ACCELERATING SCIENTIFIC DISCOVERY WITH AUTONOMOUS GOAL-EVOLVING AGENTS

Anonymous authors
Paper under double-blind review

ABSTRACT

There has been unprecedented interest in developing agents that expand the boundary of scientific discovery, primarily by optimizing quantitative objective functions specified by scientists. However, for grand challenges in science, these objectives may only be imperfect proxies. We argue that automating objective function design is a central, yet unmet need for scientific discovery agents. In this work, we introduce the Scientific Autonomous Goal-evolving Agent (SAGA) to address this challenge. SAGA employs a bi-level architecture in which an outer loop of LLM agents analyzes optimization outcomes, proposes new objectives, and converts them into computable scoring functions, while an inner loop performs solution optimization under the current objectives. This bi-level design enables systematic exploration of the space of objectives and their trade-offs, rather than treating them as fixed inputs. We demonstrate the framework in the context of functional DNA sequence design, showing that automating objective formulation can substantially improve the effectiveness of scientific discovery agents.

1 INTRODUCTION

Scientific discovery has been driven by human ingenuity through iterations of hypothesis, experimentation, and observation, but is increasingly bottlenecked by the vast space of potential solutions to explore and the high cost of experimental validation. Previous work has sought to address these challenges by developing optimization models that automatically find solutions maximizing a manually defined set of quantitative objectives, such as drug efficacy, protein expression, and material stability. These approaches, ranging from traditional generative models to more recent LLM-based methods, have demonstrated the ability to efficiently optimize against fixed objectives in domains including drug design and synthesis (Cavanagh et al., 2024; Loeffler et al., 2024; Sun et al., 2025), algorithm discovery (Novikov et al., 2025), and materials design (Zeni et al., 2025; Liu et al., 2025b).

However, these optimization models operate under a critical assumption: that the right set of objective functions is known upfront. In practice, this assumption is seldom known completely *a priori*. Just as scientific discovery requires iterations of hypothesis, experimentation, and observation, determining the appropriate objectives for a discovery task is itself an iterative search process. Scientists must constantly tweak objectives based on intermediate results, domain knowledge, and practical constraints that emerge during exploration (Figure 1(a)). This iterative refinement is particularly crucial in experimental disciplines such as drug discovery, materials design, and protein engineering, where many critical properties can only be approximated through predictive models. Without this evolving process, the discovery suffers from *reward hacking issues* (van den Broek et al., 2025): they exploit gaps between models and reality, producing solutions that maximize predicted scores while missing important practical considerations that experts would recognize. The search space for objectives and their relative weights is itself combinatorially large (Figure 1(b)), making it extremely difficult to specify the right objectives from the outset. As a result, while existing optimization models can solve the low-level optimization problem efficiently, scientific discovery remains bottlenecked by the high-level objective search process that relies on manual trial-and-error.

In this work, we introduce SAGA as our first concrete step toward automating this iterative objective evolving process. SAGA is designed to navigate the combinatorial search space of objectives by integrating high-level objective planning in the outer loop with low-level optimization in the inner loop (Figure 1(c)). The outer loop comprises four agentic modules: a planner that proposes new objec-

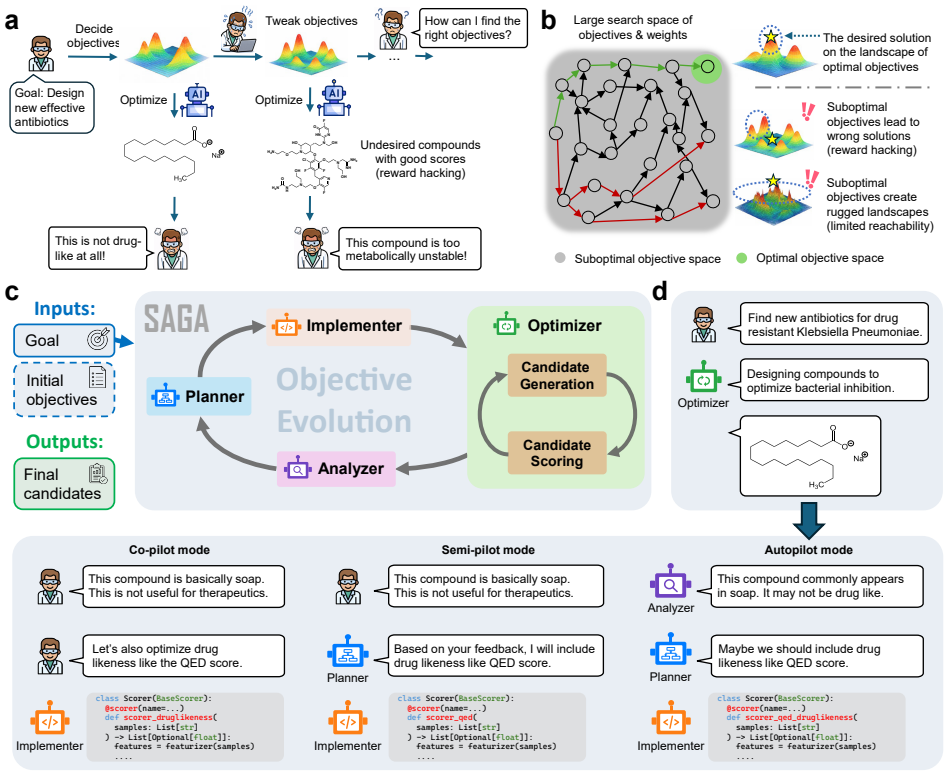

Figure 1: *The SAGA framework and the examples of scientific applications.* (a) Scientists constantly suffer from reward hacking issues, where optimization agents exploit the approximation error of objective functions and propose undesirable solutions with good scores. (b) Finding optimal objectives that bypass reward hacking issues is difficult due to the large search space of objectives and their relative weights. (c) We propose the SAGA framework to automatically discover optimal objectives and candidate solutions through a bi-level procedure. (d) SAGA operates at three different levels of automation, allowing scientists to steer the objective discovery process in various ways.

tives based on the task goal and current progress, an implementer that converts proposed objectives into executable scoring functions, an optimizer that searches for candidate solutions maximizing the specified objectives, and an analyzer that examines the optimization results and identifies areas for improvement. Within the optimizer module, an inner loop employs any optimization methods (e.g., genetic algorithms or reinforcement learning) to iteratively evolve candidate solutions toward the current objectives. Importantly, SAGA is a flexible framework supporting different levels of human involvement. It offers three modes (Figure 1(d)): (1) co-pilot mode, where scientists collaborate with both the planner and analyzer to reflect on results and determine new objectives; (2) semi-pilot mode, where scientists provide feedback only to the analyzer; and (3) autopilot mode, where both analysis and planning are fully automated. This design allows scientists to interact with SAGA in ways that best suit their expertise and preferences.

2 Demonstration: SAGA for Functional DNA Sequence Design

Programmed, highly precise, and cell-type-specific enhancers and promoters are fundamental to the development of reporter constructs, genetic therapeutics, and gene replacement strategies (Gosai et al., 2024). Such regulatory control is particularly important in HepG2, a human hepatocellular carcinoma cell line that retains key hepatic functions within a single cell type, including plasma protein synthesis and xenobiotic drug metabolism (Moyers et al., 2023). Although enhancers play a central role in establishing cell-type-specific gene expression programs (Andersson & Sandelin, 2020), their rational design remains challenging due to the vast combinatorial space of possible functional DNA sequences. This task can be naturally formulated as an optimization problem with

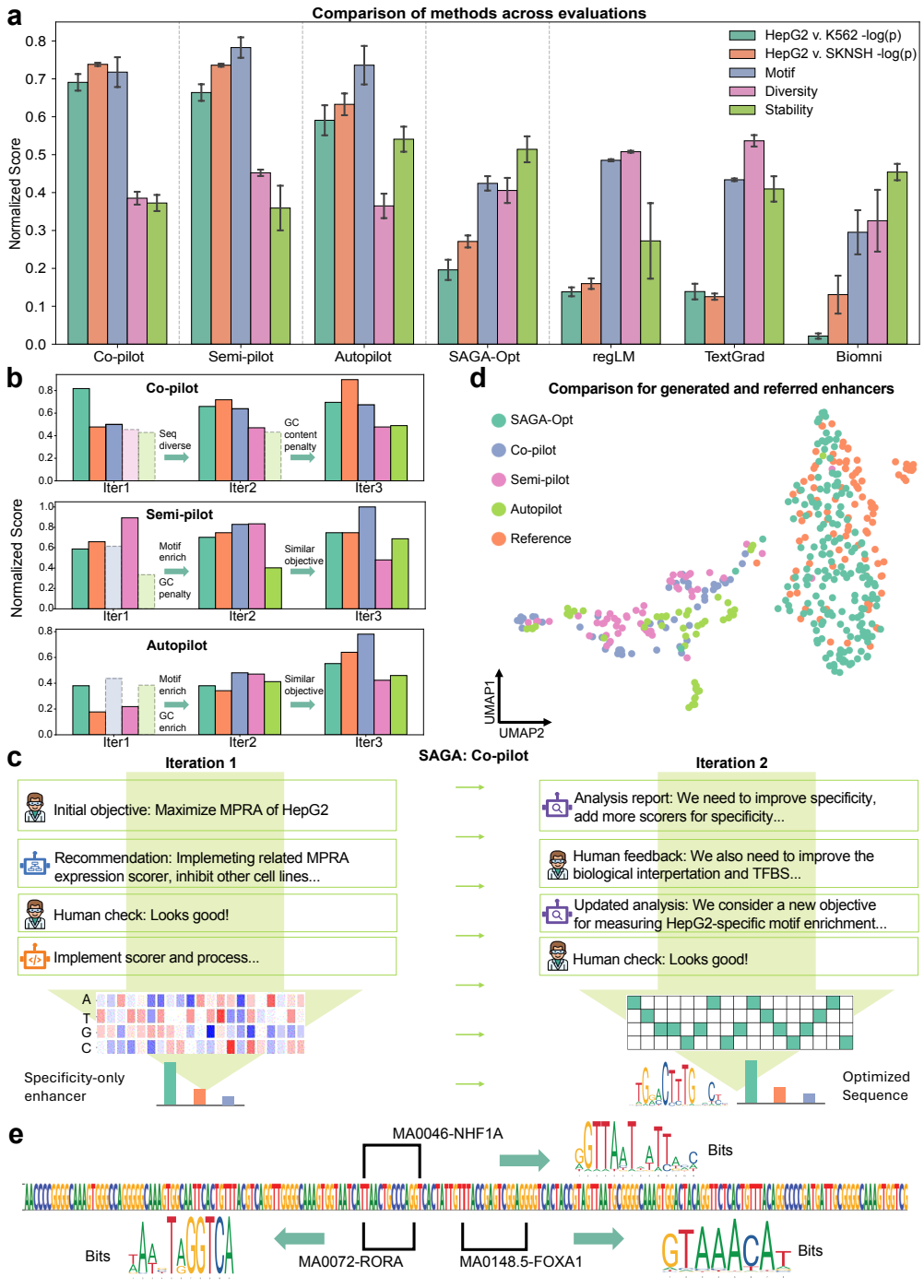

Figure 2: *Results for functional sequence design.* (a) Comparisons between different levels of SAGA and selected baselines with evaluation metrics (both average score (higher is better) and scaled standard error are reported (lower is better)). Our selected task is to design HepG2 (a cell line of epithelial-like cells from liver) specific enhancers. (b) The comparisons of different iterations of two different levels with the same held-out metrics. Each iteration will create new objectives. The solid line means objectives address the evaluation metrics, and the dash line means the metric has not been addressed. (c) An example of SAGA to correct the issues in previous iterations. (d) UMAP visualization of enhancers designed by SAGA, SAGA-Opt, and from references. (e) HepG2-specific motif visualization.

predefined oracle functions, such as DNA expression level predictor (Liu et al., 2025a). However, optimizing solely against expression-based oracles often results in sequences that generalize poorly with respect to biologically relevant constraints, including transcription factor motif enrichment, sequence diversity, and DNA stability. To address these limitations, we apply SAGA to discover novel cell-type-specific enhancers while iteratively refining the optimization objectives. Here, the SAGA framework is initialized using cell-type-specific expression measurements obtained from Massively Parallel Reporter Assays (MPRA) (Agarwal et al., 2025) and subsequently performs optimization with respect to an initial set of objectives. Crucially, SAGA closes the design loop by systematically analyzing deficiencies in the designed sequences and adaptively modifying the objective functions to guide subsequent exploration.

SAGA effectively discovers biologically plausible functional DNA sequences. We compare SAGA's discovery capabilities by benchmarking it against established domain-specific models and AI agents (Huang et al., 2025a; Yuksekgonul et al., 2025; Lal et al., 2024). Figure 2(a) reveals that our agents in different modes surpass selected baselines on metrics probing both statistical validity and biological function by 176.2% at most and 19.2% at least, based on an average comparison. Under controlled conditions where all baselines targeted the same objectives, our system exhibits marked improvements in MPRA specificity (by at least 48.0%), motif enrichment (by at least 47.9%), and sequence stability (by at least 1.7%). To further demonstrate the superiority of the multi-objective optimization method proposed by SAGA, we utilize the analyzer to examine the differences between enhancers produced by the Optimizer of SAGA with initial objectives only (SAGA-Opt) and SAGA. These results suggest that SAGA effectively captures the complex interplay between statistical likelihood and biological constraints.

SAGA proposes reasonable and helpful objectives to assist human scientists for enhancer design. As already shown for drug discovery and materials design, the inclusion of human feedback via Co-pilot and Semi-pilot leads to marked improvements in biologically meaningful outcomes for DNA enhancer design as illustrated in Figure 2(b). For example, explicitly prioritizing transcription factor motif enrichment and sequence stability can be guided by expert input (Figure 2(c)), which results in enhanced biological validity of the designed sequences. SAGA's Autopilot mode can also fully and automatically design enhancers, achieving overall performance comparable to that obtained with human intervention, particularly with respect to HepG2 specificity and improvements in sequence diversity and stability, and consistently outperforms other fully automated AI-agent baselines across all evaluated metrics (Figures 2(a) and (b)).

SAGA uncovers both novel enhancer candidates and known biological patterns. As shown in Figure 2(d), we compare the distributions from SAGA and SAGA-Opt with sampled HepG2-specific enhancers from a known experimental pool (Gosai et al., 2024). The enhancers discovered by SAGA exhibit distinctly different distributions, and given their outstanding performance in held-out metric evaluations, we can leverage SAGA from different modes to design more enhancers with high quality. Moreover, SAGA also recapitulates key biological principles, recovering multiple liver-specific transcription factor motifs (Sandelin et al., 2004) (shown in Figure 2(e)), supporting the biological plausibility of the designed sequences. When being evaluated on more biological-relevant multimodal regulatory readouts, including Cap Analysis Gene Expression sequencing (CAGE-seq) (Shiraki et al., 2003) and DNase I hypersensitive sites sequencing (DNase-seq) (Boyle et al., 2008) predictions, the designed enhancers again display strong HepG2 specificity, and they also show higher HepG2-specific expression levels compared with baseline methods. These results highlight SAGA 's ability to leverage information encoded in pre-trained sequence-to-function models such as Enformer (Avsec et al., 2021) to capture multimodal regulatory signals. In cell types where enhancers are active, lineage-defining and signal-responsive transcription factors bind to the enhancer sequence and recruit chromatin remodeling complexes, leading to localized chromatin opening and elevated DNase I hypersensitivity, captured by CAGE-seq (DaSilva et al., 2024).

3 Conclusion

In this paper, we introduced SAGA, a generalist autonomous agent for scientific discovery. Through comprehensive experiments, we find that iterating objectives is the key driver of progress in achieving a novel discovery with practical viability. A clear advantage of SAGA comes from its alignment with scientific practice, and decide which constraints matter the most at a given stage.

REFERENCES

Josh Abramson, Jonas Adler, Jack Dunger, Richard Evans, Tim Green, Alexander Pritzel, Olaf Ronneberger, Lindsay Willmore, Andrew J Ballard, Joshua Bambrick, et al. Accurate structure prediction of biomolecular interactions with alphafold 3. *Nature*, 630(8016):493–500, 2024.

Vikram Agarwal, Fumitaka Inoue, Max Schubach, Dmitry Penzar, Beth K Martin, Pyaree Mohan Dash, Pia Keukeleire, Zicong Zhang, Ajuni Sohota, Jingjing Zhao, et al. Massively parallel characterization of transcriptional regulatory elements. *Nature*, 639(8054):411–420, 2025.

Robin Andersson and Albin Sandelin. Determinants of enhancer and promoter activities of regulatory elements. *Nature Reviews Genetics*, 21(2):71–87, 2020.

Žiga Avsec, Vikram Agarwal, Daniel Visentin, Joseph R Ledsam, Agnieszka Grabska-Barwinska, Kyle R Taylor, Yannis Assael, John Jumper, Pushmeet Kohli, and David R Kelley. Effective gene expression prediction from sequence by integrating long-range interactions. *Nature methods*, 18 (10):1196–1203, 2021.

L T Biegler, I E Grossmann, and A W Westerberg. *Systematic methods for chemical process design*. Prentice Hall, Old Tappan, NJ (United States), 12 1997. URL https://www.osti.gov/biblio/293030.

Alan P Boyle, Sean Davis, Hennady P Shulha, Paul Meltzer, Elliott H Margulies, Zhiping Weng, Terrence S Furey, and Gregory E Crawford. High-resolution mapping and characterization of open chromatin across the genome. *Cell*, 132(2):311–322, 2008.

Eric D. Brown and Gerard D. Wright. Antibacterial drug discovery in the resistance era. *Nature*, 529(7586):336–343, January 2016. ISSN 1476-4687. doi: 10.1038/nature17042. URL https://www.nature.com/articles/nature17042.

Darko Butina. Unsupervised data base clustering based on daylight's fingerprint and tanimoto similarity: A fast and automated way to cluster small and large data sets. *Journal of Chemical Information and Computer Sciences*, 39(4):747–750, 1999.

J. M. Cavanagh, K. Sun, A. Gritsevskiy, D. Bagni, T. D. Bannister, and T. Head-Gordon. Smileyllama: Modifying large language models for directed chemical space exploration. *ArXiv*, 2409.02231(in review), 2024.

Junwu Chen, Jeff Guo, Edvin Fako, and Philippe Schwaller. Accelerating inverse materials design using generative diffusion models with reinforcement learning. *arXiv preprint arXiv:2511.03112*, 2025a.

Junwu Chen, Xu Huang, Cheng Hua, Yulian He, and Philippe Schwaller. A multi-modal transformer for predicting global minimum adsorption energy. *Nature Communications*, 16(1):3232, 2025b.

Xingyu Chen, Shihao Ma, Runsheng Lin, Jiecong Lin, and Bo Wang. Ctrl-dna: Controllable cell-type-specific regulatory dna design via constrained rl. *arXiv preprint arXiv:2505.20578*, 2025c.

Lucas Ferreira DaSilva, Simon Senan, Zain Munir Patel, Aniketh Janardhan Reddy, Sameer Gabbita, Zach Nussbaum, César Miguel Valdez Córdova, Aaron Wenteler, Noah Weber, Tin M Tunjic, et al. Dna-diffusion: leveraging generative models for controlling chromatin accessibility and gene expression via synthetic regulatory elements. *Biorxiv*, 2024.

Justas Dauparas, Ivan Anishchenko, Nathaniel Bennett, Hua Bai, Robert J Ragotte, Lukas F Milles, Basile IM Wicky, Alexis Courbet, Rob J de Haas, Neville Bethel, et al. Robust deep learning–based protein sequence design using proteinmpnn. *Science*, 378(6615):49–56, 2022.

Carl G de Boer and Jussi Taipale. Hold out the genome: a roadmap to solving the cis-regulatory code. *Nature*, 625(7993):41–50, 2024.

Wenli Du and Shaoyi Yang. The potential and challenges of large language model agent systems in chemical process simulation: from automated modeling to intelligent design. *Frontiers of Chemical Science and Engineering*, 19(10):99, 2025.

Qinghe Gao and Artur M Schweidtmann. Deep reinforcement learning for process design: Review and perspective. *Current Opinion in Chemical Engineering*, 44:101012, 2024.

Michael W Gaultois, Taylor D Sparks, Christopher KH Borg, Ram Seshadri, William D Bonificio, and David R Clarke. Data-driven review of thermoelectric materials: performance and resource considerations. *Chemistry of Materials*, 25(15):2911–2920, 2013.

Sager J Gosai, Rodrigo I Castro, Natalia Fuentes, John C Butts, Kousuke Mouri, Michael Alasoadura, Susan Kales, Thanh Thanh L Nguyen, Ramil R Noche, Arya S Rao, et al. Machine-guided design of cell-type-targeting cis-regulatory elements. *Nature*, 634(8036):1211–1220, 2024.

Quirin Göttl, Jonathan Pirnay, Jakob Burger, and Dominik G Grimm. Deep reinforcement learning enables conceptual design of processes for separating azeotropic mixtures without prior knowledge. *Computers & Chemical Engineering*, 194:108975, 2025.

Kexin Huang, Serena Zhang, Hanchen Wang, Yuanhao Qu, Yingzhou Lu, Yusuf Roohani, Ryan Li, Lin Qiu, Gavin Li, Junze Zhang, et al. Biomni: A general-purpose biomedical ai agent. *biorxiv*, 2025a.

Xu Huang, Junwu Chen, Yuxing Fei, Zhuohan Li, Philippe Schwaller, and Gerbrand Ceder. Cascade: Cumulative agentic skill creation through autonomous development and evolution. *arXiv preprint arXiv:2512.23880*, 2025b.

Anubhav Jain, Shyue Ping Ong, Geoffroy Hautier, Wei Chen, William Davidson Richards, Stephen Dacek, Shreyas Cholia, Dan Gunter, David Skinner, Gerbrand Ceder, et al. Commentary: The materials project: A materials genome approach to accelerating materials innovation. *APL Materials*, 1(1):011002, 2013.

Seung-Hoon Jhi, Jisoon Ihm, Steven G Louie, and Marvin L Cohen. Electronic mechanism of hardness enhancement in transition-metal carbonitrides. *Nature*, 399(6732):132–134, 1999.

Wolfgang Kabsch and Christian Sander. Dictionary of protein secondary structure: pattern recognition of hydrogen-bonded and geometrical features. *Biopolymers: Original Research on Biomolecules*, 22(12):2577–2637, 1983.

Anthousa Kythreotou, Abdul Siddique, Francesco A Mauri, Mark Bower, and David J Pinato. Pd-l1. *Journal of clinical pathology*, 71(3):189–194, 2018.

Avantika Lal, David Garfield, Tommaso Biancalani, and Gokcen Eraslan. Designing realistic regulatory dna with autoregressive language models. *Genome Research*, 34(9):1411–1420, 2024.

Tianyu Liu, Tinglin Huang, Lijun Wang, Yingxin Lin, Rex Ying, and Hongyu Zhao. Unicorn: Towards universal cellular expression prediction with a multi-task learning framework. *Nature Communications*, 16(1):9455, 2025a.

Yunsheng Liu, Joseph M Cavanagh, Kunyang Sun, Jacob Toney, Chung-Yueh Yuan, Andrew Smith, Roland St Michel II, Paul A. Graggs, F. Dean Toste, Heather Kulik, and Teresa Head-Gordon. Exploring transition metal complexes with large language models. *ChemRxiv*, 2025b. doi: 10.26434/chemrxiv-2025-hm3zb.

Hannes H Loeffler, Jiazhen He, Alessandro Tibo, Jon Paul Janet, Alexey Voronov, Lewis H Mervin, and Ola Engkvist. Reinvent 4: modern ai–driven generative molecule design. *Journal of Cheminformatics*, 16(1):20, 2024.

Aria Mansouri Tehrani, Anton O Oliynyk, Marcus Parry, Zeshan Rizvi, Samantha Couper, Feng Lin, Lowell Miyagi, Taylor D Sparks, and Jakoah Brgoch. Machine learning directed search for ultraincompressible, superhard materials. *Journal of the American Chemical Society*, 140(31):9844–9853, 2018.

Amil Merchant, Simon Batzner, Samuel S Schoenholz, Muratahan Aykol, Gowoon Cheon, and Ekin Dogus Cubuk. Scaling deep learning for materials discovery. *Nature*, 624(7990):80–85, 2023.

Belle A Moyers, E Christopher Partridge, Mark Mackiewicz, Michael J Betti, Roshan Darji, Sarah K Meadows, Kimberly M Newberry, Laurel A Brandsmeier, Barbara J Wold, Eric M Mendenhall, et al. Characterization of human transcription factor function and patterns of gene regulation in hepg2 cells. *Genome Research*, 33(11):1879–1892, 2023.

Serge Muyldermans. Applications of nanobodies. *Annual review of animal biosciences*, 9(1):401–421, 2021.

Alexander Novikov, Ngân Vũ, Marvin Eisenberger, Emilien Dupont, Po-Sen Huang, Adam Zsolt Wagner, Sergey Shirobokov, Borislav Kozlovskii, Francisco JR Ruiz, Abbas Mehrabian, et al. Alphaevolve: A coding agent for scientific and algorithmic discovery. *arXiv preprint arXiv:2506.13131*, 2025.

Shyue Ping Ong, William Davidson Richards, Anubhav Jain, Geoffroy Hautier, Michael Kocher, Shreyas Cholia, Dan Gunter, Vincent L Chevrier, Kristin A Persson, and Gerbrand Ceder. Python materials genomics (pymatgen): A robust, open-source python library for materials analysis. *Computational Materials Science*, 68:314–319, 2013.

Lisa E Pangilinan, Shanlin Hu, Spencer G Hamilton, Sarah H Tolbert, and Richard B Kaner. Hardening effects in superhard transition-metal borides. *Accounts of Materials Research*, 3(1):100–109, 2021.

Hyunsoo Park, Zhenzhu Li, and Aron Walsh. Has generative artificial intelligence solved inverse materials design? *Matter*, 7(7):2355–2367, 2024.

Saro Passaro, Gabriele Corso, Jeremy Wohlwend, Mateo Reveiz, Stephan Thaler, Vignesh Ram Somnath, Noah Getz, Tally Portnoi, Julien Roy, Hannes Stark, et al. Boltz-2: Towards accurate and efficient binding affinity prediction. *BioRxiv*, 2025.

Sophia Rupprecht, Qinghe Gao, Tanuj Karia, and Artur M Schweidtmann. Multi-agent systems for chemical engineering: A review and perspective. *arXiv preprint arXiv:2508.07880*, 2025.

Albin Sandelin, Wynand Alkema, Pär Engström, Wyeth W Wasserman, and Boris Lenhard. Jaspar: an open-access database for eukaryotic transcription factor binding profiles. *Nucleic acids research*, 32(suppl_1):D91–D94, 2004.

Hatim Sati, Elena Carrara, Alessia Savoldi, Paul Hansen, Jacopo Garlasco, Enrica Campagnaro, Simone Boccia, Juan Antonio Castillo-Polo, Eugenia Magrini, Pilar Garcia-Vello, Eve Wool, Valeria Gigante, Erin Duffy, Alessandro Cassini, Benedikt Huttner, Pilar Ramon Pardo, Mohsen Naghavi, Fuad Mirzayev, Matteo Zignol, Alexandra Cameron, Evelina Tacconelli, and WHO Bacterial Priority Pathogens List Advisory Group. The who bacterial priority pathogens list 2024: a prioritisation study to guide research, development, and public health strategies against antimicrobial resistance. *The Lancet Infectious Diseases*, 25(9):1033–1043, 2025. doi: 10.1016/S1473-3099(25)00118-5. Epub 2025-04-14.

Toshiyuki Shiraki, Shinji Kondo, Shintaro Katayama, Kazunori Waki, Takeya Kasukawa, Hideya Kawaji, Rimantas Kodzius, Akira Watahiki, Mari Nakamura, Takahiro Arakawa, et al. Cap analysis gene expression for high-throughput analysis of transcriptional starting point and identification of promoter usage. *Proceedings of the National Academy of Sciences*, 100(26):15776–15781, 2003.

Alexander N Shivanyuk, Sergey V Ryabukhin, A Tolmachev, AV Bogolyubsky, DM Mykytenko, AA Chupryna, W Heilman, and AN Kostyuk. Enamine real database: Making chemical diversity real. *Chemistry today*, 25(6):58–59, 2007.

Yasir Sohail, Chongle Zhang, Dezhen Xue, Jinyu Zhang, Dongdong Zhang, Shaohua Gao, Yang Yang, Xiaoxuan Fan, Hang Zhang, Gang Liu, et al. Machine-learning design of ductile fenicoalta alloys with high strength. *Nature*, pp. 1–6, 2025.

Hannes Stark, Felix Faltings, MinGyu Choi, Yuxin Xie, Eunsu Hur, Timothy O'Donnell, Anton Bushuiev, Talip Uçar, Saro Passaro, Weian Mao, et al. Boltzgen: Toward universal binder design. *bioRxiv*, pp. 2025–11, 2025.

Laura Stops, Roel Leenhouts, Qinghe Gao, and Artur M Schweidtmann. Flowsheet generation through hierarchical reinforcement learning and graph neural networks. *AIChE Journal*, 69(1): e17938, 2023.

Kunyang Sun, Dorian Bagni, Joseph M. Cavanagh, Yingze Wang, Jacob M. Sawyer, Bo Zhou, Andrew Gritsevskiy, Oufan Zhang, and Teresa Head-Gordon. Synllama: Generating synthesizable molecules and their analogs with large language models. *ACS Central Science*, 11 (11):2108–2120, 2025. ISSN 2374-7943. doi: 10.1021/acscentsci.5c01285. URL https://doi.org/10.1021/acscentsci.5c01285.

Richard Turton, Richard C Bailie, Wallace B Whiting, and Joseph A Shaeiwitz. *Analysis, synthesis and design of chemical processes*. Pearson Education, 2008.

Remco L van den Broek, Shivam Patel, Gerard JP van Westen, Willem Jespers, and Woody Sherman. In search of beautiful molecules: a perspective on generative modeling for drug design. *Journal of chemical information and modeling*, 65(18):9383–9397, 2025.

Patricia J Wittkopp and Gizem Kalay. Cis-regulatory elements: molecular mechanisms and evolutionary processes underlying divergence. *Nature Reviews Genetics*, 13(1):59–69, 2012.

Chengyu Xiao, Mengqi Liu, Kan Yao, Yifan Zhang, Mengqi Zhang, Max Yan, Ya Sun, Xianghui Liu, Xuanyu Cui, Tongxiang Fan, et al. Ultrabroadband and band-selective thermal meta-emitters by machine learning. *Nature*, 643(8070):80–88, 2025.

Han Yang, Chenxi Hu, Yichi Zhou, Xixian Liu, Yu Shi, Jielan Li, Guanzhi Li, Zekun Chen, Shuizhou Chen, Claudio Zeni, et al. MatterSim: A deep learning atomistic model across elements, temperatures and pressures. *arXiv preprint arXiv:2405.04967*, 2024.

Zhenpeng Yao, Yanwei Lum, Andrew Johnston, Luis Martin Mejia-Mendoza, Xin Zhou, Yonggang Wen, Alán Aspuru-Guzik, Edward H Sargent, and Zhi Wei Seh. Machine learning for a sustainable energy future. *Nature Reviews Materials*, 8(3):202–215, 2023.

Xiaoang Yuan, Bo Zhu, Chunbo Zhang, Qifan Zheng, Enlai Gao, and Qian Shao. Accelerated discovery of ultraincompressible, superhard materials via physics-enhanced active learning. *Materials Horizons*, 2025.

Mert Yuksekgonul, Federico Bianchi, Joseph Boen, Sheng Liu, Pan Lu, Zhi Huang, Carlos Guestrin, and James Zou. Optimizing generative ai by backpropagating language model feedback. *Nature*, 639(8055):609–616, 2025.

Tong Zeng, Srivathsan Badrinarayanan, Janghoon Ock, Cheng-Kai Lai, and Amir Barati Farimani. Llm-guided chemical process optimization with a multi-agent approach. *Machine Learning: Science and Technology*, 2025.

Claudio Zeni, Robert Pinsler, Daniel Zügner, Andrew Fowler, Matthew Horton, Xiang Fu, Zilong Wang, Aliaksandra Shysheya, Jonathan Crabbé, Shoko Ueda, et al. A generative model for inorganic materials design. *Nature*, pp. 1–3, 2025.

A SAGA FOR INORGANIC MATERIALS DESIGN

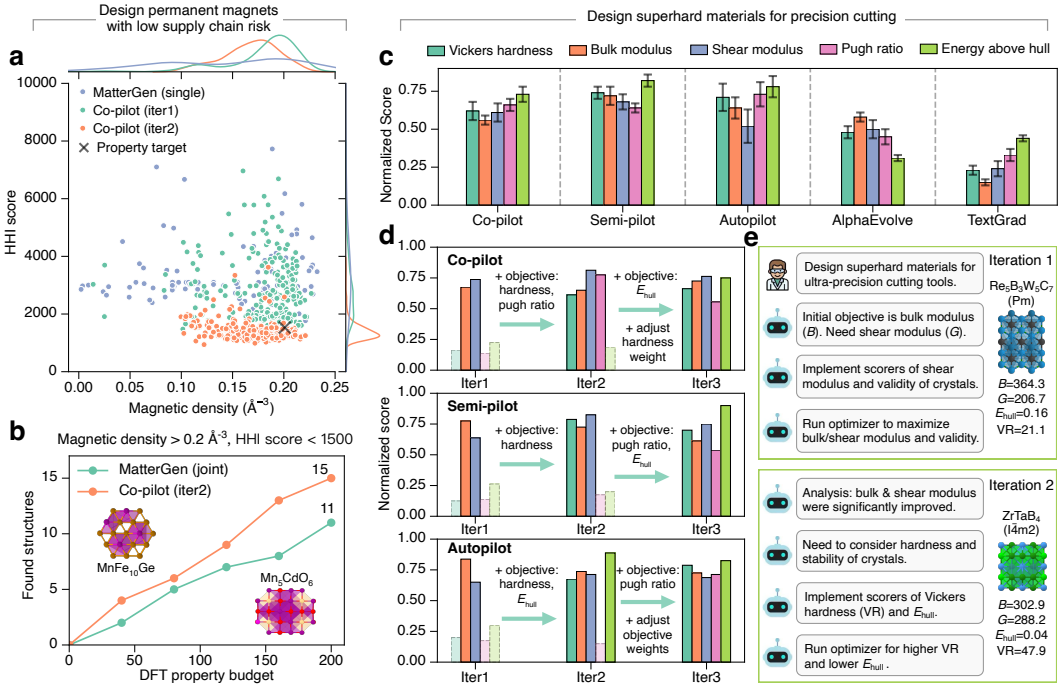

Figure 3: *Results for inorganic materials design.* (a) Property distributions of generated structures from co-pilot across different iterations and from MatterGen (single) targeting only high magnetic density. (b) Number of stable and novel structures satisfying property requirements found by co-pilot and MatterGen (joint) within 200 DFT property calculations, for targets with magnetic density above 0.2 Å^{-3} and HHI score below 1500. It also displays 3D visualizations of two crystal structures proposed by the co-pilot mode that satisfy the design goal. (c) Comparisons between different levels of SAGA and selected baselines on the design task of superhard materials for precision cutting. All evaluation metrics are normalized, with higher scores representing better performance. (d) Comparison of different SAGA modes over three iterations with the same held-out metrics. In each iteration, SAGA analyze the optimized crystal structures, propose new objectives, run property optimization, and select the best candidates across all current iterations. Text annotations highlight specific agent feedback on objective evolution that drives the improvement in metric scores across iterations. The solid line means objectives address the evaluation metrics, and the dash line means the metric has not been addressed. (e) An example of the autopilot feedback loop. SAGA identifies issues and dynamically evolves objectives, successfully proposed novel structures exhibiting high hardness, high elastic modulus, and thermodynamic stability.

The discovery of novel materials is critical for driving technological innovation across diverse fields, including catalysis, energy, electronics, and advanced manufacturing (Huang et al., 2025b; Zeni et al., 2025; Merchant et al., 2023; Sohail et al., 2025; Yao et al., 2023; Chen et al., 2025a;b). Most material design tasks involve multiple objectives encompassing electronic, mechanical and physicochemical properties, as well as production costs (Zeni et al., 2025; Chen et al., 2025a). These design objectives are often intricately interrelated and may exhibit competitive or even conflicting trade-offs (Park et al., 2024; Chen et al., 2025a; Xiao et al., 2025). Optimization with fixed objectives may overlook other important material properties or fail to refine optimization objectives based on deficiencies identified in proposed candidates. To address this challenge, we apply SAGA to design the desired novel materials for specific applications through iterative optimization with dynamic objectives. SAGA can guide LLMs to search materials with desired properties, iteratively analyzing and adjusting optimization objectives, while automatically programming scoring functions to evaluate the new objectives and provide feedback. We propose two design tasks to assess the SAGA's effectiveness.

SAGA enables efficient magnet materials design. First, we evaluate SAGA on the task of designing permanent magnets with low supply chain risk, and compare against MatterGen (Zeni et al., 2025), one of the state-of-the-art generative models for inorganic materials design. In this task, two objectives are specified: magnetic density higher than 0.2 $Å^{-3}$ and Herfindahl–Hirschman index (HHI) score less than 1500, where a lower HHI score indicates lower supply chain risk and the absence of rare earth elements (Zeni et al., 2025; Gaultois et al., 2013). The SAGA Co-pilot mode is deployed with iteratively refined objectives: maximizing magnetic density in the first iteration, followed by the addition of HHI score minimization in the second. Performance was compared against the MatterGen model that targets only high magnetic density (single) or both properties (joint) (Zeni et al., 2025). As shown in Figure 3(a), Co-pilot proposes crystal structures with high magnetic density after the first iteration, exhibiting a higher distribution density near the target value of 0.2 $Å^{-3}$ compared to MatterGen (single). However, these structures display a broad range of HHI scores, with over 80 % exceeding 2000. After the second iteration, Co-pilot successfully discovers crystal structures with both high magnetic density and low HHI scores. The majority of proposed structures exhibit magnetic density above 0.15 $Å^{-3}$ and HHI scores below 1500. This demonstrates that SAGA's Co-pilot mode can continuously and iteratively optimize material properties with human feedback to accomplish multi-objective tasks. Moreover, within a computational budget of 200 DFT property evaluations (Figure 3b), Co-pilot mode identify 15 novel and stable structures satisfying the desired properties, outperforming MatterGen (11 structures). These results demonstrate that SAGA can continuously optimize dynamic objectives, potentially outperforming specialized generative models that are constrained to fixed objectives.

SAGA enables efficient superhard materials design. Subsequently, we evaluate SAGA on the task of designing superhard materials for precision cutting and compare with an LLM-based optimization algorithm, TextGrad (Yuksekgonul et al., 2025), as MatterGen (Zeni et al., 2025) requires fine-tuning on large amounts of DFT-labeled data when switching tasks. This task involves more than three target material properties, whereas conventional methods that optimize with fixed targets may only achieve high scores on certain metrics but ignore other important properties of the designed materials. As shown in Figure 3(c), the crystal structures designed by three modes (co-pilot, semi-pilot, autopilot) achieve high scores on all metrics. Benefiting from iterative optimization and dynamic objective refinement, all SAGA modes successfully propose novel structures exhibiting high hardness, high elastic modulus, appropriate brittleness, and thermodynamic stability. In contrast, the TextGrad approach, which employs fixed optimization objectives, demonstrated moderate performance for energy above hull and Pugh ratio but achieved much lower scores for hardness and elastic modulus. These results demonstrate that SAGA's iterative optimization and dynamic objective strategy are effective for complex multi-objective tasks. Furthermore, we analyze the crystal structures proposed by SAGA in the final iteration and found that the underlying patterns correlate with key factors for superhard material formation reported in experimental studies. More than 90 % of the proposed crystals contain light elements such as boron, carbon, nitrogen, and oxygen, aligning with experimental findings that light elements are essential for superhard materials because their small atomic radii enable short, directional covalent bonds with high electron density (Pangilinan et al., 2021; Jhi et al., 1999). In addition, over 75 % of the proposed crystals are transition metal carbides, nitrides, and borides. Correspondingly, experimental studies have demonstrated that the combination of light elements (boron, carbon, nitrogen) with electron-rich transition metals can form dense covalent networks and enhance material hardness (Pangilinan et al., 2021; Jhi et al., 1999).

SAGA proposes reasonable and important objectives aligning with materials scientists. The co-pilot and semi-pilot modes incorporate human input into the agent's decision-making process to review and refine candidate analyses and proposed objectives for subsequent iterations. As shown in Figure 3(d), integration of human feedback enabled SAGA to consider additional relevant objectives across multiple iterations, resulting in comprehensive performance improvements of the designed materials. For instance, explicit prioritization of Vickers hardness, elastic modulus and Pugh ratio (Mansouri Tehrani et al., 2018; Yuan et al., 2025), guided by expert input, led to substantial enhancement of the mechanical properties in proposed crystalline materials. The results demonstrate that SAGA can effectively integrate human feedback through adaptive objective formulation, improving the overall performance of proposed materials. Moreover, SAGA's autopilot mode can analyze results and set objectives autonomously without human intervention. By analyzing materials proposed in the current iteration, autopilot mode can identify their weaknesses and adaptively refines objectives for iterative improvement. As illustrated in Figure 3(d), autopilot mode can propose important

optimization objectives for the design goal, similar to expert guidance. Autopilot achieves excellent overall performance comparable to co-pilot and semi-pilot across all five metrics, underscoring its remarkable intelligence and automation capabilities. Figure 3e demonstrates that autopilot can correctly understand the design goal and analyze properties of proposed materials, subsequently proposing appropriate and highly relevant new objectives (e.g., Vickers hardness, Pugh ratio, and energy above hull) targeting mechanical performance and stability. For newly proposed objectives, SAGA implements property evaluators through web search and automated programming, leveraging publicly available pretrained models or empirical methods. Upon analyzing designed structures and determining that a particular objective has been sufficiently optimized, SAGA automatically adjusts the optimization weight for that objective. Specifically, SAGA employs scaling or truncation of material property values to prevent over-optimization of individual objectives while neglecting others. Overall, these results demonstrate that SAGA enables automated materials design with different levels of human intervention through dynamic iterative optimization.

B SAGA FOR ANTIBIOTIC DESIGN

Antimicrobial resistance (AMR) is rapidly eroding our ability to treat common Gram-negative infections, one of which is Klebsiella pneumoniae (*K. pneumoniae*), ranked as a critical priority pathogen by the World Health Organization (WHO) (Brown & Wright, 2016; Sati et al., 2025). However, designing novel inhibitors for Gram-negative bacteria is notoriously difficult, as optimization agents suffer from generating chemically unreasonable compounds that lack the necessary given objectives van den Broek et al. (2025). To address this challenge, we use SAGA to design novel *K. pneumoniae* inhibitors. Rather than relying on a static scoring function that attempts to encode every rule upfront, SAGA begins with only primary biological objectives for maximizing potency and minimizing toxicity, along with a constraint to avoid existing scaffolds. From this foundation, SAGA dynamically constructs a suite of auxiliary objectives that steer the generative process toward realistic chemical space at all three levels of automation. This strategy allows SAGA to learn the specific constraints of the Gram-negative landscape in an interpretable, iterative manner. Ultimately, SAGA produces more valid candidates that satisfy rigorous external evaluations and align with scientists' intuition.

SAGA discovers computationally selective and chemically reasonable candidates. We run SAGA at all three levels of automation with the same prompt and primary biological objectives. SAGA then iterates at different levels of automation until optimization is complete. To evaluate the quality of proposed candidates, we selected three biological evaluations: *K. pneumoniae* (KP) Activity defined by the Antibiotic activity score, a Novelty score, and Safety defined by (1 - the Toxicity score), as well as two chemical evaluations: Drug likeness defined by the Quantitative estimate of drug likeness (QED) score, and Synthesizability defined by the Synthetical Accessibility (SA) score.

As illustrated in Figure 4(a), SAGA achieves a significantly higher percentage of candidates passing all evaluations compared to state-of-the-art language model baselines. In contrast to SAGA, these baselines exhibit distinct failure modes. Several language models struggle to overcome the optimization difficulty of the KP activity score alone, resulting in chemically valid but inactive molecules. Conversely, the Optimizer agent (SAGA-Opt), which does not have the capacity to dynamically evolve objectives, achieves high KP activity but suffers a catastrophic drop in medicinal chemistry quality. Furthermore, SAGA successfully balances the scores of both biological objectives and standard medicinal chemistry filters, discovering drug-like molecules with high predicted activity across all 3 modes of operation. As seen in Figure 4(c), while SAGA candidates consistently score above the Drug likeness Threshold, SAGA-Opt's population distribution falls almost entirely below it, designing unrealistically large and undrug-like molecules. This observation confirms that, without dynamic auxiliary objectives, language models either fail to optimize the primary biological goal or, like SAGA-Opt, exploit the scoring function to propose active but chemically invalid structures. SAGA, on the other hand, successfully aligns with the desired distribution of realistic drug candidates, producing the most promising candidates.

SAGA can effectively analyze, propose, and implement informative and necessary objectives across different levels of automation. In addition to raw performance, SAGA enables explainable and robust optimization by dynamically evolving its scoring functions, reorienting the trajectory to align with both the global *K. pneumoniae* optimization goals and chemical intuition. As illustrated in

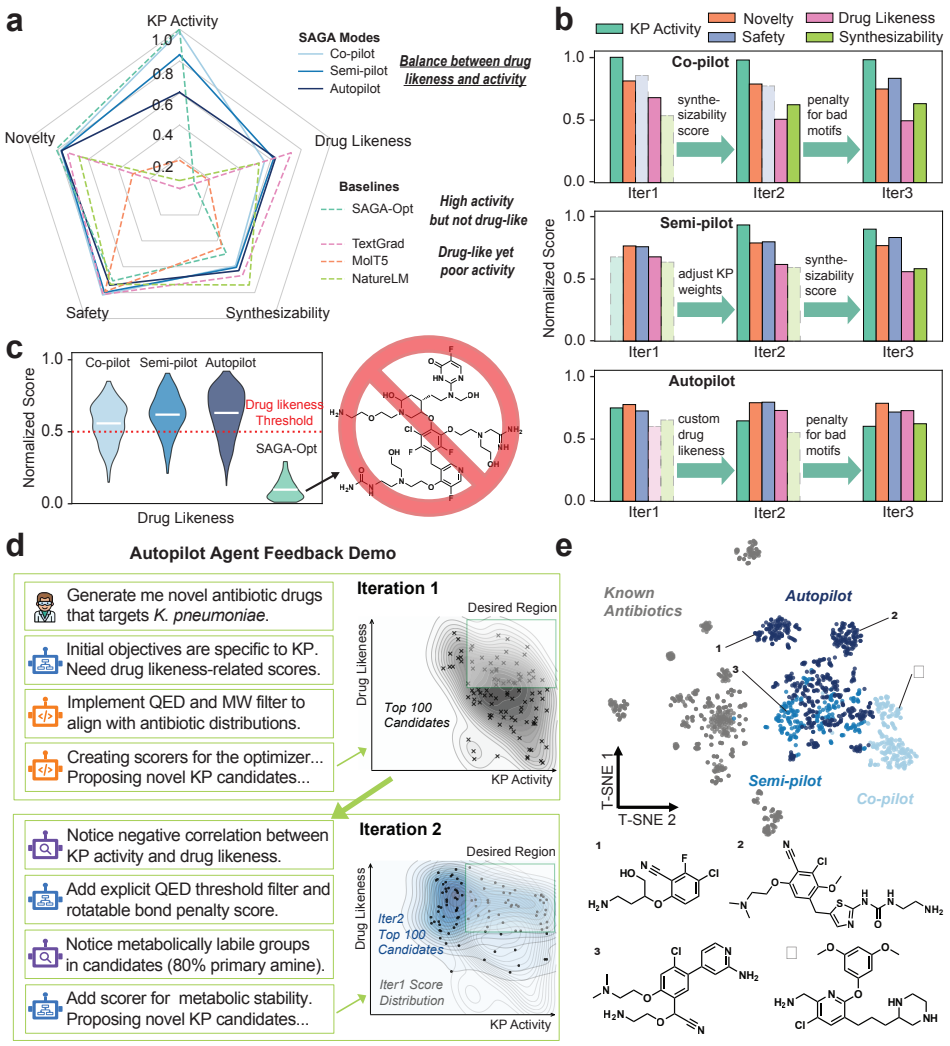

Figure 4: *Results for antibiotic design.* (a) Comparisons between SAGA and language model baselines. Candidates from all SAGA modes achieve the "drug likeness and activity sweet spot", whereas baselines struggle to balance biological objectives, especially KP activity, with chemical assessments like Drug likeness and Synthesizability. (b) Comparisons across SAGA iterations. Text annotations highlight specific agent feedback on objective evolution that drives the improvement in metric scores across iterations. The solid line means objectives address the evaluation metrics, and the dash line means the metric has not been addressed. (c) Distribution of drug likeness score. Most molecules from SAGA surpass the drug likeness threshold (red dashed line), while AlphaEvolve falls below it, demonstrating its critical misalignment with final objectives. (d) An example of the autopilot feedback loop. The analyzer identifies issues and the planner dynamically evolves objectives, shifting the distribution of the Top 100 Candidate more to the Desired Region of high activity and drug likeness. (e) T-SNE plot of SAGA molecules against known antibiotics. Numbered structures (1-4) are examples of molecules passing all evaluation metrics. They contain novel scaffolds distinct from known antibiotic clusters.

Figure 4(b), the components from SAGA demonstrate context-awareness regarding the optimization landscape. While the co-pilot mode incorporates nuanced human feedback to address low synthesizability, the semi-pilot agent intelligently defers adding strict chemical constraints in early iterations to instead adjust weights and prioritize the optimization of the KP activity objective. In the autopilot mode, the analyzer agent provides chemical insights that anticipate expert concerns. As shown in Figure 4(d), SAGA goes beyond individual molecular analysis and identifies population-level

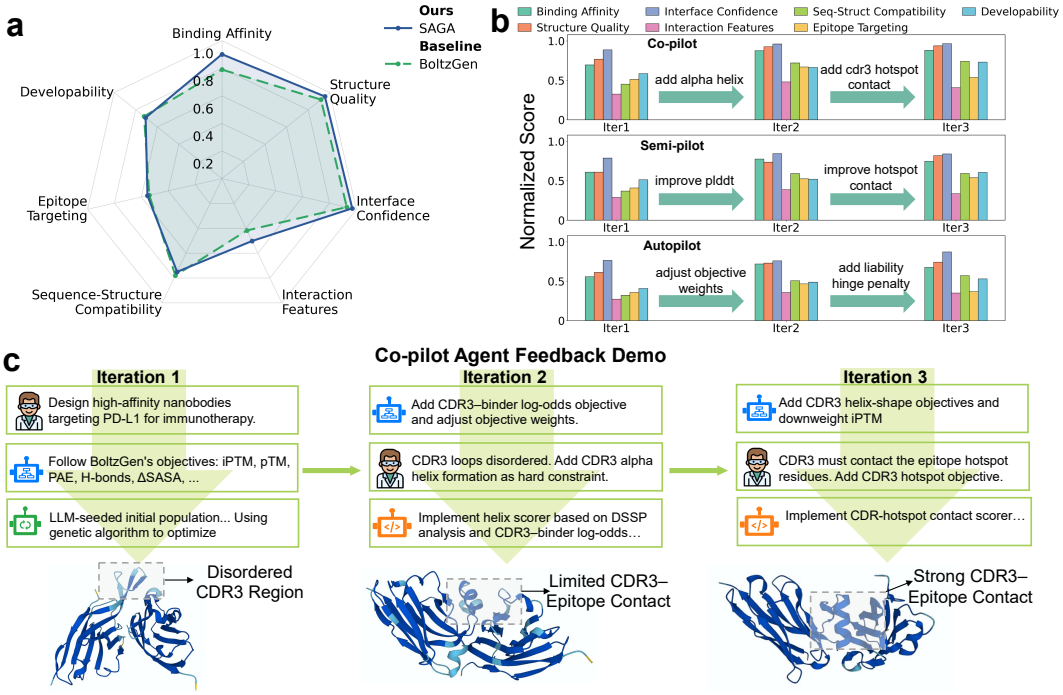

Figure 5: *Results for nanobody binder design against PD-L1.* (a) Multi-objective performance comparison between SAGA, BoltzGen, and baseline designs. Radar plots summarize seven complementary evaluation axes, including predicted binding affinity, structure quality, interface confidence, interaction features, sequence–structure compatibility, epitope targeting, and developability. SAGA achieves a more balanced profile across binding, structural integrity, and developability metrics. (b) Iterative performance improvement across three levels of human–agent collaboration. Bar plots show normalized metric scores over successive optimization iterations for the co-pilot, semi-pilot, and autopilot modes. Text annotations highlight representative objective updates introduced at each iteration that drive metric improvements. (c) Demonstration of the co-pilot feedback loop. Human experts provide high-level design goals and critiques, which SAGA translates into concrete objectives, such as CDR3 helix-shape constraints, CDR3–binder log-odds scoring, and epitope hotspot contact objectives. Structural snapshots illustrate the progressive transition from disordered CDR3 regions to geometrically well-formed CDR3 helices with strong epitope engagement.

trends, such as “negative correlation between KP activity and drug likeness”. Furthermore, it performs granular structural analysis to pinpoint specific over-represented metabolically labile groups, insights that typically require systematic review to uncover. In response, SAGA autonomously constructs filters and scorers that steer the generated population to the ”Desired Region” of the physicochemical space, leading to a higher passing rate for external chemical motif alerts. Collectively, these examples demonstrate the practical utility of dynamic objective evolution in solving the hard, multi-objective optimization problem, generating final candidates that show more promises.

SAGA discovers computationally performant molecules with novel, synthetically accessible scaffolds distinct from existing antibiotics. A primary goal of *de novo* design is to discover potent, drug-like candidates that diverge from the existing antibiotics space. Therefore, after SAGA finishes optimization, we first filter all proposed candidates by applying a set of stringent evaluation cutoffs to select molecules with high probability of experimental success and then assess how similar they are to existing antibiotics. As shown in Figure 4(e), the selected molecules occupy diverse regions distinct from the tight clusters of over 500 known antibiotics. Specific examples (Structures 1–4) further illustrate that, rather than only optimizing around one fixed scaffold, SAGA generalizes the rules of bacterial inhibition to assemble novel backbone architectures and halobenzene cores.

C SAGA FOR ANTIBODY BINDER DESIGN

Antibody binder design is a cornerstone of modern therapeutic discovery and a canonical setting for automating scientific discovery. A successful design must simultaneously satisfy target binding, structural integrity, epitope engagement, and developability constraints. Although the field has accumulated a rich toolbox of *in silico* proxies and increasingly standardized pipelines for generation, inverse folding, and post hoc filtering and ranking, recent community-scale experience suggests that this space remains methodologically unsettled. Post-competition analyses of the Adaptyv Nipah *de novo* binder challenge further underscore that progress in *in silico* scoring does not yet translate into a reliable recipe for experimental success. In practice, candidates that score highly under standard computational criteria can still fail wet-lab validation, suggesting that the target-relevant signals captured by current metrics, and the appropriate way to weight them, remain unclear. Collectively, these observations expose a central bottleneck. We still lack a principled understanding of which computational objectives are reliably predictive and how to balance them to consistently yield experimental binders. In practice, this gap is often compensated by brute-force candidate generation at the scale of tens of thousands of sequences per target, followed by extensive filtering and ranking, which is both time-consuming and compute intensive.

To address this challenge, we apply SAGA to *de novo* binder design through dynamic objective evolution. Rather than committing to a static scoring function that attempts to encode every rule upfront, SAGA begins from a minimal set of primary design goals and iteratively constructs auxiliary objectives that diagnose and correct failure modes as they emerge. It adds, removes, and reweights objectives to steer generation toward a realistic desired region of sequence, structure, and function space. Crucially, SAGA supports three levels of human-agent collaboration, enabling the optimization process to incorporate expert hypotheses when available while remaining capable of autonomous exploration when expert feedback is limited. In this section, we focus on nanobody design against PD-L1, an immune checkpoint ligand with major clinical relevance in cancer immunotherapy, as a representative high-impact target to demonstrate how SAGA leverages the background knowledge of a large language model, human expert interaction, and objective discovery to produce promising nanobody candidates (Kythreotou et al., 2018).

SAGA discovers computationally strong nanobody candidates while substantially reducing search cost. We run SAGA with the same prompt and the same initial design objective as BoltzGen (Stark et al., 2025) and compare its final candidates against the 15 PD-L1 nanobodies reported by BoltzGen. For a controlled evaluation, we select 15 SAGA candidates from the end of optimization that are consistently strong across objectives. We evaluate candidates along seven complementary axes that reflect practical binder requirements: binding affinity (ipTM, pTM), structure quality (pLDDT for the full binder and CDRs), interface confidence (minimum PAE), interaction features (hydrogen bonds, salt bridges, and ΔSASA), sequence–structure compatibility (ProteinMPNN score and recovery (Dauparas et al., 2022)), epitope targeting (CDR–hotspot contacts), and developability (liability scores). To reduce oracle-specific bias, we predict structures for each sequence using both AlphaFold3 (Abramson et al., 2024) and Boltz2 (Passaro et al., 2025) and report metrics averaged across the two predictors. As shown in Figure 5(a), SAGA matches or exceeds BoltzGen across most dimensions, yielding a more balanced profile across binding, structural integrity, interface confidence, and developability. Notably, SAGA achieves this performance with substantially fewer generated sequences, producing 12,000 total candidates compared to BoltzGen's 60,000 per target. This suggests that dynamic objective evolution can compress the effective search space while maintaining strong multi-objective performance.

SAGA can effectively diagnose optimization bottlenecks and evolve informative, necessary objectives across three levels of automation. Beyond final scores, SAGA makes optimization progress interpretable by explicitly exposing objective updates across iterations. As shown in Figure 5(b), the co-pilot, semi-pilot, and autopilot settings differ in the source of guidance, yet share a common workflow: identify the dominant bottleneck from population-level trends and intervene through targeted objective changes. In co-pilot mode, experts can translate mechanistic hypotheses into constraints and scorers. For example, when early generations exhibit disordered CDR3 conformations or weak epitope engagement, SAGA implements CDR3 helix-shape objectives and hotspot-contact scoring in the subsequent iteration, improving structure/interface signals while increasing CDR3–epitope interactions (Figure 5(c)). In semi-pilot mode, experts provide only high-level critique and SAGA operationalizes it into concrete objectives (e.g., pLDDT-focused terms or

explicit hotspot-contact targets). In autopilot mode, SAGA proposes objective additions and weight adjustments purely from iteration-to-iteration trade-offs. Across all settings, SAGA also adapts objective weights to balance competing goals; for instance, when topology metrics saturate early, it can down-weight pTM and up-weight structural-confidence terms to prioritize local reliability without sacrificing global fold quality.

D SAGA FOR CHEMICAL PROCESS DESIGN

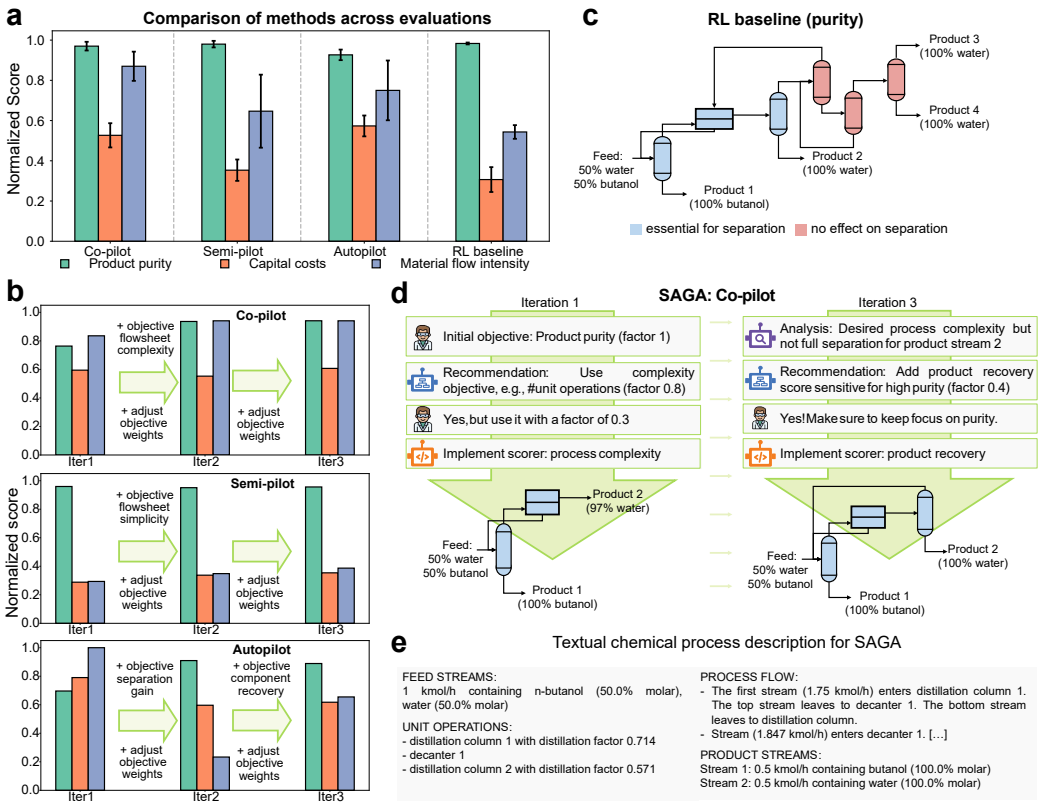

Figure 6: *Results for chemical separation process design.* (a) Comparisons between different levels of SAGA and baseline RL (trained to maximize product purity) with evaluation metrics (higher is better). Our selected task is to design process flowsheets for separation of an azeotropic butanol/water mixture with varying feed compositions (between 2%/98%). (b) The comparisons of different iterations of the three SAGA levels ran for three iterations with the same evaluation metrics. In each iteration, SAGA will analyze the processes, create new objectives, run the RL optimization, and select the best candidates across all current iterations. (c) Exemplary designed process by baseline RL agent for separating a 50%/50% butanol-water mixture, demonstrating that maximizing the product purity only leads to full separation but also in applying unit operations that do not have an effect on the separation quality (marked in red), as the RL agent is not penalized for using unnecessary operations. (d) Workflow for using SAGA co-pilot with agent and user actions for two iterations, resulting in optimal process design for separating a 50%/50% butanol-water mixture. (e) Text description of an exemplary chemical process that is used by SAGA.

Finally, we consider the use of SAGA for chemical process engineering applications, which is of high practical relevance within the chemical industry. While chemical process engineering has historically developed various heuristics and optimization-based approaches for the design of process flowsheets over the last decades, cf. (Biegler et al., 1997; Turton et al., 2008), more recently generative ML models combined with Reinforcement Learning (RL) have been investigated as a promising approach to automate chemical process design, with exemplary applications in reaction synthesis, separation, and extraction processes (Gao & Schweidtmann, 2024; Göttl et al., 2025; Stops et al.,

2023). However, the focus on single design objectives predefined by human experts (Gao & Schweidtmann, 2024) can result in process flowsheets that lack characteristics of practical relevance and thus require iterative refinement in subsequent manual steps. It has also proven challenging to the LLM/agentic domain (Rupprecht et al., 2025; Du & Yang, 2025), and only a few recent studies have utilized LLMs to optimize parameters for given chemical processes, e.g., in (Zeng et al., 2025). Here SAGA is used to advance automation of the chemical process design loop for separation of mixtures by proposing objectives that lead to more practical designs.

SAGA finds practically relevant processes by refining objectives in RL-based flowsheet design. As shown in Figure 6(a) and (c), using only the key objective of product purity for designing a separation process, i.e., the baseline, incentivizes the baseline RL agent to propose a flowsheet that results in optimal purity. However, without considering further objectives, such as capital costs, the RL agent might place unit operations that do not have an effect on the separation quality, or use a more complex flowsheet structure than needed. When using SAGA, we observe increased objective scores for capital costs and material flow intensity compared to the RL baseline, while the purity is at a high level that is close to ideal separation (Figure 6(a)). SAGA effectively assists human experts (at the co- and semi-pilot levels) in the iterative refinement and addition of objectives which includes balancing multiple objective weighting factors, illustrated exemplary with the human-agent interaction in Figure 6(d) and quantified along the iterations in Figure 6(b). Also at the autopilot level, we observe significantly increased process performance compared to the RL-based chemical process design, such that SAGA enables automation of practically improved chemical processes.

SAGA identifies and implements objectives that align with early-stage chemical process design. Starting with maximizing the product purity as an initial objective, SAGA proposes a diverse set of useful objectives, such as process complexity, component recovery, and material efficiency, to be considered in the design. In fact, we find that SAGA identifies and implements suitable process objectives and scoring functions at all levels (co-, semi- and autopilot), leading to higher overall scores on the evaluation metrics, cf. Figure 6(a). Adding additional objectives to the optimization also requires setting appropriate objective weights, see, e.g., Figure 6 (b) and (d), since we combine multiple objectives into one reward for the RL design agent. As the design is sensitive to these objective weights, we see larger variation in individual objectives with less human intervention, particularly, for material flow intensity at semi- and autopilot level, see Figure 6(a). Notably, the product purity across all levels also shows some slight variations, which can be explained by partly conflicting objectives, as SAGA achieves high gains in capital costs and material flow intensity compared to the baseline. Therefore, SAGA is able to enrich the chemical process design by relevant early-stage objectives.

SAGA effectively analyzes chemical processes based on text representations. To analyze the flowsheet designs and propose new objectives, SAGA requires a text representation of the chemical processes. As indicated in Figure 6(e), we represent the flowsheets as natural text description with the four categories: feed streams, unit operations, process flow, and product streams. SAGA successfully utilizes this representation to analyze process design potentials, e.g., by highlighting suboptimal product purity, as shown in Figure 6(d), and identifying unit operations and flowsheet patterns that result in desired separation. These examples highlight the capability of SAGA to capture complex process context and advance automated chemical process design.

E METHODS

E.1 SAGA FRAMEWORK

E.1.1 OVERVIEW

SAGA transforms open-ended scientific discovery into structured, iterative optimization by dynamically decomposing the high-level discovery goal into computable objectives and scoring functions. The framework comprises two nested loops: an *outer loop* that explores and evolve objectives for the optimization; and an *inner loop* that systematically optimize candidates against the scoring functions of the specified objectives.

The workflow proceeds as follows (Figure 1(c)): users provide a *high-level goal* in natural language, such as “design novel antibiotic small molecules that are highly effective against Klebsiella pneu-

moniae bacteria," and can optionally provide more context information such as task background or specific requirements, as well as initial objectives and initial candidates as the starting points. The system then iterates through four core agentic modules: (1) *Planner* formulates measurable objectives aligned with the overarching goal and informed by previous analysis; (2) *Implementer* realizes executable scoring functions for proposed objectives; (3) *Optimizer* optimizes candidates by iteratively generating and assessing candidates that maximize the objective scores, as the inner loop; and (4) *Analyzer* assesses progress and determines whether to continue optimization or terminate upon goal satisfaction.

E.1.2 Core Modules

Planner. This agent decomposes the scientific goal into concrete optimization objectives at each iteration. Given the goal and current candidate analysis, it identifies gaps between the present state and desired outcome, proposing computable objectives with associated names, descriptions, optimization directions (e.g., maximize or minimize), and (optional) objective weights.

Implementer. This agent instantiates callable scoring functions for proposed objectives. When the implementations are not provided with the initial objectives, it develops custom implementations by conducting web-based research on relevant computational methods and software packages, then implements and validates the scoring function within a standardized Docker environment to ensure executability and correctness.

Optimizer. This module constitutes the inner optimization loop. Given objectives and their scoring functions, it employs established optimization algorithms to generate improved candidates. The process alternates between batch evaluation using objective scoring functions and generation of new candidates designed to outperform previous iterations. The architecture accommodates diverse optimization strategies, such as prompted language models, trained reinforcement learning agents, or any optimization algorithms, enabling flexible tuning. The default optimizer for SAGA is a simple LLM-based evolutionary algorithm with three essential steps: (1) candidate proposal: LLM proposes new candidates based on the current candidate pool, (2) candidate scoring: scores all proposed new candidates, and (3) candidate selection: constructs the updated pool from the previous pool and new candidates.

Analyzer. This agent evaluates optimization outcomes and recommends subsequent actions. It examines objective score trajectories, investigates candidate properties using computational tools, and synthesizes insights into actionable reports. The analyzer also determines whether candidates satisfy the goal and can trigger early termination when success criteria are met.

E.1.3 Autonomy Levels

SAGA aligns with human scientific discovery workflows and seamlessly supports human intervention at varying levels. We define three operational modes based on the degree of autonomy (Figure 1(d)):

- **Co-pilot**: Human scientists collaborate closely with both the planner and analyzer. At each iteration, these agents generate proposals (i.e., new objectives from the planner, and analysis from the Analyzer), which scientists can either accept directly or revise based on domain expertise. The implementer and optimizer operate autonomously within the outer loop, executing the human-approved objectives. This mode maximizes human control while automating implementation details.
- **Semi-pilot**: Human intervention is limited to the analyzer stage. Scientists review progress reports and optimization outcomes, providing feedback that guides the planner's subsequent objective proposal. The planner, implementer, and optimizer function autonomously, but strategic decisions about continuation, termination, or pivoting remain human-guided. This mode balances automation with critical oversight at decision points.
- **Autopilot**: All four modules operate fully autonomously without human intervention. The system independently plans objectives, implements scoring functions, optimizes candidates, and analyzes results. This mode enables large-scale automated exploration when domain constraints are well-specified and trust in the system is established.

This tiered design ensures scientists can interact with SAGA in ways that maximize productivity for their specific research context, from hands-on collaboration to fully autonomous discovery.

E.2 Task Configurations

Antibiotic discovery. We formulate this task to discover novel small-molecule antibiotics against *K. pneumoniae*. In practice, we set the high-level discovery objective as designing candidates with strong predicted antibacterial efficacy while maintaining structural novelty, favorable predicted mammalian-cell safety, avoidance of dominant known-antibiotic motifs, and practical feasibility aligned with purchasable-like chemical space for wet-lab validation. Both the high-level goal and the accompanying contextual information explicitly encode our design target and related constraints. For each SAGA instance, our initial objectives are always to maximize antibiotic activity, molecule novelty, and synthesizability, while minimizing toxicity to human and similarity to known antibiotic motifs in the designed molecules. During the loop of optimization, we use the default LLM-based evolutionary algorithm. The initial populations are selected from the Enamine REAL Database (Shivanyuk et al., 2007), which also serves as the first group of molecules in the parent node. We provide molecules from the parent node, individual score from each objective, and an aggregated score (by product individual scores) to the LLM, and generate new molecules after crossover operation. We then select the top molecules based on the list containing both generated molecules and molecules from the parent node. To encourage diversity, we consider a cluster-based selection strategy (Butina cluster-based selection (Butina, 1999)). Finally, we combine all scoring functions into a single scalar value by product of expert to discourage ignoring any objective and select top molecules across all iterations. We use the standard implementation of the planner, implementer, optimizer, and analyzer modules.

Inorganic materials discovery. We consider two materials inverse design tasks. The first task aims to design permanent magnets with low supply chain risk, specified by two objectives: magnetic density higher than 0.2 Å^{-3} and HHI score below 1500. The initial objective is set to maximize magnetic density. The SAGA Co-pilot mode is deployed with iteratively refined objectives: maximizing magnetic density in the first iteration, followed by the addition of HHI score minimization in the second. This task provides a direct comparison with MatterGen (Zeni et al., 2025). The second task is to design superhard materials for precision cutting, requiring high hardness, high elastic modulus, appropriate brittleness, and thermodynamic stability. The high-level goal and contextual information explicitly encode design requirements and constraints. For each SAGA experiments of superhard materials design, initial objectives are set to maximize bulk modulus and shear modulus, which are important indicators for screening superhard materials (Mansouri Tehrani et al., 2018). The optimization loop employs a default LLM-based evolutionary algorithm. Initial populations are randomly sampled from the Materials Project database (Jain et al., 2013), which also serves as the first group of crystals in the parent node. Based on LLM-proposed chemical formulas, pretrained diffusion models provide 3D crystal structures, with geometric optimization performed using universal ML force fields (Yang et al., 2024). Evaluators assign objective scores based on the 3D structure of each crystal. Chemical formulas from the parent node and individual score of each objective were provided to the LLM, which generated new formulas through crossover operations. Optimal structures are then selected via Pareto front analysis from a combined pool of generated and parent crystals. The standard implementation of the planner, implementer, optimizer, and analyzer modules are used for all materials design tasks.

Functional DNA sequence design. Functional DNA sequences, also referred to as cis-regulatory elements (CREs), primarily include enhancers and promoters and play a central role in regulating gene expression levels (Wittkopp & Kalay, 2012; de Boer & Taipale, 2024). We focus on the *de novo* design of cell-type-specific enhancers and promoters across multiple cellular contexts, including HepG2 (enhancer and promoter), K562 (enhancer and promoter), SKNSH (enhancer and promoter), A549 (promoter only), and GM12878 (promoter only). The selection of these cell lines is constrained by the availability of high-quality, publicly accessible datasets. We formulate the discovery task using a high-level natural-language prompt that specifies the objective of generating functional DNA sequences with strong cell-type specificity. Both the high-level goal and the accompanying contextual information explicitly encode target and off-target cell-type constraints. During optimization, the primary objectives are to maximize predicted expression in the target cell line while suppressing activity in non-target cell lines. For optimization, we employ a default LLM-

based evolutionary algorithm. The initial population is selected by sampling from a pool of random DNA sequences. During candidate selection, we keep all candidates that satisfy the expression selectivity. Moreover, we also keep top 50% diverse candidates measured by average pairwise Hamming distance. Finally, we use the standard implementation of the outer loop and the analyzer, planner, and implementer agents.

Nanobody design. Nanobodies, also known as single-domain antibodies derived from camelids, represent a promising therapeutic modality due to their small size, high stability, and ease of engineering (Muyldermans, 2021). We formulate this task as the *de novo* design of high-affinity nanobodies targeting PD-L1 (Programmed Death-Ligand 1), a key immune checkpoint exploited by tumors for immune evasion. Our high-level discovery objective specifies designing candidates with strong predicted binding affinity, favorable interface quality, and practical sequence developability. Both the high-level goal and the accompanying contextual information explicitly encode the binding target and key residue contacts on the PD-L1 epitope. We adopt the nanobody scaffold provided by BoltzGen, based on caplacizumab (PDB: 7EOW), which defines the framework regions and three designable Complementarity-Determining Region (CDR) loops with variable-length insertions. The initial population is sampled from LLM-generated random nanobody sequences. For each SAGA instance, we initialize the design objective to match BoltzGen, aiming to maximize binding interface quality (protein iPTM and pTM) and interface interaction features (hydrogen bonds, salt bridges, and buried surface area), while minimizing interface prediction uncertainty (PAE), hydrophobicity, and sequence liabilities. During optimization, we employ a genetic algorithm with hybrid crossover operators consisting of 40% CDR swap, 40% single-point crossover, and 20% uniform crossover, together with random CDR mutation. To encourage diversity while maintaining quality, we use tournament selection for parent pairing and elitism-aware survival selection. Candidate evaluation uses structure prediction with Boltz2. We apply diversity filtering based on CDR-only sequence similarity, rejecting any candidate with more than 50% CDR identity to a selected sequence. Finally, we combine objectives using normalized weighted aggregation and select top candidates based on rank-based scoring across all iterations. We use the standard implementation of the planner, implementer, optimizer, and analyzer modules.

Chemical process design. We use SAGA for the design of chemical process, more specifically separation process flowsheets, which is a central task in chemical engineering (Biegler et al., 1997; Turton et al., 2008; Gao & Schweidtmann, 2024). The high-level goal is formulated as a natural language prompt targeting the design process flowsheets for the steady-state separation of an azeotropic binary mixture of water and ethanol at different feed compositions into high purity streams. For the optimizer designing process flowsheets in the inner loop, we use an RL agent based on the separation process design framework by Göttl et al. (2025). The the action space of the RL agent comprises the (1) selection of suitable unit operations, such as decanters, distillation columns, and mixers with their specifications, and (2) determination of the material flow structures (including recycles) that connect the unit operations. We translate the RL-internal matrix representation of process flowsheets to a text description. We use the standard implementation of the analyzer, planner and implementer, and the text description is provided to the agents in the outer loop. The proposed new objectives – with corresponding weighting factors to aggregate the objective values into one reward value – are automatically added to the RL framework and used for the next iteration of process design, which always starts from scratch without an initial population, whereby the initial objective for the first iteration is the product purity. We thus focus on the iterative addition and refinement of suitable chemical process design objectives and their weighting factors.

E.3 Task Evaluations

We validate the performance of SAGA on each individual task by setting up a set of evaluation metrics. The evaluation metrics are unseen during the online running procedure of SAGA. Below, we briefly discuss the evaluation procedures for each task.

Antibiotic discovery. To mimic real-world lab experiment, we consider evaluating the candidates from the perspectives of biological objectives, synthesizability, and drug likeness. These three areas can be covered with 11 different computational metrics. To evaluate generated molecules with biological objectives, we consider antibiotic activity score, novelty score, toxicity score, and known motif filter score as metrics. For synthesizability, we consider a synthetic accessibility score as the metric. Last but not least, to evaluate drug likeness, we consider QED score, DeepDL prediction

score, molecular weight score, PAINS filter score, BRENK filter score, and RING score as metrics. When evaluating candidates proposed by baselines and SAGA, we compute both the absolute score and pass rate of the top 100 molecules selected using each model's optimization objectives for a fair comparison.

Inorganic materials design. To evaluate material properties, density functional theory (DFT) calculations were employed to determine the electronic, magnetic, and mechanical properties of generated materials, as well as energy above hull (Zeni et al., 2025; Chen et al., 2025a). HHI scores were computed using the pymatgen package (Ong et al., 2013). In the task of designing permanent magnets with low supply chain risk, two objectives were specified: magnetic density higher than 0.2 Å^{-3} and HHI score less than 1500. In the task of designing superhard materials for precision cutting, the evaluation metrics include Vickers hardness, bulk modulus, shear modulus, Pugh ratio, and energy above hull.

Functional DNA sequence design. To emulate real-world experimental evaluation, we adopt a blind computational assessment framework based on five established computational oracles drawn from prior studies (Gosai et al., 2024; DaSilva et al., 2024; Chen et al., 2025c; Lal et al., 2024). As a representative task, we focus on the design of HepG2-specific enhancer sequences. The evaluation metrics include statistical comparisons of MPRA-measured expression between the target cell line and non-target cell lines (e.g., HepG2 vs. K562 and HepG2 vs. SKNSH), together with knowledge-driven criteria such as transcription factor motif enrichment, sequence diversity, and sequence stability.

Nanobody design. To emulate real-world therapeutic antibody development, we adopt a comprehensive computational assessment framework spanning structural quality, binding interface characteristics, epitope engagement, and sequence developability. Structural quality is evaluated using confidence metrics from structure prediction, including interface predicted TM-score (iPTM), overall predicted TM-score (pTM), and predicted local distance difference test (pLDDT) for the full binder, the CDR regions, and the CDR3 loop, together with predicted aligned error (PAE) at the binding interface. Binding interface characteristics are assessed using physically interpretable interaction metrics computed on predicted complex structures, including the number of hydrogen bonds, salt bridges, and the change in solvent-accessible surface area upon binding (ΔSASA). We further quantify epitope engagement using CDR-hotspot and CDR3-hotspot contact counts, measuring how many CDR residues fall within contact distance of predefined PD-L1 epitope residues. Sequence–structure compatibility is assessed with ProteinMPNN (Dauparas et al., 2022) by computing the negative log-likelihood score and expected sequence recovery on the predicted structure. We validate CDR3 secondary structure using DSSP assignment (Kabsch & Sander, 1983) on predicted structures, verifying proper alpha-helical content within the specified positional constraints. Sequence developability is evaluated with a liability score that penalizes known sequence liabilities such as deamidation sites, oxidation-prone residues, and aggregation motifs. To improve robustness to predictor-specific biases, we perform structure prediction with both AlphaFold3 and Boltz2. When evaluating candidates proposed by baselines and SAGA, we report metrics computed under both structure prediction backends and select top candidates using rank-based aggregation across objectives.

Chemical process design. To cover early-stage process design goals, we utilize the short-cut simulations models developed in (Göttl et al., 2025) and calculate three process performance indicators. These are used as the evaluation metrics and include the product purity, capital costs, and material flow intensity. The product purity corresponds to the average purity of the product streams received from the simulation. The capital costs represent the sum of individual unit operation costs estimated on a simple heuristic, similar to (Göttl et al., 2025). For the material flow intensity, we calculate the recycle ratios and introduce penalty terms for excessive ratios and very small streams ($< 1\%$ of the feed stream).