CAML: A Conflict-Aware Molecular Language Model Merging Framework for Multi-Constraint Molecular Generation

Xuanbai Ren[1,2], Luoda Tan[1,2], Pei Liu[1,3], Tengfei Ma[1,2], Xiangzheng Fu[1], Longyue Wang[4], Yiping Liu[1,3,*], Xiangxiang Zeng[1,2,3,*]

[1]College of Computer Science and Electronic Engineering, Hunan University
[2]State Key Laboratory of Chemo and Biosensing, Hunan University
[3]Yuelushan Center for Industrial Innovation, Changsha
[4]Alibaba International Digital Commerce

Correspondence*: ypliu@hnu.edu.cn, xzeng@hnu.edu.cn

Abstract

Transfer learning has demonstrated efficacy in single-property constraint molecular generation. However, real-world drug discovery demands molecules to satisfy multiple property constraints. Existing paradigms often struggle with this challenge due to catastrophic forgetting or gradient conflicts. To address this, we propose a **c**onflict-**a**ware **m**olecular **l**anguage model merging framework (CAML). CAML generates multiple constraints molecular as a cooperative game among property-specific fine-tune models (expert models). Specifically, we formulate a Stability-Aware Covariance Matrix Adaptation Evolution Strategy (SACMA-ES) to dynamically optimize the fusion strategy. This algorithm searches for a Nash-equilibrium–like solution that minimizes conflicts among properties by exploring the optimal combination of the importance of the task parameter (intrinsic scale) and relative fusion weights of each expert (fusion coefficient), yielding a multi-constraint molecular property generation model without revisiting the training data. Extensive experiments demonstrate that CAML achieves state-of-the-art performance in complex multi-constraint scenarios. Our results validate that this training-free paradigm offers a robust and efficient solution for resolving intrinsic property conflicts in *de novo* molecular design.

1 Introduction

Discovering molecules that satisfy specific multi-attribute requirements is a cornerstone of modern drug discovery and materials science (Butler et al., 2018; Meyers et al., 2021; Wan et al., 2020; Li et al., 2024). However, the chemical space is astronomically vast—estimated at 10^{60} (Zeng et al., 2022). Traditional expert-driven exploration is a protracted and high-risk endeavor. With the advent of deep learning, generative artificial intelligence has emerged as a promising paradigm to navigate this space efficiently (Li et al., 2024; Ren et al., 2024).

The dominant paradigm for constraint-based molecular generation follows a transfer learning approach (Ren et al., 2024; Bagal et al., 2021). Typically, a generative model is pre-trained on large-scale general molecular datasets, then fine-tuned on a smaller, property-labeled dataset to align the generated distribution with desired attributes. However, this paradigm suffers from two critical deficiencies: 1) Data Sparsity (Figure 1a): The scarcity of real-world molecules annotated with multiple validated properties makes it difficult for a single model to learn comprehensive constraints, especially as the number of labels increases. 2) Property Conflicts (Figure 1b): Viable drug candidates are rarely defined by a single property. Instead, they require a balance of multiple physicochemical properties that often exhibit conflicting relationships (e.g., bioactivity vs. toxicity), making it difficult for a single model to satisfy these competing attribute constraints through simple fine-tuning.

Multi-Objective Optimization (MOO) strategies have been proposed to address property conflicts. These include scalarization methods that transform multi-objective problems into single-objective ones using weighted sums (Abels et al., 2019; SV et al., 2022), approaches that identify Pareto sets via two-stage processes (Yasonik, 2020; Verhellen, 2022), and techniques employing Bayesian optimization to enhance sampling efficiency (Xie et al., 2021; Gao et al., 2022). While these approaches alleviate conflicts and data scarcity to varying degrees, they introduce new bottlenecks. Specifically, multi-objective algorithms suffer from steeply increasing search latency and significantly reduced success rates as the number of objectives grows. Furthermore, existing MOO methods rely heavily on iterative search-based strategies, lacking plug-and-play functionality. This rigidity complicates future extensions or the integration of new constraints

Proceedings of the 64th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers), pages 19578–19594
July 2-7, 2026

without retraining or extensive reconfiguration.

Model merging techniques have been proposed to efficiently synthesize domain knowledge from distinct fine-tuned models, offering a promising remedy to the multi-label data scarcity inherent in the transfer learning paradigm (Ilharco et al., 2022; Yadav et al., 2023; Du et al., 2024; Marczak et al., 2025). Nevertheless, the direct application of existing fusion algorithms to multi-property molecular generation is hindered by significant methodological gaps. Primarily, these strategies rely on static heuristic thresholds for parameter scaling or dropping, lacking the dynamic adaptability required to navigate the complex, non-linear sensitivity between model parameters and molecular properties. Furthermore, current approaches typically ascribe inter-model conflicts to discrepancies in parameter signs and spatial geometric variations derived from matrix decompositions. However, the properties of molecules generated by different expert models are in stark conflict (Figure 1c), extending beyond the scope of problems these methods were designed to address.

To address these challenges, we introduce CAML, a novel model merging method designed to minimize conflicts of molecular property expert models. In this framework, we define molecular property expert model merging as a cooperative game (Holt and Roth, 2004). To minimize molecular property conflicts, we set two parameters in the model merging: intrinsic scale (k) and fusion coefficient (α). To solve for the optimal merging strategy, we propose the Stability-Aware Covariance Matrix Adaptive Evolution Algorithm (SACMA-ES) as a global solver. By defining a composite utility objective function that maximizes overall attribute enhancement while imposing strict penalties on property conflicts and single-property degradation, we ensure both global stability and individual rationality (i.e., no single property is sacrificed). SACMA-ES efficiently navigates high-dimensional non-convex parameter spaces without requiring gradient information. By adaptively updating the covariance matrix of the fusion distribution, it searches for solutions satisfying minimal conflict conditions.

Our contributions to multi-property constrained molecular generation are summarized as follows:

- We formulate a multi-property model merging as a cooperative Nash game. By modeling properties as rational experts, our approach fundamentally resolves parameter conflicts and ensures individual rationality, surpassing the limitations of static linear model merging.
- We propose a SACMA-ES strategy to search the high-dimensional, non-convex parameter space efficiently. By integrating conflict penalties and degradation constraints, our gradient-free approach robustly locates stable conflict minimization solutions.
- We propose a framework for multi-property constrained molecular generation that efficiently merges diverse expert models. This method supports plug-and-play extensibility, enabling seamless adaptation to new properties and providing a scalable solution for complex multi-constraint AI-assisted drug discovery.

2 Related Work

2.1 Molecular Design

Current approaches for molecular optimization design generally fall into three categories. **Transfer learning** (Bagal et al., 2021; Li et al., 2024) excels in single-property tasks but struggles to reconcile intrinsic conflicts in multi-objective settings. **Deep generative models** (e.g., VAEs, GANs, Diffusion) (Jin et al., 2018; Hoogeboom et al., 2022) model property distributions effectively but are heavily reliant on scarce labeled data. Meanwhile, **multi-objective optimization methods**, including Reinforcement Learning (Jin et al., 2020), Evolutionary Algorithms (Nigam et al., 2020a), and Bayesian Optimization (Gómez-Bombarelli et al., 2018), seek global optima within high-dimensional latent spaces. However, these search-based strategies are computationally intensive and lack the modularity required for scalable, plug-and-play deployment, limiting their flexibility in dynamic drug discovery pipelines.

2.2 Model Merging

Model merging integrates fine-tuned models derived from a common pre-trained base into a unified multi-task network (Radford et al., 2021; Ye et al., 2025a,b). The foundational method, Task Arithmetic (TA) (Ilharco et al., 2022), aggregates task vectors via linear combination. To address the performance degradation caused by parameter interference in TA, subsequent studies have introduced advanced interference mitigation strategies.

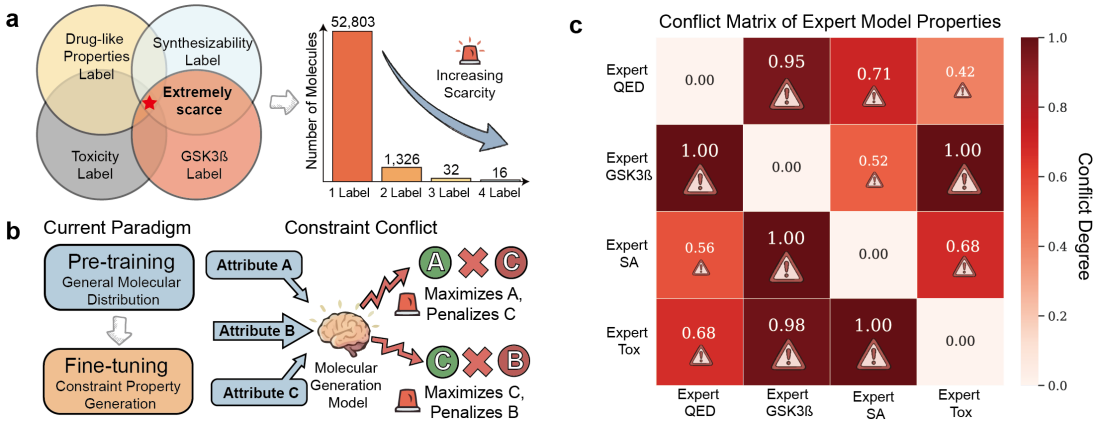

Figure 1: a.High-quality multi-label data is scarce (more details in Appendix A.1). b.The transfer learning paradigm struggles to balance conflicting attributes among molecules. c.Heatmap of property conflicts between expert models. Derived by evaluating single-property experts across all property tasks, each cell (i, j) quantifies the conflict exerted by Expert i on Property j. Scores are normalized to $[0, 1]$, where 0 indicates compatibility and 1 indicates maximum incompatibility. Darker red regions highlight significant inter-task conflicts (more details in Appendix D).

For instance, TIES (Yadav et al., 2023) and Model Breadcrumbs (Davari and Belilovsky, 2024) focus on filtering noise by eliminating sign conflicts and parameter outliers, respectively. PCB-merging (Du et al., 2024) refines this further by weighing parameters based on cross-task importance weights. More recently, Iso-merging (Marczak et al., 2025) introduces a geometric perspective, utilizing singular value decomposition (SVD) to treat models as angular relationships in latent space rather than flat vectors, thereby offering a more robust metric for conflict evaluation. However, directly transposing these paradigms to molecular generation reveals two critical shortcomings. First, reliance on static heuristics lacks the adaptability needed to navigate the non-convex chemical space. Second, existing geometric alignment strategies address interference only superficially, failing to resolve fundamental conflicts between opposing molecular properties.

3 Preliminaries

In this section, we present the general framework of the transfer learning molecular generation model, describe the formulation of task vectors, and establish the notation used throughout the paper.

3.1 Molecular Generation Framework

We formulate *de novo* molecular generation as an autoregressive sequence modeling task. A molecule is represented as a sequence of tokens $x = (x_1, \ldots, x_T)$ from a vocabulary $\mathcal{V}$. Utilizing a GPT-based architecture parameterized by θ, the joint probability of generating x is factorized as:

$$p_\theta(x) = \prod_{t=1}^{T} p_\theta(x_t \mid x_{<t}), \tag{1}$$

where $x_{<t}$ denotes the context of preceding tokens. The model training follows a standard two-stage paradigm: **Pre-training.** The model is initially trained on a large-scale unlabeled molecular datasets $\mathcal{D}_{pre}$ to learn general chemical syntax. The objective is to minimize the negative log-likelihood (NLL):

$$\mathcal{L}(\theta) = -E_{x\sim\mathcal{D}}\left[\sum_{t=1}^{T} \log p_\theta(x_t \mid x_{<t})\right]. \tag{2}$$

Fine-tuning. To construct property-specific expert models, we further fine-tune the pre-trained parameters on a domain-specific dataset $\mathcal{D}_{ft}$ containing molecules with desired property (e.g., high QED). This process optimizes the same NLL objective over $\mathcal{D}_{ft}$, yielding a specialized expert model θ_{ft} tailored for downstream constraints.

3.2 Task Vectors and Model Merging

Definition. Building upon the transfer learning paradigm, we utilize task vectors to manipulate domain-specific capabilities. Let θ_{pre} denote the parameter vector of the foundational, general molecular generation model. For a specific property i (e.g., QED or GSK3), we obtain a specialized expert model $\theta_{ft}^{(i)}$ via fine-tuning. The task vector τ_i is formally defined as the element-wise difference between the expert and pre-trained parameters:

$$\tau_i = \theta_{ft}^{(i)} - \theta_{pre}. \tag{3}$$

Model Merging. Model merging aims to synthesize a multi-property generator by aggregating these task vectors. In a standard linear merging framework (e.g., TIES), the fused parameter set θ_{fuse} is computed by adding the weighted sum of task vectors back to the pre-trained base:

$$\theta_{fuse} = \theta_{pre} + \sum_{i=1}^{N} \lambda_i \tau_i, \tag{4}$$

where N is the number of target attributes and $\lambda_i \in R$ is a scalar scaling coefficient of the i-th task vector.

4 Method

4.1 Overview

In this section, we present CAML, a framework designed to resolve property conflicts in multi-constraint molecular generation based on model merging. We build upon a set of property-specific expert models, for which the detailed descriptions of pre-training data, fine-tuning datasets, and training protocols are provided in Appendix A.1 and A.2. Departing from rigid heuristic merging strategies, we reconceptualize the model merging process as a cooperative game among these experts. Crucially, we employ a Nash Equilibrium formulation to theoretically derive the optimal fusion strategy as a weighted centroid of expert parameters, providing a rigorous mathematical foundation for our approach.

Guided by this theoretical insight and operating on the task vectors defined in Section 3.2 (as illustrated in Figure 2), we parameterize the contribution of each expert model using intrinsic scale (k) and fusion coefficient (α). This transforms the discrete selection problem into a continuous optimization task. To solve this high-dimensional, non-convex game, we propose SACMA-ES to search for the practical Nash Equilibrium, a stable state where the collective utility of all attributes is maximized without sacrificing individual rationality. The following subsections detail the game-theoretic formulation, the parameterization strategy, and the conflict-aware optimization process.

4.2 Nash Equilibrium Formulation for Model Merging

To resolve the property conflicts among expert models, we formulate the merging process as a cooperative game (more details in Appendix A.4). In this setting, the fused task vector τ represents the conflict minimization scenario, and each expert model i acts as a player aiming to minimize the deviation between the consensus vector τ and its own optimal parameter state τ_i.

We define the utility of the fusion system as minimizing the aggregate weighted disagreement cost. Let $w_i \in R^+$ denote the importance coefficient of the i-th expert model. The global objective function $\mathcal{L}(\tau)$ is defined as the weighted sum of squared distances between the fused vector and each expert vector:

$$\mathcal{L}(\tau) = \sum_{i=1}^{N} w_i \|\tau - \tau_i\|^2, \tag{5}$$

where N is the number of properties. This objective function represents a strictly convex quadratic optimization problem. To find the optimal fusion strategy τ^* that satisfies the Nash equilibrium condition (i.e., a state where the collective dissatisfaction is minimized), we solve for the stationary point where the gradient of $\mathcal{L}(\tau)$ with respect to τ equals zero:

$$\frac{\partial \mathcal{L}}{\partial \tau} = \sum_{i=1}^{N} 2w_i(\tau - \tau_i) = 0. \tag{6}$$

Rearranging the terms yields:

$$\tau \cdot \sum_{i=1}^{N} w_i = \sum_{i=1}^{N} w_i \tau_i. \tag{7}$$

Thus, we derive the closed-form analytical solution for the optimal fused vector τ_{fused}:

$$\tau_{fused} = \frac{\sum_{i=1}^{N} w_i \tau_i}{\sum_{i=1}^{N} w_i}. \tag{8}$$

Equation 8 reveals that the optimal Nash equilibrium strategy is mathematically equivalent to the weighted centroid of the expert task vectors. This theoretical insight implies that the ideal fused model lies within the convex combination of the experts, governed by the optimal weight profile $\mathbf{w} = \{w_1, \ldots, w_N\}$.

We distinguish our Nash equilibrium-based weight allocation from traditional heuristic fusion methods (such as uniform averaging or empirical scalar weighting). Heuristic methods blindly interpolate parameters, often leading to destructive gradient interference when tasks conflict. In contrast, our equilibrium formula constructs the fusion process as a cooperative game, mathematically guaranteeing convergence to Pareto optimality.

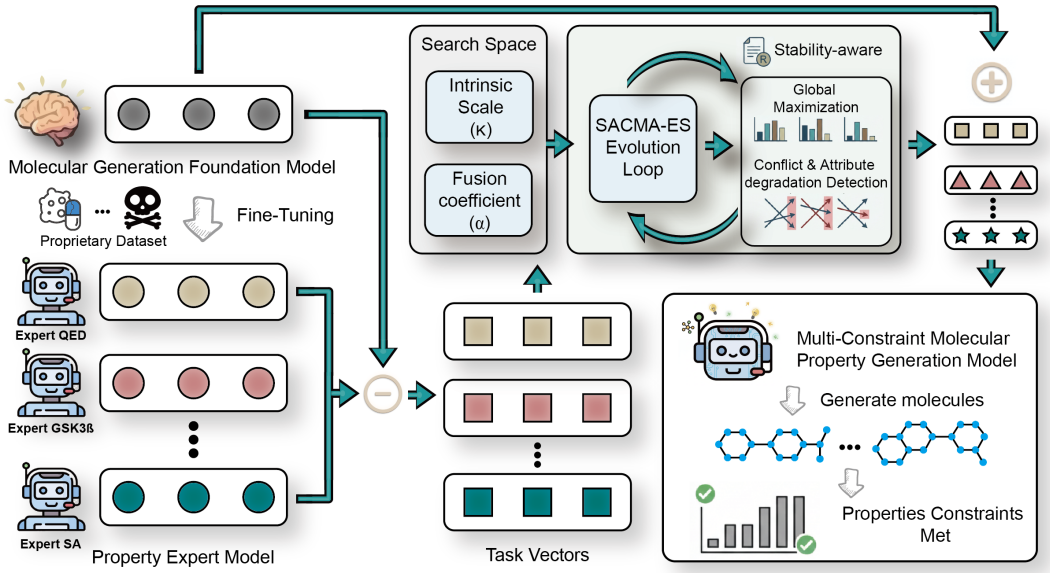

Figure 2: The overall framework of the proposed multi-constraint molecular generation method. Different color blocks represent distinct property expert models derived from fine-tuning. The framework begins with a pre-trained foundation model and multiple expert models to extract task vectors. A SACMA-ES algorithm is employed to search for optimal configurations, specifically determining the intrinsic scale (k) and fusion coefficient (α) for each expert. Finally, the modulated task vectors are merged into the pre-trained model to construct the final model capable of generating molecules satisfying multiple constraints.

4.3 Parameterization Strategy

While Equation 8 provides the theoretical optimum, directly applying it is impractical due to two challenges: (1) the disparate scales of raw task vectors (τ_i) caused by varying convergence behaviors during fine-tuning, and (2) the unknown optimal distribution of weights. To operationalize the Nash solution, we propose a dual-parameter decomposition strategy.

We reformulate the theoretical weights into two distinct learnable scalar components: the **Intrinsic Scale** (k_i) and the **Fusion Coefficient** (α_i).

1. Intrinsic Scale ($k_i \in [0, 1]$): This parameter acts as a calibration factor for the magnitude of the task vector. Since gradients from different experts may vary significantly in norm, a vector with a larger norm could unintentionally dominate the fusion. k_i calibrates the confidence or intensity of the i-th expert without altering its direction, ensuring that all experts contribute on a comparable scale.

2. Fusion Coefficient ($\alpha_i \in R^+$): This parameter approximates the normalized weight term $\frac{w_i}{\sum w_j}$ in Eq. 8. It represents the strategic priority of the i-th property within the multi-constraint game, determining the mixing ratio of the calibrated vectors.

By substituting these parameters into the linear merging framework, the practical fused task vector is expressed as:

$$\tau_{fused} = \sum_{i=1}^{N} \alpha_i \cdot (k_i \cdot \tau_i). \tag{9}$$

This parameterization transforms the fusion problem into finding the optimal set $\Theta = \{k_1, \ldots, k_N, \alpha_1, \ldots, \alpha_N\}$. Since the molecular generation task involves discrete token sampling and the evaluation metrics (e.g., QED, Docking Score) are typically non-differentiable black-box functions, gradient-based optimization is inapplicable.

To address this, we employ SACMA-ES, a robust derivative-free optimization algorithm. SACMA-ES efficiently explores the continuous parameter space of Θ to maximize the global utility (defined by property scores). This approach enables a *training-free* paradigm: once the optimal parameters are identified, the expert models are dynamically merged to generate high-quality molecules without computationally expensive retraining of the backbone network.

4.4 Stability Aware Covariance Matrix Adaptation Evolution Strategy

The covariance matrix adaptation evolution strategy (CMA-ES) (Hansen et al., 2003) relies on a scalar

fitness function to guide the evolution of the covariance matrix. However, a naive linear combination of property scores (e.g., weighted sum) often fails in molecular generation tasks. Such approaches tend to bias the search towards easy-to-optimize properties (e.g., rapidly increasing QED) while neglecting harder constraints. Furthermore, without explicit constraints, the algorithm may sacrifice one property to artificially boost another, violating the principle of individual rationality.

To address these limitations, we propose a Conflict-Regularized Fitness Function, denoted as $\mathcal{F}(\Theta)$. This function is designed to enforce Nash equilibrium conditions by integrating multi-property gains with explicit penalties for conflict and degradation.

Before formulating the global fitness, the raw evaluation metric $S_i(\Theta)$ of the i-th property is mapped to a normalized reward score $R_i(\Theta) \in (0, 1]$ to unify disparate numerical scales. A small constant $\epsilon = 10^{-8}$ is inherently added to R_i to ensure numerical stability for subsequent logarithmic operations. The fitness function is formulated as follows:

$$\mathcal{F}(\Theta) = \sum_{i=1}^{N} \alpha_i \log(R_i) - \lambda_c \mathcal{L}_{conf} - \lambda_d \mathcal{L}_{drop}; \tag{10}$$

where α_i is the preference weight for the i-th property, and λ_c, λ_d are regularization coefficients. The three components of this objective function serve distinct roles in guiding the search:

Weighted Log-Product (Global Maximization): This term $\sum \alpha_i \log(R_i)$ acts as the primary optimization engine to maximize the collective payoff. While simple summation focuses on total quantity, the logarithmic form intrinsically favors a fair distribution of rewards, ensuring that the system prioritizes lifting low-scoring attributes to maximize the geometric mean of the global objective.

Conflict Penalty ($\mathcal{L}_{conf}$): While the first term encourages high scores, it does not guarantee that properties improve at the same pace. This term minimizes the variance of improvement rates across tasks to align the optimization directions of all experts:

$$\mathcal{L}_{conf} = Var(R_1, R_2, \ldots, R_N). \tag{11}$$

It acts as a regularizer to enforce co-elevation, penalizing scenarios where one property improves rapidly while another stagnates.

Degradation Penalty ($\mathcal{L}_{drop}$): This term serves as a strict safety anchor. Even with the above mechanisms, trade-offs might occasionally favor a massive gain in one attribute at the slight cost of another. This penalty imposes a hard constraint to ensure that no property performance drops below its initial expert baseline score, denoted as R_i^{base}:

$$\mathcal{L}_{drop} = \sum_{i=1}^{N} \max\left(0, R_i^{base} - R_i\right). \tag{12}$$

By explicitly penalizing performance drops, this term preserves the individual rationality of each expert model.

By maximizing $\mathcal{F}(\Theta)$, the SACMA-ES algorithm is effectively constrained to explore the cooperative subspace of the parameters (details in Appendix A.5 and A.6), locating a robust fusion strategy that satisfies all multi-attribute requirements simultaneously.

5 Experiments and Results

We conduct extensive experiments to evaluate the effectiveness of CAML in generating molecules under multi-property constraints. We designed four increasingly complex experimental scenarios to evaluate the properties of 10,000 generated molecules, testing the model's ability to balance conflicting objectives: **1) QED + Penalized LogP:** A standard non-biological benchmark targeting drug-like molecules with optimal hydrophobicity while penalizing structural complexity and excessive rings. **2) QED + GSK3β:** A biological task aiming to generate drug-like inhibitors for GSK3β, a key target in Alzheimer's disease therapy. **3) QED + SA + GSK3β:** An extension of the previous task that additionally enforces Synthetic Accessibility (SA) to ensure chemical feasibility. **4) QED + SA + Tox + GSK3β:** A comprehensive setting that simultaneously optimizes for bioactivity, drug-likeness, and synthesizability, while strictly constraining toxicity (tox) to ensure safety.

5.1 Baselines

To comprehensively evaluate our framework, we compare it with three representative categories of methods: **1) Transfer Learning:** The standard paradigm where a general molecular generator (ChemGPT) (Frey et al., 2022) is fine-tuned on property-labeled data (details in Appendix A.7). **2) Multi-Objective Optimization:** RL-based approaches, including **PPO** (Schulman et al., 2017)

Method	QED+PLogP	QED+GSK3β	QED+SA+GSK3β	QED+SA+Tox+GSK3β
ChemGPT (transfer learning) (Frey et al., 2022)	0.304 ± 0.007	0.346 ± 0.007	0.411 ± 0.005	0.450 ± 0.012
ChemGPT-PPO (Frey et al., 2022)	0.550 ± 0.001	0.467 ± 0.001	0.357 ± 0.010	0.657 ± 0.006
RationaleRL (Jin et al., 2020)	0.363 ± 0.011	0.323 ± 0.008	0.589 ± 0.017	0.603 ± 0.023
TIES (Yadav et al., 2023)	0.214 ± 0.004	0.321 ± 0.004	0.342 ± 0.011	0.300 ± 0.003
PCB (Davari and Belilovsky, 2024)	0.245 ± 0.003	0.321 ± 0.003	0.357 ± 0.004	0.341 ± 0.010
Iso-c (Marczak et al., 2025)	0.208 ± 0.005	0.357 ± 0.005	0.371 ± 0.004	0.417 ± 0.013
Iso-cts (Marczak et al., 2025)	0.239 ± 0.005	0.369 ± 0.003	0.380 ± 0.004	0.437 ± 0.014
CAML (Ours)	$\mathbf{0.145 \pm 0.004}$	$\mathbf{0.233 \pm 0.001}$	$\mathbf{0.253 \pm 0.010}$	$\mathbf{0.239 \pm 0.010}$

Table 1: Comparison of different methods on NCD ↓.

Method	QED+PLogP	QED+GSK3β	QED+SA+GSK3β	QED+SA+Tox+GSK3β
ChemGPT (transfer learning) (Frey et al., 2022)	0.53 ± 0.003	0.31 ± 0.006	0.38 ± 0.003	0.42 ± 0.007
ChemGPT-PPO (Frey et al., 2022)	0.31 ± 0.001	0.24 ± 0.001	0.44 ± 0.001	0.34 ± 0.003
RationaleRL (Jin et al., 2020)	0.42 ± 0.009	$\mathbf{0.57 \pm 0.007}$	0.40 ± 0.014	0.46 ± 0.013
TIES (Yadav et al., 2023)	0.56 ± 0.002	0.38 ± 0.005	0.44 ± 0.008	0.50 ± 0.001
PCB (Davari and Belilovsky, 2024)	0.54 ± 0.004	0.36 ± 0.002	0.45 ± 0.002	0.48 ± 0.005
Iso-c (Marczak et al., 2025)	0.51 ± 0.014	0.31 ± 0.004	0.42 ± 0.003	0.45 ± 0.006
Iso-cts (Marczak et al., 2025)	0.49 ± 0.003	0.30 ± 0.002	0.41 ± 0.002	0.44 ± 0.007
CAML (Ours)	$\mathbf{0.59 \pm 0.002}$	0.40 ± 0.001	$\mathbf{0.49 \pm 0.005}$	$\mathbf{0.54 \pm 0.006}$

Table 2: Comparison of different methods on WAU ↑.

(details in Appendix A.8), use proximal policy optimization algorithms for ChemGPT, and **RationaleRL** (Jin et al., 2020), which extends molecule rationales into complete molecules using reinforcement learning (details in Appendix A.9). **3) Model Merging:** Recent state-of-the-art parameter merging techniques, specifically **TIES** (Yadav et al., 2023), **PCB-Merging** (Davari and Belilovsky, 2024), and **Iso-Merging** (Marczak et al., 2025).

We excluded search-based methods (Liu et al., 2025; Sun et al., 2022; Nigam et al., 2020b) (e.g., GA and MCMC) because they rely on instance-specific optimization rather than learning generalized generative policies. Furthermore, to ensure rigorous and fair comparisons, we did not include recent LLM-based methods (e.g., M4olGen (Li et al., 2026) and TSMMG (Zhou et al., 2025)) in our main evaluation due to the significant differences in pre-training data size (e.g., TSMMG uses 40GB of data, while our standard dataset is only 524MB). All major baseline models in this paper are strictly aligned and trained on the same pre-training corpus.

5.2 Evaluation Metrics

To quantitatively assess the performance and statistical robustness of the proposed framework, we employ three key metrics focusing on constraint satisfaction, conflict minimization, and chemical space exploration, reporting the mean and standard deviation across five independent runs with different random seeds. Detailed mathematical definitions and calculation procedures for all metrics are provided in Appendix B.

Nash Convergence Distance (NCD): This metric measures the Euclidean distance between the average property score of the generated batch and the Ideal Point formed by the consensus of individual expert models. A lower NCD signifies better alignment with the collective expert optimum, indicating the model has successfully converged to a state of minimal conflict.

Weighted Average Utility (WAU): To quantify the comprehensive quality of the molecules, we utilize a weighted scalar score derived from normalized property values. This metric reflects the overall utility of the generated batch, balancing multiple conflicting objectives based on task-specific importance.

Comprehensive Discovery Score (CDS): A composite metric designed to evaluate the critical trade-off between exploitation (quality) and exploration (diversity). CDS is calculated as the weighted sum of **Success Rate (SR)**, **Diversity (Div)**, and **Novelty (Nov)**. This metric serves as a holistic indicator to ensure the model generates high-quality candidates without collapsing into repetitive modes.

5.3 Comparison with Baselines

We present a comprehensive analysis focusing on three critical dimensions: convergence stability (NCD ↓, Table 1), multi-constraint utility (WAU ↑, Table 2), and the overall discovery capability (CDS ↑, Table 3).

Convergence Stability (NCD). As constraint complexity increases, reinforcement learning (RL) baselines exhibit significant optimization

Method	QED+PLogP	QED+GSK3β	QED+SA+GSK3β	QED+SA+Tox+GSK3β
ChemGPT (transfer learning) (Frey et al., 2022)	0.75 ± 0.015	0.53 ± 0.007	0.49 ± 0.002	0.47 ± 0.002
ChemGPT-PPO (Frey et al., 2022)	0.38 ± 0.003	0.45 ± 0.001	0.47 ± 0.001	0.40 ± 0.007
RationaleRL (Jin et al., 2020)	0.62 ± 0.012	0.54 ± 0.007	0.43 ± 0.022	0.41 ± 0.017
TIES (Yadav et al., 2023)	0.66 ± 0.003	0.50 ± 0.002	0.46 ± 0.012	0.45 ± 0.007
PCB (Davari and Belilovsky, 2024)	0.64 ± 0.005	0.50 ± 0.004	0.48 ± 0.010	0.46 ± 0.005
Iso-c (Marczak et al., 2025)	0.69 ± 0.005	0.51 ± 0.004	0.49 ± 0.005	0.47 ± 0.005
Iso-cts (Marczak et al., 2025)	0.68 ± 0.003	0.50 ± 0.002	0.49 ± 0.001	0.46 ± 0.003
CAML (Ours)	$\mathbf{0.81 \pm 0.003}$	$\mathbf{0.60 \pm 0.004}$	$\mathbf{0.57 \pm 0.013}$	$\mathbf{0.50 \pm 0.006}$

Table 3: Comparison of different methods on CDS ↑.

drift. In the most challenging scenario (Task 4: QED+SA+Tox+GSK3β), ChemGPT-PPO and RationaleRL record extremely high NCD values of 0.657 and 0.603, respectively. This indicates a severe deviation from the expert consensus due to gradient conflicts. In contrast, CAML achieves the lowest NCD of 0.239, significantly outperforming static merging methods like TIES (0.300). This validates that our dynamic parameterization effectively minimizes conflicts, anchoring the generated distribution to the area with the minimum conflict.

Utility and Discovery Trade-off (WAU & CDS). Regarding utility, CAML demonstrates state-of-the-art performance in high-conflict settings. In Task 4, our method achieves a WAU of 0.54, surpassing the strongest baseline TIES (0.51) and RL-based PPO (0.34).

More importantly, the Comprehensive Discovery Score (CDS) reveals the superiority of our method in balancing quality with exploration. While RationaleRL achieves a higher WAU in the simpler Task 2 (0.57 vs. 0.40), its CDS is significantly lower than ours (0.54 vs. 0.60). This discrepancy suggests that RL methods tend to over-optimize rewards by collapsing into a few repetitive modes (low diversity). In contrast, CAML consistently achieves the highest CDS across all tasks (e.g., 0.50 in Task 4 vs. 0.45 for TIES), proving that it generates high-quality molecules while maintaining rich chemical diversity and novelty without requiring training.

5.4 Ablation Study

To validate the contribution of each component in the CAML framework, we conducted an ablation study comparing the full model against variants with fixed parameters or without regularization. We focus primarily on the NCD, which quantifies the model's ability to align with the collective expert consensus.

As shown in Figure 3, dynamic parameterization is critical for minimizing optimization drift. In the most complex scenario (Task 4), removing the adaptive preference weights (*w/o Fusion coefficient* α) results in a sharp increase in NCD to 0.333, compared to 0.239 for the full model. Similarly, fixing the intrinsic scale (*w/o Intrinsic Scale* k) degrades stability, yielding an NCD of 0.314. Even the stability aware regularization proves necessary, as its removal increases NCD to 0.259. These results confirm that simply averaging gradients, even with sign conflict resolution like TIES (NCD 0.300), is insufficient. The k and α are essential for effectively anchoring the generated distribution in the area with the minimum conflict. Additional analyses on utility (WAU) and discovery capability (CDS) are provided in Appendix C.

6 Conclusion

We introduced CAML, a unified framework that resolves parameter conflicts in multi-attribute molecular generation through the lens of cooperative game theory. By utilizing the principle of Nash equilibrium and a stability-aware evolutionary search strategy, our method effectively harmonizes conflicting expert models without computationally expensive re-training. Empirical results show that CAML achieves state-of-the-art performance in complex constraints (e.g., QED+SA+Tox+GSK3β), surpassing existing merging and RL baselines in NCD and other evaluation methods. This approach provides a flexible, efficient, and explainable paradigm for navigating high-dimensional objective spaces in *de novo* drug design.

Limitations

Our proposed CAML framework, driven by SACMA-ES, demonstrates superior efficacy in generating multi-constraint molecules. However, it being a training-free paradigm, the search cost at inference time remains non-negligible. Future work could alleviate this problem by using deep learning predictors or theoretically analyzing the mapping

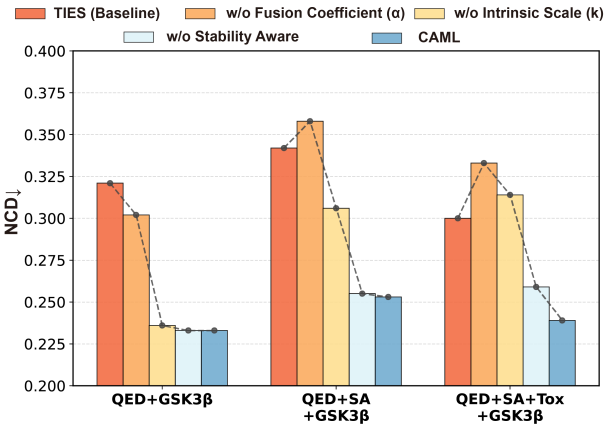

Figure 3: Ablation Study on NCD ↓.

between molecular property generation models and parameters to more quickly approximate the optimal fusion weights.

And, the performance of the fused model is inherently bounded by the quality of the initial fine-tuning expert models. Our method excels at resolving conflicts and finding the Pareto-optimal integration of existing capabilities, but it cannot synthesize new knowledge that is absent from the constituent experts. In the future, we may explore leveraging commonalities in knowledge across expert models to achieve greater improvements for a specific attribute while maintaining the performance of other attributes.

Acknowledgments

This work was supported by the National Natural Science Foundation of China (Grant No. 62425204, U22A2037, 62450002, 62432011, 62522110, 62472152, 62372158, 62572178), the Hunan Provincial Natural Science Foundation of China (Grant No. 2024JJ4015) and the Project of Yuelushan Center for Industrial Innovation (Grant No. 2025YCII0214), Fundamental and Interdisciplinary Disciplines Breakthrough Plan of the Ministry of Education of ChinaGrant No. JYB2025XDXM602)

References

Axel Abels, Diederik Roijers, Tom Lenaerts, Ann Nowé, and Denis Steckelmacher. 2019. Dynamic weights in multi-objective deep reinforcement learning. In *International conference on machine learning*, pages 11–20. PMLR.

Viraj Bagal, Rishal Aggarwal, PK Vinod, and U Deva Priyakumar. 2021. Molgpt: molecular generation using a transformer-decoder model. *Journal of chemical information and modeling*, 62(9):2064–2076.

Keith T Butler, Daniel W Davies, Hugh Cartwright, Olexandr Isayev, and Aron Walsh. 2018. Machine learning for molecular and materials science. *Nature*, 559(7715):547–555.

MohammadReza Davari and Eugene Belilovsky. 2024. Model breadcrumbs: Scaling multi-task model merging with sparse masks. In *European Conference on Computer Vision*, pages 270–287. Springer.

Guodong Du, Junlin Lee, Jing Li, Runhua Jiang, Yifei Guo, Shuyang Yu, Hanting Liu, Sim Kuan Goh, Ho-Kin Tang, Daojing He, and Min Zhang. 2024. Parameter competition balancing for model merging. In *Advances in Neural Information Processing Systems 38: Annual Conference on Neural Information Processing Systems 2024, NeurIPS 2024, Vancouver, BC, Canada, December 10 - 15, 2024*.

Nathan Frey, Ryan Soklaski, Simon Axelrod, Siddharth Samsi, Rafael Gomez-Bombarelli, Connor Coley, and Vijay Gadepally. 2022. Neural scaling of deep chemical models. *ChemRxiv*.

Wenhao Gao, Tianfan Fu, Jimeng Sun, and Connor Coley. 2022. Sample efficiency matters: a benchmark for practical molecular optimization. *Advances in Neural Information Processing Systems*, 35:21342–21357.

Rafael Gómez-Bombarelli, Jennifer N Wei, David Duvenaud, José Miguel Hernández-Lobato, Benjamín Sánchez-Lengeling, Dennis Sheberla, Jorge Aguilera-Iparraguirre, Timothy D Hirzel, Ryan P Adams, and Alán Aspuru-Guzik. 2018. Automatic chemical design using a data-driven continuous representation of molecules. *ACS central science*, 4(2):268–276.

Nikolaus Hansen, Sibylle D Müller, and Petros Koumoutsakos. 2003. Reducing the time complexity of the derandomized evolution strategy with covariance matrix adaptation (cma-es). *Evolutionary computation*, 11(1):1–18.

Charles A Holt and Alvin E Roth. 2004. The nash equilibrium: A perspective. *Proceedings of the National Academy of Sciences*, 101(12):3999–4002.

Emiel Hoogeboom, Vıctor Garcia Satorras, Clément Vignac, and Max Welling. 2022. Equivariant diffusion for molecule generation in 3d. In *International conference on machine learning*, pages 8867–8887. PMLR.

Gabriel Ilharco, Marco Tulio Ribeiro, Mitchell Wortsman, Suchin Gururangan, Ludwig Schmidt, Hannaneh Hajishirzi, and Ali Farhadi. 2022. Editing models with task arithmetic. *arXiv preprint arXiv:2212.04089*.

Wengong Jin, Regina Barzilay, and Tommi Jaakkola. 2020. Multi-objective molecule generation using interpretable substructures. In *International conference on machine learning*, pages 4849–4859. PMLR.

Wengong Jin, Kevin Yang, Regina Barzilay, and Tommi Jaakkola. 2018. Learning multimodal graph-to-graph translation for molecule optimization. In *International Conference on Learning Representations*.

Tingting Li, Xuanbai Ren, Xiaoli Luo, Zhuole Wang, Zhenlu Li, Xiaoyan Luo, Jun Shen, Yun Li, Dan Yuan, Ruth Nussinov, Xiangxiang Zeng, Junfeng Shi, and Feixiong Cheng. 2024. A foundation model identifies broad-spectrum antimicrobial peptides against drug-resistant bacterial infection. *Nature Communications*, 15(1):7538.

Yizhan Li, Florence Cloutier, Sifan Wu, Ali Parviz, Boris Knyazev, Yan Zhang, Glen Berseth, and Bang Liu. 2026. Mˆ 4olgen: Multi-agent, multi-stage molecular generation under precise multi-property constraints. *arXiv preprint arXiv:2601.10131*.

Yiping Liu, Jiahao Yang, Xuanbai Ren, Xinyi Zhang, Yuansheng Liu, Bosheng Song, Xiangxiang Zeng, and Hisao Ishibuchi. 2025. Multi-objective molecular design through learning latent pareto set. In *Thirty-Ninth AAAI Conference on Artificial Intelligence, Thirty-Seventh Conference on Innovative Applications of Artificial Intelligence, Fifteenth Symposium on Educational Advances in Artificial Intelligence, AAAI 2025, Philadelphia, PA, USA, February 25 - March 4, 2025*, pages 19006–19014. AAAI Press.

Daniel Marczak, Simone Magistri, Sebastian Cygert, Bartłomiej Twardowski, Andrew D Bagdanov, and Joost van de Weijer. 2025. No task left behind: Isotropic model merging with common and task-specific subspaces. *arXiv preprint arXiv:2502.04959*.

Joshua Meyers, Benedek Fabian, and Nathan Brown. 2021. De novo molecular design and generative models. *Drug Discovery Today*, 26(11):2707–2715.

AkshatKumar Nigam, Pascal Friederich, Mario Krenn, and Alán Aspuru-Guzik. 2020a. Augmenting genetic algorithms with deep neural networks for exploring the chemical space. *ICLR 2020*.

AkshatKumar Nigam, Pascal Friederich, Mario Krenn, and Alán Aspuru-Guzik. 2020b. Augmenting genetic algorithms with deep neural networks for exploring the chemical space. In *8th International Conference on Learning Representations, ICLR 2020, Addis Ababa, Ethiopia, April 26-30, 2020*. OpenReview.net.

Alec Radford, Jong Wook Kim, Chris Hallacy, Aditya Ramesh, Gabriel Goh, Sandhini Agarwal, Girish Sastry, Amanda Askell, Pamela Mishkin, Jack Clark, Gretchen Krueger, and Ilya Sutskever. 2021. Learning transferable visual models from natural language supervision. In *Proceedings of the 38th International Conference on Machine Learning*, volume 139 of *Proceedings of Machine Learning Research*, pages 8748–8763. PMLR.

Xuanbai Ren, Jiaying Wei, Xiaoli Luo, Yuansheng Liu, Kenli Li, Qiang Zhang, Xin Gao, Sizhe Yan, Xia Wu, Xingyue Jiang, Mingquan Liu, Dongsheng Cao, Leyi Wei, Xiangxiang Zeng, and Junfeng Shi. 2024. Hydrogelfinder: A foundation model for efficient self-assembling peptide discovery guided by non-peptidal small molecules. *Advanced Science*, 11(26):2400829.

John Schulman, Filip Wolski, Prafulla Dhariwal, Alec Radford, and Oleg Klimov. 2017. Proximal policy optimization algorithms. *arXiv preprint arXiv:1707.06347*.

Mengying Sun, Jing Xing, Han Meng, Huijun Wang, Bin Chen, and Jiayu Zhou. 2022. Molsearch: Search-based multi-objective molecular generation and property optimization. In *KDD '22: The 28th ACM SIGKDD Conference on Knowledge Discovery and Data Mining, Washington, DC, USA, August 14 - 18, 2022*, pages 4724–4732. ACM.

Shree Sowndarya SV, Jeffrey N Law, Charles E Tripp, Dmitry Duplyakin, Erotokritos Skordilis, David Biagioni, Robert S Paton, and Peter C St. John. 2022. Multi-objective goal-directed optimization of de novo stable organic radicals for aqueous redox flow batteries. *Nature Machine Intelligence*, 4(8):720–730.

Jonas Verhellen. 2022. Graph-based molecular pareto optimisation. *Chemical Science*, 13(25):7526–7535.

Chengzhang Wan, Xiangfeng Duan, and Yu Huang. 2020. Molecular design of single-atom catalysts for oxygen reduction reaction. *Advanced Energy Materials*, 10(14):1903815.

Yutong Xie, Chence Shi, Hao Zhou, Yuwei Yang, Weinan Zhang, Yong Yu, and Lei Li. 2021. Mars: Markov molecular sampling for multi-objective drug discovery. In *International Conference on Learning Representations*.

Prateek Yadav, Derek Tam, Leshem Choshen, Colin A Raffel, and Mohit Bansal. 2023. Ties-merging: Resolving interference when merging models. *Advances in Neural Information Processing Systems*, 36:7093–7115.

Jacob Yasonik. 2020. Multiobjective de novo drug design with recurrent neural networks and nondominated sorting. *Journal of Cheminformatics*, 12(1):14.

Guanghui Ye, Huan Zhao, Zixing Zhang, and Zhihua Jiang. 2025a. Unide: A multi-level and low-resource framework for automatic dialogue evaluation via llm-based data augmentation and multitask learning. *Inf. Process. Manag.*, 62(3):104035.

Guanghui Ye, Huan Zhao, Zhixue Zhao, Xupeng Zha, Yang Liu, and Zhihua Jiang. 2025b. Knowledge image matters: Improving knowledge-based visual reasoning with multi-image large language models. In *Proceedings of the 63rd Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers), ACL 2025, Vienna, Austria, July 27 - August 1, 2025*, pages 21883–21896. Association for Computational Linguistics.

Xiangxiang Zeng, Fei Wang, Yuan Luo, Seung-gu Kang, Jian Tang, Felice C Lightstone, Evandro F Fang, Wendy Cornell, Ruth Nussinov, and Feixiong Cheng. 2022. Deep generative molecular design reshapes drug discovery. *Cell Reports Medicine*, 3(12).

Peng Zhou, Jianmin Wang, Chunyan Li, Zixu Wang, Yiping Liu, Siqi Sun, Jianxin Lin, Leyi Wei, Xibao Cai, Houtim Lai, Wei Liu, Longyue Wang, Yuansheng Liu, and Xiangxiang Zeng. 2025. Instruction multi-constraint molecular generation using a teacher-student large language model. *BMC biology*, 23(1):105.

A Training Details

A.1 Data Preparation and Analysis

The quality and distribution of training data are crucial for training expert models. We utilize the PubChem dataset as our primary source, which is a large-scale database containing approximately 6.6 million molecules.

Data Preprocessing. We obtained the following molecular property scores using RDKit and a random forest model.

Table 4: **Statistics of Single-Property High-Quality Subsets.** We filtered the PubChem dataset to create training sets for individual experts. Note that single-property data is abundant.

Property	Filtering Criteria	Count
Penalized logP	Score ≤ 2	1,821,160
QED	Score ≥ 0.9	229,983
GSK3β	Inhibition ≥ 0.5 (Active)	52,803
SA Score	Score ≤ 2 (Easy)	413,524
Toxicity	Tox Prob ≤ 0.2 (Non-toxic)	1,469,911

Table 5: **Scarcity of Multi-Property Data .** The number of molecules satisfying *all* constraints simultaneously drops to near zero as task complexity increases.

Scenario	Combined Constraints	Count
Task 1	QED + PLogP	24,743
Task 2	QED + GSK3β	1,326
Task 3	QED + SA + GSK3β	32
Task 4	**QED + SA + Tox + GSK3β**	**16**

Single-Property vs. Multi-Property Data Scarcity. A core motivation for our CAML framework is the scarcity of perfect molecules that satisfy all constraints simultaneously in the training distribution.

- **Abundance of Single-Property Data:** As shown in Table 4, it is relatively easy to curate high-quality subsets for individual properties. For instance, we identified **229,983** molecules with high QED scores and **52,803** GSK3β inhibitors, providing sufficient data to train robust single-task expert models.
- **The Multi-Property Data Void:** In contrast, finding molecules that simultaneously satisfy multiple constraints is exponentially more difficult. As detailed in Table 5, the intersection of high-quality labels shrinks drastically as the number of constraints increases. For the most complex scenario (Task 4), only **16** molecules (approx. $<$ **0.000002**%) in the entire dataset meet all criteria (High QED + Low SA + Non-toxic + Active). This data void necessitates our strategy of merging separate experts rather than training a single model on non-existent multi-label data.

A.2 Pre-training and Fine-tuning Strategy

Our foundation model is based on the ChemGPT architecture, a Transformer-based decoder-only model (1.9 million parameters).

Pre-training Phase. We pre-trained the model on the PubChem dataset to learn the general syntax of SELFIES strings and the distribution of drug-like chemical space. The objective was standard Next Token Prediction (NTP), minimizing the cross-entropy loss:

$$\mathcal{L}_{pre} = -\sum_{t=1}^{T} \log P(x_t | x_{<t}; \theta) \tag{13}$$

where x is the SELFIES sequence, we used the Adam optimizer with a learning rate of **1e-6** and a batch size of **128** for **20** epochs.

Expert Fine-tuning (SFT). To create the single-property experts (e.g., θ_{QED}, θ_{Tox}), we fine-tuned the pre-trained weights on the specific high-quality subsets described in Section A.1. We employed early stopping based on the validation set performance of the specific property to prevent overfitting and catastrophic forgetting of chemical validity.

A.3 Computational Efficiency and Time Cost

All our SACMA-ES searches were conducted on a machine equipped with 2$\times$ NVIDIA RTX 2080 Ti GPUs. As highlighted in Table 6, a full 30-generation search takes approximately 42 to 45 minutes depending on the number of objectives. This contrasts sharply with traditional fine-tuning methods (including the total time for SFT and PPO), which require 3 to 40 hours of intensive training for each combination of objectives and must be completely repeated from scratch if the objective attributes change.

A.4 Theoretical Background: Nash Equilibrium in Model Merging

To provide a rigorous theoretical foundation for our proposed framework, we briefly review the concept of Nash Equilibrium and elucidate its adaptation to the multi-constraint model merging problem.

Table 6: **Time Cost Comparison.** Comparison of the computational time required to adapt the framework to new property constraints (evaluated on 2× 2080Ti GPUs).

Method	Time Cost
2 objectives	$\sim$ 42 minutes
3 objectives	$\sim$ 43 minutes
4 objectives	$\sim$ 45 minutes

A.4.1 General Definition

In game theory, a **Nash Equilibrium (NE)** is a solution concept for a non-cooperative game involving two or more players. Let there be N players, where each player i chooses a strategy s_i from their strategy set S_i. The combined strategy profile is denoted as $S = (s_1, \ldots, s_N)$. Let $u_i(S)$ be the utility function (payoff) for player i, which depends on the strategies of all players.

A strategy profile $S^* = (s_1^*, \ldots, s_N^*)$ constitutes a Nash Equilibrium if no player can unilaterally increase their utility by changing only their own strategy while the other players keep theirs unchanged. Mathematically, for every player i:

$$u_i(s_i^*, s_{-i}^*) \geq u_i(s_i, s_{-i}^*) \quad \forall s_i \in S_i, \tag{14}$$

where s_{-i}^* denotes the strategies of all players except i. In a cooperative setting (or a bargaining game), the equilibrium represents a state where the collective agreement satisfies certain axioms of fairness and efficiency (e.g., Pareto optimality).

A.4.2 Adaptation to Molecular Generation

In our CAML framework, we map these game-theoretic concepts to the model merging process as follows:

- **Players:** Each fine-tuned expert model (e.g., the QED expert, the GSK3β expert) acts as a player in the game.
- **Utility (u_i):** The goal of each expert is to minimize the distance between the final fused model parameters and its own optimal parameters. This is equivalent to maximizing the preservation of its specific domain knowledge (property constraint).
- **Conflict & Cooperation:** Since maximizing one property (e.g., high potency) might degrade another (e.g., high toxicity), the experts have conflicting interests. The merging process is thus a negotiation to find a weight configuration where no single property is severely compromised for the sake of another.
- **The Equilibrium State:** The solution derived in Eq. 8 (the weighted centroid) represents the Nash Equilibrium of this continuous game. At this point, the gradient forces pulling the fused vector towards different experts cancel out (sum to zero). This implies that the system has reached a stable compromise where the marginal gain of moving closer to one expert is exactly offset by the marginal loss of moving away from others, thereby achieving the optimal trade-off on the Pareto frontier.

A.5 SACMA-ES Optimization Process

The core of our CAML framework involves finding the optimal merging coefficients using the SACMA-ES. Unlike standard CMA-ES, which assumes an unbounded search space, SACMA-ES explicitly incorporates boundary constraints to ensure the physical validity of the fusion parameters.

Optimization Targets and Constraints. We optimize the joint parameter vector $\Theta = [k, \alpha] \in R^{2M}$, subject to the feasible region Ω:

1. **Intrinsic Scale ($k \in [0, k_{\max}]$):** The scale factors must be non-negative to preserve the gradient direction. We set $k_{\max} =$ **1.0**.
2. **Fusion Coefficient ($\alpha \in [0, \alpha_{\max}]$):** The weights must be non-negative to ensure a valid convex-like combination. We set $\alpha_{\max} =$ **10.0**.

The Optimization Loop. The process follows these steps:

1. **Initialization:** We initialize the population center $\mu^{(0)}$ with uniform weights and a unit covariance matrix $\Sigma^{(0)} = I$.
2. **Sampling:** At generation g, a population of λ candidate parameter sets (offspring) is sampled from the multivariate normal distribution:

$$z_i \sim \mathcal{N}(\mu^{(g)}, \sigma^{(g)2}\Sigma^{(g)}), \quad i = 1, \ldots, \lambda \tag{15}$$

3. **Constraint Handling (Boundary Repair):** Since the raw sampled candidates z_i may violate the boundaries (e.g., negative weights),

we apply a projection function $\mathcal{P}_\Omega$ to map them into the feasible region Ω:

$$\Theta_i = Clip(z_i, \mathbf{0}, [k_{\max}, \alpha_{\max}]) \quad (16)$$

This ensures that all evaluated parameters possess valid physical meanings.

4. **Evaluation:** Each valid candidate set Θ_i is applied to fuse the expert models. We compute the loss function by calculating the difference between the mean values of each attribute generated by the model and the mean values from the expert model.

5. **Selection and Update:** The top μ candidates are used to update the mean vector $\mu^{(g+1)}$ and the covariance matrix $\Sigma^{(g+1)}$ for the next generation, guiding the search distribution towards the optimal Nash Equilibrium within the constrained space.

Hyperparameters. We set the population size $\lambda = \mathbf{15}$, initial step size $\sigma = \mathbf{0.3}$, and ran the optimization for a maximum of **30** iterations.

A.6 Hyperparameter Sensitivity and Robustness

To assess the stability of our framework, we conducted a rigorous sensitivity analysis on the core objective and regularization hyperparameters during the most complex 4-property generation task. Based on our empirical tuning, the default configuration is set to $\alpha_i = 5$, $\lambda_c = 1$, and $\lambda_d = 1$.

As shown in Table 7, generation performance (measured by NCD, with lower values being better) improves smoothly with varying parameter values. This gradual and predictable improvement confirms the robustness of CAML and its reliance on a stable optimization environment, rather than fragile and highly specific tuning.

Table 7: **Sensitivity Analysis of Hyperparameters.** Evaluated on the 4-property task. The results demonstrate stable improvements (NCD ↓) across different hyperparameter values.

Varied Parameter	Tested Values	NCD ↓
α_i	$1, 3, 5$	$0.287, 0.250, 0.239$
λ_c	$0, 0.3, 1$	$0.263, 0.247, 0.239$
λ_d	$0, 0.5, 1$	$0.263, 0.257, 0.239$

A.7 Transfer Learning Baseline Details

As a foundational baseline for multi-constraint molecular generation, we implemented a standard transfer learning approach (often referred to as Supervised Fine-Tuning). This method relies on directly updating the parameters of a pre-trained molecular language model (e.g., ChemGPT) using a dataset that exclusively contains molecules satisfying the target constraints.

Dataset Curation. For each multi-objective task (Task 1 to Task 4), we constructed a task-specific fine-tuning dataset by filtering the global ChEMBL dataset. A molecule is included in the fine-tuning set if and only if it simultaneously satisfies all the property thresholds defined for that specific task.

Training Objective. The pre-trained backbone model is fine-tuned on these curated subsets using the standard causal language modeling objective (next-token prediction). Given a molecular sequence $x = (x_1, x_2, \ldots, x_L)$, the model is trained to minimize the negative log-likelihood:

$$\mathcal{L}_{TL} = -\sum_{t=1}^{L} \log P(x_t | x_{<t}; \theta) \quad (17)$$

where θ represents the model parameters. The training proceeds for a predefined number of epochs with early stopping based on validation loss.

A.8 Reinforcement Learning (PPO) Details

For the baseline comparisons (e.g., ChemGPT-PPO), we employed Proximal Policy Optimization (PPO). A critical step in RL for molecular generation is the **scalarization** of diverse chemical properties into a unified reward function.

Property Scalarization. Since raw property values have different scales and optimization directions (minimization vs. maximization), we normalized them into a range of $[0, 1]$:

- **QED & GSK3β:** Used directly as they are defined in $[0, 1]$ (maximization).
- **Synthetic Accessibility (SA):** The raw SA score ranges from 1 (easy) to 10 (hard). We transformed it to a maximization reward:

$$R_{SA}(x) = clip\left(\frac{10 - SA(x)}{9}, 0, 1\right) \quad (18)$$

- **Toxicity:** Modeled as a binary classifier probability $P(tox)$. The reward for safety is:

$$R_{Tox}(x) = 1 - P(tox) \quad (19)$$

Reward Aggregation. For multi-objective tasks, the scalar rewards are aggregated using a weighted sum or geometric mean. The PPO algorithm then optimizes the policy to maximize this cumulative reward while constrained by a KL-divergence penalty to prevent the model from deviating too far from the pre-trained prior.

A.9 RationaleRL Baseline Details

RationaleRL (Jin et al., 2020) is a prominent learning-based framework that formulates multi-objective molecular generation as a graph completion task. It first extracts molecular fragments (rationales) associated with desired properties and then employs reinforcement learning to extend these subgraphs into complete molecules.

Data Adaptation and Retraining. To ensure a rigorous and fair comparison, we did not rely on the pre-trained checkpoints provided by the original authors, as their training distribution and target property combinations differ significantly from our problem formulation. Instead, we completely retrained the RationaleRL framework using our own curated dataset.

We processed our dataset to extract property-specific rationales, molecular subgraphs that satisfy the distinct positive thresholds for our target properties (e.g., high GSK3β affinity and low Toxicity). The base graph generative model was then explicitly trained on these specific substructures to learn how to expand them into complete, valid molecules.

B Evaluation Metrics Details

In this section, we provide the detailed mathematical formulations for the three evaluation metrics used in our experiments.

B.1 Nash Convergence Distance (NCD)

NCD quantifies the distance to the Pareto-optimal consensus. It is defined as:

$$NCD = \sqrt{\sum_{j=1}^{M} \left(\bar{s}_j - s_j^*\right)^2} \tag{20}$$

where M is the number of target properties. $\bar{s}_j$ denotes the average normalized score of the generated batch for the j-th property, calculated as $\frac{1}{N}\sum_{i=1}^{N} Norm(x_{i,j})$. s_j^* represents the Normalized Ideal Point for the j-th property, derived from the mean performance of the corresponding single-property expert model (i.e., $s_j^* = Norm(\mu_{expert}^{(j)})$).

B.2 Weighted Average Utility (WAU)

WAU assesses the overall multi-objective performance. It is calculated as:

$$WAU = \frac{\sum_j w_j \cdot Norm(P_j)}{\sum_j w_j} \tag{21}$$

where P_j is the raw value of the j-th property for a generated molecule, w_j is the task-specific weight assigned to that property, and $Norm(\cdot)$ is a normalization function (e.g., Min-Max scaling) that maps all properties to the range $[0, 1]$.

B.3 Comprehensive Discovery Score (CDS)

CDS evaluates the balance between quality and exploration. It is computed as the weighted sum of three sub-metrics:

$$CDS = \lambda_1 \cdot SR + \lambda_2 \cdot Div + \lambda_3 \cdot Nov \tag{22}$$

The components are defined as follows:

- **Success Rate (SR):** The proportion of generated molecules that satisfy all specified constraints simultaneously.

$$SR = \frac{1}{N}\sum_{i=1}^{N} I(x_i satisfies all constraints) \tag{23}$$

- **Diversity (Div):** The internal diversity measured by the average pairwise Tanimoto distance based on Morgan fingerprints.

$$Div = 1 - \frac{2}{N(N-1)}\sum_{i=1}^{N}\sum_{j=i+1}^{N} Sim(x_i, x_j) \tag{24}$$

- **Novelty (Nov):** The fraction of valid generated molecules not present in the training set $\mathcal{D}_{train}$.

$$Nov = \frac{|\{x_i | x_i \in Generated \wedge x_i \notin \mathcal{D}_{train}\}|}{N} \tag{25}$$

In our experiments, we set equal weights $\lambda_1 = \lambda_2 = \lambda_3 = 1/3$ to prioritize quality, diversity, and novelty equally.

C Detailed Ablation Analysis

In this section, we provide a comprehensive analysis of the ablation study, extending the discussion to Multi-Objective Utility (WAU) and the Comprehensive Discovery Score (CDS).

C.1 Impact on Utility (WAU)

Beyond convergence stability, the proposed components significantly contribute to the overall quality of the generated molecules. As detailed in the experimental data:

- **Role of Adaptive Weights (α):** In Task 4, the full CAML model achieves a WAU of **0.54**. Removing the adaptive weighting (α) causes this score to drop to **0.51**, which is comparable to the static TIES baseline (**0.50**). This suggests that static averaging fails to prioritize properties effectively when gradients conflict.
- **Role of Intrinsic Importance (k):** Fixing the scale parameter k results in a WAU of **0.52**. This indicates that correctly estimating the "confidence" or scale of each expert's gradient is necessary to maximize the joint utility.

C.2 Impact on Discovery Capability (CDS)

We further evaluated the Comprehensive Discovery Score (CDS), which balances success rate with diversity and novelty.

- **Preventing Mode Collapse:** The full model achieves the highest CDS of **0.50** in Task 4. In contrast, the TIES baseline records a lower CDS of **0.45**, indicating that while it may find some valid solutions, it struggles to explore the chemical space as effectively as the Nash-optimized model.
- **Synergy of Components:** The ablation variants (*w/o* α and *w/o* k) yield lower CDS scores of **0.48** and **0.49**, respectively. This demonstrates that our game-theoretic formulation not only finds a high-performing solution but also maintains a diverse generative distribution, avoiding the mode collapse often observed in optimization-based approaches.

D Quantification of Expert Property Conflicts

To systematically analyze the trade-offs between different molecular properties, we computed a *Property Conflict Matrix*. Since the evaluated properties have different scales (e.g., QED $\in [0, 1]$, SA $\in [1, 10]$) and optimization directions (i.e., "higher-is-better" vs. "lower-is-better"), we first normalized the raw performance scores to a unified $[0, 1]$ interval.

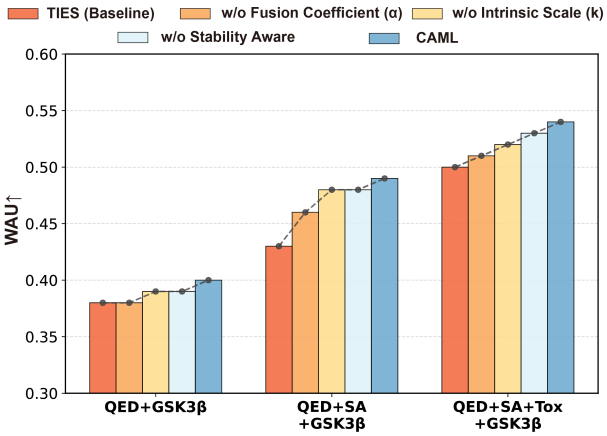

Figure 4: Ablation Study on WAU ↑.

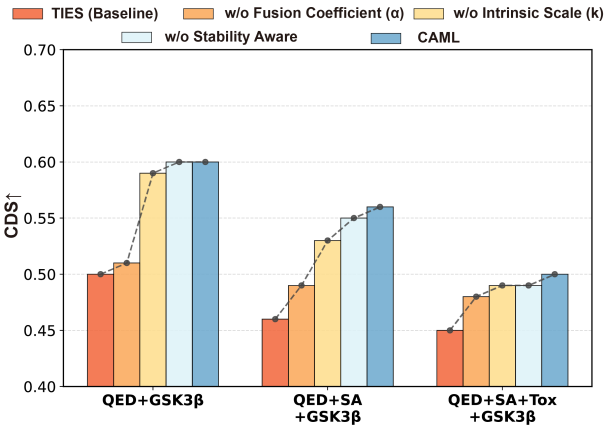

Figure 5: Ablation Study on CDS ↑.

Let $\mathcal{E} = \{E_1, \ldots, E_N\}$ denote the set of single-property expert models, and $\mathcal{P} = \{P_1, \ldots, P_N\}$ denote the corresponding set of properties. Let $R_{i,j}$ be the raw score of Expert E_i evaluated on Property P_j.

1. Min-Max Normalization We defined the normalized score $S_{i,j}$ to represent the relative performance of Expert i on Property j, where $S_{i,j} = 1$ indicates the best performance observed among all experts, and $S_{i,j} = 0$ indicates the worst.

- For properties where **higher is better** (e.g., QED, GSK3β):

$$S_{i,j} = \frac{R_{i,j} - \min_k(R_{k,j})}{\max_k(R_{k,j}) - \min_k(R_{k,j})} \quad (26)$$

- For properties where **lower is better** (e.g., SA, Toxicity):

$$S_{i,j} = \frac{\max_k(R_{k,j}) - R_{i,j}}{\max_k(R_{k,j}) - \min_k(R_{k,j})} \quad (27)$$

2. Conflict Calculation The conflict value $C_{i,j}$ quantifies the **performance degradation** of Property j when using Expert i, relative to the optimal performance achieved by the group. It is defined as:

$$C_{i,j} = 1 - S_{i,j} \tag{28}$$

A value of $C_{i,j} \approx 0$ indicates minimal conflict (Expert i is compatible with Property j), whereas $C_{i,j} \approx 1$ indicates severe conflict (Expert i significantly degrades Property j).