCc3ccccc3)C[C@H]21 without using carcinogens.
C18	Carcinogen	Find a synthesis route for Cn1oc(=O)nc1/C(=N\OCc1cccc(NC(=O)OCCc2ccccc2)n1)c1ccccc1 that avoids all known or suspected carcinogens.
C19	Carcinogen	Plan the shortest synthesis route for CC(C)(C)OC(=O)NC1(c2nc(NCc3ccccn3)c3c(Cl)ccn3n2)CC1. Do not use any carcinogens in the route.

Continued on next page

ID	Constraint Type	Task
C20	Carcinogen	Generate a synthesis plan for the following compound without using any carcinogens: CC[C@@H](OC(=O)c1ccccc1)[C@H]1CCCN(C(=O)OC(C)(C)C)C1
C21	Carcinogen	Plan the synthesis of CC(C)(C)OC(=O)N[C@@H]1c2cccnc2[C@H](O)CC[C@H]1c1cccc(F)c1F. Avoid carcinogenic materials.
C22	Carcinogen	Find the shortest synthesis route that doesn't use carcinogens for the following product: CC(C)(C)OC(=O)N[C@@H]1c2cccnc2[C@H](N)CC[C@H]1c1cccc(F)c1F.
C23	Carcinogen	Design a synthesis path for CC(C)(C)OC(=O)N[C@@H]1c2cccnc2C(=O)CC[C@H]1c1cccc(F)c1F. Do not use any carcinogens.
C24	Carcinogen	Plan the shortest possible synthesis route for COCCCc1cc(CN(C(=O)[C@H]2CNCC[C@@H]2c2ccc(OCCOc3c(Cl)cc(C)cc3Cl)cc2)C2CC2)cc(OCCOC)c1, but do not use any carcinogens in the route.
C25	Carcinogen	Find the shortest synthesis path for COCCCc1cc(CN(C(=O)[C@H]2CN(C(=O)OC(C)(C)C)CC[C@@H]2c2ccc(OCCOc3c(Cl)cc(C)cc3Cl)cc2)C2CC2)cc(OCCOC)c1. Do not use any carcinogens.
C26	Carcinogen	Generate a synthesis plan without carcinogens for CC#CCn1c(Br)nc(C=O)c1C(=O)OC.
C27	Carcinogen	Find the shortest synthesis path for COC(=O)c1ccc2c(c1)C=CC(=CCl)CO2 that does not use carcinogenic substances.
C28	Carcinogen	Find a synthesis path for O=C(Nc1cccc(Cl)c1)N1CCc2[nH]nc(C(=O)N3CC(F)CO3)c2C1 that doesn't use carcinogens.
P1	Pyrophoric	Identify a synthesis route for C[C@H](O[Si](C)(C)C(C)(C)C)[C@@H]1CC(=O)CC(C)(C)N1 that does not use pyrophoric substances.
P2	Pyrophoric	Find the shortest synthesis plan for C[C@H](c1ccccc1)N1C[C@]2(C(=O)OC(C)(C)C)C=CC[C@@H]2C1=S that avoids all pyrophoric and water-reactive substances.
P3	Pyrophoric	Plan the shortest synthesis route for CC(=O)c1ccc2c(c1)C=CC(O)(CO)CO2 without using any pyrophoric or water reactive substances.
P4	Pyrophoric	Synthesize CC[C@@H](OC(=O)c1ccccc1)[C@H]1CCCN(C(=O)OC(C)(C)C)C1 without using any pyrophoric or water-reactive substances.
P5	Pyrophoric	Perform synthesis planning for CC1=NC2(N=C1N)c1cc(Br)ccc1CCC21CC1, avoiding pyrophoric materials (substances that ignite in moisture or air) in your synthesis route.
P6	Pyrophoric	Plan a synthesis route for COc1cc2ncc3c(N)nc(-c4cncc(OCCN(Cc5ccc(F)cc5)C(=O)OC(C)(C)C)c4)cc3c2cc1OC that uses no pyrophoric or water-reactive substances.

Continued on next page

ID	Constraint Type	Task
P7	Pyrophoric	Perform synthesis planning for O[C@H]1C[C@H](c2cnn3c(N[C@H]4CCc5ccccc54)ncnc23)C=C1COCc1ccccc1 without using pyrophoric substances in your synthesis plan.
P8	Pyrophoric	Synthesize CC(=O)N1c2ccc(N3CCNCC3)cc2[C@H](Nc2ccccc2)[C@@H](C)[C@@H]1C without using any pyrophoric or water-reactive reagents.
P9	Pyrophoric	Find the shortest synthesis route for COc1cc2c(=O)[nH]c(=O)n([C@@H]3O[C@H](CO)[C@H]4OC(C)(C)O[C@H]43)c2cc1OC, avoiding pyrophoric substances.
P10	Pyrophoric	Identify a synthesis route for Oc1ccc2c3c(ccc2c1)Cc1ccccc1OC3c1ccc(OCCN2CCCCC2)cc1 that does not use pyrophoric substances.
P11	Pyrophoric	Plan the synthesis of C[C@@H](O)C[C@H]1OC[C@@H](C2CCCCC2)N(c2cc(C#CC(C)(C)C)sc2C(=O)O)C1=O. Do not use any pyrophoric substances.
P12	Pyrophoric	Find the shortest synthesis route for OC[C@H]1C[C@@H](c2cnn3c(N[C@H]4CCc5ccccc54)ncnc23)C[C@@H]1O that avoids using any pyrophoric substances.
S1	User-Specified	Find the shortest synthesis path for C[C@@H]1CCCN1CCc1nnc2cc(Br)ccc2c1O, but avoid using C=C[Sn](CCCC)(CCCC)CCCC in your synthesis route.
S2	User-Specified	Plan a synthesis route for CC(=O)NC[C@H]1CN(c2ccc3c(c2)CCCc2c(C(C)C)n[nH]c2-3)C(=O)O1. Avoid using phosgene (O=C(Cl)Cl) in your synthesis.
S3	User-Specified	Find the shortest synthesis route for C[C@@H]1CNC(=O)c2cc3cc(OCCCN4CCCCC4)ccc3n21 that does not use trimethyl borate (COB(OC)OC).
S4	User-Specified	Plan the synthesis of COCCCc1cc(CN(C(=O)[C@H]2CNCC[C@@H]2c2ccc(OCCOc3c(Cl)cc(C)cc3Cl)cc2)C2CC2)cc(OCCOC)c1 without using hexane (CCCCCC).
S5	User-Specified	Identify a synthesis plan for COc1cc2c(=O)[nH]c(=O)n([C@@H]3O[C@H](CO)[C@H]4OC(C)(C)O[C@H]43)c2cc1OC that does not use methanol (CO).
S6	User-Specified	Plan the synthesis of CC(C)c1ccc2c(c1)OC1(O)c3ccccc3C(=O)C21NC(=O)c1cc(-c2ccccc2)n[nH]1. Do not use phenol (Oc1ccccc1).
S7	User-Specified	Synthesize CCOC(=O)/C(N)=N/Nc1cc(Cl)ccc1[N+](=O)[O-] without using nitric acid (O=[N+]([O-])O).
S8	User-Specified	Find the shortest synthesis route for C#CC1(O)C(C)=CC2(CC1(C)C(F)(F)F)OC(C)C(C)O2 that does not use toluene (Cc1ccccc1).

B LARC System Instructions

Here we present the system instructions for implementing the Evaluator in $\text{LARC}_{\text{Mistral}}$ and $\text{LARC}_{\text{Claude}}$. Each set of system instructions was carefully engineered to elicit good instruction following from Mistral Nemo and Claude 3.5 Sonnet, respectively.

B.1 LARC$_{\text{Mistral}}$ System Instructions

The following system instructions are used to implement the EVALUATOR in LARC$_{\text{Mistral}}$. Values surrounded by curly braces {} are replaced with task-specific information. Please note, reaction evaluation is broken into two separate instructions: one to call tools and another to generate a score based on the results.

Evaluation planning. This set of instructions instructs MISTRAL NEMO to perform evaluation planning, generating the evaluation instructions.

You are an expert chemist evaluating a reaction. Your task is to use tools to determine if the user's provided reaction SMILES (reactants>>product) satisfies only the following constraint:

To help you evaluate the constraint, you may use any of the following tools:

- AIExpert(question): Asks a self-contained question of an AI chemistry expert to get information not available from the other tools or infer reaction conditions (e.g. catalysts, temperature, etc.).
- Carcinogenicity(SMILES1, SMILES2, ...): returns a probabilistic score for each molecule indicating whether the provided SMILES represents a carcinogenic molecule. Can be called with one or more molecules; batching improves efficiency.
- Pyrophoricity(SMILES1, SMILES2, ...): returns a score for each molecule indicating the Tanimoto similarity between the provided SMILES and its nearest neighbor on an authoritative list of pyrophoric substances (materials that spontaneously ignite in air or water). A score of 1.000 indicates the substance is on the list. Can be called with one or more molecules. The term "pyrophoric" includes water-reactive substances, as these may ignite in air due to the presence of water vapor.
- Similarity(SMILES1, SMILES2): returns a similarity between exactly 2 molecules, represented by SMILES.

Create a plan of the actions you would take to evaluate this specific constraint for any user-provided reaction. Use as few actions as possible to evaluate the constraint, it is acceptable to use a single action. Explicitly identify actions taken using the following template:

```
ActionName(comma-separated action inputs)
ActionName(comma-separated action inputs)
...
```

Action names must be one of: AIExpert, Carcinogenicity, Pyrophoricity, Similarity.

In place of actual inputs, place descriptions of the values you would use. Be efficient and use only the tools necessary. Then, create a rubric for assigning a score of 1-5 where 5 is assigned if the constraint is completely satisfied and 1 is assigned if the constraint is completely violated.

Constraint: {constraint}

Reaction evaluation 1. This set of instructions instructs MISTRAL NEMO to follow the evaluation instructions to select tools to evaluate a specific reaction. It uses the messages from evaluation planning (including the generated evaluation instructions) as context.

{evaluation planning context}

User-provided reaction:
Reactants: {reactants}
Product: `{product}`

When multiple reactants are present, they are separated by dots.
Fill in your plan with the specific inputs for this reaction. Format your actions using the following format, one per line, with inputs wrapped in quotes:

```
ActionName(ActionInput1, ActionInput2, ...)
ActionName(ActionInput1, ActionInput2, ...)
...
```

Do not include any other information. Await the results.

Reaction evaluation 2. This set of instructions instructs Mistral Nemo to generate a score between 1 and 5. It uses the evaluation planning messages, the messages from Reaction Evaluation 1 and the tool outputs as context.

{evaluation planning context}
{reaction evaluation context}
{tool outputs}
Analyze these results and provide your final score. Use the following format:
Reasoning: (your analysis)
Answer: N
where N is between 1 and 5.

B.2 LARC$_{\text{Claude}}$ System Instructions

The following system instructions are used to implement the Evaluator in LARC$_{\text{Claude}}$. Values surrounded by curly braces {} are replaced with task-specific information. Please note, reaction evaluation is broken into two separate instructions: one to call tools and another to generate a score based on the results.

Evaluation planning. This set of instructions instructs Claude 3.5 Sonnet to perform evaluation planning, generating the evaluation instructions.

You are an expert chemist assisting in retrosynthesis planning. You aim to determine if the user's provided reaction (reactants>>product) satisfies the following constraint:
{constraint}
Note that there may be reagents omitted from the user's provided reaction.
To help you answer the user question, you have access to these tools:

- Carcinogenicity(SMILES1, SMILES2, ...): returns a probabilistic score for each molecule indicating whether the provided SMILES represents a carcinogenic molecule. Can be called with one or more molecules; batching improves efficiency.
- Pyrophoricity(SMILES1, SMILES2, ...): returns a score for each molecule indicating the Tanimoto similarity between the provided SMILES and its nearest neighbor on an authoritative list of pyrophoric substances (materials that spontaneously ignite in air or water). A score of 1.000 indicates the substance is on the list. Can be called with one or more molecules. The term "pyrophoric" includes water-reactive substances, as these may ignite in air due to the presence of water vapor.
- Similarity(SMILES1, SMILES2): returns a similarity between exactly 2 molecules, represented by SMILES.
- AIExpert(question): Asks a self-contained question of an AI chemistry expert, for instance,

to infer reaction conditions or reagents not listed in the reaction, or to get information not available from the other tools. Include all relevant details, including reaction information and SMILES strings.

- Answer(value): Answer with a score of 1-5 based on only the constraint above is satisfied, where 5 is completely satisfied and 1 is not at all satisfied. Keep in mind these scores will be used to prioritize further planning.

Create a plan of the actions you would take to evaluate this specific constraint for any user-provided reaction. Explicitly identify actions taken using the following template:

Action: ActionName(ActionInput)

In place of actual inputs, place descriptions of the values you would use. Use only the tools necessary. Think step by step.

Reaction evaluation 1. This set of instructions instructs Claude 3.5 Sonnet to follow the evaluation instructions to select tools to evaluate a specific reaction.

You are an expert chemist assisting in retrosynthesis planning. You aim to determine if the user's provided reaction (reactants>>product) satisfies the following constraint:

{constraint}

To help you answer the user question, you have access to these tools:

- Carcinogenicity(SMILES1, SMILES2, ...): returns a probabilistic score for each molecule indicating whether the provided SMILES represents a carcinogenic molecule. Can be called with one or more molecules; batching improves efficiency.
- Pyrophoricity(SMILES1, SMILES2, ...): returns a score for each molecule indicating the Tanimoto similarity between the provided SMILES and its nearest neighbor on an authoritative list of pyrophoric substances (materials that spontaneously ignite in air or water). A score of 1.000 indicates the substance is on the list. Can be called with one or more molecules. The term "pyrophoric" includes water-reactive substances, as these may ignite in air due to the presence of water vapor.
- Similarity(SMILES1, SMILES2): returns a similarity between exactly 2 molecules, represented by SMILES.
- Answer(value): Answer with a score of 1-5 based on only the constraint above is satisfied, where 5 is completely satisfied and 1 is not at all satisfied. Keep in mind these scores will be used to prioritize further planning.

In one response, specify all of the actions you would take to gather the necessary information to evaluate the user's reaction. Follow this plan:

{evaluation instructions}

Follow the plan step by step. Replace all references to the AIExpert tool with your own expert chemistry knowledge on the user's reaction. Keep in mind there may be multiple correct answers to each question (e.g. multiple ways to catalyze a reaction), and you should discuss every possibility in detail. Reactants and products provided by the user should be considered required for the reaction, changing these would be considered a different reaction. Provide accurate and valid SMILES representations for molecules in your answer.

For the tools, construct a single, unified code block wrapped in triple backticks at the end of your response, including specific inputs based on the user's provided reaction. Ignore the Answer tool until you have results from your code block actions. Wrap all SMILES in backticks.

Reaction evaluation 2. This set of instructions instructs Claude 3.5 Sonnet to generate a score between 1 and 5. It uses the messages from Reaction Evaluation 1 and the tool outputs as context.

{reaction evaluation context}
{tool outputs}
Given this information, please provide a final score for the reaction using ```Answer(X)```, where X is your score between 1-5 on only this constraint:
{constraint}
Do not worry about any other constraints, as we may assess these separately.

C Additional Related Work

LLM agents for chemistry. LLM agents have recently shown great promise in chemistry applications. CoScientist [5] and ChemCrow [6], are notable examples, combining LLM reasoning with cheminformatics tools, web search, and robotic laboratory equipment to perform chemistry tasks, including retrosynthesis planning. However, neither of these methods are well-equipped for constrained retrosynthesis planning. CoScientist relies on general-purpose LLMs for retrosynthesis planning, which lack accuracy on chemical reaction tasks compared to models with specialized training [45]. ChemCrow generates synthetic routes using an external tool for unconstrained retrosynthesis planning, making it incapable of incorporating constraints directly into the planning process. CACTUS [31] and ChemToolAgent [46] are similar to ChemCrow, but with different focuses. CACTUS focuses only on molecule property prediction with in silico tools, and ChemToolAgent focuses on analyzing the impact of tool use across various chemistry tasks. [19] introduces an LLM agent co-scientist similar to CoScientist, but designed for broad applicability across multiple scientific disciplines, not specifically for chemistry or retrosynthesis planning. LIDDiA [2] is an LLM agent for in silico drug discovery that automates generation, screening, and optimization of molecules, but it does not consider retrosynthesis planning. While these efforts demonstrate the promise of LLM agents in chemistry applications, none of the existing LLM agents can perform constrained retrosynthesis planning.

Additional LLM-based efforts, such as TextGrad [48], STELLA [25], and BIOMNI [22], have also been used in biomedical and chemistry applications; however, none of these methods are equipped for retrosynthesis planning tasks.

D Implementation Details for Reproducibility

For the experiments in this paper, we used the following hyperparameters and settings. To control cost and runtime, we enforced a limit on the number of expansions and evaluations. The search terminates after 500 expansions, following the established convention in multi-step retrosynthesis literature [10, 49, 27]. Additionally, after 300 evaluations, all subsequent evaluations were replaced with the optimistic default score (i.e. 5 on the 1-5 scale). To unify scaling, the raw values from MEEA*'s V_{A^*} are min-max normalized to a $[0, 1]$ scale at each selection step. The MEEA* simulation step already included min-max scaling, so no change was required for V_{MCTS}. Additionally, the evaluator scores $S(r)$ are each normalized to a $[0, 1]$ scale, where 0 indicates complete constraint violation and 1 indicates complete constraint satisfaction. For hyperparameters, we used $\lambda = 2$ to ensure the constraint was followed. While λ can be tuned as needed in practice, we observed during development that LARC$_{\text{Mistral}}$ had relatively high success rates for of $\lambda \in \{2, 3\}$, moderate success rates for $\lambda \in \{4, 5\}$, and low success rate with $\lambda = 1$. Additionally, planning time generally increased with larger λ, thus our choice of $\lambda = 2$. We set $K = 5$ to control the number of candidate routes subject to evaluation. Following the MEEA* paper, we used a UCB scaling term of 4.

Algorithm 1 LARC

Require: Target molecule p; constraints c; commercially available molecule set B;
unconstrained MCTS value function V_{MCTS}; unconstrained A* value function V_{A^*};
expansion function EXPAND;
expansion limit N_{exp}; evaluation limit N_{eval};
number of simulations K; constraint weight λ; default score S_{def}

Ensure: Synthetic route R_*

$n_{exp} \leftarrow 0$ ▷ Initialize expansion count
$T \leftarrow$ Tree(nodes={p}, edges={}) ▷ Initialize search tree
$S \leftarrow \{\}$ ▷ Initialize reaction evaluations

$P \leftarrow$ EVALUATOR-plan(c) ▷ Evaluation planning generates evaluation instructions P

while R_* is empty $\wedge$ $n_{exp} < N_{exp}$ **do**
  $\mathcal{C} \leftarrow$ SYNTHESIZER-MCTS$(T, V_{MCTS}, S, S_{\text{def}}, \lambda, K)$ ▷ Simulate to get candidate set $\mathcal{C}$

  for all $(m, R) \in \mathcal{C}$ **do**
    for all $r \in R$ **do**
      if $|S| < N_{eval} \wedge r \notin S$ **then**
        $S(r) \leftarrow$ EVALUATOR-evaluate(r, P) ▷ Reaction evaluation
      else if $r \notin S$ **then**
        $S(r) \leftarrow S_{\text{def}}$ ▷ Use default after evaluation limit
      end if
    end for
  end for

  $(m_{exp}, R_{exp}) \leftarrow$ SYNTHESIZER-A*$(\mathcal{C}, V_{\text{A}^*}, S, \lambda)$ ▷ Selection of (m_{exp}, R_{exp}) from $\mathcal{C}$

  $\mathcal{R} \leftarrow$ EXPAND(T, R_{exp}, m_{exp}) ▷ Expansion updates T with set of reactions $\mathcal{R}$
  $n_{exp} \leftarrow n_{exp} + 1$

  for all $r \in \mathcal{R}$ **do**
    if $R_{exp} \oplus r$ is a complete route from B to p **then** ▷ Check for complete route
      $R_* \leftarrow R_{exp} \oplus r$
    end if
  end for
end while

return R_*

The algorithm for LARC is presented in Algorithm 1, illustrating how LARC uses both the EVALUATOR and SYNTHESIZER to perform constrained retrosynthesis planning. LARC's EVALUATOR has two steps: evaluation planning (denoted EVALUATOR-plan in Algorithm 1) and reaction evaluation (denoted EVALUATOR-evaluate). Both of these steps are implemented using the system instructions. LARC's SYNTHESIZER adapts MEEA*'s two step process:

MCTS simulation (denoted SYNTHESIZER-MCTS) to generate K candidate routes, and A* search (denoted SYNTHESIZER-A*) to select the best of the K candidates. LARC's constrained adaptations of these steps are given in Algorithms 2 and 3, respectively. LARC's SYNTHESIZER uses the same expansion function (denoted EXPAND in Algorithm 1) as MEEA* [49]. Please note, these algorithms assume V_{A^*} and $S(r)$ are already normalized, per the implementation details above. A full implementation of LARC can be found at https://github.com/ninglab/LARC.

Algorithm 2 SYNTHESIZER-MCTS

Require: search tree T; unconstrained value function (includes UCB) V_{MCTS};
EVALUATOR scores S; default score S_{def} constraint weight λ; number of simulations K;

Ensure: expansion candidates $\mathcal{C} = \{(m, R), ...\}$ of intermediate-route pairs

$\mathcal{C} \leftarrow \{\}$
$V'_{MCTS}(m, R) := V_{MCTS}(m, R) + \lambda \sum_{r \in R} \big(S(r) \text{ if } r \in S \text{ else } S_{\text{def}}\big)$ ▷ Equation 1
for $k = 1$ to K **do**
  $R \leftarrow \langle\,\rangle$; $m \leftarrow \text{root}(T)$ ▷ start from target molecule and empty route
  $A \leftarrow$ reactions that produce m in T
  while $|A| > 0$ **do** ▷ Simulate route from reactions in T
    $r^*, m^* \leftarrow \underset{r \in A,\ m \in \text{reactants}(r)}{\arg\max} V'_{MCTS}(m, R \oplus r)$ ▷ select reaction
    $R \leftarrow R \oplus r^*$; $m \leftarrow m^*$ ▷ update route
    $A \leftarrow$ reactions that produce intermediate m in T
  end while
  $\mathcal{C} \leftarrow \mathcal{C} \cup \{(m, R)\}$ ▷ add candidate
end for
return $\mathcal{C}$

Algorithm 3 SYNTHESIZER-A*

Require: candidate routes $\mathcal{C}$; unconstrained value function V_{A^*};
EVALUATOR scores S; constraint weight λ;

Ensure: intermediate molecule m_*, route R_*

$\mathcal{C} \leftarrow \{\}$ ▷ Initialize empty candidate set
$V'_{\text{A}^*}(m, R) := V_{\text{A}^*}(m, R) + \lambda \sum_{r \in R} S(r)$ ▷ Equation 1
$m_*, R_* \leftarrow \underset{(m,R) \in \mathcal{C}}{\arg\max}\, V'_{\text{A}^*}(m, R)$ ▷ Select best candidate based on constrained A* policy
return m_*, R_*

E Additional Experimental Settings

Constrained retrosynthesis planning by LLMs. We use general-purpose LLMs, including Claude 3.5 Sonnet [1], GPT-4o [34], DeepSeek R1 [15], and Mistral Nemo [32] as the baselines for constrained retrosynthesis planning. Claude 3.5 Sonnet and GPT-4o are representative state-of-the-art closed-source LLMs, which have both shown promising results on chemistry knowledge benchmarks [43, 37, 46]. DeepSeek R1 serves as a representative of open-source reasoning LLMs, leveraging an internal chain of thought [44] process to improve its responses. Mistral Nemo serves as a representative of small, open-source LLMs, selected due to its instruction-following capabilities [32]. Note that baselines do not include models specifically designed for unconstrained retrosynthesis planning, such as MEEA. This is because the benchmark is constructed by selecting the tasks for which MEEA can generate valid routes but violate the constraints. As MEEA is the state of the art for unconstrained retrosynthesis planning, we believe other unconstrained retrosynthesis planning methods [10, 27, 17, 38] will not succeed on the benchmark data.

F Additional Results

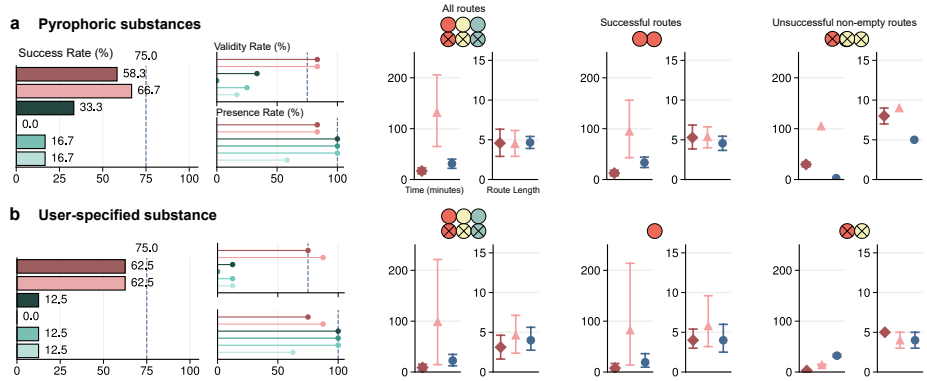


Figure F4: Evaluation **a,b** success rate, validity rate, presence rate, planning time, and route length for the benchmark tasks, organized by constraints for avoiding pyrophoric substances, and a user-specified substance, respectively. The symbols and colors in this figure follow the same conventions as Figure 2.

F.1 Pyrophoricity-constrained retrosynthesis planning

Figure F4**a** presents the results for retrosynthesis planning tasks that are constrained to avoid pyrophoric substances. $\text{LARC}_{\text{Mistral}}$ and $\text{LARC}_{\text{Claude}}$ continue to achieve high success rates of 58.3% and 66.7%, respectively. $\text{LARC}_{\text{Claude}}$ moderately underperforms Expert, who secures a success rate of 75.0%. Meanwhile, $\text{LARC}_{\text{Mistral}}$ is significantly faster, with 12.58 minutes on average to generate one successful route, compared to Expert (33.48 minutes) ($p=0.020$, two-sided two-sample t-test) and $\text{LARC}_{\text{Claude}}$ (94.81 minutes) ($p=0.026$). This suggests that, for retrosynthesis planning avoiding pyrophoricity, LARC is still behind human experts, but could be a decent option with satisfactory/acceptable success rates, considering its efficiency and automation. LARC can also serve as an alternative to human experts, and provide additional, different solutions, which human experts can further select, utilize, or optimize based on their domain knowledge.

The LLM baselines fall short on these tasks, with the best LLM baseline, CLAUDE 3.5 SONNET, achieving only 33.3% success rate. Pyrophoric substances can vary greatly in their composition and the underlying mechanism of pyrophoricity [18], demanding a nuanced understanding of chemical reactivity. These results demonstrate LARC's ability and efficiency to avoid a broad set of dangerous materials in its retrosynthesis planning compared to general-purpose LLMs.

F.2 User-specified constrained retrosynthesis planning

As Figure F4**b** shows, $\text{LARC}_{\text{Claude}}$ and $\text{LARC}_{\text{Mistral}}$ achieve the same success rate of 62.5% when planning 8 routes that must each avoid a single, user-specified substance. LLMs still struggle, with a very low success rate of 12.5%, due to the lack of ability to generate valid synthetic routes. On the other hand, LARC generalizes well to the highly diverse constraint types, fulfilling a key need in the dynamic context of retrosynthesis planning. EXPERT achieves a higher success rate of 75.0% with an average 22.47 minutes on generating one route. $\text{LARC}_{\text{Mistral}}$ is still faster on average than EXPERT (8.82 minutes), though the difference is not very significant (p=0.089, two-sided two-sample t-test).

F.3 Case Studies

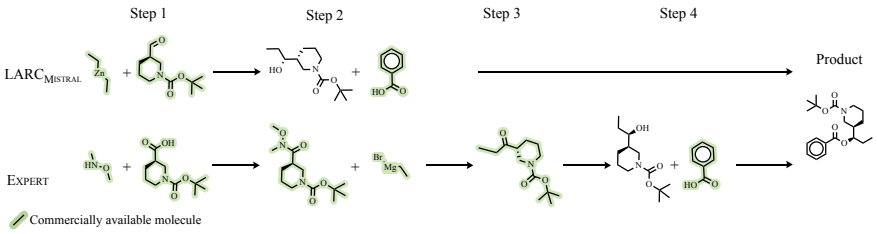


Figure F5: Synthesis route comparison of $\text{LARC}_{\text{Mistral}}$ against EXPERT tert-butyl (R)-3-((R)-1-(benzoyloxy)propyl)-piperidine-1-carboxylate.

F.3.1 $\text{LARC}_{\text{Mistral}}$ can generate better routes than human experts

Figure F5 compares routes from $\text{LARC}_{\text{Mistral}}$ and EXPERT for the synthesis of tert-butyl (R)-3-((R)-1-(benzoyloxy)propyl)-piperidine-1-carboxylate. $\text{LARC}_{\text{Mistral}}$ proposed a very concise two-step reaction to the product using commercially available reagents: diethyl zinc, tert-butyl (S)-3-formylpiperidine-1-carboxylate, and benzoic acid. Its Step 1 involves a stereoselective, nucleophilic addition of an ethyl group to tert-butyl (S)-3-formylpiperidine-1-carboxylate using the diethylzinc reagent. This results in an alcohol that can be converted to the product (an ester) using benzoic acid in Step 2.

In contrast, EXPERT proposed a four-step reaction to the same product. Step 1 involves conversion of commercially available (S)-1-(tert-butoxycarbonyl)piperidine-3-carboxylic acid to the Weinreb amide intermediate, tert-butyl (S)-3-(methoxy(methyl)carbamoyl)piperidine-1-carboxylate. This step was unnecessary, as its product is also commercially available. Step 2 is the Grignard addition of an ethyl group to obtain tert-butyl (S)-3-propionylpiperidine-1-carboxylate. Stereoselective reduction of the ketone in Step 3 results in the alcohol intermediate obtained in $\text{LARC}_{\text{Mistral}}$'s Step 1. A final ester formation, similar to that proposed by $\text{LARC}_{\text{Mistral}}$, yields the product.

One potential reason for Expert's longer route is the Expert's inherent bias towards the stability of some functional groups and their commercial availability. For example, it is a generally accepted notion that aldehydes are not very stable [35], so chemists tend to synthesize aldehydes and use them as needed. In most cases, an aldehyde equivalent such as the Weinreb amide, which was proposed in Step 1 by Expert, is used instead, followed by a reduction step when the product is an alcohol. Thus, as the Weinreb amide is stable and easily synthesizable, Expert may tend to propose a synthesis of this intermediate in the synthetic route. This may be another reason why Expert proposed this route, relying on their familiarity with this synthesis strategy rather than referring to the commercial availability of proposed intermediates. In contrast, $\text{LARC}_{\text{Mistral}}$ benefits from Synthesizer's thorough checks for commercial availability while simultaneously relying on feedback from the Evaluator to ensure the constraint is satisfied.

F.4 Study on tooling in LARC

For $\text{LARC}_{\text{Mistral}}$, tools are critical: across all 48 benchmark tasks, $\text{LARC}_{\text{Mistral}}$ achieves a success rate of 72.9% using tools in addition to Evaluator's internal knowledge, and 45.8% without tooling. Specifically, $\text{LARC}_{\text{Mistral}}$ with tooling generates 43 valid routes (validity rate of 89.6%), of which 35 are successful, and without tooling, 41 valid routes (validity rate of 85.4%), of which 22 are successful. This indicates that tooling primarily impacts LARC through deliberately assessing and enforcing constraint satisfaction to ensure successful routes. Moreover, with tooling, $\text{LARC}_{\text{Mistral}}$ generates successful routes faster (14.53 minutes on average) than without tooling (22.08 minutes), whereas in the latter case, Evaluator has to act as the AI expert and conduct reasoning via Mistral Nemo.

Evaluator's interpretation of results from different tools also significantly impacts LARC performance. For example, in pyrophoricity-constrained retrosynthesis planning, $\text{LARC}_{\text{Claude}}$ performs better than $\text{LARC}_{\text{Mistral}}$ (success rate of 66.7% vs 58.3%). One reason for this lies in how Claude 3.5 Sonnet- and Mistral Nemo-based Evaluator interprets and leverages evaluation results from the pyrophoricity predictor. Although with similar instructions, Claude 3.5 Sonnet considers a high value (e.g., close to 1.0) from the predictor as an indicator of high pyrophoricity, and thus, a low $S(r)$. This allows Synthesizer to explore more extensively to avoid pyrophoricity, and thus, a high success rate in the generated routes. However, $\text{LARC}_{\text{Mistral}}$ could interpret a relatively low value (e.g., 0.333) for high pyrophoricity, thus discouraging the exploration of routes that satisfy the constraint, resulting in more failed routes.

G Additional Discussion

Constrained retrosynthesis planning is highly challenging in the practice of synthetic chemistry, as it requires finding synthetic routes that not only lead to the target molecule but also satisfy additional, often complex, user-specified requirements. These constraints, such as limiting the number of steps, avoiding specific reagents or reaction types, or adhering to cost, safety, or environmental guidelines, can drastically narrow and fragment the feasible search space. Moreover, verifying constraint satisfaction is non-trivial: it may involve detailed reaction feasibility assessments, availability checks for intermediates, or compliance with regulatory and safety standards. Computational tools for such verifications can be resource-intensive to run, less reliable, or even unavailable, while manual verification can be very slow, biased, and inconsistent – it is not uncommon that chemists disagree with each other in retrosynthesis planning [40, 29].