FROM TARGET MOLECULE TO SIMULATION-READY DIGITAL MICROBIAL LIFE

Anonymous authors
Paper under double-blind review

ABSTRACT

Engineering microorganisms to produce target molecules is a central goal of synthetic biology, yet designing a simulation-ready microbial system remains labor-intensive and dependent on expert knowledge. Although large language model (LLM)-based scientific agents can automate parts of this workflow, generating executable biological designs and evaluating their quality remain challenging. We propose *digital microbial life design* as a new end-to-end problem of transforming a target molecule into a simulation-ready microbial design. To solve this problem, we introduce **Origo**, a multi-agent system that structures the design workflow using an expert-designed primitive task pool. Each primitive task is bound to specialized MCP tools, while hierarchical multi-agent coordination assigns tasks to specialized agents and coordinates their execution to construct GEMs and identify enzyme candidates. To address the corresponding evaluation gap, we introduce **DLBench**, a benchmark for evaluating digital microbial life design through network-level executability and reaction-level biological plausibility, with 40 target products and 22 rate-limiting enzymes. Experiments on DLBench show that Origo outperforms general-purpose LLMs and biomedical agents across both tasks. This work provides a foundation for AI-driven microbial design, downstream simulation, and hypothesis generation for wet-lab validation.

1 INTRODUCTION

Engineering microorganisms to produce a target molecule is a central goal of synthetic biology and metabolic engineering. Microbial fermentation provides a scalable and greener alternative to chemical synthesis (Liu & Nielsen, 2019). The design process typically starts with a target molecule and seeks a biosynthetic pathway that enables a microbial host to produce it (Ko et al., 2020). In practice, this process is slow and highly dependent on expert knowledge (Cho et al., 2022). Recent advances in large language models (LLMs) and tool-augmented agents show promise for automating parts of this workflow (M. Bran et al., 2024; Roohani et al., 2025). This motivates a broader challenge: can we autonomously construct simulation-ready *digital microbial life* by starting from a target molecule?

In this paper, we use digital microbial life to denote a computationally specified microbial design that connects a target molecule to the metabolic machinery required to produce it. Such a design requires a genome-scale metabolic model (GEM) that represents the metabolic network for target production, together with enzyme candidates for the reactions along the biosynthetic pathway. Existing LLM-based biomedical agents are not designed for this problem. Biomedical agents such as Biomni (Huang et al., 2025) and STELLA (Jin et al., 2025) provide broad tool inventories but leave task decomposition and tool selection to the LLM. This requires the agent to infer the task dependencies, including how pathway planning guides GEM construction and identifies reactions requiring enzyme candidates. Incorrect task decomposition can produce incomplete or inconsistent designs, while irrelevant tools increase the burden of tool selection.

Beyond system design, evaluation must establish whether the outputs constitute simulation-ready digital microbial life, rather than plausible pathway descriptions alone. At the network level, a biosynthetic pathway must connect to a coherent, mass-balanced metabolic network (Zhang & Hua, 2016; Gu et al., 2019). The resulting GEM must be encoded in a machine-readable format such as Systems Biology Markup Language (SBML) (Hucka et al., 2019; Olivier & Bergmann, 2018) and support flux balance analysis (FBA) (Orth et al., 2010) to assess growth and target production. At the

Reviewers: please read the Reviewer Guidelines (iclr.cc/Conferences/2027/ReviewerGuidelines) and the AI Policy for Reviewers (iclr.cc/Conferences/2027/AIPolicyForReviewers). If you used AI to expand, edit, or polish your review, please provide the input text to the LLM. Better still, consider skipping the LLM and submitting your original text: we, and the authors, are much more interested in your unedited thoughts than in what an LLM has to say. AI-assisted or not, you are putting your name and reputation behind your review: LLM-generated falsehoods, hallucinations or misrepresentations are subject to disciplinary action, which may include desk-rejecting all papers you have authored.

reaction level, enzyme candidates must be evaluated for their ability to catalyze the required reactions, not merely for the validity of their protein identifiers. A dedicated benchmark is therefore needed to jointly assess network-level executability and reaction-level biological plausibility (Blazeck & Alper, 2010; Simeonidis & Price, 2015).

To address these system and evaluation challenges, we introduce **Origo**, a multi-agent system for constructing simulation-ready digital microbial life from target molecules, and **DLBench**, a benchmark for evaluating this design capability. The main contributions of our work are summarized as follows:

- We propose *digital microbial life design* as a new end-to-end problem of transforming a target molecule into a simulation-ready microbial design, and formulate it through the complementary tasks of GEM generation and reaction-level enzyme mining.
- We introduce **Origo**, a multi-agent system that structures the design workflow using an expert-designed primitive task pool. Each primitive task is bound to specialized MCP tools, while hierarchical multi-agent coordination assigns tasks to specialized agents and coordinates their execution to construct GEMs, and identify enzyme candidates.
- We introduce **DLBench**, a benchmark for evaluating *digital microbial life design* through network-level executability and reaction-level biological plausibility, with 40 target products and 22 rate-limiting enzymes.
- Experiments on DLBench show that Origo outperforms general-purpose LLMs and biomedical agents in both GEM construction and reaction-level enzyme mining.

2 Related Work

Agentic Systems for Scientific Reasoning and Scientific Workflows. Recent research has moved beyond simple prompting toward agentic systems that integrate planning, retrieval, tool use, and iterative refinement. For literature-centered scientific reasoning, OpenScholar (Asai et al., 2024) emphasizes retrieval-augmented synthesis over scientific literature, while PaperQA2 (Skarlinski et al., 2024) uses explicit query planning and evidence aggregation for scientific question answering. SciAgents (Ghafarollahi & Buehler, 2025) further extends this paradigm to multi-agent scientific ideation and hypothesis generation. Domain-specific agents have also emerged in chemistry and biomedicine. ChemAgent (Yu et al., 2025) highlights the importance of task orchestration and external-tool integration in chemistry. In biomedicine and biology, STELLA (Jin et al., 2025) and Biomni (Huang et al., 2025) demonstrate the use of multi-agent architectures and expanding tool ecosystems for scientific research, with Biomni also serving as a representative biology-focused agent in recent benchmarking and safety studies (Nasri et al., 2026; Provatas et al., 2026).

Computational Systems for Pathway Design and Metabolic Model Construction. Computational metabolic engineering has traditionally relied on explicit model representations, curated pathway knowledge, and executable metabolic reconstructions. Pathway Tools (Karp et al., 2002; 2016; 2021) solve this problem by integrating pathway prediction, database construction, and systems biology analysis. More recent methods like ART (Radivojević et al., 2020) recommends synthetic biology designs under experimental constraints, while COVRECON (Li et al., 2023) combines genome-scale metabolic reconstruction with inverse modeling to infer metabolic network changes from metabolomics data. Genome-scale reconstruction tools such as CarveMe (Machado et al., 2018) and ModelSEED (Seaver et al., 2021) similarly derive candidate networks from annotated genome sequences and curated biochemistry, followed by retrieval-based filtering and completion. In contrast, de novo construction from a target molecule must decide which metabolic infrastructure and biosynthetic reactions to introduce without a sequenced organism as the starting point, making the search space and biological-grounding requirements qualitatively different.

AI Methods for Reaction and Enzyme Modeling. Reaction- and enzyme-level models provide biological grounding for pathway design, since pathway-level plausibility does not ensure that individual reactions are feasible. RXNRECer (Shi et al., 2026) uses active learning and protein language models for fine-grained enzyme function annotation, while CLIPZyme (Mikhael et al., 2024) aligns protein and reaction representations through contrastive learning for enzyme-reaction prediction. Other approaches target enzyme-reaction retrieval and catalytic properties: FGW-CLIP (Zhou et al., 2025) formulates enzyme-reaction screening as a contrastive learning problem, and

EnzyCLIP (Khan et al., 2025) uses a cross-attention dual encoder to predict kinetic constants. Together, these methods complement pathway-level reasoning by providing finer-grained models of reaction feasibility and enzyme function.

3 METHOD

3.1 PROBLEM FORMULATION: DESIGNING DIGITAL MICROBIAL LIFE

Given a target molecule x, the goal of *digital microbial life design* is to construct a simulation-ready microbial specification that connects cellular metabolism to the production of x. We represent such a design as

$$\mathcal{D}(x) = \big(\mathcal{G}_x, \mathcal{E}_x\big), \tag{1}$$

where $\mathcal{G}_x$ is a machine-readable GEM containing a feasible biosynthetic route to x, and $\mathcal{E}_x$ is a set of candidate enzymes that can support the constituent reaction steps. Origo infers and assigns biologically reasonable host chassis and growth medium settings from the structural and metabolic properties of the target molecule. Unlike pathway suggestion, the desired output is a computational artifact that can be parsed, simulated, inspected, and subsequently refined for experimental validation.

We operationalize this problem through two complementary tasks:

GEM generation maps a target molecule to an SBML-encoded metabolic network,

$$x \longmapsto \mathcal{G}_x = (\mathcal{M}, \mathcal{R}, \mathbf{S}, \mathbf{lb}, \mathbf{ub}, r_{\text{bio}}, r_x), \tag{2}$$

where $\mathcal{M}$ and $\mathcal{R}$ denote metabolite and reaction sets, $\mathbf{S}$ is the stoichiometric matrix, $\mathbf{lb}$ and $\mathbf{ub}$ are flux bounds, and r_{bio} and r_x are the biomass and target-production reactions. The inferred chassis and medium consequently determine organism-specific metabolic capabilities, biomass composition, and the available exchange environment. A successful model must be structurally valid, mass-balanced where chemically specified, growth-competent, and capable of carrying a positive target-production flux.

Enzyme mining maps a reaction-level enzyme query q which is specified by a reference enzyme and its biochemical context to a ranked candidate set,

$$q \longmapsto \mathcal{E}_q = \{(e_j, s_j)\}_{j=1}^{K}, \tag{3}$$

where e_j is a verifiable protein candidate and s_j is its evidence-based ranking score. Candidates should be consistent with the query in catalytic function.

The two tasks address different levels of the same design problem. GEM generation establishes whether the target can be connected to a coherent, simulatable cellular network; enzyme mining grounds selected reaction steps in implementable protein candidates. Together, they transform a molecular objective into the network-level and reaction-level components required for simulation-ready digital microbial life.

3.2 ORIGO: A MULTI-AGENT SYSTEM FOR DIGITAL LIFE DESIGN

The overall architecture of Origo is shown in Figure 1A, which organizes the system into a user interface (UI) layer, an agent layer, and a tool layer. The UI layer receives queries from the users and presents the resulting designs to users. The tool layer provides standardized access to biological databases, computational models, and analytical services through Model Context Protocol (MCP) (Anthropic, 2024). In the agent layer, Origo designs a **primitive task pool** and **task-scoped tool binding** that constrain and structure agent reasoning. On top of these task and tool constraints, **hierarchical multi-agent coordination** assigns each primitive task to a specialized execution agent and coordinates task-level reasoning, execution, and validation across agents.

3.2.1 PRIMITIVE TASK POOL

Instead of generating an unconstrained task decomposition for every user query, Origo represents the complex design objectives as compositions of predefined primitive tasks. Each primitive task specifies a task description, expected inputs and outputs, and an associated set of tools required for its execution. The primitive task pool $\mathcal{P}$ is defined as $\mathcal{P} = \{T_1, T_2, \ldots, T_N\}$, where each T_i

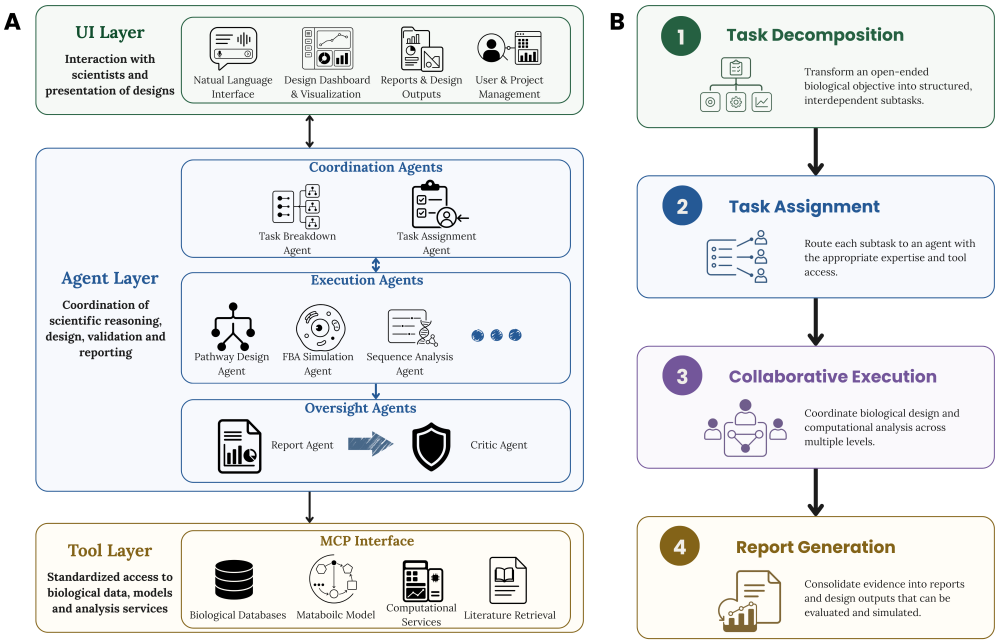

Figure 1: Origo Multi-Agent System Overview. **(A)** The Origo architecture consists of three layers: Agent Layer, Tool Layer, and UI Layer. **(B)** The system completes digital microbial life design in four steps: task decomposition, task allocation, collaborative execution, and report generation.

corresponds to a reusable primitive task. Primitive tasks are designed by domain experts to capture recurring operations in digital microbial design, such as pathway design, GEM assembly, sequence analysis, and enzyme screening. Each primitive task is represented as

$$T_i = (D_i, I_i, O_i, \mathcal{M}_i), \tag{4}$$

where D_i is the task description optimized by domain experts, I_i and O_i denote the expected input and output specifications, and $\mathcal{M}_i$ denotes the subset of MCP tools associated with the task. This primitive-task design provides a standardized interface between planning and execution.

3.2.2 Task-Scoped Tool Binding

Origo integrates more than 30 specialized MCP tools covering metabolic pathway design, GEM construction and enzyme mining. Let $\mathcal{M}$ denote the complete tool inventory. For each primitive task T_i, Origo defines a task-specific tool subset $\mathcal{M}_i \subseteq \mathcal{M}$. Only the tools in $\mathcal{M}_i$ are exposed to the execution agent when T_i is active.

Formally, if the agent action space under the complete tool inventory is $\mathcal{A}(\mathcal{M})$, execution of primitive task T_i is restricted to

$$\mathcal{A}(T_i) = \mathcal{A}(\mathcal{M}_i). \tag{5}$$

This restriction reduces irrelevant tool selection and limits the search space of agent actions, while preserving access to the specialized capabilities required by the current task.

3.2.3 Hierarchical Multi-Agent Coordination

The agent layer combines specialized expertise with centralized coordination through three tiers: Coordination Agents, Execution Agents, and Oversight Agents. The Coordination Agents, comprising the **Task Breakdown Agent** and **Task Assignment Agent**, decompose the overall objective and route subtasks to the appropriate agents. The Execution Agents consist of domain-expert agents that carry out the assigned subtasks and return the resulting design proposals. The Oversight Agents, consisting of the **Report Agent** and **Critic Agent**, review and synthesize the outputs into coherent results. This hierarchical architecture allows each agent to specialize in a distinct scientific role while preserving consistency across the overall engineering task.

Agent collaboration follows a four-stage workflow (Figure 1B). In the **task decomposition** stage, an open-ended biological objective is transformed into a structured set of interdependent subtasks. In **task assignment**, each subtask is routed to the agent with the appropriate expertise and tool access. During **collaborative execution**, biological design and computational analysis are coordinated across multiple agents and levels of abstraction. Finally, **report generation** consolidates the resulting evidence into reports and design outputs that are ready for evaluation and simulation.

3.3 GEMs Generation

When Origo receives a request to generate a genome-scale metabolic model, the system enters the task-planning stage. The **Task Breakdown Agent** reads the predefined task pool and decomposes the user query according to the mandatory task chain specified by the matched knowledge template. This process produces four sequential subtasks: **GEMs Pathway Design**, **GEMs Assemble**, **GEMs Repair**, and **GEMs Refinement** (Figure S1).

GEMs Pathway Design. The execution agent first retrieves the available KEGG pathways (Kanehisa et al., 2023) (Table S2) and selects the core pathways required to support the biological model, such as glycolysis, the tricarboxylic acid (TCA) cycle, and pyruvate metabolism. It then adds pathways specific to the target product class, such as terpenoid or flavonoid biosynthesis, and saves the initial pathway design locally.

GEMs Assemble. Using the pathway design from the preceding stage, the agent maps reactions against MetaNetX (Moretti et al., 2021) and SEED (Overbeek et al., 2014), adds exchange reactions and transporters, and assembles the model in SBML format. It subsequently runs FBA and produces a structured diagnostic report describing the state of the assembled model.

GEMs Repair. If the diagnostic report indicates that repair is required, the agent traces the reaction network backward to identify each blocking dead-end precursor, its depth, and the associated pathway chain. It also checks the completeness of transport and secretion reactions. Then the agent solve each blocking precursor using multiple feasible strategies. The system then recursively re-evaluates the network until the target product is connected to central metabolism.

GEMs Refinement. The final execution agent evaluates the generated GEM using FBA simulations, sets biologically reasonable flux bounds, and removes blocked reactions.

3.4 Enzyme Mining

The workflow of enzyme mining is shown in Figure S3. Enzyme mining is divided into four tasks: **Enzyme Structure Analysis**, **Enzyme Sequence Analysis**, **Enzyme Candidate Clustering**, and **Enzyme Screening**. Rather than treating enzyme identification as a single-step retrieval problem, each stage narrows the candidate space using the evidence accumulated from the stages before it, moving from structural and sequence homology, through evolutionary organization, to final candidate selection.

Enzyme Structure Analysis. Origo retrieves the query enzyme using the input UniProt ID and uses Foldseek (Van Kempen et al., 2024) to search for proteins that are structurally homologous to the query enzyme. This step provides an initial structure-based filter for potential enzyme candidates.

Enzyme Sequence Analysis. Origo subsequently uses MMseqs2 (Steinegger & Söding, 2017) to search the Swiss-Prot (Boeckmann et al., 2003) database for proteins that are sequence-homologous to the candidates obtained in the previous stage. The resulting sequences are then clustered to obtain a non-redundant set of candidate proteins.

Enzyme Candidate Clustering. Origo computes structure-aware sequence embeddings using ProstT5 (Heinzinger et al., 2024) and uses these embeddings to construct a phylogenetic tree. The cluster containing the query enzyme is then selected as the representative cluster.

Enzyme Screening. The candidates in the selected cluster are subsequently screened based on structural similarity. Specifically, Origo predicts the 3D structure of each candidate, after which TM-align (Zhang & Skolnick, 2005) is used to compare the predicted structure with the query structure and compute a TM-score (Zhang & Skolnick, 2004). Origo uses the TM-score as a final structure-based criterion for ranking the candidate set.

4 DLBENCH: BENCHMARKING DIGITAL MICROBIAL LIFE DESIGN

We introduce **DLBench** (Digital Life Bench), a benchmark for evaluating *digital microbial life design* as defined in Section 3.1. DLBench evaluates two complementary tasks: GEM generation and reaction-level enzyme mining.

4.1 GEM GENERATION

For GEM generation, DLBench includes 40 expert-defined target products (Table S4) and requires a GEM in SBML format for each target product. Each GEM is assessed along five criteria: SBML validation, growth validation, product flux validation, Memote score, and expert quality assessment. Full implementation details are given in Appendix B.2.

SBML Validation. Each de novo GEM is parsed with COBRApy (Ebrahim et al., 2013). A model is considered structurally valid if the SBML file, including its Flux Balance Constraints (FBC) package, is parsed without raising an exception and the in-memory stoichiometric matrix, reactions, metabolites, and compartments are instantiated. Models that fail to load are excluded from downstream analyses without manual repair, and each source is scored by the fraction of the 40 target compounds for which it produces a structurally valid model.

Growth Validation. We evaluate the predicted growth phenotype by flux balance analysis (FBA), solving the stoichiometric linear program

$$\mu^* \;=\; \max_{\mathbf{v}} \; v_{\text{biomass}} \quad \text{s.t.} \quad \mathbf{Sv} = \mathbf{0}, \quad \mathbf{lb} \le \mathbf{v} \le \mathbf{ub}, \tag{6}$$

where $\mathbf{S}$ is the stoichiometric matrix, $\mathbf{v}$ the reaction flux vector, and $\mathbf{lb}$ and $\mathbf{ub}$ the flux bounds declared in the SBML file. The biomass reaction is identified by case-insensitive substring search for "biomass" in reaction identifiers. A model is classified as *growth-competent* if its maximized biomass flux is positive ($\mu^* > 0$); non-optimal solutions and models without an identifiable biomass reaction count as not growth-competent. Each source is scored by the fraction of growth-competent models among its submissions.

Product Flux Validation. The export reaction of the target product is identified uniformly for all sources by a deterministic filter that first matches the target compound's MetaNetX or ModelSEED identifier among boundary reactions and then applies exchange/transport naming rules (Appendix B.2). If no matching boundary reaction exists, the product flux is recorded as zero rather than falling back to an arbitrary boundary reaction. An unconstrained FBA scenario is solved per model, each starting from the pristine loaded state. In the unconstrained setting, the product export flux is maximized as a means of validating that the target product pathway is unobstructed:

$$v_{\text{prod}}^{\text{unc}} \;=\; \max_{\mathbf{v}} \; v_{\text{ex}} \quad \text{s.t.} \quad \mathbf{Sv} = \mathbf{0}, \quad \mathbf{lb} \le \mathbf{v} \le \mathbf{ub}. \tag{7}$$

Each source is scored by the fraction of production-competent models among its submissions.

Memote Score. Overall model quality is assessed with the Memote test suite (Lieven et al., 2020), using its official continuous score, a hierarchical weighted average of per-test compliance scores $s_t = 1 - m_t$ over the 27 tests across five scored sections: consistency, as well as metabolite, reaction, gene, and Systems Biology Ontology (SBO) annotations, with per-test weights w_t and section weights W_s:

$$S_s = \frac{\sum_{t \in \mathcal{T}_s} w_t \, s_t}{\sum_{t \in \mathcal{T}_s} w_t}, \qquad S_{\text{Memote}} = \frac{\sum_{s \in \mathcal{S}} W_s \, S_s}{\sum_{s \in \mathcal{S}} W_s} \;\in\; [0, 1]. \tag{8}$$

The precise test inventory, weight assignment, and handling of skipped or missing tests are documented in Appendix B.2. Each source is scored by the mean Memote score over its SBML-loadable models.

Expert Quality Assessment. Complementary to the computational criteria, each model was evaluated by an LLM acting as an expert metabolic engineer using a fixed eight-dimension rubric, with each dimension scored from 1 to 10 (Appendix B.2). The rubric assesses the model as a *de novo* living system, covering pathway design, central metabolism, subsystem completeness, reaction-level plausibility, SBML structure, production logic, and design originality. The overall-quality score is anchored to central metabolism, subsystem completeness, and reaction-level plausibility, as these

dimensions capture the core biological quality of the generated organism, while the remaining dimensions provide complementary evidence for the overall assessment. Overall scores are reported in Figure 2, with per-dimension scores in Figure S2.

4.2 Reaction-Level Enzyme Mining

For reaction-level enzyme mining, we select 22 expert-defined rate-limiting enzymes from diverse biosynthetic pathways as query enzymes (Table S5). The enzyme candidates are assessed along four primary criteria: candidate validity, structural confidence, EC classification accuracy, and kinetic properties.

Candidate Validity. We verify every returned UniProt identifier against UniProt (Consortium, 2015). The candidate validity score is defined as the proportion of the ten returned identifiers that correspond to retrievable UniProt entries, averaged across the 22 query enzymes. Invalid, duplicated, or unresolvable identifiers are retained as failures and are not manually replaced.

Structural Confidence. Each candidate protein is paired with the substrate SMILES associated with its query enzyme, and Boltz (Passaro et al., 2025) is used to predict the corresponding protein–ligand complex. Overall structural confidence is assessed using pTM and pLDDT, while interface-level confidence is assessed using ipTM and ipLDDT. For queries with multiple substrates, the metrics are first computed for each protein–substrate pair and then aggregated for the candidate.

EC Classification Accuracy. We apply CLEAN (Yu et al., 2023) to each candidate sequence and compare its predicted Enzyme Commission (EC) number with the reference annotation of the corresponding query enzyme. Accuracy is computed hierarchically at EC levels 1–4, where EC_1 measures agreement in the broad enzyme class and EC_4 requires an exact four-level functional match. This evaluation measures whether a retrieved candidate is functionally consistent with the queried reaction rather than merely similar at the sequence or structure level.

Kinetic Properties. We use CatPred (Boorla & Maranas, 2025) to predict k_{cat}, K_m, and catalytic efficiency k_{cat}/K_m. To avoid comparing proteins assigned to different reactions, this analysis is restricted to candidates whose predicted EC numbers match the reference annotation at all four levels. For enzymes with multiple substrates, predictions are averaged across substrates for each candidate before system-level aggregation.

Structural Similarity Analysis. As a complementary evaluation, we extract the protein chains and ligand-binding pockets from the predicted complexes and align them to the corresponding query enzyme using US-align. We report both global protein TM-score and pocket TM-score, with the binding pocket defined as residues containing at least one heavy atom within 5 Å of any ligand atom. These metrics assess whether candidates preserve both the overall fold and the local active-site geometry of the reference enzyme.

5 Experiments

We compare Origo with seven general-purpose LLMs and two biomedical agents, Biomni (Huang et al., 2025) and STELLA (Jin et al., 2025) on DLBench. Origo uses DeepSeek V4 Flash as the underlying LLM for its coordination, execution, and oversight agents. The detailed configurations of Origo and baselines are shown in Table S1. Claude declines to answer GEM generation requests due to safety restrictions, so its results are unavailable for this task and it is included only in the reaction-level enzyme mining evaluation.

5.1 GEM Generation Evaluation

We evaluate the remaining nine baselines on the 40-target DLBench GEM generation task introduced in Section 4. Each baseline receives the same target-level request and returns one SBML model per target; no invalid output is manually repaired. Product export reactions are detected with the same deterministic identifier-matching protocol for every source. Figure 2 reports the five benchmark metrics, while Appendix B.2 provides the complete implementation and per-dimension expert scores.

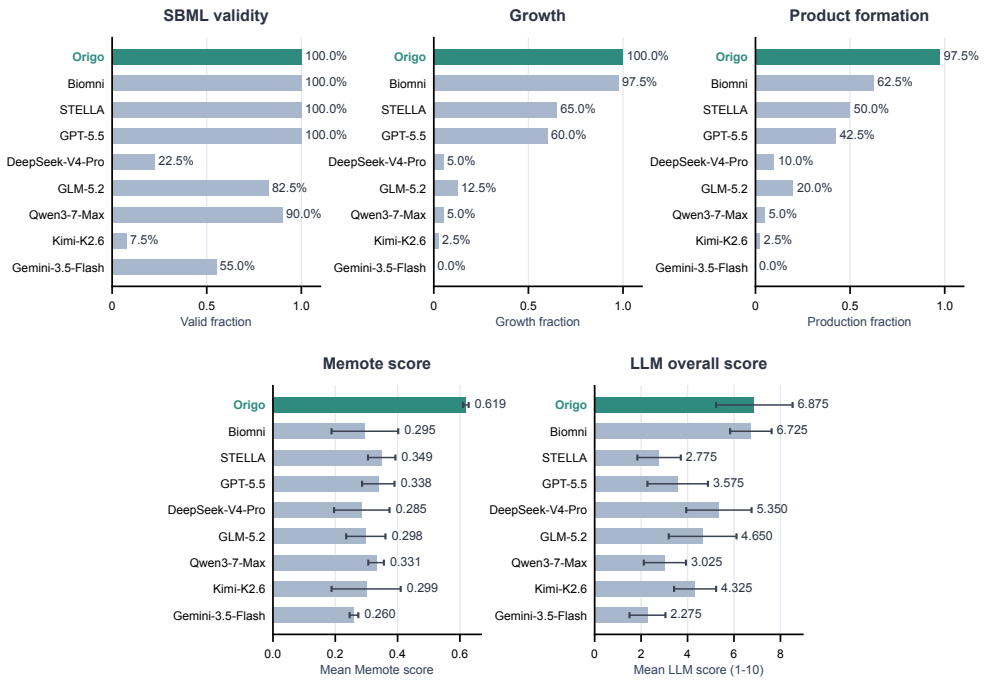

Figure 2: Figure 2: Benchmark of de novo GEM generation across five criteria. SBML validity, growth, and product formation are reported as the fraction of 40 target compounds meeting each criterion. Memote and LLM overall quality are reported as mean $\pm$ SD. **Origo** is highlighted.

SBML Validity. Origo produced structurally valid SBML models for all 40 targets, tied with Biomni, STELLA, and GPT-5.5 at 100% validity. Validity dropped sharply for the remaining LLMs (Qwen3.7-Max 90%, GLM-5.2 82.5%, Gemini-3.5-Flash 55%, DeepSeek-V4-Pro 22.5%, Kimi-K2.6 7.5%), indicating that general-purpose LLMs frequently fail to produce machine-readable models that downstream tools can parse without manual repair.

Growth. All 40 Origo models were growth-competent, followed by Biomni (97.5%), STELLA (65%), and GPT-5.5 (60%); the remaining LLMs rarely produced growing models ($\leq 12\%$). Origo therefore assembles networks that sustain biomass production, rather than merely syntactically valid files.

Product Formation. Origo was production-competent for 39 of the 40 targets (97.5%), substantially ahead of Biomni (62.5%), STELLA (50%), and GPT-5.5 (42.5%), while the other LLMs reached at most 20%. This indicates that Origo's designs route carbon to the target product rather than only supporting growth.

Memote Score. Origo attained a mean Memote score of 0.619, $1.8\times$ that of the next-best source (STELLA, 0.349) and more than twice that of Biomni (0.295). This reflects higher curation quality, annotation completeness, and stoichiometric consistency across the generated models.

Expert Quality Assessment. Under the eight-dimension expert rubric, Origo achieved the highest mean overall-quality score (6.875, compared with 6.725 for Biomni and 5.350 for the next-best LLM, DeepSeek-V4-Pro), indicating that its models are judged as more coherent, plausible, and systematically organized than those of the baselines.

Overall, these results show that Origo produces GEMs that are not only syntactically valid but also executable, more complete, and suitable for downstream flux analysis.

5.2 Enzyme Mining Evaluation

We evaluate all baselines on the 22-query DLBench enzyme track introduced in Section 4. Each baseline returns ten UniProt candidates per query under the same output requirements.

Table 1: Comparison of validity score, confidence score, and EC classification accuracy across different models.

Model	Validity Score	Confidence Score				EC Classification Accuracy			
		pTM	pLDDT	ipTM	ipLDDT	EC_1	EC_2	EC_3	EC_4
Claude-Opus-4.8	0.9000	0.8789	0.8448	0.8388	0.7510	0.6334	0.5591	0.5136	0.3364
DeepSeek-V4-Pro	0.7727	0.8568	0.7921	0.8101	0.6656	0.3409	0.2909	0.2727	0.1909
Gemini-3.5-Flash	0.8136	0.8489	0.7961	0.8019	0.6643	0.3682	0.2727	0.2636	0.1955
GPT-5.5	0.9091	0.8652	0.8380	0.8190	0.7271	0.5045	0.4455	0.4364	0.3000
Kimi-K2.6	0.7318	0.8352	0.7839	0.7745	0.6510	0.2955	0.2091	0.1818	0.1273
Qwen3.7-Max	0.8136	0.7974	0.7621	0.7592	0.6264	0.2273	0.1273	0.1227	0.0773
GLM-5.2	0.7182	0.8367	0.7834	0.7896	0.6297	0.2545	0.1591	0.1545	0.0955
STELLA	0.9773	0.8955	0.8598	0.8584	0.7612	0.8818	0.8227	0.8045	0.5500
Biomni	**1.0000**	0.9137	0.8878	**0.8895**	0.8151	0.9636	0.9318	0.8818	0.5773
Origo	**1.0000**	**0.9293**	**0.9017**	0.8822	**0.8218**	**0.9955**	**0.9636**	**0.9227**	**0.6591**

Candidate Validity. As shown in Table 1, Origo achieves a validity score of 1.0, indicating that all returned UniProt identifiers are verifiable. It outperforms STELLA and every general-purpose LLM while matching Biomni, demonstrating reliable candidate generation.

Structural Confidence. As shown in Table 1, Origo achieves the highest pTM, pLDDT, and ipLDDT scores and remains comparable to Biomni on ipTM. These results indicate stronger overall structural confidence and protein–ligand interface quality among the candidates retrieved by Origo.

EC Classification Accuracy. As shown in Table 1, Origo achieves the highest CLEAN-based EC classification accuracy at all four levels. At the most specific EC_4 level, its accuracy reaches 0.6591, compared with 0.5773 for the next-best method, Biomni. This advantage shows that Origo retrieves candidates that more closely match the queried catalytic function at fine-grained resolution.

Kinetic Properties. The results of k_{cat}/K_m is shown in Figure 3. Among agent-based systems, Origo achieves the highest predicted k_{cat}/K_m on candidates with exact EC_4 agreement. This result suggests that its functionally consistent candidates also have stronger predicted catalytic efficiency. Full k_{cat}, K_m results are provided in Figure S5.

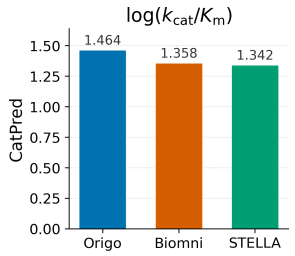

Figure 3: Comparison of the kinetic parameter k_{cat}/K_m.

6 Conclusion

This work formulated digital microbial life design as a new end-to-end problem and decomposed it into GEM generation and enzyme mining. We introduced Origo, a multi-agent system that solves these tasks through structured task decomposition, scoped biochemical tools, and hierarchical coordination, and DLBench, a benchmark that evaluates both network-level executability and protein-level plausibility. Across the 40 target products and 22 rate-limiting enzymes in DLBench, Origo achieved the best or tied-best results on all five GEM criteria and led most enzyme-mining metrics relative to general-purpose LLMs and biomedical agents. These findings show that structured multi-agent coordination can move beyond pathway suggestions toward executable models and biologically grounded enzyme candidates.

Limitations and Future Work. The evaluation is computational and does not establish that the proposed designs function in living cells. Although reaction additions are checked for formula stoichiometry and charge balance, standard FBA does not enforce Gibbs-energy directionality or exclude thermodynamically infeasible internal loops; consequently, positive growth or production flux should be interpreted as network feasibility rather than proof of energetic consistency. Future work should validate selected designs experimentally and expand the benchmark across molecules, enzyme families, and microbial hosts. Incorporating thermodynamic constraints, regulation, multi-omics data, toxicity, and enzyme-expression compatibility would enable more context-aware designs.

AI USE STATEMENT

Generative AI tools were used for manuscript writing assistance (language polishing and drafting of the manuscript text). They were also used as part of the benchmark methodology: DeepSeek V4 Pro served as the automated LLM expert evaluator that scored the 360 generated GEMs under the fixed rubric described in Appendix B.2. The LLM was not used for ideation or experimental design. All AI-assisted text and evaluation outputs were reviewed and verified by the authors, who take full responsibility for the correctness, originality, and integrity of all content in this paper.

REPRODUCIBILITY STATEMENT

The code is available at `https://anonymous.4open.science/r/Origo-D7BB`.

ETHICS STATEMENT

This work is purely computational: it involves no human subjects, animal experiments, or wet-lab manipulation of microorganisms. All data used in this study come from public databases (KEGG, MetaNetX, SEED, UniProt, PubChem). The generated designs are *in silico* metabolic models; any future wet-lab validation should follow applicable institutional biosafety and biosecurity regulations.

REFERENCES

Anthropic. Introducing the Model Context Protocol. `https://www.anthropic.com/news/model-context-protocol`, November 2024. Accessed: 2026-05-13.

Akari Asai, Jacqueline He, Rulin Shao, Weijia Shi, Amanpreet Singh, Joseph Chee Chang, Kyle Lo, Luca Soldaini, Sergey Feldman, Mike D'arcy, et al. Openscholar: Synthesizing scientific literature with retrieval-augmented lms. *arXiv preprint arXiv:2411.14199*, 2024.

John Blazeck and Hal Alper. Systems metabolic engineering: Genome-scale models and beyond. *Biotechnology journal*, 5(7):647–659, 2010.

Brigitte Boeckmann, Amos Bairoch, Rolf Apweiler, Marie-Claude Blatter, Anne Estreicher, Elisabeth Gasteiger, Maria J Martin, Karine Michoud, Claire O'Donovan, Isabelle Phan, et al. The swiss-prot protein knowledgebase and its supplement trembl in 2003. *Nucleic acids research*, 31(1):365–370, 2003.

Veda Sheersh Boorla and Costas D Maranas. Catpred: a comprehensive framework for deep learning in vitro enzyme kinetic parameters. *Nature communications*, 16(1):2072, 2025.

Jae Sung Cho, Gi Bae Kim, Hyunmin Eun, Cheon Woo Moon, and Sang Yup Lee. Designing microbial cell factories for the production of chemicals. *Jacs Au*, 2(8):1781, 2022.

UniProt Consortium. Uniprot: a hub for protein information. *Nucleic acids research*, 43(D1): D204–D212, 2015.

CrewAI Inc. Crewai. GitHub repository, 2024. URL `https://github.com/crewAIInc/crewAI`. Accessed: 2026-08-27.

Ali Ebrahim, Joshua A Lerman, Bernhard O Palsson, and Daniel R Hyduke. Cobrapy: constraints-based reconstruction and analysis for python. *BMC systems biology*, 7(1):74, 2013.

Alireza Ghafarollahi and Markus J Buehler. Sciagents: automating scientific discovery through bioinspired multi-agent intelligent graph reasoning. *Advanced Materials*, 37(22):2413523, 2025.

Changdai Gu, Gi Bae Kim, Won Jun Kim, Hyun Uk Kim, and Sang Yup Lee. Current status and applications of genome-scale metabolic models. *Genome biology*, 20(1):121, 2019.

Janna Hastings, Gareth Owen, Adriano Dekker, Marcus Ennis, Namrata Kale, Venkatesh Muthukrishnan, Steve Turner, Neil Swainston, Pedro Mendes, and Christoph Steinbeck. ChEBI in 2016: Improved services and an expanding collection of metabolites. *Nucleic Acids Research*, 44(D1): D1214–D1219, 2016. doi: 10.1093/nar/gkv1031.

Michael Heinzinger, Konstantin Weissenow, Joaquin Gomez Sanchez, Adrian Henkel, Milot Mirdita, Martin Steinegger, and Burkhard Rost. Bilingual language model for protein sequence and structure. *NAR Genomics and Bioinformatics*, 6(4):lqae150, 2024.

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*, 2025.

Michael Hucka, Frank T Bergmann, Claudine Chaouiya, Andreas Dräger, Stefan Hoops, Sarah M Keating, Matthias König, Nicolas Le Novère, Chris J Myers, Brett G Olivier, et al. The systems biology markup language (sbml): language specification for level 3 version 2 core release 2. *Journal of integrative bioinformatics*, 16(2):20190021, 2019.

Kristian Jensen, João G. R. Cardoso, and Nikolaus Sonnenschein. Optlang: An algebraic modeling language for mathematical optimization. *The Journal of Open Source Software*, 2(9):139, 2017. doi: 10.21105/joss.00139.

Ruofan Jin, Mingyang Xu, Fei Meng, Guancheng Wan, Qingran Cai, Yize Jiang, Jin Han, Yuanyuan Chen, Wanqing Lu, Mengyang Wang, et al. Stella: Towards a biomedical world model with self-evolving multimodal agents. *bioRxiv*, pp. 2025–07, 2025.

Minoru Kanehisa, Miho Furumichi, Yoko Sato, Masayuki Kawashima, and Mari Ishiguro-Watanabe. KEGG for taxonomy-based analysis of pathways and genomes. *Nucleic Acids Research*, 51(D1): D587–D592, 2023. doi: 10.1093/nar/gkac963.

Peter D Karp, Suzanne Paley, and Pedro Romero. The pathway tools software. *Bioinformatics*, 18: S225–S232, 2002.

Peter D Karp, Mario Latendresse, Suzanne M Paley, Markus Krummenacker, Quang D Ong, Richard Billington, Anamika Kothari, Daniel Weaver, Thomas Lee, Pallavi Subhraveti, et al. Pathway tools version 19.0 update: software for pathway/genome informatics and systems biology. *Briefings in bioinformatics*, 17(5):877–890, 2016.

Peter D Karp, Peter E Midford, Richard Billington, Anamika Kothari, Markus Krummenacker, Mario Latendresse, Wai Kit Ong, Pallavi Subhraveti, Ron Caspi, Carol Fulcher, et al. Pathway tools version 23.0 update: software for pathway/genome informatics and systems biology. *Briefings in bioinformatics*, 22(1):109–126, 2021.

Anas Aziz Khan, Md Shah Fahad, Ramesh Chandra, Guransh Singh, et al. Enzyclip: A cross-attention dual encoder framework with contrastive learning for predicting enzyme kinetic constants. *arXiv preprint arXiv:2512.00379*, 2025.

Sunghwan Kim, Jie Chen, Tiejun Cheng, Asta Gindulyte, Jia He, Siqian He, Qingliang Li, Benjamin A. Shoemaker, Paul A. Thiessen, Bo Yu, Leonid Zaslavsky, Jian Zhang, and Evan E. Bolton. PubChem 2023 update. *Nucleic Acids Research*, 51(D1):D1373–D1380, 2023. doi: 10.1093/nar/gkac956.

Zachary A. King, Justin Lu, Andreas Dräger, Philip Miller, Stephen Federowicz, Joshua A. Lerman, Ali Ebrahim, Bernhard O. Palsson, and Nathan E. Lewis. BiGG Models: A platform for integrating, standardizing and sharing genome-scale models. *Nucleic Acids Research*, 44(D1):D515–D522, 2016. doi: 10.1093/nar/gkv1049.

Yoo-Sung Ko, Je Woong Kim, Jong An Lee, Taehee Han, Gi Bae Kim, Jeong Eum Park, and Sang Yup Lee. Tools and strategies of systems metabolic engineering for the development of microbial cell factories for chemical production. *Chemical Society Reviews*, 49(14):4615–4636, 2020.

Jiahang Li, Steffen Waldherr, and Wolfram Weckwerth. Covrecon: automated integration of genome- and metabolome-scale network reconstruction and data-driven inverse modeling of metabolic interaction networks. *Bioinformatics*, 39(7):btad397, 2023.

Christian Lieven, Moritz E Beber, Brett G Olivier, Frank T Bergmann, Meric Ataman, Parizad Babaei, Jennifer A Bartell, Lars M Blank, Siddharth Chauhan, Kevin Correia, et al. Memote for standardized genome-scale metabolic model testing. *Nature biotechnology*, 38(3):272–276, 2020.

Yi Liu and Jens Nielsen. Recent trends in metabolic engineering of microbial chemical factories. *Current opinion in biotechnology*, 60:188–197, 2019.

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, 2024.

Daniel Machado, Sergej Andrejev, Marco Tramontano, and Kiran Raosaheb Patil. Fast automated reconstruction of genome-scale metabolic models for microbial strains and communities. *PLoS Computational Biology*, 14(9):e1006786, 2018. doi: 10.1371/journal.pcbi.1006786.

Andrew Makhorin. Gnu linear programming kit (glpk), 2024. URL `https://www.gnu.org/software/glpk/`. Version 5.0.

Peter G Mikhael, Itamar Chinn, and Regina Barzilay. Clipzyme: Reaction-conditioned virtual screening of enzymes. *arXiv preprint arXiv:2402.06748*, 2024.

Sébastien Moretti, Van Du T. Tran, Florence Mehl, Mark Ibberson, and Marco Pagni. MetaNetX/MNXref: unified namespace for metabolites and biochemical reactions in the context of metabolic models. *Nucleic Acids Research*, 49(D1):D570–D574, 2021. doi: 10.1093/nar/gkaa992.

Ferdous Nasri, Sarah Gurev, Patrick Varilly, Krithik Ramesh, Nuala A O'Leary, Jonah Cool, Bernhard Y Renard, Pardis C Sabeti, and Laura Luebbert. Deterministic access to global viral sequence data enables robust agentic scientific discovery. *arXiv preprint arXiv:2606.06749*, 2026.

Brett G Olivier and Frank T Bergmann. Sbml level 3 package: flux balance constraints version 2. *Journal of integrative bioinformatics*, 15(1):20170082, 2018.

Jeffrey D Orth, Ines Thiele, and Bernhard Ø Palsson. What is flux balance analysis? *Nature biotechnology*, 28(3):245–248, 2010.

Ross Overbeek, Robert Olson, Gordon D. Pusch, Gary J. Olsen, James J. Davis, Terry Disz, Robert A. Edwards, Svetlana Gerdes, Bruce Parrello, Maulik Shukla, Veronika Vonstein, Alice R. Wattam, Fangfang Xia, and Rick Stevens. The SEED and the Rapid Annotation of microbial genomes using Subsystems Technology (RAST). *Nucleic Acids Research*, 42(D1):D206–D214, 2014. doi: 10.1093/nar/gkt1226.

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.

Kimon Antonios Provatas, Avery Self, Ioannis Mouratidis, and Ilias Georgakopoulos-Soares. Bioveil matrix: Uncovering and categorizing vulnerabilities of agentic biological ai scientists. *arXiv preprint arXiv:2605.00927*, 2026.

Tijana Radivojević, Zak Costello, Kenneth Workman, and Hector Garcia Martin. A machine learning automated recommendation tool for synthetic biology. *Nature communications*, 11(1):4879, 2020.

Yusuf Roohani, Andrew Lee, Qian Huang, Jian Vora, Zachary Steinhart, Kexin Huang, Alexander Marson, Percy Liang, and Jure Leskovec. Biodiscoveryagent: An ai agent for designing genetic perturbation experiments. In *International Conference on Learning Representations*, volume 2025, pp. 26417–26466, 2025.

Samuel M. D. Seaver, Filipe Liu, Qizhi Zhang, James Jeffryes, José P. Faria, Janaka N. Edirisinghe, Michael Mundy, Nicholas Chia, Elad Noor, Moritz E. Beber, Aaron A. Best, Matthew DeJongh, Jeffrey A. Kimbrel, Patrik D'haeseleer, Sean R. McCorkle, Jay R. Bolton, Erik Pearson, Shane Canon, Elisha M. Wood-Charlson, Robert W. Cottingham, Adam P. Arkin, and Christopher S. Henry. The ModelSEED Biochemistry Database for the integration of metabolic annotations and the reconstruction, comparison and analysis of metabolic models for plants, fungi and microbes. *Nucleic Acids Research*, 49(D1):D575–D588, 2021. doi: 10.1093/nar/gkaa746.

Zhenkun Shi, Jun Zhu, Dehang Wang, BoYu Chen, Qianqian Yuan, Zhitao Mao, Fan Wei, Weining Wu, Xiaoping Liao, and Hongwu Ma. Rxnrecer enables fine-grained enzymatic function annotation through active learning and protein language models. *arXiv preprint arXiv:2603.12694*, 2026.

Evangelos Simeonidis and Nathan D Price. Genome-scale modeling for metabolic engineering. *Journal of Industrial Microbiology and Biotechnology*, 42(3):327–338, 2015.

Michael D Skarlinski, Sam Cox, Jon M Laurent, James D Braza, Michaela Hinks, Michael J Hammerling, Manvitha Ponnapati, Samuel G Rodriques, and Andrew D White. Language agents achieve superhuman synthesis of scientific knowledge. *arXiv preprint arXiv:2409.13740*, 2024.

Martin Steinegger and Johannes Söding. Mmseqs2 enables sensitive protein sequence searching for the analysis of massive data sets. *Nature biotechnology*, 35(11):1026–1028, 2017.

Michel Van Kempen, Stephanie S Kim, Charlotte Tumescheit, Milot Mirdita, Jeongjae Lee, Cameron LM Gilchrist, Johannes Söding, and Martin Steinegger. Fast and accurate protein structure search with foldseek. *Nature biotechnology*, 42(2):243–246, 2024.

Banghao Wu, Bozitao Zhong, Lirong Zheng, Runye Huang, Shifeng Jiang, Mingchen Li, Liang Hong, and Pan Tan. Harnessing protein language model for structure-based discovery of highly efficient and robust pet hydrolases. *Nature Communications*, 16(1):6211, 2025.

Lifeng Yang, Mengliang Yang, Xiangting Zheng, Qian Ping, Yunfei Song, Yongfu Xie, Yijian Rao, Jinsong Shi, and Yan Zhang. Selective biosynthesis and evaluation of rebaudioside M8 with improved organoleptic property and significant inhibition on inflammatory factor TNF-α. *Food Bioscience*, 59:103898, 2024. doi: 10.1016/j.fbio.2024.103898.

Botao Yu, Frazier N Baker, Ziru Chen, Garrett Herb, Boyu Gou, Daniel Adu-Ampratwum, Xia Ning, and Huan Sun. Tooling or not tooling? the impact of tools on language agents for chemistry problem solving. In *Findings of the Association for Computational Linguistics: NAACL 2025*, pp. 7620–7640, 2025.

Tianhao Yu, Haiyang Cui, Jianan Canal Li, Yunan Luo, Guangde Jiang, and Huimin Zhao. Enzyme function prediction using contrastive learning. *Science*, 379(6639):1358–1363, 2023.

Cheng Zhang and Qiang Hua. Applications of genome-scale metabolic models in biotechnology and systems medicine. *Frontiers in physiology*, 6:413, 2016.

Yang Zhang and Jeffrey Skolnick. Scoring function for automated assessment of protein structure template quality. *Proteins: Structure, Function, and Bioinformatics*, 57(4):702–710, 2004.

Yang Zhang and Jeffrey Skolnick. Tm-align: a protein structure alignment algorithm based on the tm-score. *Nucleic acids research*, 33(7):2302–2309, 2005.

Gengmo Zhou, Feng Yu, Wenda Wang, Zhifeng Gao, Guolin Ke, Zhewei Wei, and Zhen Wang. Fused gromov-wasserstein contrastive learning for effective enzyme-reaction screening. *arXiv preprint arXiv:2512.08508*, 2025.

A BASELINES

As shown in Table S1, we compare Origo with a diverse set of strong baselines: the general-purpose LLMs Claude Opus 4.8, DeepSeek V4 Pro, Gemini 3.5 Flash, GPT-5.5, Kimi-K2.6, Qwen3.7-Max, and GLM-5.2, and the biomedical agent systems STELLA and Biomni.

Table S1: Configurations of the models evaluated in our experiments.

Model	**Configuration**
Claude Opus 4.8	`reasoning_effort=high`, `temperature=0`
DeepSeek V4 Pro	`reasoning_effort=high`, `temperature=0`
Gemini 3.5 Flash	`reasoning_effort=high`, `temperature=0`
GPT-5.5	`reasoning_effort=medium`, `temperature=0`
Kimi-K2.6	`thinking.type=enabled`, `temperature=0`
Qwen3.7-Max	`thinking.type=adaptive`, `temperature=0`
GLM-5.2	`reasoning_effort=max`, `temperature=0`
STELLA	Local deployment, run with `-no_template`
Biomni	Web UI
Origo agent backbone	DeepSeek V4 Flash, `temperature=0`
LLM expert evaluator	DeepSeek V4 Pro, `reasoning_effort=high`, `temperature=0`

B GEMS GENERATION SUPPLEMENTARY INFORMATION

B.1 GEMS GENERATION WORKFLOW

The user initiates a session through the Origo interactive interface by submitting a natural language research query:

> *"Generate a bacterial SBML model for {target_product} synthesis."*

Origo uses a **Knowledge Template Router** to perform semantic classification over the user query and map it to the de novo SBML generation domain template (de novo SBML generation). This template encapsulates an expert-curated workflow that specifies the mandatory four-task chain: design $\rightarrow$ assemble $\rightarrow$ repair $\rightarrow$ refinementFigure S1.

B.1.1 TASK DECOMPOSITION.

The Task Breakdown Agent consumes the matched knowledge template and decomposes the user query into a sequence of sub-tasks:

- **GEMs Pathway Design**
  - **Description:** Design the metabolic foundation of the target organism. Given the target product, select the core pathways that any living cell requires—central carbon metabolism (glycolysis, the TCA cycle, the pentose phosphate pathway), amino-acid and nucleotide biosynthesis, cofactor and lipid metabolism, cell-wall and energy metabolism—together with biosynthetic pathways specific to the product class, and specify the growth medium and the product export. The design balances completeness against tractability: enough pathways to sustain growth and biosynthesis, without importing the entire catalog.
  - **Expected output:** A pathway design specification covering the core metabolic network, the product-specific pathway, and the growth medium.
- **GEMs Assemble**
  - **Description:** Assemble the selected pathways into an executable, machine-readable genome-scale metabolic model. Map the pathway selection onto curated reaction databases, connect the network to the growth medium via exchange and transport reactions, and evaluate the assembled model by flux balance analysis. Assess whether the model grows and whether it can produce the target product.

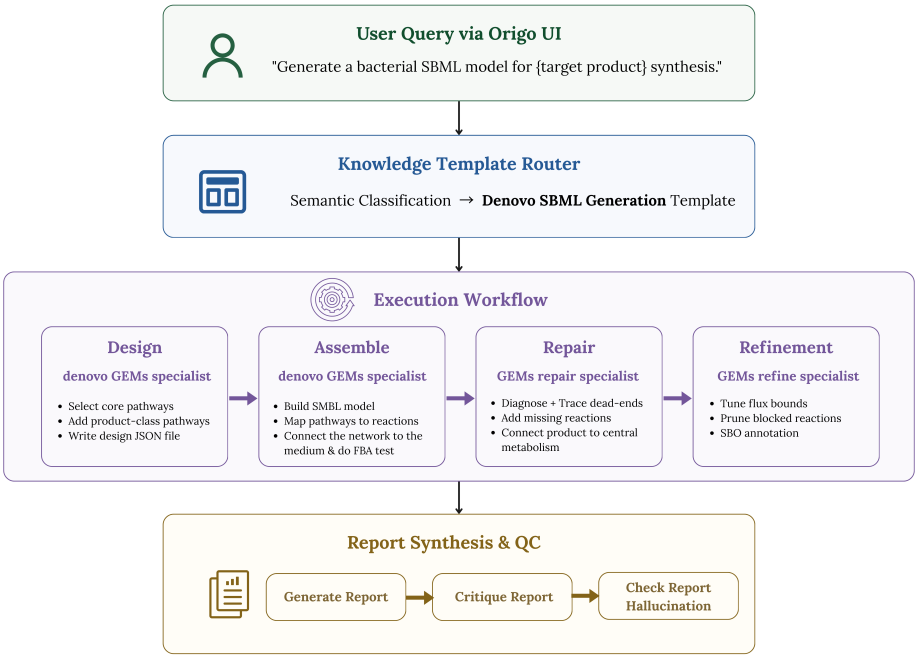

Figure S1: Illustration of the enzyme mining workflow.

  - **Expected output:** An assembled SBML model and a diagnostic report on growth and product formation.

- **GEMs Repair**
  - **Description:** Repair the model until it can biosynthesize and export the target product. Work backward from the product to locate each blocking dead-end precursor—the metabolites the current network cannot produce—and resolve them in a prioritized order: curated database reactions first, then literature evidence, then biochemical inference from compound class (terpenoids from IPP/DMAPP, phenylpropanoids from Phe/Tyr, glycosides from aglycone plus UDP-sugar), and lumped pathway proposals only as a last resort. Chemically balance each added reaction with proper cofactors, and re-evaluate the network recursively until the product connects to central metabolism.
  - **Expected output:** A repaired model with confirmed growth and positive product export flux.
- **GEMs Refinement**
  - **Description:** Polish the repaired model into a curated, simulation-ready genome-scale metabolic model. Prune reactions that carry no flux, keep product-flux bounds biologically realistic, and enrich the model with standard annotations—SBO terms, database cross-references, gene–protein–reaction associations, chemical formulas, and subsystem notes—so that the final model is not only executable but also auditable and reusable by the community. Introduce no new biosynthesis chemistry at this stage.
  - **Expected output:** The finalized, annotated model with confirmed growth and product formation.

B.1.2 Task Assignment.

The Task Assignment Agent assigns each sub-task to the most suitable expert agent from a predefined agent pool based on the role descriptions:

- **denovo GEMs specialist**
  - **Role:** denovo GEMs specialist

  - **Goal:** Build complete metabolic models from curated pathway knowledge, ensuring that the target product can be biosynthesized.
  - **Backstory:** You assemble de novo genome-scale metabolic models. You select pathways, map them onto curated reaction databases, build SBML models, and verify by flux balance analysis that the assembled network sustains growth and produces the target compound. When curated databases lack the biosynthetic route, you draw on AI retrosynthesis to fill the gap.
  - **Assigned sub-tasks: GEMs Pathway Design** and **GEMs Assemble**.
- **GEMs repair specialist**
  - **Role:** GEMs repair specialist
  - **Goal:** Add the missing product biosynthesis pathway to the model without rebuilding it—augment, never replace.
  - **Backstory:** You specialize in fixing metabolic models that cannot yet produce their target compound. Working outward from each dead-end precursor, you search the literature and curated databases for known biosynthetic routes, infer reactions from the chemistry of the compound class, and add them step by step with proper cofactors and balanced stoichiometry. Approximate reactions are acceptable as a last resort: a model with small but positive product flux is far more useful than a model with none.
  - **Assigned sub-tasks: GEMs Repair**.
- **GEMs refine specialist**
  - **Role:** GEMs refine specialist
  - **Goal:** Polish and tune metabolic models—remove dead reactions, set realistic flux bounds, and curate annotations for model quality.
  - **Backstory:** You polish metabolic models into curated, simulation-ready reconstructions. You remove reactions that cannot carry flux, keep export fluxes within biologically plausible bounds, and enrich the model with standard annotations so that its quality is verifiable and its provenance is traceable.

The MCP tools available to each agent are controlled per task by the tool configuration, so that each stage reasons over only the capabilities it requires; the complete tool inventory is provided in Table S7.

B.1.3 COLLABORATIVE EXECUTION.

The four sub-tasks are executed under a sequential process orchestrated by CrewAI(CrewAI Inc., 2024). The output of each preceding task is persisted as files in the working directory, so that downstream tasks retrieve and build on upstream results through the file system: `design_spec.json` $\rightarrow$ assembled `model.xml` $\rightarrow$ repaired `model.xml` $\rightarrow$ refined `model.xml`.

GEMs Pathway Design. The agent calls `list_denovo_pathways()` to obtain the curated pathway catalog (Table S2), selects core biomass-supporting pathways (glycolysis/gluconeogenesis, TCA cycle, pentose phosphate pathway, pyruvate metabolism, oxidative phosphorylation, purine and pyrimidine metabolism, amino-acid metabolism, vitamin and cofactor metabolism, fatty-acid and lipid metabolism, peptidoglycan biosynthesis, nitrogen metabolism, and the terpenoid backbone), plus pathways specific to the product class, and writes `design_spec.json`. The specification contains the model identifier, the selected pathways, the growth medium (glucose, ammonia, phosphate, sulfate, and oxygen at 10–20 mM, with `product_export` set to the target product name), and empty lists for extra, custom, and MetaNetX reactions Moretti et al. (2021).

GEMs Assemble. The agent calls `build_denovo_model(design_spec_path)` tool. The tool maps the selected KEGG pathways Kanehisa et al. (2023) (Table S2) to MetaNetX Moretti et al. (2021) and SEED Overbeek et al. (2014) reactions through the ModelSEED-compatible bridge table Seaver et al. (2021), adds exchange reactions (`EX_`) and transporters (`TRANS_`), assembles the model in SBML Level 3 format with the FBC package, runs an FBA simulation with the GLPK solver Makhorin (2024), and returns a structured diagnostic report. The `BUILD_RESULT` block reports the verdict (`OK` or `NEEDS_REPAIR`), reaction and metabolite counts, growth, and product-flux status (`product_has_producer`, `product_flux`); the verdict is set to `NEEDS_REPAIR` whenever the product has no producer or its flux is zero.

GEMs Repair. The agent first calls `diagnose_denovo_model`; if `product_flux` > 0, the task completes immediately. Otherwise it calls `trace_product_pathway`, which backtracks the reaction network from the product toward central metabolism (automatically skipping cofactors and high-connectivity hub metabolites) to pinpoint each blocking dead-end precursor together with its depth and pathway chain, and then calls `fix_product_export` to complete transport and secretion reactions. For each blocking precursor—starting from the one closest to the product—missing reactions are searched in a prioritized order: (a) database retrieval with `search_metabolite_reactions` followed by `add_reactions_to_sbml`; (b) literature research and AI-assisted retrosynthesis via `search_academic_content`, `plan_product_route`, and `search_retrosynthesis_route`; (c) biochemical class inference, in which a balanced overall reaction is written from the compound's structural features using the LLM's own knowledge. After every addition the network is re-traced recursively until the product is connected to central metabolism.

GEMs Refinement. The agent diagnoses the model, removes blocked (zero-flux) reactions with `prune_reactions_from_sbml` while protecting product-pathway reactions. It then enriches the model with SBO terms, identifiers.org annotations from ModelSEED Overbeek et al. (2014), gene products with gene–protein–reaction rules, chemical formulas, and subsystem notes, and confirms that the final model satisfies growth > 0 and `product_flux` > 0. No new biosynthetic reactions are added at this stage.

After the four sub-tasks are completed, Origo enters the report-synthesis and quality-control stage (`generate_report`, `critique_report`, `check_report_hallucination`) and delivers the quality-controlled report together with the finalized model.

B.1.4 Pathway Catalog

The pathway-design stage selects from a curated catalog of 165 KEGG pathways (Kanehisa et al., 2023) (Table S2), grouped into twelve functional categories and mapped to ModelSEED (Seaver et al., 2021), MetaNetX (Moretti et al., 2021), and SEED (Overbeek et al., 2014) reactions through the bridge table behind `list_denovo_pathways`.

Table S2: The curated pathway catalog used by Origo's pathway-design stage: all 165 KEGG pathways (Kanehisa et al., 2023) whose reactions are mapped to ModelSEED (Seaver et al., 2021), MetaNetX (Moretti et al., 2021), and SEED (Overbeek et al., 2014) identifiers, grouped by functional category. Reactions: number of mapped reactions per pathway.

#	Pathway ID	Pathway	Reactions
Carbohydrate metabolism			
1	map00010	Glycolysis / Gluconeoge nesis	54
2	map00020	Citrate cycle (TCA cycle)	29
3	map00030	Pentose phosphate pathway	51
4	map00040	Pentose and glucuronat e interconve rsions	78
5	map00051	Fructose and mannose metabolism	74
6	map00052	Galactose metabolism	54
7	map00053	Ascorbate and aldarate metabolism	57
8	map00500	Starch and sucrose metabolism	51
9	map00562	Inositol phosphate metabolism	64
10	map00566	Sulfoquino vose metabolism	27
11	map00620	Pyruvate metabolism	71
12	map00630	Glyoxylate and dicarboxyl ate metabolism	70
13	map00640	Propanoate metabolism	56
14	map00650	Butanoate metabolism	60
15	map00660	C 5-Branched dibasic acid metabolism	20
Energy metabolism			
16	map00190	Oxidative phosphoryl ation	7
17	map00680	Methane metabolism	102
18	map00710	Carbon fixation by Calvin cycle	24

#	Pathway ID	Pathway	Reactions
19	map00720	Other carbon fixation pathways	57
20	map00910	Nitrogen metabolism	41
21	map00920	Sulfur metabolism	53
Lipid metabolism			
22	map00061	Fatty acid biosynthes is	64
23	map00062	Fatty acid elongation	34
24	map00071	Fatty acid degradatio n	42
25	map00073	Cutin, suberine and wax biosynthes is	18
26	map00074	Mycolic acid biosynthes is	28
27	map00100	Steroid biosynthes is	59
28	map00120	Primary bile acid biosynthes is	51
29	map00140	Steroid hormone biosynthes is	132
30	map00561	Glycerolip id metabolism	44
31	map00564	Glyceropho spholipid metabolism	85
32	map00565	Ether lipid metabolism	22
33	map00590	Arachidoni c acid metabolism	50
34	map00591	Linoleic acid metabolism	13
35	map00592	alpha-Linolenic acid metabolism	29
36	map00600	Sphingolip id metabolism	48
37	map01040	Biosynthes is of unsaturate d fatty acids	41
Nucleotide metabolism			
38	map00230	Purine metabolism	139
39	map00240	Pyrimidine metabolism	100
Amino acid metabolism			
40	map00220	Arginine biosynthes is	32
41	map00250	Alanine, aspartate and glutamate metabolism	44
42	map00260	Glycine, serine and threonine metabolism	71
43	map00270	Cysteine and methionine metabolism	90
44	map00280	Valine, leucine and isoleucine degradatio n	46
45	map00290	Valine, leucine and isoleucine biosynthes is	24
46	map00300	Lysine biosynthes is	37
47	map00310	Lysine degradatio n	56
48	map00330	Arginine and proline metabolism	96
49	map00340	Histidine metabolism	35
50	map00350	Tyrosine metabolism	78
51	map00360	Phenylalan ine metabolism	52
52	map00380	Tryptophan metabolism	67
53	map00400	Phenylalan ine, tyrosine and tryptophan biosynthes is	48
Metabolism of other amino acids			
54	map00410	beta-Alanine metabolism	29
55	map00430	Taurine and hypotaurin e metabolism	22
56	map00440	Phosphonat e and phosphinat e metabolism	39
57	map00450	Selenocomp ound metabolism	19
58	map00460	Cyanoamino acid metabolism	49
59	map00470	D-Amino acid metabolism	57
60	map00480	Glutathion e metabolism	47
Glycan biosynthesis and metabolism			
61	map00510	N-Glycan biosynthes is	34
62	map00512	Mucin type O-glycan biosynthes is	12
63	map00513	Various types of N-glycan biosynthes is	27
64	map00514	Other types of O-glycan biosynthes is	20

#	Pathway ID	Pathway	Reactions
65	map00515	Mannose type O-glycan biosynthes is	13
66	map00520	Amino sugar and nucleotide sugar metabolism	77
67	map00531	Glycosamin oglycan degradatio n	19
68	map00532	Glycosamin oglycan biosynthes is - chondroiti n sulfate / dermatan sulfate	8
69	map00534	Glycosamin oglycan biosynthes is - heparan sulfate / heparin	9
70	map00540	Lipopolysa ccharide biosynthes is	35
71	map00541	Biosynthes is of various nucleotide sugars	123
72	map00550	Peptidogly can biosynthes is	25
73	map00552	Teichoic acid biosynthes is	33
74	map00563	Glycosylph osphatidyl inositol (GPI) -anchor biosynthes is	13
75	map00571	Lipoarabin omannan (LAM) biosynthes is	9
76	map00572	Arabinogal actan biosynthes is - Mycobacter ium	6
77	map00601	Glycosphin golipid biosynthes is - lacto and neolacto series	54
78	map00603	Glycosphin golipid biosynthes is - globo and isoglobo series	19
79	map00604	Glycosphin golipid biosynthes is - ganglio series	24
Metabolism of cofactors and vitamins			
80	map00130	Ubiquinone and other terpenoid-quinone biosynthes is	61
81	map00670	One carbon pool by folate	46
82	map00730	Thiamine metabolism	28
83	map00740	Riboflavin metabolism	27
84	map00750	Vitamin B 6 metabolism	28
85	map00760	Nicotinate and nicotinami de metabolism	63
86	map00770	Pantothena te and CoA biosynthes is	29
87	map00780	Biotin metabolism	24
88	map00785	Lipoic acid metabolism	37
89	map00790	Folate biosynthes is	67
90	map00830	Retinol metabolism	24
91	map00860	Porphyrin metabolism	144
Metabolism of terpenoids and polyketides			
92	map00253	Tetracycli ne biosynthes is	19
93	map00522	Biosynthes is of 12-, 14- and 16-membered macrolides	75
94	map00523	Polyketide sugar unit biosynthes is	60
95	map00900	Terpenoid backbone biosynthes is	52
96	map00902	Monoterpen oid biosynthes is	48
97	map00903	Limonene degradatio n	16
98	map00904	Diterpenoi d biosynthes is	100
99	map00905	Brassinost eroid biosynthes is	28
100	map00906	Carotenoid biosynthes is	116
101	map00907	Pinene, camphor and geraniol degradatio n	28
102	map00908	Zeatin biosynthes is	13
103	map00909	Sesquiterp enoid and triterpeno id biosynthes is	84
104	map00981	Insect hormone biosynthes is	25
105	map01051	Biosynthes is of ansamycins	19

#	Pathway ID	Pathway	Reactions
106	map01053	Biosynthes is of siderophor e group nonribosom al peptides	5
107	map01054	Nonribosom al peptide structures	1
108	map01055	Biosynthes is of vancomycin group antibiotic s	13
109	map01056	Biosynthes is of type II polyketide backbone	18
110	map01057	Biosynthes is of type II polyketide products	99
111	map01059	Biosynthes is of enediyne antibiotic s	45
Biosynthesis of other secondary metabolites			
112	map00232	Caffeine metabolism	31
113	map00254	Aflatoxin biosynthes is	27
114	map00261	Monobactam biosynthes is	21
115	map00311	Penicillin and cephalospo rin biosynthes is	14
116	map00331	Clavulanic acid biosynthes is	7
117	map00332	Carbapenem biosynthes is	17
118	map00333	Prodigiosi n biosynthes is	17
119	map00401	Novobiocin biosynthes is	24
120	map00402	Benzoxazin oid biosynthes is	8
121	map00403	Indole diterpene alkaloid biosynthes is	28
122	map00404	Staurospor ine biosynthes is	47
123	map00405	Phenazine biosynthes is	21
124	map00521	Streptomyc in biosynthes is	15
125	map00524	Neomycin, kanamycin and gentamicin biosynthes is	78
126	map00525	Acarbose and validamyci n biosynthes is	29
127	map00901	Indole alkaloid biosynthes is	62
128	map00940	Phenylprop anoid biosynthes is	71
129	map00941	Flavonoid biosynthes is	77
130	map00942	Anthocyani n biosynthes is	51
131	map00943	Isoflavono id biosynthes is	45
132	map00944	Flavone and flavonol biosynthes is	24
133	map00945	Stilbenoid, diarylhept anoid and gingerol biosynthes is	20
134	map00946	Degradatio n of flavonoids	27
135	map00950	Isoquinoli ne alkaloid biosynthes is	79
136	map00960	Tropane, piperidine and pyridine alkaloid biosynthes is	29
137	map00965	Betalain biosynthes is	9
138	map00966	Glucosinol ate biosynthes is	53
139	map00975	Biosynthes is of various siderophor es	31
140	map00996	Biosynthes is of various alkaloids	19
141	map00997	Biosynthes is of various other secondary metabolite s	31
142	map00998	Biosynthes is of various antibiotic s	79
143	map00999	Biosynthes is of various plant secondary metabolite s	94
Xenobiotics biodegradation and metabolism			
144	map00361	Chlorocycl ohexane and chlorobenz ene degradatio n	60
145	map00362	Benzoate degradatio n	84
146	map00363	Bisphenol degradatio n	2
147	map00364	Fluorobenz oate degradatio n	23
148	map00365	Furfural degradatio n	6
149	map00621	Dioxin degradatio n	31
150	map00622	Xylene degradatio n	38

#	Pathway ID	Pathway	Reactions
151	map00623	Toluene degradatio n	38
152	map00624	Polycyclic aromatic hydrocarbo n degradatio n	36
153	map00625	Chloroalkane and chloroalkene degradation	29
154	map00626	Naphthalen e degradatio n	26
155	map00627	Aminobenzo ate degradatio n	70
156	map00633	Nitrotolue ne degradatio n	9
157	map00642	Ethylbenze ne degradatio n	12
158	map00643	Styrene degradatio n	21
159	map00791	Atrazine degradatio n	14
160	map00930	Caprolacta m degradatio n	15
161	map00980	Metabolism of xenobiotic s by cytochrome P 450	72
162	map00982	Drug metabolism - cytochrome P 450	48
163	map00983	Drug metabolism - other enzymes	42
164	map00984	Steroid degradatio n	20
Translation			
165	map00970	Aminoacyl-tRNA biosynthes is	32

B.2 Detailed Evaluation Procedures

This appendix documents the complete experimental protocol behind the five-criterion GEM evaluation of Figure 2, at a level of detail sufficient for independent re-implementation, together with the target-compound information collected at the end of the appendix.

B.2.1 Evaluation Corpus and Software Environment

The evaluation corpus comprised 360 de novo SBML genome-scale metabolic models generated for 40 expert-defined target compounds (Table S4) by nine model sources: six general-purpose LLMs (DeepSeek-V4-Pro, Gemini-3.5-Flash, GLM-5.2, GPT-5.5, Kimi-K2.6, Qwen3.7-Max), Biomni, STELLA, and Origo. Each source contributed 40 models; SBML-validity, FBA, and Memote values were analyzed with the same protocol described below, including the same deterministic product-export identification procedure. All analyses were performed with COBRApy v0.31.1 (Ebrahim et al., 2013) (optlang interface(Jensen et al., 2017), GLPK simplex solver (Makhorin, 2024)) and Memote v0.17.0 (Lieven et al., 2020) under Python 3.13.

B.2.2 SBML Validation

Each model file was loaded with `cobra.io.read_sbml_model`, which parses the SBML Level 3 document (including the Flux Balance Constraints (FBC) package) through libSBML and instantiates the in-memory stoichiometric matrix, reaction list, metabolite list, gene list, and compartment definitions. A model was marked structurally valid (`sbml_valid` = `true`) if and only if this call returned without raising an exception; models triggering parsing errors (malformed XML, unresolved species references, duplicate identifiers, or FBC violations) were marked invalid. No manual curation or repair of invalid files was performed, and invalid models were excluded from all FBA and Memote analyses.

B.2.3 FBA: Objective Identification and Solver Settings

For each structurally valid model, growth FBA was run as follows. If the model carried a non-empty active objective after parsing, that objective was retained as-is. Otherwise, the objective was set to the first reaction whose identifier contained the substring "biomass" (case-insensitive). Models with neither an active objective nor a biomass-named reaction were excluded from growth FBA

and recorded as "No biomass objective found". The linear program of Eq. (6) was solved with COBRApy's `optimize()` using default settings (plain LP simplex via GLPK (Makhorin, 2024); parsimonious FBA disabled), with flux bounds taken exactly as declared in the SBML file; no external growth medium was imposed. When the solver status was `optimal`, the objective value was recorded rounded to six decimals; for any non-optimal status (e.g., infeasible or unbounded), a growth rate of zero was recorded together with the solver status. For the cross-source comparison, a model was classified as growth-competent if its maximized biomass flux was positive ($\mu^* > 0$), and each source was scored by the fraction of growth-competent models among its 40 submissions. Each FBA scenario was executed inside a COBRApy model context, so every run (growth, unconstrained product) started from the pristine loaded state and were mutually independent.

B.2.4 Product Exchange Reaction Detection

The target product's export reaction was identified uniformly for every source. Candidate boundary reactions were first matched by the compound's MetaNetX or ModelSEED identifier, with reaction names used only as an explicit secondary check for the same target compound. The remaining candidates were collected as reactions whose identifier starts with the exchange prefix `EX_`, plus phylo-style reactions whose identifier ends with the suffix `tex` or `exchange`. Reactions corresponding to standard growth-medium components were discarded: glucose (`glc`, `glc__D`), O_2, NH_4^+/NH_3, sulfate, phosphate, H_2O, CO_2, inorganic ions (K^+, Fe^{2+}, Mg^{2+}, Ca^{2+}, Cl^-, Na^+, Mn^{2+}, Zn^{2+}, Cu^{2+}, Co^{2+}, molybdate), and common cofactors (THF, NAD, NADP, CoA, ATP, ubiquinone, glycogen), including their ModelSEED compound identifiers and the corresponding MetaNetX aliases (Moretti et al., 2021) (`cpd00027`, `cpd00013`, `cpd00009`, `cpd00017`, `cpd00007`, `cpd00001`, `cpd00067`, `cpd00011`). To avoid false positives, identifiers of length ≥ 3 were excluded by substring match whereas 1–2-character identifiers had to match the whole base name. Among the remaining candidates, only reactions with a positive upper bound (i.e., permitting export) were kept, and the identifier-matched target reaction was selected deterministically. If no matching boundary reaction existed, the product flux was set to zero; no generic boundary reaction was used as a fallback. For the cross-source comparison, a model was classified as production-competent if its unconstrained product export flux in Eq. (7) was positive, and each source was scored by the fraction of production-competent models among its 40 submissions.

B.2.5 Memote Scoring in Detail

Memote was executed via its command-line interface (`memote report snapshot`), and the official continuous per-section and total scores embedded in the report were parsed and stored. The Memote v0.17.0 suite comprises 78 individual test functions, organized into cards covering SBML format, basic model information, metabolite and reaction information, gene–protein–reaction associations, annotation, stoichiometric consistency, network topology, biomass, energy metabolism, matrix conditioning, and experimental-data comparison. Of these, only the 27 tests in the five scored sections enter the score: consistency (5 tests), metabolite annotation (4), reaction annotation (4), gene annotation (3), and SBO-term annotation (11).

Each scored test reports an imperfection metric $m_t \in [0, 1]$ with $m_t = 0$ denoting full compliance: binary tests yield $m_t \in \{0, 1\}$; coverage tests report the fraction of violating entities (e.g., the proportion of metabolites lacking a chemical formula or a charge); parametrized tests average over parameter instances; and skipped tests retain the default $m_t = 1$. The test score is $s_t = 1 - m_t$. Per-test weights w_t are 1.0 for all scored tests except the stoichiometric-consistency test, which carries $w_t = 3.0$. Section scores are computed as the weighted average $S_s = \sum_{t \in \mathcal{T}_s} w_t s_t / \sum_{t \in \mathcal{T}_s} w_t$, and the total Memote score is the hierarchical weighted average in Eq. (8), with section weights $W_s = 3.0$ (consistency), 2.0 (SBO-term annotation), and 1.0 (all other scored sections). Scored tests that could not be collected for a model (e.g., when no biomass reaction exists) contribute $s_t = 0$ at their full weight, so that missing content lowers the score rather than inflating it. Models that failed SBML loading were recorded as Memote failures without a numerical score.

The annotation tests verify MIRIAM-compliant cross-references of metabolites, reactions, and genes against community databases, including MetaNetX/MNXref (Moretti et al., 2021), the SEED database (Overbeek et al., 2014), KEGG (Kanehisa et al., 2023), BiGG Models (King et al., 2016), ChEBI (Hastings et al., 2016), and UniProt (Consortium, 2015). These checks operate on the database

identifiers attached to model components, so they measure the curation quality of the generated models rather than pathway correctness.

B.2.6 LLM-BASED EXPERT EVALUATION

Complementary to the computational criteria, each of the 360 models was additionally reviewed by a large language model acting as an expert metabolic engineer. The review followed a fixed rubric of eight dimensions scored from 1 to 10 (Table S3). The rubric evaluates the model as a *de novo* living system rather than as a parts list: dimensions D_1–D_7 assess, respectively, the biosynthetic route chosen for the target compound, the completeness of central metabolism, the systematic coverage of functional subsystems, the biological plausibility of individual reactions, the structural organization of the SBML file, the production logic and carbon routing, and the originality of the design. Three rules of evidence are imposed. First, claimed subsystems must be verified against actual reaction stoichiometry rather than against reaction names, subsystem tags, or annotations: a multi-step pathway collapsed into a single net reaction is treated as a missing subsystem rather than a simplification. Second, chemistry is judged on its own terms (correct cofactors, elementally balanced stoichiometry, and mechanistically plausible transformations) rather than on the presence of database cross-references or on agreement with canonical pathways. Third, each dimension carries explicit anchors (10 = fully satisfied, 5 = directionally correct but partial, 1 = absent or fundamentally implausible), and the overall score D_8 is anchored to the mean of the organism-quality dimensions, $D_8 \approx (D_2 + D_3 + D_4)/3$, adjusted upward by at most 2 points for exceptional pathway design or production logic and downward by at most 2 points for terminal-pathway flaws. The judge returned, for every model and dimension, a numeric score together with a written justification citing specific reaction identifiers; the per-dimension means over the 40 models of each source are shown in Figure S2.

Table S3: The eight dimensions of the LLM expert-judgment rubric, each scored from 1 to 10. The "Rubric prompt" column reproduces verbatim excerpts of the judge prompt for each dimension.

#	Dimension	What is assessed	Rubric prompt (verbatim excerpts)
1	Pathway design (D_1)	Biochemical soundness of the biosynthetic route to the target compound: plausible chemistry, correct cofactors, and connection of central metabolism to the product.	"Evaluate the biosynthetic route chosen for the target compound. Is the route biochemically sound — does the chemistry make sense (correct cofactors, plausible transformations, no impossible reactions)? Does the route successfully connect central metabolism to the target compound? Do NOT penalize a route for being non-standard. If the model uses an alternative precursor, a promiscuous enzyme, or a non-canonical pathway to achieve the correct product, that is valid engineering. What matters is whether the chemistry is plausible, not whether it matches the textbook plant/microbial pathway. 10 = Biochemically sound route that successfully produces the target, well-matched to the chassis metabolic context. 5 = Route is directionally correct but has minor biochemical imprecisions (wrong cofactor count, missing intermediate detail). 1 = Fundamentally impossible chemistry or complete absence of a production route."

#	Dimension	What is assessed	Rubric prompt (verbatim excerpts)
2	Central metabolism completeness (D_2)	Presence of the core metabolic network (glycolysis, TCA cycle, pentose phosphate pathway) as individual, stoichiometrically correct reactions rather than lumped net reactions.	"Does the model include the core metabolic network needed to sustain a living cell? Are glycolysis, TCA cycle (or suitable alternative), and pentose phosphate pathway present as individual, stoichiometrically correct reactions? Do not accept reaction names as evidence. A single lumped reaction called 'glycolysis' covering glucose → pyruvate is NOT a glycolytic pathway — it is a black box. Verify that the individual enzymatic steps exist. Can the network generate biomass precursors (not just name them, but actually produce them through connected reactions)? 10 = Core metabolism fully represented at the individual reaction level — not lumped, not merely labeled. 5 = Core pathways are partially present as individual reactions, but some steps are lumped, missing, or represented only by placeholder names rather than actual chemistry. 1 = Central metabolism severely incomplete."
3	Model systematicity (D_3)	Coverage and interconnection of the functional subsystems of a living cell (central carbon, energy, amino acids, nucleotides, cofactors, lipids, transport) with actual multi-step biochemistry.	"Evaluate the model as a living system: does it cover the functional subsystems a cell needs with actual biochemical reactions, not just labels? What metabolic subsystems are present: central carbon, energy, amino acids, nucleotides, cofactors, lipids, transport? For each claimed subsystem, trace the actual reactions. Are subsystems connected through shared metabolites, or are there disconnected islands? Are there critical gaps preventing self-sustainability? 8–10 = Comprehensive, well-integrated metabolic system where each claimed subsystem is backed by actual multi-step biochemistry. 5–7 = Partial coverage: central carbon is real, but several peripheral subsystems (multiple amino acids, nucleotides, cofactors, lipids) are represented by lumped net reactions rather than individual enzymatic steps. 1–4 = Severely incomplete."

#	Dimension	What is assessed	Rubric prompt (verbatim excerpts)
4	Biological plausibility (D_4)	Whether individual reactions constitute real biochemistry: correct cofactor usage, balanced stoichiometry, and mechanistically plausible transformations, without recurring invented balancing reactions.	"Do the individual reactions look like real biochemistry? Correct cofactors (NAD+ vs NADP+, ATP vs GTP), proper stoichiometric balance? Does the chemistry make thermodynamic and mechanistic sense? Do NOT penalize reactions for lacking EC numbers, gene associations, or database cross-references — these are annotation details, not biochemistry. Invented chemistry is not minor. Reactions that are elementally balanced but have no enzymatic basis — such as 'NADPH + CO_2 $\rightarrow$ NADP + formate' or biosynthetic steps that release formate as a generic carbon accounting byproduct — are NOT biochemically plausible. Elemental balance alone does not make a reaction real. 10 = Biochemically authentic: correct cofactors, plausible mechanisms, proper stoichiometry. 5–6 = Mostly reasonable with some questionable reactions. 1–4 = Systematic biochemical problems: recurring invented chemistry (formate shunts, non-enzymatic redox), pervasive lumped placeholder reactions, or widespread implausible cofactor/co-substrate choices."
5	SBML structure (D_5)	Structural organization of the SBML file: compartments, units, flux bounds, objective function, and readability.	"Evaluate the SBML structure. Clear compartments, proper units, flux bounds, objective function? Readable and well-organized? 10 = Professionally organized: logical grouping, clear compartments, proper formatting. 5 = Adequate with organizational issues. 1 = Chaotic or malformed."
6	Production logic (D_6)	Carbon routing from substrate to product and the metabolic infrastructure (energy, redox, precursors) supporting production, including product export.	"How well does the model route carbon from substrate to product? Can carbon flow from substrate to product through the network? Is there a demand reaction or exchange that allows product accumulation? Does the metabolic infrastructure (energy, redox, precursors) support production? Missing export/exchange reactions are minor configuration issues, not production logic failures. If the intracellular product pathway is complete and connected, the production logic is sound. 10 = Production pathway correctly embedded within a comprehensive metabolic system; full precursor/energy/redox support; product can accumulate. 5 = The metabolic foundation is solid, the product pathway is connected, but minor terminal details are missing or imprecise. 1 = Product pathway completely absent or fundamentally disconnected."

#	Dimension	What is assessed	Rubric prompt (verbatim excerpts)
7	Originality (D_7)	Evidence of purposeful, informed assembly of the complete organism, rather than a trivial placeholder or a chaotic collection of reactions.	"Evaluate the design as an act of de novo organism construction. The primary act of creativity in de novo life design is building the complete, coherent organism. Do NOT penalize a model for resembling known biochemistry. All correct de novo designs converge to the same fundamental metabolic solutions (glycolysis, TCA cycle, PPP). Do NOT penalize a model for including database annotations (EC numbers, gene IDs, KEGG/BiGG/MetaCyc cross-references). Judge whether the model shows evidence of purposeful, informed assembly — does it read as if someone who understands metabolism built this organism? 10 = A purposeful, comprehensive organism design demonstrating expert metabolic knowledge. 5 = Competent but conventional assembly; functional but does not stand out. 1 = No evidence of purposeful design."
8	Overall quality (D_8)	Holistic judgment of the model as a living system, anchored to $D_8 \approx (D_2 + D_3 + D_4)/3$ with adjustments of at most ± 2 points for pathway design and terminal-pathway flaws.	"Synthesize all dimensions into a holistic judgment of the de novo organism design. The primary question: does this model succeed as a living system? The dimensions that answer this are central metabolism (D2), systematicity (D3), and biological plausibility (D4). Scoring formula: The overall score should be anchored to the average of D2, D3, and D4 (the 'organism quality' baseline). Adjust upward by 0–2 points for exceptional pathway design (D1) or production logic (D6). Adjust downward by 0–2 points for terminal pathway flaws. 10 = Exceptional organism. 7–9 = Strong organism. 5–6 = Adequate organism. 1–4 = Not a functional organism."

B.3 Target Compound Information

Table S4: The 40 expert-defined target products of the GEM generation benchmark, with identity. All data were retrieved from PubChem Kim et al. (2023) (accessed 18 August 2026); CID, PubChem Compound identifier; superscript citations denote classic discovery literature.

#	Name	Formula	Category	CID
1	Alpha-Neoclovene	$C_{15}H_{24}$	Sesquiterpene (clovene type)	15973057
2	Alpha-Santalene	$C_{15}H_{24}$	Sesquiterpene	94164
3	Beta-Carotene	$C_{40}H_{56}$	Tetraterpene (carotenoid)	5280489
4	Beta-Hydroxy-beta-methylbutyrate	$C_5H_{10}O_3$	Branched-chain organic acid (leucine metabolite)	69362
5	Bisabolene	$C_{15}H_{24}$	Sesquiterpene	62346
6	Caffeic acid	$C_9H_8O_4$	Hydroxycinnamic acid (phenylpropanoid)	689043
7	Camptothecin	$C_{20}H_{16}N_2O_4$	Quinoline alkaloid	24360
8	Chlorogenic acid	$C_{16}H_{18}O_9$	Caffeoylquinic acid (phenolic acid ester)	1794427

#	Name	Formula	Category	CID
9	Cordycepin	$C_{10}H_{13}N_5O_3$	Purine nucleoside analogue	6303
10	Evodiamine	$C_{19}H_{17}N_3O$	Indoloquinazoline alkaloid	442088
11	Farnesene	$C_{15}H_{24}$	Sesquiterpene	5281516
12	Ferulic acid	$C_{10}H_{10}O_4$	Hydroxycinnamic acid	445858
13	Gadusol	$C_8H_{12}O_6$	Mycosporine-like cyclitol	195955
14	Gastrodin	$C_{13}H_{18}O_7$	Phenolic glycoside	115067
15	Geranial	$C_{10}H_{16}O$	Monoterpenoid aldehyde	638011
16	Geraniol	$C_{10}H_{18}O$	Monoterpenoid alcohol	637566
17	Huperzine A	$C_{15}H_{18}N_2O$	Lycopodium alkaloid	854026
18	Isoprenol	$C_5H_{10}O$	Hemiterpenoid alcohol	12988
19	Limonene	$C_{10}H_{16}$	Monoterpene	22311
20	Matrine	$C_{15}H_{24}N_2O$	Quinolizidine alkaloid	91466
21	Myrcene	$C_{10}H_{16}$	Monoterpene	31253
22	Nerol	$C_{10}H_{18}O$	Monoterpenoid alcohol	643820
23	Nuciferine	$C_{19}H_{21}NO_2$	Aporphine alkaloid	10146
24	p-Anisaldehyde	$C_8H_8O_2$	Aromatic aldehyde	31244
25	p-Coumaric acid	$C_9H_8O_3$	Hydroxycinnamic acid	637542
26	Perillic acid	$C_{10}H_{14}O_2$	Monoterpenoid carboxylic acid	1256
27	Perillyl alcohol	$C_{10}H_{16}O$	Monoterpenoid alcohol	10819
28	Phloretin	$C_{15}H_{14}O_5$	Dihydrochalcone (flavonoid)	4788
29	Rebaudioside D	$C_{50}H_{80}O_{28}$	Steviol glycoside (diterpenoid)	71773169
30	Rebaudioside M	$C_{56}H_{90}O_{33}$	Steviol glycoside (diterpenoid)	92023628
31	Rebaudioside M8	$C_{56}H_{90}O_{33}$[†]	Steviol glycoside (diterpenoid)	Yang et al. (2024)
32	Resveratrol	$C_{14}H_{12}O_3$	Stilbenoid	445154
33	Salidroside	$C_{14}H_{20}O_7$	Phenylethanoid glycoside	159278
34	Salvianic acid A	$C_9H_{10}O_5$	Phenolic acid (phenylpropanoid-derived)	11600642
35	Sclareol	$C_{20}H_{36}O_2$	Labdane diterpenoid	163263
36	Sinapic acid	$C_{11}H_{12}O_5$	Hydroxycinnamic acid	10743
37	Sinensal	$C_{15}H_{22}O$	Sesquiterpenoid aldehyde	5281535
38	Spathulenol	$C_{15}H_{24}O$	Sesquiterpenoid alcohol	92231
39	Squalene	$C_{30}H_{50}$	Triterpene	638072
40	Tanshinone IIA	$C_{19}H_{18}O_3$	Abietane diterpenoid quinone	164676

[†] Not deposited in PubChem (accessed 18 August 2026); identity from Yang et al. (2024). The molecular formula corresponds to the rebaudioside M isomer bearing six β-glucose residues.

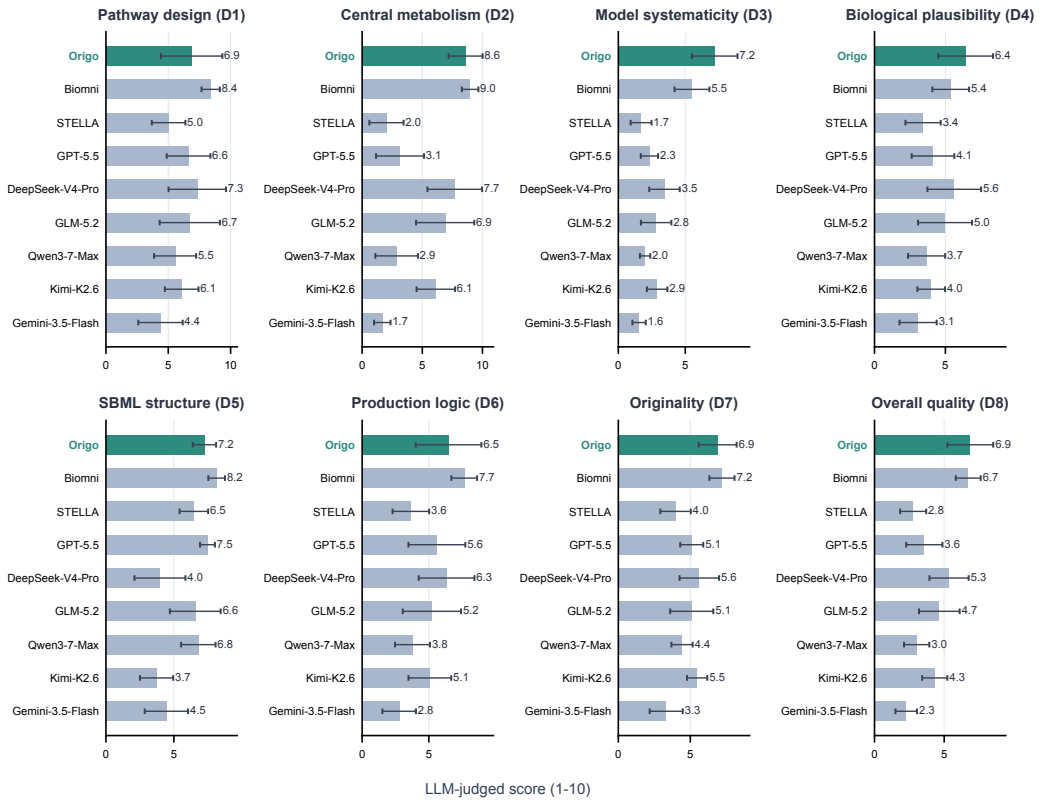

Figure S2: Per-dimension LLM-judged scores (1–10) of the nine model sources on the eight evaluation dimensions of the expert rubric (D_1 pathway design, D_2 central metabolism completeness, D_3 model systematicity, D_4 biological plausibility, D_5 SBML structure, D_6 production logic, D_7 originality, D_8 overall quality). Bars show mean $\pm$ SD over $n = 40$ scored models per source. Origo is highlighted.

C Enzyme Mining Supplementary Information

C.1 Enzyme Information

Table S5: Enzyme Information

UniProt ID	Enzyme Name	EC Number
Q8L5K3	(+)-Limonene synthase	4.2.3.20
A0A0A1C3I2	HMGR (HMG-CoA reductase)	1.1.1.34
P15496	IDI (IPP delta-isomerase)	5.3.3.2
P08524	ERG20 (FPP synthase)	2.5.1.10
Q92236	GGPPS (Geranylgeranyl diphosphate synthase)	2.5.1.1
P24481	PAL (Phenylalanine ammonia-lyase)	4.3.1.24
Q3IWB0	TAL (Tyrosine ammonia-lyase)	4.3.1.23
Q9S725	4CL (4-coumarate-CoA ligase)	6.2.1.12
P30074	CHS (Chalcone synthase)	2.3.1.74
Q8VZW3	CHI (Chalcone isomerase)	5.5.1.6
Q9S818	F3H (Flavanone 3-hydroxylase)	1.14.11.9
Q96330	FLS (Flavonol synthase)	1.14.20.6
Q9FI21	BBE (Berberine bridge enzyme)	1.21.3.3
P32282	NRDA (Ribonucleoside-diphosphate reductase subunit alpha)	1.17.4.1
P06785	CDC21 (Thymidylate synthase)	2.1.1.45
P11156	NRDB (Ribonucleoside-diphosphate reductase subunit beta)	1.17.4.1
P54973	Beta-carotene 3-hydroxylase (CrtZ)	1.14.15.24
O65572	Carotenoid cleavage dioxygenase 1 (CCD1)	1.13.11.69
O24540	4-coumarate-CoA ligase (4CL)	6.2.1.12
Q6QDB6	UDP-glycosyltransferase 73B6 (UGT73B6)	2.4.1.x
Q9Y7X5	UDP-glucose 4-epimerase (UGE)	5.1.3.2
O65677	3-ketoacyl-CoA synthase 17 (KCS17)	2.3.1.199

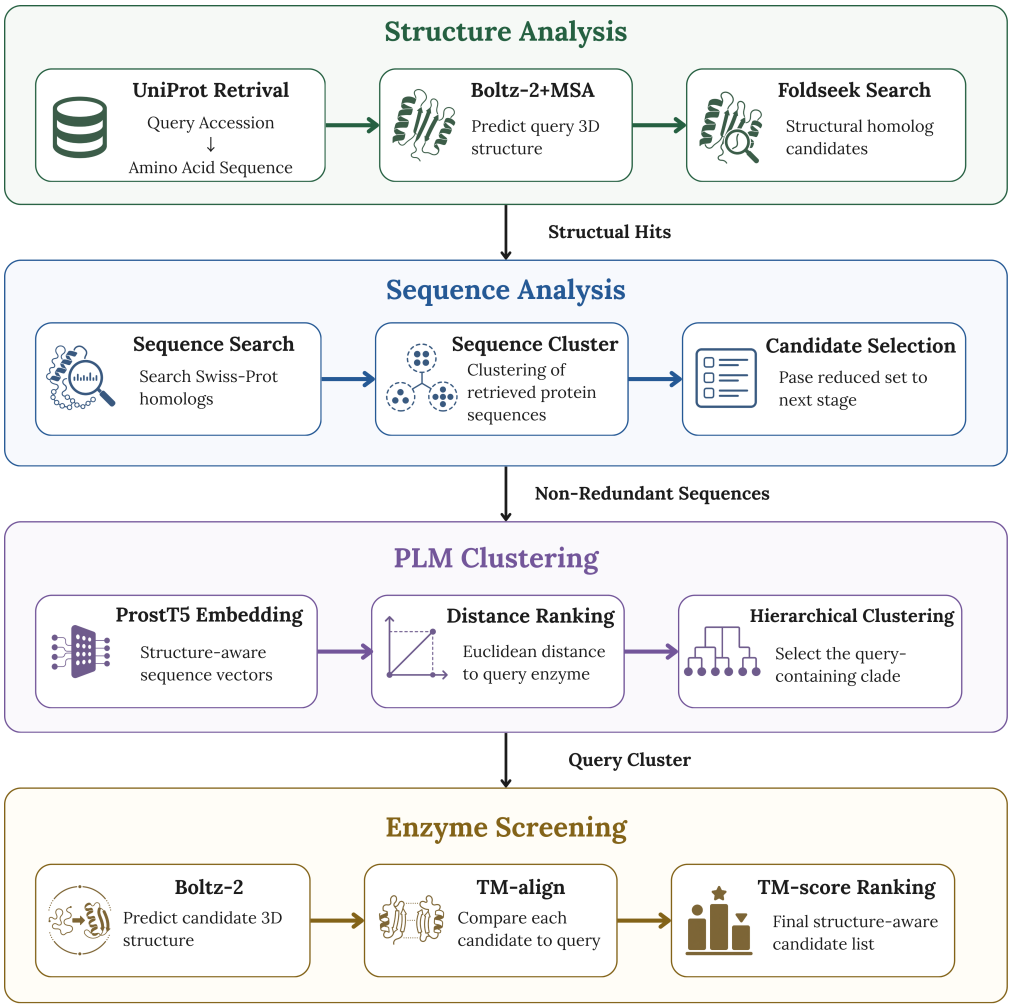

Figure S3: Illustration of the enzyme mining workflow.

C.2 Enzyme Mining Workflow

The user initiates a session through the `Origo` interactive interface by submitting a natural language research query:

> *"Identify specific protein candidates that are likely to possess EC {ec_number}-like activity and are functionally related to {enzyme_name} (UniProt: {reference_uniprot}). Return the result as valid JSON. For each candidate, provide 10 UniProt accessions and concise evidence for selection."*

The variables *ec_number*, *enzyme_name*, and *reference_uniprot* are provided in Table S5. Origo uses a **Knowledge Template Router** to perform semantic classification over the user query and map the query to the **Enzyme Mining** template. This template encapsulates expert-curated workflows for computational enzyme discovery, thereby guiding the subsequent planning and execution with expert knowledge. Inspired by the work of Wu et al. (2025), the enzyme mining workflow is illustrated in Figure S3.

C.2.1 Task Decomposition.

The **Task Breakdown Agent** consumes the matched knowledge template and decomposes the user query into a sequence of sub-tasks:

- **Enzyme Structure Analysis**

  - **Description:** Conduct a comprehensive structural analysis of the user query. Focus on three-dimensional biomolecular structures and structure-based insights. Depending on the input, perform relevant analyses such as: predicting or retrieving 3D structures of proteins, nucleic acids, or complexes; identifying structural homologs using structure similarity search; analyzing molecular interfaces, binding pockets, active sites, or ligand interactions; evaluating structural features relevant to molecular function, engineering, or synthetic biology applications. Select appropriate structural bioinformatics tools and provide biologically meaningful interpretations rather than only reporting raw tool outputs.
  - **Expected output:** List the tool used and provide a summary based on the tool output.
- **Enzyme Sequence Analysis**
  - **Description:** Conduct a comprehensive sequence-based analysis of the user query. Focus on enzyme sequence information and sequence-derived biological insights. Depending on the input, perform relevant analyses such as: sequence similarity search and homolog identification; multiple sequence alignment and evolutionary conservation analysis; domain, motif, and functional annotation; sequence-based prediction of biological properties or engineering potential. Select appropriate sequence analysis tools and interpret the biological meaning of the obtained results instead of only collecting raw outputs.
  - **Expected output:** List the tool used and provide a summary based on the tool output.
- **Enzyme Candidates Clustering**
  - **Description:** Cluster and organize protein candidates based on sequence similarity and evolutionary relationships. This task analyzes retrieved homologous proteins or candidate sequences using approaches such as: embedding-based clustering with protein language models; sequence similarity clustering; phylogenetic analysis using multiple sequence alignment and tools such as IQ-TREE. The goal is to identify distinct protein families, evolutionary groups, and representative candidates, enabling diversity-aware selection of proteins for downstream functional analysis or engineering.
  - **Expected output:** List the tool used and provide a summary based on the tool output.
- **Enzyme Screening**
  - **Description:** Select and prioritize enzyme candidates from clustered protein groups using structure-based or functional relevance criteria. This task evaluates candidate enzymes obtained from sequence or embedding-based clustering and identifies the most promising representatives by integrating: structural similarity metrics such as TM-score, RMSD, and fold-level similarity; active site and catalytic residue conservation; distances and geometric relationships between key functional residues; ligand-binding pocket similarity and protein-ligand interaction patterns; sequence, evolutionary, and biochemical information when available. The goal is to select diverse enzyme candidates that maintain the desired functional characteristics while maximizing structural and functional relevance for downstream characterization or engineering.
  - **Expected output:** List the tool used and provide a summary based on the tool output.

C.2.2 TASK ASSIGNMENT.

The **Task Assignment Agent** assigns each sub-task to the most suitable expert agent from a predefined agent pool based on the role descriptions:

- **Bioinformatics Researcher**
  - **Role:** Bioinformatics Researcher
  - **Goal:** To provide a clean, comprehensive, and biologically meaningful data layer that enables predictive modeling, pathway design, and AI learning, acting as the foundational knowledge base for the Design and Learn phases.
  - **Backstory:** You are a seasoned bioinformatics scientist with an exceptional talent for sifting through vast amounts of biological data—from genomic sequences to metabolomic profiles. You are adept at identifying key information, performing intricate sequence and functional analyses, and applying advanced computational methods to uncover hidden patterns in complex omics data. Your expertise is crucial in collecting, curating, and annotating biological

information, transforming raw data into actionable insights that guide every design, optimization, and prediction in the DBTL (Design–Build–Test–Learn) cycle. Your work establishes the foundational knowledge essential for any new synthetic biology design or experiment.
  - **Assigned sub-tasks: Enzyme Structure Analysis**, **Enzyme Sequence Analysis**, and **Enzyme Candidates Clustering**.
- **Molecular Engineering Specialist**
  - **Role:** Molecular Engineering Specialist
  - **Goal:** To translate conceptual design hypotheses and pathway predictions into executable, standardized strain-construction plans and protocols, ensuring feasibility, reproducibility, and safety for wet-lab implementation.
  - **Backstory:** You are a highly skilled specialist in synthetic biology strain engineering. With deep expertise in molecular biology, metabolic engineering, and cutting-edge genome editing techniques like CRISPR and recombineering, you are the bridge between computational design and physical construction. For each organism, you meticulously develop standard operating procedures for every step involved in designing and building an engineered strain. Given a target molecule, you excel at elucidating the required pathways and crafting a detailed plan to construct these pathways within the specified host strain. Your core focus is to transform a desired function for an engineered strain into a testable biological construct, always prioritizing safety, reproducibility, and data traceability.
  - **Assigned sub-tasks: Enzyme Screening**.

C.2.3 COLLABORATIVE EXECUTION

The four sub-tasks are executed under a sequential process orchestrated by CrewAI(CrewAI Inc., 2024). The output of each preceding task is propagated as contextual input to the subsequent task. Each agent interacts with external bioinformatics tool servers via the MCP server to perform computations and analyses. The detailed execution flow is as follows:

Enzyme Structure Analysis. Origo retrieves the query enzyme sequence from UniProt using the provided accession ID. The sequence is fed into Boltz v2.2.1 with the online MSA server to model the 3D structure. Subsequently, Origo invokes Foldseek through the online Foldseek server (`https://search.foldseek.com/search`).

Enzyme Sequence Analysis. Origo subsequently uses MMseqs2 (Steinegger & Söding, 2017) to search the Swiss-Prot (Boeckmann et al., 2003) database for proteins that are sequence-homologous to the candidates obtained in the previous stage. Specifically, we use `mmseqs easy-search` with the command `mmseqs easy-search query database output --format-output "query,target,pident,fident,nident,alnlen,bits,tseq,evalue" tmp --threads 64 --num-iterations 2 --max-seqs 2500 --search-type 1`. The resulting sequences are then clustered using `mmseqs easy-cluster` with the command `mmseqs easy-cluster query output tmp --min-seq-id 0.5 --threads 96`, yielding a non-redundant set of candidate proteins.

Enzyme Candidate Clustering. Origo computes structure-aware sequence embeddings using ProstT5 (Heinzinger et al., 2024). It first ranks the mined candidates and EC-annotated enzymes according to their Euclidean distance to the query enzymes and selects the closest sequences for downstream analysis. The selected representations are then subjected to Ward hierarchical clustering to construct a phylogenetic tree. The cluster containing the query enzyme is subsequently selected as the representative cluster.

Enzyme Screening. The candidates in the selected cluster are subsequently screened based on structural similarity. Specifically, Origo uses Boltz v2.2.1 (Passaro et al., 2025) to predict the 3D structure of each candidate, after which TM-align (Zhang & Skolnick, 2005) is used to compare the predicted structure with the query structure and compute a TM-score (Zhang & Skolnick, 2004). Origo uses the TM-score as a final structure-based criterion for ranking the candidate set.

C.3 Additional Results

C.3.1 Structural Confidence

We show the Boltz confidence score and binding-related metrics in Table S6. Origo achieves the best overall structural confidence, with the highest pTM (0.9293) and pLDDT (0.9017) and the lowest pDE (0.6421). At the protein–ligand interface, Origo achieves the best ipLDDT (0.8218) and ranks second on both ipTM (0.8822) and ipDE (2.7217), only marginally behind Biomni. Moreover, Origo achieves the second-best affinity (0.9707) and the second-highest binding probability (0.4968), remaining close to the respective leaders (DeepSeek-V4-Pro and Biomni). Relative to STELLA and the LLM baselines, Origo consistently improved structural confidence, interface quality, and binding-associated metrics.

Table S6: Comparison of different models across confidence score and binding-related metrics.

Model	**pTM** ↑	**pLDDT** ↑	**pde** ↓	**ipTM** ↑	**ipLDDT** ↑	**ipde** ↓	**Affinity** ↓	**Binding Prob.** ↑
Claude-Opus-4.8	0.8789	0.8448	0.8459	0.8388	0.7510	3.5398	1.0431	0.4435
DeepSeek-V4-Pro	0.8568	0.7921	0.8515	0.8101	0.6656	3.5132	**0.9379**	0.4291
Gemini-3.5-Flash	0.8489	0.7961	0.8683	0.8019	0.6643	3.6973	1.0348	0.4092
GPT-5.5	0.8652	0.8380	0.8731	0.8190	0.7271	3.8811	1.0461	0.4365
Kimi-K2.6	0.8352	0.7839	0.9694	0.7745	0.6510	4.4529	1.0253	0.4138
Qwen3.7-Max	0.7974	0.7621	1.1573	0.7592	0.6264	4.8473	1.0729	0.3900
GLM-5.2	0.8367	0.7834	0.9013	0.7896	0.6297	4.0153	1.0169	0.3889
STELLA	0.8955	0.8598	0.7941	0.8584	0.7612	3.2700	1.1002	0.4431
Biomni	0.9137	0.8878	0.6871	**0.8895**	0.8151	**2.7146**	1.0386	**0.4988**
Origo	**0.9293**	**0.9017**	**0.6421**	0.8822	**0.8218**	2.7217	0.9707	0.4968

C.4 EC Classification Accuracy

We show the per-enzyme EC_4 accuracy across methods in Figure S4. Origo consistently outperforms the LLM baselines on all enzymes and is more accurate than Biomni and STELLA on most enzymes. The accuracy of UGT73B6 whose EC number is 2.4.1.x is recorded as the EC_3 accuracy.

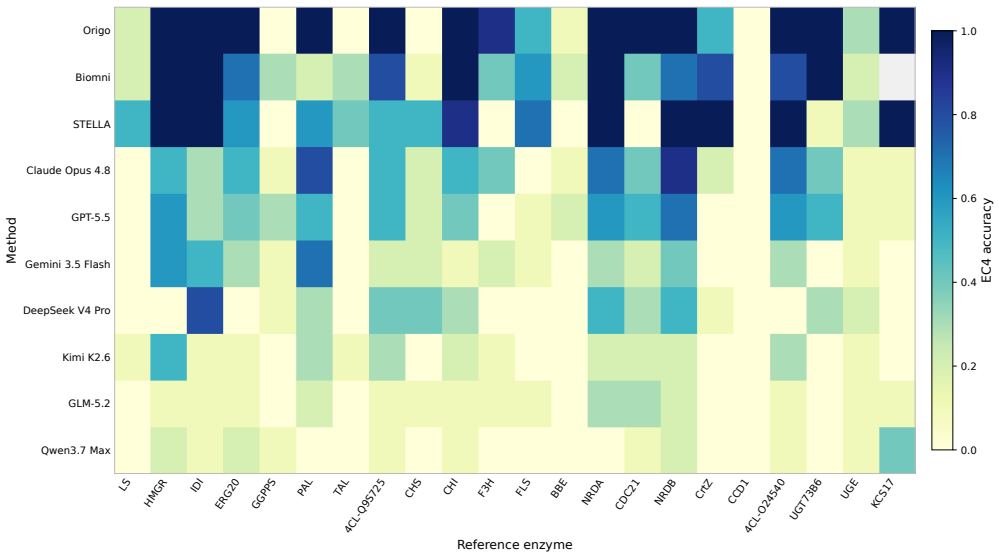

Figure S4: EC_4 prediction accuracy for each candidate enzyme of Origo and baselines.

C.5 KINETIC PROPERTIES

The results of k_{cat}, K_m and k_{cat}/K_m are shown in Figure S5. We omit the LLM-mined candidates from the evaluation: general LLMs exhibit relatively low EC_4 accuracy, indicating that enzymes constitute only a small fraction of their outputs. Evaluating such candidates with CatPred, an enzyme-specific kinetic parameter predictor, would therefore be uninformative. Origo achieves the highest predicted k_{cat} and k_{cat}/K_m compared with baselines.

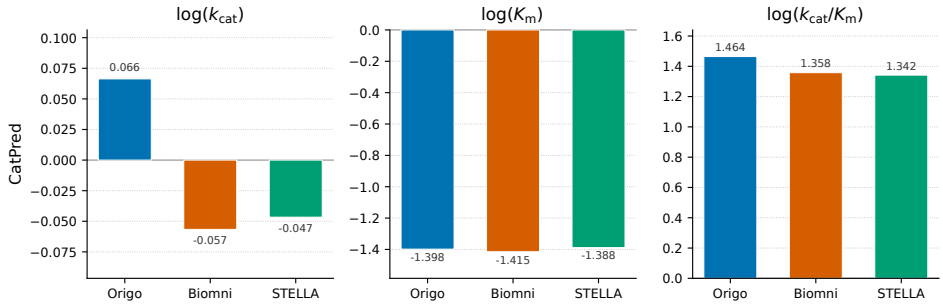

Figure S5: The kinetic parameters of k_{cat}, K_m and k_{cat}/K_m predicted by CatPred.

C.6 STRUCTURAL SIMILARITY

We first predict the complex structures of the candidate proteins and their substrates with Boltz v2.2.1. The protein chain is then extracted from each complex and aligned to the query enzyme with US-align to calculate the structural TM-score. The binding pocket is defined as residues with at least one heavy atom within 5 Å of any ligand atom. Pocket residues from the query and candidate were then aligned with US-align to obtain the pocket TM-score. Higher values indicate closer structural and active-site similarity to the reference enzyme. As shown in Figure S6, Origo achieves the best structural and pocket similarity compared with the baselines.

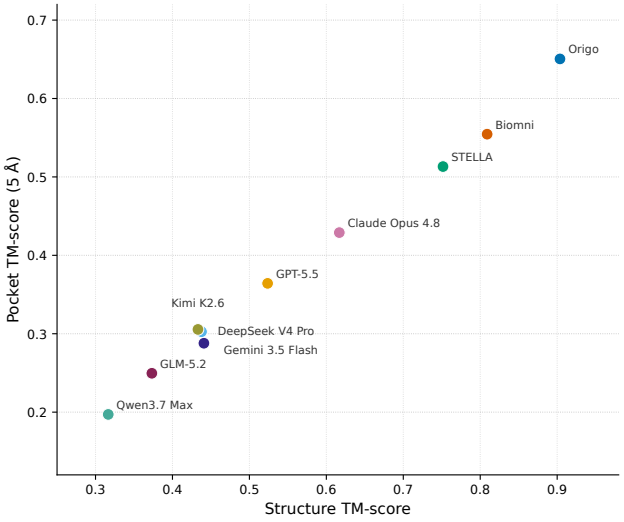

Figure S6: Comparison of the structural and pocket TM-scores of Origo and the baselines.

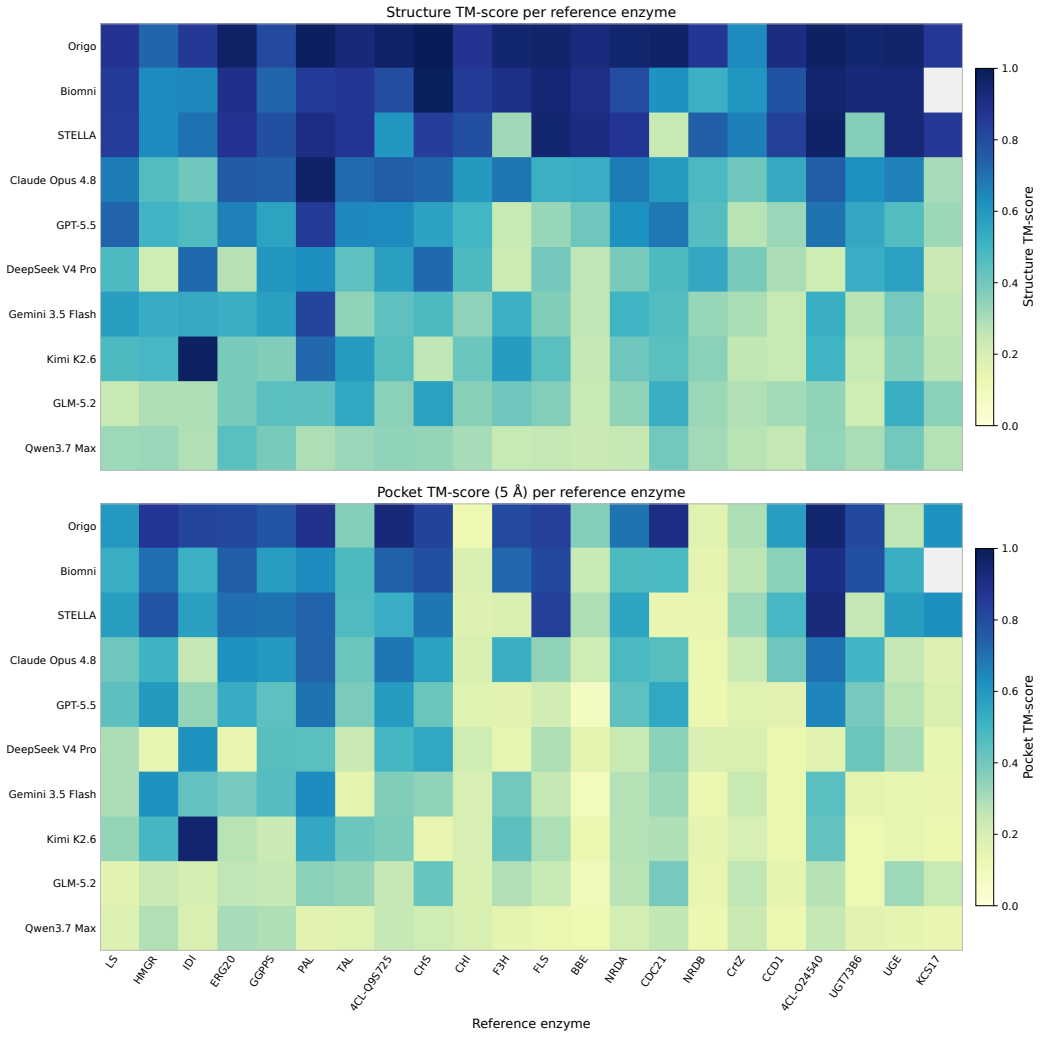

Figure S7: Structure and pocket TM-scores for each candidate enzyme of Origo and baselines.

D MCP TOOL LIST

Table S7: MCP tool list of Origo.

Tool Name	Backend	Description
Pathway Design & Model Assembly		
`list_denovo_pathways`	KEGG	List available KEGG pathways for de novo SBML assembly, grouped by category.
`build_denovo_model`	COBRApy	One-shot de novo SBML assembly from a design specification.
`patch_denovo_model`	COBRApy	Apply modifications to a de novo SBML model and re-diagnose.
Model Diagnosis & Gap Repair		
`diagnose_denovo_model`	COBRApy	Run diagnosis on an existing de novo SBML model.
`trace_product_pathway`	COBRApy	Trace biosynthesis pathway from product back to central metabolism.
`fix_product_export`	ElementTree	Fix a product export that points to a dummy or wrong species.
`search_metabolite_reactions`	MetaNetX	Search for reactions involving a given metabolite.
`search_transport_for_metabolite`	MetaNetX	Search for transport reactions for a specific metabolite.
`search_academic_content`	Exa	Search academic literature for biosynthesis pathways of a compound.
`plan_product_route`	MetaNetX (SQLite)	Find complete biosynthetic routes using MetaNetX reaction topology.
`search_retrosynthesis_route`	READRetro	AI-driven retrosynthesis using READRetro.
`search_seed_compounds`	SEED	Fuzzy-search the SEED compound database by name.
`resolve_compound_id`	SEED	Look up compound IDs by name or partial ID for use in `add_reactions_to_sbml`.
`add_reactions_to_sbml`	RDKit + ElementTree	Validate and add balanced reactions to SBML XML.
Model Refinement & Annotation		
`tune_product_flux_bounds`	COBRApy	Set realistic flux bounds on the product export reaction.
`prune_reactions_from_sbml`	COBRApy	Remove blocked (zero-flux) and/or dead-end reactions from an SBML model.
`FormatCompleter`	ElementTree	Normalize SBML model formatting and metadata representations.
Data Retrieval		
`download_uniprot_seq`	UniProt	Download protein sequence in FASTA format by UniProt ID.
Homolog Search & Alignment		
`run_mmseqs_search`	MMseqs2	Sequence similarity search against a protein database.
`run_mmseqs_cluster`	MMseqs2	Cluster protein sequences by sequence similarity.
`run_mmseqs_msa`	MMseqs2	Generate multiple sequence alignment from search results.
`run_foldseek_search`	Foldseek	Structure-based search of query structures against structural databases.
`run_tmalign_against_directory`	TM-align	Batch structural alignment; filter by TM-score and download homolog sequences.
Structure Prediction & Analysis		
`run_boltz_protein_structure_prediction`	Boltz-2	Predict 3D structure of a single protein sequence.
`run_boltz_protein_structure_prediction_mul`	Boltz-2	Batch structure prediction for multiple sequences; returns PAE scores.
`run_boltz_complex_structure_prediction`	Boltz-2	Predict protein–ligand complex structures.
`extract_pocket`	BioPython	Extract binding pocket residues from a protein–ligand complex.
`filter_cif_by_atom_distance`	BioPython	Sort and filter CIF structures by inter-atomic distance.
Phylogenetics & Enzyme Mining		
`run_prostt5`	ProstT5	Generate protein sequence embeddings.
`cluster_sequences_and_build_tree`	ProstT5	Cluster embeddings and build phylogenetic tree for homolog discovery.