CRYSTALREASONER: REASONING AND RL FOR PROPERTY-CONDITIONED CRYSTAL STRUCTURE GENERATION

Anonymous authors
Paper under double-blind review

ABSTRACT

Generative modeling has emerged as a promising approach for crystal structure discovery. However, existing LLM-based generative models struggle with low-level atomic precision, while diffusion-based methods fall short in integrating high-level scientific knowledge. As a result, generated structures are often invalid, unstable, or do not possess desirable properties. To address this gap, we propose CrystalReasoner (CrysReas), an end-to-end LLM framework that generates crystal structures from natural language instructions through reasoning and alignment. Before emitting atomic coordinates, CrysReas writes a thinking trace that states the crystallographic symmetry, the local coordination environments, and the physical properties of the target structure, so that the textual knowledge a language model already carries is turned into constraints on the coordinates it is about to write. CrysReas then employs reinforcement learning (RL) with a multi-objective, dense reward function to align generation with downstream desired properties such as physical validity, chemical consistency, and thermodynamic stability. For property-conditioned tasks, we design task-specific reward functions and train specialized models for discrete constraints (e.g., space group) and continuous properties (e.g., elasticity, thermal expansion). On the MP-20 benchmark, CrysReas raises the share of generated structures that are stable, unique, and novel from 2.5% to 3.7% over the strongest prior language-model baseline re-implemented in the same pipeline, and each of the three property specialists improves adherence to its target constraint over the unconditioned model. Our work demonstrates the potential of leveraging thinking traces and RL for generating valid, stable, and property-conditioned crystal structures[1].

1 INTRODUCTION

Modern technologies increasingly rely on the development of new materials, such as solid-state electrolytes for batteries (Zhao et al., 2020), high-performance catalysts (Goldsmith et al., 2018), and functional semiconductors (Davies et al., 2018). Traditional computational methods for crystal structure discovery such as random search (Pickard & Needs, 2011) and particle swarm optimization (Wang et al., 2010) are computationally intensive due to explicit energy evaluation in each search iteration. In contrast, generative models offer a scalable alternative by bypassing the costly search and energy evaluation steps (De Breuck et al., 2025) to propose reasonable initial structures.

Despite the progress in generative modeling, existing generative models for crystal structures are limited. For example, diffusion-based models (Yang et al., 2023; Xie et al., 2021; Chen et al., 2025; Jiao et al., 2023; 2024; Kelvinius et al., 2025; Joshi et al., 2025) operate in the 3D structure or latent space, and can not easily integrate rich textual knowledge (e.g., compositions, properties from textbooks). To incorporate scientific knowledge, some works (Yang et al., 2024c; Inizan et al., 2025; Khastagir et al., 2025) use LLMs to generate formulas followed by diffusion for structures conditioned on chemical formulas. However, these decoupled architectures separate semantic reasoning

[1]The entire pipeline trains in fewer than 40 GPU hours on two A100s. Code is available at: `https://anonymous.4open.science/r/CrystalReasoner/`

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.

and structural generation into distinct modules, preventing end-to-end training and joint optimization.

On the other hand, fine-tuning LLMs to directly generate Crystallographic Information Files (CIFs) holds great promise for integrating scientific knowledge, as the pretraining corpus of models already describes crystals in this vocabulary (Antunes et al., 2024; Gruver et al., 2024). However, recent attempts (Antunes et al., 2024; Gruver et al., 2024; Mohanty et al., 2026; Gan et al., 2025; Xu et al., 2025; Hong et al., 2025; Cao et al., 2025; Cao & Wang, 2026) face a critical challenge: the LLM tokenizer flattens 3D coordinates into strings, losing symmetry and spatial constraints, thereby leading to inconsistency between the symmetry of the generated structure and user requirements. Furthermore, LLM based approaches generally suffer from a lack of precision in generated atom locations, and they lack mechanisms to enforce physical validity, stability, and property conditioning in the generated structures.

To address this gap, we draw insights from the development of LLMs around reasoning and RL alignment with verifiable feedback. We propose CrystalReasoner (CrysReas), an end-to-end framework that converts textual instructions into high-fidelity crystal structures through reasoning and alignment. Specifically, we curate thinking traces based on the physical knowledge of a structure, following the logical sequence of symmetry, local coordination environments, formation energy, and other properties. The hope is that by introducing textual knowledge of the 3D structure in the thinking trace, LLMs can learn to reason about the 3D structure before generating the structure itself, making structure generation more tractable.

To further improve precision of the generated atom locations, we apply online RL with a carefully designed multi-objective dense reward function covering physical validity, chemical validity, and thermodynamic stability, guiding generation toward valid, low-energy configurations. To enable property-conditioned generation, CrysReas employs RL with property-specific rewards calculated using surrogate Machine Learning Interatomic Potentials (MLIPs) (Yang et al., 2024b). By combining stability rewards with property-specific objectives, CrysReas can be specialized for diverse material design scenarios without architectural modifications.

Against CrystalTextLLM and the Wyckoff representation of PLAID++, both re-implemented in the same pipeline with the same data, backbone, and hyperparameters, CrysReas produces the largest share of structures that are simultaneously stable, unique and novel. Thinking traces and RL turn out to address different failure modes: the traces enhance instruction following, and RL optimizes energy, with a Density Functional Theory (DFT) recomputation (Hohenberg & Kohn, 1964; Kresse & Furthmüller, 1996) confirming the energy ordering that the surrogate potential gives. The same RL machinery extends to property-conditioned generation, at a small cost in general structural validity.

2 Preliminaries

In this section, we define notations and provide background on LLMs for crystal structure generation and RL for LLMs.

2.1 LLMs for Crystal Structure Generation

Prior work (Gruver et al., 2024; Antunes et al., 2024) casts crystal structure generation as token-sequence generation with an LLM π_θ. Given a natural language description c that states target physical properties of the structure (e.g., formula, space group), the LLM autoregressively emits a token sequence $a_{0:N}$ encoding lattice parameters and atomic coordinates. Generated structures are conventionally scored along four axes: structural validity (geometric constraints) and chemical validity (oxidation states consistent with electroneutrality), both from symbolic checkers (Xie et al., 2021; Davies et al., 2019); composition consistency, whether the reduced formula matches the request; and thermodynamic stability, the distance to the convex hull, usually estimated with an MLIP.

2.2 RL for Language Models

RL is an effective technique for refining LLMs, ensuring that the models are specifically optimized for targeted objectives, human preferences (Ouyang et al., 2022), or verifiable rewards (Shao et al.,

2024). Among different RL algorithms, Group Relative Policy Optimization (GRPO) (Shao et al., 2024) demonstrates significant utility in addressing domains that necessitate long thinking traces, most notably in the context of mathematical reasoning. GRPO is a policy gradient RL algorithm for LLMs that eliminates the need for a critic network by comparing multiple outputs sampled from the same input. For each input c, it samples G candidate outputs $\{a_i\}$, each receiving a reward R_i, and optimizes a clipped objective with KL regularization to a reference policy:

$$\mathcal{J}(\theta) = \mathbb{E}_{c\sim\mathcal{D},\{a_i\}\sim\pi_\theta}\left[\frac{1}{G}\sum_{i=1}^{G}\left(\mathcal{L}_{\text{clip},i}(\theta) - \beta\mathbb{D}_{\text{KL}}(\pi_\theta(\cdot|c)||\pi_{\text{ref}}(\cdot|c))\right)\right]$$

where $\mathcal{L}_{\text{clip},i}(\theta)$ is the standard PPO-style clipped surrogate objective (Schulman et al., 2017) adapted with GRPO's group-relative advantage, using the normalized rewards $\tilde{R}_i = \frac{R_i - \text{mean}(R)}{\text{std}(R)}$ within the group of G samples.

3 Method

We first state the generation problem and the conditioning protocol (Section 3.1). Section 3.2 then describes how we generate ground-truth thinking traces to enable the model to think, and Section 3.3 describes the online RL stage that optimizes the trace and the structure jointly against a dense, verifiable reward.

3.1 Problem Setting

A crystal is a periodic arrangement $C = \big(L, \{(z_i, x_i)\}_{i=1}^{N}\big)$, where $L \in \mathbb{R}^{3\times3}$ collects the three lattice vectors, z_i is the atomic number of the i-th atom of the unit cell and $x_i \in [0,1)^3$ is its fractional coordinate. A generation request is a textual specification c, built from a random subset of a pool of optional physical constraints

$$\mathcal{P} = \big\{\text{composition},\ \text{space group},\ E_{\text{hull}},\ E_f,\ E_g,\ (K_{\text{VRH}}, G_{\text{VRH}}),\ \alpha_V\big\}.$$

Here E_{hull} is the energy above the convex hull, E_f the formation energy per atom, E_g the band gap, K_{VRH} and G_{VRH} the Voigt–Reuss–Hill bulk and shear moduli, and α_V the volumetric coefficient of thermal expansion at 300 K. Only the chemical formula is always present; the remaining conditions are drawn independently per example. The task is to learn a policy $\pi_\theta(a \mid c)$ over token strings such that the decoded structure $C(a)$ is physically valid, consistent with every constraint present in c, and thermodynamically stable, i.e., at or below the convex hull of the known phases.

3.2 Thinking by Progressively Shrinking the Crystallographic Solution Space

A structure with N atoms in the unit cell has $3N + 6$ continuous degrees of freedom. However, a crystal is specified far more compactly by its space group G, by an assignment of the composition to the Wyckoff orbits of G, and by the free parameters of the asymmetric unit; the remaining atoms then follow deterministically from the symmetry operations of G (see Appendix C for an example). There are also implicit constraints from the physical requirements that a stable structure has to satisfy. Resolving all of these in a single pass is hard for a transformer-based autoregressive model that emits lattice parameters and fractional coordinates directly, because it has no mechanism for imposing the discrete symmetry that makes a solid a crystal.

A similar difficulty arises when a model works on math problems or on general reasoning, and people usually let the model think before it answers (Wei et al., 2022), requiring a large and diverse corpus of human-written reasoning traces. Training on such traces not only strengthens reasoning in natural-language space, but also lets the model reuse argumentative patterns it has already seen during pre-training.

For crystallography, however, no corpus of human-written reasoning exists. We therefore *synthesize* the supervision: a deterministic crystallographic program reads the ground-truth structure and emits the reasoning by stating the facts of the structure in a proper order. As illustrated in Figure 1, the trace proceeds through three segments: (i) symmetry information, including space group, Wyckoff letters

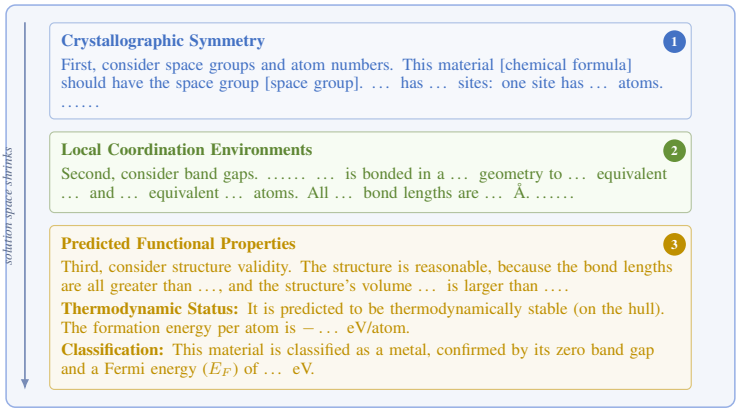


Figure 1: LLMs are required to generate thinking tokens before outputting atomic coordinates. The first part encodes abstract physical knowledge (e.g., formula, space group). The second characterizes local coordination environments (e.g., bond length distribution). The last part reasons about physical properties (e.g., structural volume, stability, and electronic properties). Finally, the model outputs the crystal structure in a simplified Crystallographic Information File similar to that in CrystalTextLLM (Gruver et al., 2024). See details in Appendix B, and see Appendix C for an example.

and multiplicities, oxidation states, and the electroneutrality balance over occupied sites; (ii) local coordination environment, including coordination polyhedra, corner/edge/face connectivity, polyhedral tilt angles, and bond-length distributions; (iii) cell-level and functional properties, including cell volume, formation energy, E_{hull}, and band gap. This mirrors how a crystallographer would proceed: fix the symmetry and the orbit structure first, then settle local details such as site multiplicities, coordination, and bond lengths, and finally check the result by evaluating macroscopic quantities. The procedure for generating such traces, together with worked examples, is given in Appendix B and Appendix D.

Such thinking traces are physically correct by construction, and the whole corpus is reproducible from the Materials Project record with a fixed seed. Although the traces are templated and expose the model to a fixed reasoning schema, progressively narrowing the solution space improves both the validity and the stability of the generated structures (Table 1), and Section 4.2 isolates which segment of the trace is responsible for which gain.

3.3 Online RL with Verifiable Feedback

Although thinking traces help LLMs to provide physical reasoning and improve validity, as stochastic models, LLMs still suffer from numerical imprecision, requiring final alignment for precise atom location generation. Moreover, thinking traces only provide supplementary reasoning information, but do not guarantee that generated structures conform to the specifications in the reasoning trace.

RL for Jointly Optimizing Thinking Trace and Crystal Structure. As discussed in Section 2.2, GRPO is capable of optimizing both the output and long reasoning traces by a scalar reward per output. We therefore apply it to crystal structure generation, which allows us to jointly optimize the thinking tokens introduced in Section 3.2 based solely on the final reward of the generated structure.

Dense, Smooth, and Verifiable Rewards. It is desirable to align structure generation so that generated structures are both valid and stable. To achieve this, we score a response with the multi-objective reward

$$R_{\text{target}} = \mathbf{1}_{\text{format}}\Big[\alpha_{\text{validity}} R_{\text{validity}} + \alpha_{\text{stability}} \mathbf{1}_{\text{validity}} R_{\text{stability}}\Big], \quad R_{\text{validity}} = \sum_{k\in\{p,s,c,i\}} w_k R_k, \tag{1}$$

whose terms are listed in Table 5. $R_{\text{stability}}$ quantifies energetic favorability via energy above the hull (E_{hull}), which is calculated by an MLIP, and contributes only when $\mathbf{1}_{\text{validity}} = 1$, i.e., basic validity holds. We set $\alpha_{\text{validity}} = 1$ and $\alpha_{\text{stability}} = 10$: the validity terms are binary and saturate early in training, whereas $R_{\text{stability}}$ is continuous and keeps supplying gradient after they have saturated

Table 1: Two prior LLM-based representations and the four CrysReas variants, all re-implemented in the CrysReas pipeline with the same dataset, base checkpoint, hyperparameters, and numerical precision (8 decimal places), so the comparison is precision-matched. `PLAID++ Wyckoff` uses only the representation of PLAID++, without its direct-preference-optimization (DPO) stage; `Base` is SFT only, `Thinking` is SFT with thinking traces, `RL` is RL on the base model, and `Full` is CrysReas with both. Every column comes from the same 10,240 generated structures, and the metrics are the ones defined in Section 4.1. All entries are percentages except Energy, which is in eV/atom; $\pm$ gives the standard error over the 10 seeds. CrysReas leads on every metric but chemical validity.

	Prior representations		CrysReas variants			
Metric	**CrystalTextLLM**	**PLAID++ Wyckoff**	**Base**	**Thinking**	**RL**	**Full**
Structural Validity (↑)	87.68 ± 0.22	87.24 ± 0.33	83.76 ± 0.45	91.95 ± 0.13	91.00 ± 0.20	**95.19** ± 0.16
Chemical Validity (↑)	**91.53** ± 0.08	91.37 ± 0.07	88.11 ± 0.13	89.92 ± 0.06	89.10 ± 0.14	90.12 ± 0.03
Composition Consistency (↑)	93.76 ± 0.20	96.23 ± 0.09	90.06 ± 0.23	97.03 ± 0.11	92.76 ± 0.23	**97.43** ± 0.09
Space Group Consistency (↑)	33.52 ± 0.27	51.66 ± 0.32	52.05 ± 0.37	56.36 ± 0.35	54.38 ± 0.35	**59.07** ± 0.17
Stability (↑)	1.76 ± 0.04	2.78 ± 0.06	3.05 ± 0.08	3.80 ± 0.09	3.15 ± 0.08	**4.17** ± 0.06
Energy (eV/atom) (↓)	0.623 ± 0.006	0.540 ± 0.007	0.497 ± 0.005	0.484 ± 0.003	0.511 ± 0.005	**0.446** ± 0.003
Uniqueness (↑)	80.53 ± 0.20	80.27 ± 0.29	75.21 ± 0.40	82.68 ± 0.15	81.53 ± 0.22	**85.52** ± 0.16
S.U.N. (↑)	1.57 ± 0.04	2.53 ± 0.05	2.73 ± 0.09	3.33 ± 0.07	2.87 ± 0.09	**3.68** ± 0.06
S.S.U.N. (↑)	15.90 ± 0.16	24.27 ± 0.19	25.47 ± 0.20	28.28 ± 0.31	26.98 ± 0.21	**30.54** ± 0.15

(Appendix E). Such a reward term is computable, smooth, and bounded, which is what lets us run online RL on the distribution the policy currently produces, with no offline stage over a precomputed candidate pool. Different weight settings are explored in experiments, and additional details (e.g., the exact formulation) are in Appendix E.

RL for Property-Conditioned Crystal Structure Generation. Beyond validity and stability, it is crucial to support property-conditioned generation for target-driven material design. We support requests for a crystal structure having a specific space group, bulk modulus and shear modulus in specific ranges, or thermal expansion in a specific range, while preserving the stability of the structure. To achieve this, the final reward is a combination of stability rewards and property related rewards. We use range-constraint reward $R_{\text{property}}(P_{\text{generated}}, P_{\text{specified}} = [L, R])$, which encourages $P_{\text{generated}}$ to be in the desired range $P_{\text{specified}}$, to construct property related rewards. More details of reward functions are in Appendix E.

4 Experiments

In this section, we systematically evaluate CrysReas on the task of generating valid, stable, and property-compliant crystal structures from natural language instructions. First, we evaluate the success of end-to-end generation in Section 4.1. We then investigate the effect of individual components of CrysReas, including thinking traces (Section 4.2) and RL optimization (Section 4.3). We finally evaluate the success of property-conditioned generation in Section 4.4.

4.1 End-to-End Evaluation of Validity, Instruction Following, and Stability

Baselines and Setups. We aim to evaluate CrysReas's ability to generate unique, valid, stable structures under textual specifications. For fairness, we only compare our work with LLM-based prior works, including CrystalTextLLM (Gruver et al., 2024) and the Wyckoff-based model of PLAID++ (Xu et al., 2025). They are re-implemented inside our pipeline, sharing the dataset, the initialization (a Qwen2.5-3B (Qwen et al., 2024) model fine-tuned by MegaScience (Fan et al., 2025)), the optimization hyperparameters, and the numerical precision (8 decimal places) with our model. We also use our model variants, `CrysReas-Base` (SFT only), `CrysReas-Thinking` (SFT with thinking traces), and `CrysReas-RL` (RL on base model) as ablation baselines. See additional details of baselines in Appendix A. We compare the models on the following metrics. (i) Structural and chemical validity, following the definition of prior works (Xie et al., 2021). (ii) Instruction following for composition, space group, elasticity, and thermal expansion. A structure counts as space group consistent when the space group detected in it carries at least the requested symmetry. (iii) Stability and the hull distance behind it, computed with MatterSim (Yang et al., 2024b) against MatterGen's `ReferenceMP2020Correction` phase diagram (Zeni et al., 2023),

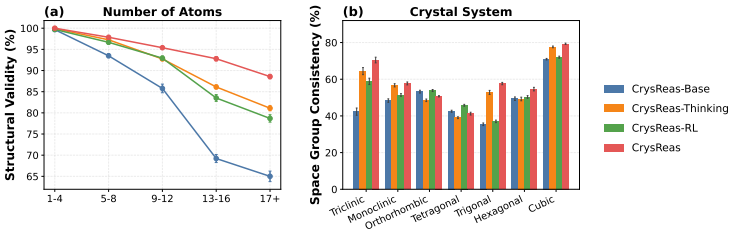


Figure 2: The four variants of Table 1 across two kinds of complexity. (a) Structural validity against the number of atoms of the requested material, in five bins that hold between 165 and 267 compositions each. (b) Space group consistency against the crystal system of the requested space group; the 1,024 compositions cover 111 distinct space groups, and every system carries at least 43 of them. Crystal systems are ordered by the highest point-group order each admits, from triclinic to cubic.

which combines Materials Project and Alexandria entries; following Xu et al. (2025), a structure is stable when its hull distance is at most 0 eV/atom, that is, when the structure lies on or below the hull. (iv) Uniqueness, the share of the 1,024 structures generated under a single seed that are not duplicates of an earlier structure of that same seed. This is different from prior works as they often evaluate unconditional generation, while our generation are property-conditioned. (v) S.U.N., the share that are stable, unique, and novel, where novel means absent from the training split; S.S.U.N. adds space group consistency to those three. Following Xu et al. (2025), S.U.N. requires $E_{\text{hull}} \leq 0$ eV/atom, whereas S.S.U.N. requires $E_{\text{hull}} \leq 0.1$ eV/atom, the metastability window established by Sun et al. (2016) and adopted by MatterGen (Zeni et al., 2023). Every number in this section comes from 10,240 structures: the 1,024 held-out compositions of the generation split, each generated once under each of 10 seeds. Every seed is a complete evaluation on its own, and the error bars are standard errors over the 10 per-seed values, so they report run-to-run variation. Appendix E gives the matchers, tolerances, and thresholds behind each metric.

Comparison Against Prior Works. We compare our model against our implementations of CrystalTextLLM (Gruver et al., 2024) and the Wyckoff Base model of PLAID++ (Xu et al., 2025) in Table 1. CrystalTextLLM has the lowest space group consistency and the lowest S.U.N., because it emits a raw CIF file and gets no geometric constraint as supervision. PLAID++'s Wyckoff representation encodes symmetry explicitly, and without thinking traces or RL it achieves higher structural validity and composition consistency than our representation. Once thinking traces and RL are added, CrysReas performs best on every metric but chemical validity.

Comparison for Model Variants. We compare four variants: baseline `CrysReas-Base` (SFT only), `CrysReas-Thinking` (SFT with thinking traces), `CrysReas-RL` (RL on base model), and full `CrysReas` (both). As shown in Table 1, the full model performs best on every metric. Comparing `CrysReas-Thinking` with `CrysReas-RL`, the traces contribute more to instruction following and to stability, while RL contributes most to hull distance.

4.2 Evaluating the Effect of Thinking Traces

We now perform ablations and qualitative analysis to better understand the effect of the thinking traces.

Varying Difficulty of Tasks. By evaluating scores for questions with different difficulties (different number of atoms, or different space group requirement), we can understand in what situations thinking traces and RL perform better, and conduct a more detailed comparison than Table 1. Figure 2(a) plots structural validity against cell size. CrysReas performs best, while `CrysReas-Thinking` and `CrysReas-RL` both outperform `CrysReas-Base`, and achieve similar results.

Figure 2(b) breaks consistency down by crystal system, with the systems ordered by the highest point-group order each of them admits. The model performs best at the two ends of that axis and worst in between. Where the requested symmetry is strong, the group itself fixes the cell metric and

places most atoms on special positions, so the model supplies only a handful of free coordinates and has correspondingly few chances to violate the request. Where it is weak, there is little to violate, and a chemically sensible cell satisfies the instruction on its own. Space groups in the middle of the axis combine real constraints with real freedom, so the model has to construct the symmetry rather than inherit it, and that is where consistency is lowest. Thinking traces help most in this middle region, but cost a few points on orthorhombic and tetragonal requests, where a model trained without traces more often over-symmetrizes the cell.

Comparing Importance of Segments in Thinking Traces. To find out which part of the trace the emitted coordinates depend on, we replace one segment at a time with the same-named segment of another randomly drawn sample, so the instruction, the format, and the length stay as they were and only the segment now describes a different material. According to Table 2, the first segment, which restates the formula and the symmetry, is the most important to formula consistency and structure validity. The second segment, which is related to local coordination, contributes to structure validity and space group consistency. This indicates that the final crystal generation draws upon every segment of the thinking traces across various aspects.

Physical Properties Are Predicted Before Generation. Appendix F compares the site counts, cell volume, and bond lengths asserted in the trace with those of the structure the model subsequently writes. The two agree closely in high-symmetry space groups and drift apart as symmetry is lowered, so the trace acts as a prior on the structure rather than as a specification the decoder always meets.

4.3 Evaluating the Effect of RL Optimization

Ablation Study on Reward Components. To understand the functionality of each component (validity and stability) of the total reward and find the best reward configuration, we compare three reward configurations (Table 3): Validity Only model ($\alpha_{\text{validity}} = 1, \alpha_{\text{stability}} = 0$), Energy Only model ($\alpha_{\text{validity}} = 0, \alpha_{\text{stability}} = 10$), and Mixed Reward model ($\alpha_{\text{validity}} = 1, \alpha_{\text{stability}} = 10$). Each configuration performs best on the metric it optimizes: Validity Only on structural validity and composition consistency, and Mixed Reward on hull distance. Validity Only also keeps the most diverse rollouts, indicating that the energy term narrows model generation. When comparing S.U.N., there is no significant difference between these configurations.

DFT Verification for Energy. Every stability number above comes from a surrogate potential, so we recompute E_{hull} with DFT for the four variants on candidates drawn from 128 held-out queries. As illustrated in the E_{hull} distributions in Figure 3(a), both RL alignment and thinking traces shift the energy distribution toward lower values, and the DFT ordering of the four variants agrees with the MatterSim ordering of Table 1.

To provide a more granular comparison, we present parity plots for Figure 3(b) `CrysReas-Base` vs. `CrysReas-Thinking` and Figure 3(c) `CrysReas-Base` vs. `CrysReas-RL`. The scatter distribution reveals that the majority of data points lie below the diagonal line $y = x$, demonstrating that in most cases both `CrysReas-Thinking` and `CrysReas-RL` achieve lower E_{hull} values compared to the `CrysReas-Base` baseline. These findings suggest that incorporating reasoning

Table 2: Ablation on the three segments of the thinking trace. Each segment is replaced by the same-named segment of another randomly drawn sample, so format, style, and length distribution stay intact and only the information the segment carries about this material is destroyed. Four kinds of replacements are listed in the table, with four kinds of metrics reported. We run this on 256 prompts the model originally answered correctly on all four metrics, with four samples per condition.

Part to replace	**Structural** (↑)	**Chemical** (↑)	**Composition** (↑)	**Space Group** (↑)
Full trace	94.63 ± 0.88	100.00 ± 0.00	99.41 ± 0.34	83.30 ± 1.67
Part 1: symmetry, formula	76.95 ± 2.06 (-17.7)	96.68 ± 0.93 (-3.3)	0.39 ± 0.39 (-99.0)	25.39 ± 2.39 (-57.9)
Part 2: coordination	84.67 ± 1.75 (-10.0)	99.12 ± 0.51 (-0.9)	96.48 ± 1.02 (-2.9)	35.25 ± 2.56 (-48.0)
Part 3: validity check	93.75 ± 1.01 (-0.9)	100.00 ± 0.00 (+0.0)	99.22 ± 0.46 (-0.2)	80.08 ± 1.84 (-3.2)
All three parts swapped	79.98 ± 1.80 (-14.6)	97.27 ± 0.94 (-2.7)	0.00 ± 0.00 (-99.4)	23.54 ± 2.37 (-59.8)

Table 3: Ablation of reward designs: Validity Only ($\alpha_{\text{validity}} = 1, \alpha_{\text{stability}} = 0$), Energy Only ($\alpha_{\text{validity}} = 0, \alpha_{\text{stability}} = 10$) and Mixed Reward ($\alpha_{\text{validity}} = 1, \alpha_{\text{stability}} = 10$, the full CrysReas). Each configuration performs best on the metric it optimizes. (a) The validity terms control the metrics they name directly. (b) The two configurations that carry the energy term are close everywhere, indicating that the energy term is what moves the discovery metrics. All entries are percentages except Energy, which is in eV/atom.

(a) Validity and instruction following

Metric	Validity	Energy	Mixed
Structural (↑)	**96.10** ± 0.19	95.87 ± 0.16	95.19 ± 0.16
Chemical (↑)	**90.13** ± 0.03	90.07 ± 0.03	90.12 ± 0.03
Composition (↑)	**97.73** ± 0.05	97.60 ± 0.05	97.43 ± 0.09
Space Group (↑)	56.53 ± 0.32	**60.27** ± 0.15	59.07 ± 0.17

(b) Stability and discovery

Metric	Validity	Energy	Mixed
Stability (↑)	4.12 ± 0.06	**4.20** ± 0.09	4.17 ± 0.06
Uniqueness (↑)	**86.48** ± 0.13	86.09 ± 0.18	85.52 ± 0.16
Energy (↓)	0.492 ± 0.003	0.449 ± 0.005	**0.446** ± 0.003
S.U.N. (↑)	3.66 ± 0.07	3.65 ± 0.08	**3.68** ± 0.06
S.S.U.N. (↑)	28.79 ± 0.24	**30.56** ± 0.12	30.54 ± 0.15

traces and policy optimization effectively guides the model toward more thermodynamically stable crystal structures.

4.4 Evaluating Property-Conditioned Generation

Property-Conditioned Generation. For conditioned generation tasks, we use `CrysReas` (with both thinking and RL) as the baseline and investigate three specialized models through property-conditioned RL: `CrysReas-SpaceGroup`, `CrysReas-ElasticProperties`, and `CrysReas-ThermalExpansion` (Figure 4). All three specialists achieve improvements on their respective conditioning targets, demonstrating that RL with specialized rewards enhances adherence to specific property constraints. However, specialization comes with trade-offs. For example, while `CrysReas-ElasticProperties` outperforms the baseline on follow-elasticity rate, it achieves slightly lower structural validity (Table 9). Since the two continuous specialists drop the validity block entirely (Appendix E), optimizing for specific property requirements may modestly impact general structural quality.

5 Related Work

Purely Diffusion-Based Crystal Generation. Diffusion models have been successfully applied to crystal structure generation, learning to reverse a noise process on atomic coordinates and lattice parameters (Yang et al., 2023; Xie et al., 2021; Chen et al., 2025; Jiao et al., 2023; 2024; Kelvinius et al., 2025; Joshi et al., 2025). These methods achieve strong performance on structural validity and serve as standard baselines. However, diffusion models operate solely on structural representations and do not explicitly integrate text-based knowledge that connects to higher-level concepts such as chemical compositions or materials semantics.

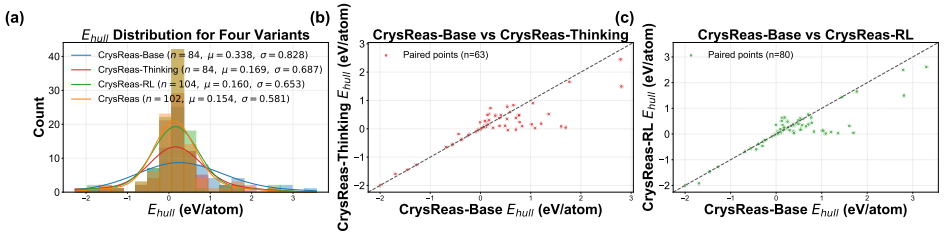


Figure 3: DFT recomputation of the energy above the hull. Candidates generated by `CrysReas-Base`, `CrysReas-Thinking`, `CrysReas-RL` and `CrysReas` for 128 held-out queries are relaxed with DFT and their hull distance E_{hull} recomputed from first principles; the legend gives the number of converged structures n, the mean μ and the standard deviation σ per variant. (a) Distribution of E_{hull} per variant: thinking traces and RL each shift mass toward the hull, and the ordering of the four variants matches the MatterSim ordering of Table 1. (b), (c) Per-query comparison of `CrysReas-Base` against `CrysReas-Thinking` and against `CrysReas-RL`; points below the diagonal are queries on which the added ingredient produced the lower-energy structure.

LLM-Related Crystal Generation. Some approaches adopt decoupled architectures where LLMs generate formulas and separate diffusion models predict structures from those formulas (Yang et al., 2024c; Inizan et al., 2025; Khastagir et al., 2025). This design enables textual priors but splits reasoning and generation into independent modules, making joint optimization infeasible. More recent end-to-end approaches use a single LLM to directly output crystal structures as flattened coordinate strings (Antunes et al., 2024; Gruver et al., 2024; Mohanty et al., 2026; Gan et al., 2025; Xu et al., 2025). However, flattening 3D coordinates disrupts crystallographic symmetries and spatial constraints, failing to comply with the space group specified in the instructions.

Reasoning Traces and RL with Verifiable Rewards. Reasoning traces improve LLM performance on structured tasks (Wei et al., 2022; Sprague et al., 2024; Yang et al., 2024a). Building on this, RL with verifiable rewards (RLVR) optimizes such models against programmatically checkable signals rather than learned preference models (Shao et al., 2024; Guo et al., 2025; Lambert et al., 2024), and has become the standard recipe for eliciting reasoning at scale. Recent works for new material discovery also apply RL to crystal generation with symmetry or stability rewards (Cao & Wang, 2026; Chen et al., 2025) and preference optimization over precomputed candidates (Xu et al., 2025). These works do not enable models to reason and do not support property-conditioned generation.

6 Conclusion

We proposed CrysReas, an end-to-end framework in which a language model states the symmetry, coordination, and properties of a crystal before writing its coordinates, and is then aligned by RL against a dense, verifiable reward. With both thinking traces and RL, the model is better on every metric we measure than either ingredient alone or than two prior works re-implemented in the same pipeline.

Limitations and Future Works. Despite these advances, our framework has several limitations that point to promising future directions. First, due to computational constraints, all models, including re-implemented prior works, are evaluated using the Qwen2.5-3B architecture, limiting direct comparison with original reported results from prior works; a more comprehensive comparison could be achieved with additional prior works and parameter tuning. Second, our framework requires training a specialized model for each property-conditioning task rather than covering all conditions in a single model; developing a multi-task or adapter-based framework may reduce training overhead for multi-task scenarios. Third, all experiments are conducted solely on the CDVAE MP-20 split, leaving generalization to other material families unvalidated; evaluating CrysReas on broader datasets with diverse chemical compositions may better assess its generalization capability. Fourth, both the stability reward and the property rewards are computed from a single machine-learned potential, enabling the model to potentially engage in reward hacking; rewarding against multiple potentials would provide a stronger safeguard.

AI use statement

In this work, we used generative AI tools (LLM-based coding agents) to implement methods, to provide feedback on our experimental methodology, and to interpret results. We have not used genera-

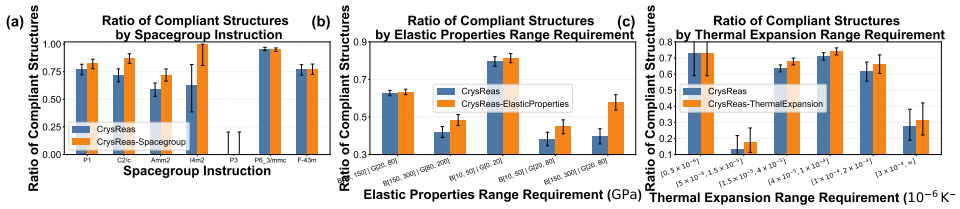


Figure 4: Performance of specialized models on three conditioning tasks: space group (left), elasticity (middle), and thermal expansion (right). Each specialist improves over the baseline `CrysReas` on its target metric, confirming that reward-shaped RL effectively enforces discrete or continuous property constraints.

tive AI tools to develop the conceptual framework of this work, to propose or refine its hypotheses, to generate synthetic data sets, to clean or reformat data, or to assist with translation; formulating mathematical claims, supplying ingredients for proofs, writing proofs, and qualitative or thematic data analysis are not applicable to this work. Additionally, we used generative AI tools to create and edit software code, to create and modify the figures, to suggest experimental parameters, to search for and summarize related literature, to format references, to suggest the structure of the manuscript, to draft parts of it, and to edit it for readability.

We have reviewed all AI-assisted work. The research question, the conceptual design of the method, and the design of every experiment reported here are the authors' own; AI assistance was used to realize them. In particular, the synthetic thinking-trace corpus contains no language-model-generated content: every statement in a trace is emitted by a deterministic crystallographic program from the ground-truth Materials Project record, and the dataset construction and cleaning steps are likewise deterministic. All AI-written code was read and tested by the authors, and every number in the paper was produced by the authors running the final pipeline end to end. AI-generated critiques of the manuscript and AI-drafted interpretations of the results were accepted only after the authors verified them against the underlying data. We take responsibility for the final content of this work, including its text, claims, and artifacts produced with the aid of generative AI.

Separately, and as the subject rather than a tool of this research, the method itself fine-tunes a pre-trained language model (Qwen2.5-3B). This is not covered by the disclosure above.

Ethics Statement

This work involves no human subjects and no personally identifiable information; all data are records of inorganic crystals drawn from the Materials Project (Jain et al., 2013) through the CDVAE MP-20 split (Xie et al., 2021), and we release our code and generated structures under CC BY 4.0, the same licence under which the upstream structures are distributed. The chief ethical consideration is one of scientific hygiene: every stability and property signal the model is optimized against comes from a machine-learned interatomic potential rather than from first principles, so a structure that CrysReas reports as stable is a hypothesis and not a verified material, and the DFT recomputation of Section 4.3 is included to keep that distinction visible. Depositing generated structures into a public materials database without independent verification would therefore be a misuse of this work. We judge the dual-use risk to be low and not specific to our contribution, since the model is conditioned on bulk inorganic crystals of at most twenty atoms per cell, emits a structure and never a synthesis route, and confers no capability beyond that of the public structure-prediction codes we compare against.

Reproducibility Statement

Everything needed to reproduce the results of this paper is either in the appendix or in the anonymous code release linked from the abstract. Appendix A specifies the data (the upstream CDVAE MP-20 split, the four derived splits by file name and size, and how the instructions are assembled from them), the models (the shared base checkpoint, the full SFT and GRPO hyperparameter settings, and the two prior-work representations re-implemented against those same settings), the evaluation protocol (which held-out compositions are generated, under how many seeds, what the reported error bars are taken over, and the sampling temperature and top-p), and the surrogate-potential configuration (the relaxation schedule, the `ReferenceMP2020Correction` phase diagram, and the elastic and quasi-harmonic workflows behind the conditioning targets). All training runs on two A100 GPUs in under 40 GPU hours in total, so reproducing the experiments does not require large-scale infrastructure.

The supervision is deterministic and therefore reproducible exactly: Appendices B and D document the trace-construction program, the templates, the library and tolerance behind each field, which fields are overwritten by MLIP predictions, and the corpus statistics of the resulting dataset, and Appendix C gives a complete prompt with the trace and structure it produces. Appendix E defines every evaluation metric and every reward term in closed form together with the matchers, tolerances and thresholds each one uses, and Table 6 states the exact objective of each RL run reported in the paper, so every row of every table can be traced back to the configuration that produced it. The code

release additionally contains the training and evaluation entry points, the split files, and the VASP inputs and per-structure outputs behind the DFT check of Section 4.3.

REFERENCES

Luis M Antunes, Keith T Butler, and Ricardo Grau-Crespo. Crystal structure generation with autoregressive large language modeling. *Nature Communications*, 15(1):10570, 2024.

Zhendong Cao and Lei Wang. Reinforcement fine-tuning for materials design. *Physical Review B*, 113(2):024106, 2026.

Zhendong Cao, Xiaoshan Luo, Jian Lv, and Lei Wang. Space group informed transformer for crystalline materials generation. *Science Bulletin*, 2025.

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

Daniel W Davies, Keith T Butler, Jonathan M Skelton, Congwei Xie, Artem R Oganov, and Aron Walsh. Computer-aided design of metal chalcohalide semiconductors: from chemical composition to crystal structure. *Chemical science*, 9(4):1022–1030, 2018.

Daniel W Davies, Keith T Butler, Adam J Jackson, Jonathan M Skelton, Kazuki Morita, and Aron Walsh. Smact: Semiconducting materials by analogy and chemical theory. *Journal of Open Source Software*, 4(38):1361, 2019.

Pierre-Paul De Breuck, Hai-Chen Wang, Gian-Marco Rignanese, Silvana Botti, and Miguel AL Marques. Generative ai for crystal structures: a review. *npj Computational Materials*, 2025.

Run-Ze Fan, Zengzhi Wang, and Pengfei Liu. Megascience: Pushing the frontiers of post-training datasets for science reasoning. *arXiv preprint arXiv:2507.16812*, 2025.

Jingru Gan, Peichen Zhong, Yuanqi Du, Yanqiao Zhu, Chenru Duan, Haorui Wang, Daniel Schwalbe-Koda, Carla P Gomes, Kristin A Persson, and Wei Wang. Matllmsearch: Crystal structure discovery with evolution-guided large language models. *arXiv preprint arXiv:2502.20933*, 2025.

Alex M Ganose and Anubhav Jain. Robocrystallographer: automated crystal structure text descriptions and analysis. *MRS Communications*, 9(3):874–881, 2019.

Bryan R Goldsmith, Jacques Esterhuizen, Jin-Xun Liu, Christopher J Bartel, and Christopher A Sutton. Machine learning for heterogeneous catalyst design and discovery. *AIChE-Journal*, 64 (7):2311–2323, 2018.

Nate Gruver, Anuroop Sriram, Andrea Madotto, Andrew Gordon Wilson, C Lawrence Zitnick, and Zachary Ulissi. Fine-tuned language models generate stable inorganic materials as text. *arXiv preprint arXiv:2402.04379*, 2024.

Daya Guo, Dejian Yang, Haowei Zhang, Junxiao Song, et al. Deepseek-r1 incentivizes reasoning in llms through reinforcement learning. *Nature*, 645(8081):633–638, 2025. doi: 10.1038/s41586-025-09422-z.

Theo Hahn, Uri Shmueli, and JC Wilson Arthur. *International tables for crystallography*, volume 1. Reidel Dordrecht, 1983.

Pierre Hohenberg and Walter Kohn. Inhomogeneous electron gas. *Physical review*, 136(3B):B864, 1964.

Zhang-Wei Hong, Nofit Segal, Aviv Netanyahu, Hoje Chun, Rafael Gomez-Bombarelli, and Pulkit Agrawal. Towards generating stable materials via large language models with reinforcement learning finetuning. In *NeurIPS 2025 AI for Science Workshop*, 2025.

Theo Jaffrelot Inizan, Sherry Yang, Aaron Kaplan, Yen-hsu Lin, Jian Yin, Saber Mirzaei, Mona Abdelgaid, Ali H Alawadhi, KwangHwan Cho, Zhiling Zheng, et al. System of agentic ai for the discovery of metal-organic frameworks. *arXiv preprint arXiv:2504.14110*, 2025.

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

Rui Jiao, Wenbing Huang, Peijia Lin, Jiaqi Han, Pin Chen, Yutong Lu, and Yang Liu. Crystal structure prediction by joint equivariant diffusion. *Advances in Neural Information Processing Systems*, 36:17464–17497, 2023.

Rui Jiao, Wenbing Huang, Yu Liu, Deli Zhao, and Yang Liu. Space group constrained crystal generation. *arXiv preprint arXiv:2402.03992*, 2024.

Chaitanya K Joshi, Xiang Fu, Yi-Lun Liao, Vahe Gharakhanyan, Benjamin Kurt Miller, Anuroop Sriram, and Zachary W Ulissi. All-atom diffusion transformers: Unified generative modelling of molecules and materials. *arXiv preprint arXiv:2503.03965*, 2025.

Filip Ekström Kelvinius, Oskar B Andersson, Abhijith S Parackal, Dong Qian, Rickard Armiento, and Fredrik Lindsten. Wyckoffdiff–a generative diffusion model for crystal symmetry. *arXiv preprint arXiv:2502.06485*, 2025.

Subhojyoti Khastagir, Kishalay Das, Pawan Goyal, Seung-Cheol Lee, Satadeep Bhattacharjee, and Niloy Ganguly. Llm meets diffusion: A hybrid framework for crystal material generation. *arXiv preprint arXiv:2510.23040*, 2025.

Georg Kresse and Jürgen Furthmüller. Efficient iterative schemes for ab initio total-energy calculations using a plane-wave basis set. *Physical review B*, 54(16):11169, 1996.

Nathan Lambert, Jacob Morrison, Valentina Pyatkin, Shengyi Huang, Hamish Ivison, Faeze Brahman, Lester James V. Miranda, et al. Tülu 3: Pushing frontiers in open language model post-training. *arXiv preprint arXiv:2411.15124*, 2024.

Trupti Mohanty, Maitrey Mehta, Hasan M Sayeed, Bat-El Oded, Itay Pitussi, Arie Borenstein, Vivek Srikumar, and Taylor D Sparks. Crystext: a generative ai approach for text-conditioned crystal structure generation using llm. *Integrating Materials and Manufacturing Innovation*, pp. 1–15, 2026.

Philipp Moritz, Robert Nishihara, Stephanie Wang, Alexey Tumanov, Richard Liaw, Eric Liang, Melih Elibol, Zongheng Yang, William Paul, Michael I Jordan, et al. Ray: A distributed framework for emerging {AI} applications. In *13th USENIX symposium on operating systems design and implementation (OSDI 18)*, pp. 561–577, 2018.

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

Long Ouyang, Jeffrey Wu, Xu Jiang, Diogo Almeida, Carroll Wainwright, Pamela Mishkin, Chong Zhang, Sandhini Agarwal, Katarina Slama, Alex Ray, et al. Training language models to follow instructions with human feedback. *Advances in neural information processing systems*, 35: 27730–27744, 2022.

Chris J Pickard and RJ Needs. Ab initio random structure searching. *Journal of Physics: Condensed Matter*, 23(5):053201, 2011.

Qwen, An Yang, Baosong Yang, Beichen Zhang, Binyuan Hui, et al. Qwen2.5 technical report. *arXiv preprint arXiv:2412.15115*, 2024.

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

Zhihong Shao, Peiyi Wang, Qihao Zhu, Runxin Xu, Junxiao Song, Xiao Bi, Haowei Zhang, Mingchuan Zhang, YK Li, Yang Wu, et al. Deepseekmath: Pushing the limits of mathematical reasoning in open language models. *arXiv preprint arXiv:2402.03300*, 2024.

Guangming Sheng, Chi Zhang, Zilingfeng Ye, Xibin Wu, Wang Zhang, Ru Zhang, Yanghua Peng, Haibin Lin, and Chuan Wu. Hybridflow: A flexible and efficient rlhf framework. *arXiv preprint arXiv: 2409.19256*, 2024.

Zayne Sprague, Fangcong Yin, Juan Diego Rodriguez, Dongwei Jiang, Manya Wadhwa, Prasann Singhal, Xinyu Zhao, Xi Ye, Kyle Mahowald, and Greg Durrett. To cot or not to cot? chain-of-thought helps mainly on math and symbolic reasoning. *arXiv preprint arXiv:2409.12183*, 2024.

Wenhao Sun, Stephen T Dacek, Shyue Ping Ong, Geoffroy Hautier, Anubhav Jain, William D Richards, Anthony C Gamst, Kristin A Persson, and Gerbrand Ceder. The thermodynamic scale of inorganic crystalline metastability. *Science advances*, 2(11):e1600225, 2016.

Atsushi Togo and Isao Tanaka. First principles phonon calculations in materials science. *Scripta materialia*, 108:1–5, 2015.

Atsushi Togo, Laurent Chaput, and Isao Tanaka. Distributions of phonon lifetimes in brillouin zones. *Physical review B*, 91(9):094306, 2015.

Atsushi Togo, Kohei Shinohara, and Isao Tanaka. Spglib: a software library for crystal symmetry search. *Science and Technology of Advanced Materials: Methods*, 4(1):2384822, 2024.

Yanchao Wang, Jian Lv, Li Zhu, and Yanming Ma. Crystal structure prediction via particle-swarm optimization. *Physical Review B-Condensed Matter and Materials Physics*, 82(9):094116, 2010.

Jason Wei, Xuezhi Wang, Dale Schuurmans, Maarten Bosma, Fei Xia, Ed Chi, Quoc V Le, Denny Zhou, et al. Chain-of-thought prompting elicits reasoning in large language models. *Advances in neural information processing systems*, 35:24824–24837, 2022.

Tian Xie, Xiang Fu, Octavian-Eugen Ganea, Regina Barzilay, and Tommi Jaakkola. Crystal diffusion variational autoencoder for periodic material generation. *arXiv preprint arXiv:2110.06197*, 2021.

Andy Xu, Rohan Desai, Larry Wang, Gabriel Hope, and Ethan Ritz. Plaid++: A preference aligned language model for targeted inorganic materials design. *arXiv preprint arXiv:2509.07150*, 2025.

Guang Yang, Yu Zhou, Xiang Chen, Xiangyu Zhang, Terry Yue Zhuo, and Taolue Chen. Chain-of-thought in neural code generation: From and for lightweight language models. *IEEE Transactions on Software Engineering*, 50(9):2437–2457, 2024a.

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

Sherry Yang, KwangHwan Cho, Amil Merchant, Pieter Abbeel, Dale Schuurmans, Igor Mordatch, and Ekin Dogus Cubuk. Scalable diffusion for materials generation. *arXiv preprint arXiv:2311.09235*, 2023.

Sherry Yang, Simon Batzner, Ruiqi Gao, Muratahan Aykol, Alexander Gaunt, Brendan McMorrow, Danilo Rezende, Dale Schuurmans, Igor Mordatch, and Ekin D Cubuk. Generative hierarchical materials search. *Advances in Neural Information Processing Systems*, 37:38799–38819, 2024c.

Claudio Zeni, Robert Pinsler, Daniel Zügner, Andrew Fowler, Matthew Horton, Xiang Fu, Sasha Shysheya, Jonathan Crabbé, Lixin Sun, Jake Smith, et al. Mattergen: a generative model for inorganic materials design. *arXiv preprint arXiv:2312.03687*, 2023.

Qing Zhao, Sanjuna Stalin, Chen-Zi Zhao, and Lynden A Archer. Designing solid-state electrolytes for safe, energy-dense batteries. *Nature Reviews Materials*, 5(3):229–252, 2020.

A Experimental Details

Data. All experiments are conducted on Materials Project (Jain et al., 2013) structures stored in our CrysReas database. We use the CDVAE MP-20 split (Xie et al., 2021) as the upstream data source. For supervised fine-tuning, we use `split_cdvae.json`, which contains 24,231 training structures and 8,141 test structures. We further construct task-specific subsets for stability optimization and property conditioning, deliberately limiting their size to avoid excessive training time. Because MLIP calculations are costly, we cap the number of samples in these subsets in order to keep the experiments tractable. For stability optimization, we select 8,000 training and 512 test structures to construct `split_rl.json`, which contains structures whose phases are valid in the `ReferenceMP2020Correction` phase diagram, so that energy above the hull can be evaluated consistently. For property-conditioned generation, we use `split_elastic.json` with 4,000 training and 256 test structures, and `split_cte.json` with 4,000 training and 256 test structures, which contain the structures that pass the corresponding MLIP calculations.

Instruction and Trace Construction. The user instructions randomly combine constraints over composition, space groups, stability-related quantities, and other physical properties, following a format similar to CrystalTextLLM (Gruver et al., 2024). For property-conditioned generation, we use empirical ranges but not arbitrary target values, ensuring that the instruction is physically reasonable and that at least one feasible reference structure exists. For thinking-trace supervision, we use Pymatgen (Ong et al., 2013) and Robocrystallographer (Ganose & Jain, 2019) to build rule-based traces that describe structural, electronic, stability, and mechanical information before emitting the final CIF-like structure. In the final CIF-like structure, lattice lengths, lattice angles and fractional coordinates are all printed to 8 decimal places.

Models and Baselines. All language-model variants are initialized from Qwen2.5-3B (Qwen et al., 2024) using the MegaScience-fine-tuned checkpoint (Fan et al., 2025), making the greatest possible use of prior scientific knowledge. We first train two SFT baselines: `CrysReas-Base` that directly emits the final CIF-like structure, and `CrysReas-Thinking` that first generates thinking traces before producing the structure. Our main stability-optimized model, `CrysReas`, starts from the thinking SFT model and is optimized with GRPO. We also train `CrysReas-RL` as an RL counterpart of the no-thinking baseline. For property-conditioned generation, we train `CrysReas-SpaceGroup`, `CrysReas-ElasticProperties`, and `CrysReas-ThermalExpansion`.

We compare against two prior-work-style representations implemented in the same CrysReas pipeline: `CrystalTextLLM` and `PLAID++ Wyckoff Base`. The three models share the dataset, the base checkpoint, the optimization hyperparameters, and numerical precision (8 decimal places).

Supervised Fine-Tuning. For SFT, we apply full-parameter fine-tuning for 2 epochs with a global batch size of 32, per-GPU micro-batch size 1, maximum sequence length 4096, learning rate 1×10^{-4}, Adam betas $(0.9, 0.95)$, weight decay 0.01, cosine learning-rate decay, 10% warmup, and gradient clipping at 1.0. Training uses FSDP, bf16 precision, and gradient checkpointing. The no-thinking and thinking models are trained on conditional structure-generation prompts. The CrystalTextLLM and PLAID++ Wyckoff Base comparison models are trained on the same structures but use their corresponding text representations; their SFT data mixes generation and infilling examples with a 66/34 ratio.

Reinforcement Learning. We use the Verl (Sheng et al., 2024) PPO trainer stack with the GRPO advantage estimator (Sheng et al., 2024). RL training runs for 1 epoch with batch size 64, group size 8, maximum prompt length 256, maximum response length 4096, actor learning rate 1×10^{-5}, PPO mini-batch size 32, per-GPU micro-batch size 1, clip ratio 0.2, entropy coefficient 0, and weight decay 0.01. The GRPO configuration uses $\gamma = 0.98$, $\lambda = 0.9$, normalized group advantages, and an adaptive KL controller with initial coefficient 0.001, target KL 0.05, and horizon 10,000. Rollouts are sampled with temperature 1.0 and top-$p = 1.0$.

Generation and Evaluation. At evaluation time, validity, instruction following and stability are measured on 1,024 held-out compositions of `split_generation.json`, a subset

of the 8,141 test compositions of the CDVAE MP-20 split, each generated once under each of 10 seeds, which gives the 10,240 structures behind Tables 1–3. The same 1,024 compositions and the same 10 seeds are used for every model, and the error bars in those tables are standard errors over the 10 seeds. Elasticity conditioned generation is evaluated on `split_generation_elastic.json` with 512 test prompts, and thermal expansion conditioned generation is evaluated on `split_generation_cte.json` with 256 test prompts, with 16 samples per prompt. We sample with temperature 1.0 and top-$p = 0.7$. Generated structures are parsed into the common CrysReas structure format and evaluated using the same downstream metric pipeline for all models. All training is performed on two A100 GPUs and requires less than 40 GPU hours in total.

Reward Calculations. To make online reward computation efficient, we use a CPU/GPU workload sharding strategy. Lightweight symbolic and structural checks, such as parsing, composition matching, space group matching, and SMACT validity, are parallelized on CPU workers with Ray by splitting the rollout batch into DataFrame chunks. Expensive MLIP-based metrics are handled separately as heavy metrics. For these metrics, structures are dispatched to Ray (Moritz et al., 2018) GPU workers, where MatterSim-based calculations are performed in batches. We also enable the Verl (Sheng et al., 2024) framework to launch reward calculation asynchronously during the computation of log probabilities under the current policy.

MLIP Settings. Direct first-principles evaluation of crystal stability and functional properties is too expensive to use during training. We therefore employ MatterSim (Yang et al., 2024b) as the MLIP backend for structure relaxation and property evaluation. In our pipeline, candidate structures are first relaxed with MatterSim, after which energy above the hull is computed from the relaxed structures and their predicted energies. For this step, we use the `ReferenceMP2020Correction` reference set from MatterGen (Zeni et al., 2023) to construct the phase diagram and evaluate hull distance.

To estimate the elastic properties at 0 K, we first perform an additional MLIP-based structural relaxation tailored for elastic analysis, allowing both atomic positions and lattice parameters to adjust until a force threshold of 10^{-4} eV/Å is reached. We then compute the stress response of the relaxed structure and estimate the full 6×6 elastic tensor in Voigt notation using symmetry-aware elastic analysis. From this tensor, we derive the bulk modulus and shear modulus.

For coefficient of thermal expansion at 300 K, we use a quasi-harmonic approximation (QHA) workflow driven by MLIP-predicted energies and forces (Togo & Tanaka, 2015; Togo et al., 2015), and report the volumetric thermal expansion coefficient.

B Thinking Traces Design

The thinking traces are produced by rule-based functions and narrow the solution space progressively, moving from the abstract to the concrete. This mirrors the way a human reasons, which both strengthens reasoning in natural-language space and lets the model exploit the similarity between this reasoning pattern and the large-scale text seen during pre-training.

First, the LLM determines the space group and appropriate Wyckoff sites (Hahn et al., 1983). This step establishes the fundamental symmetry constraints and the general symbolic arrangement of atoms, preventing the model from generating physically inconsistent configurations in subsequent stages.

Second, to map these abstract symmetry constraints to a precise geometric realization, we incorporate the descriptive logic derived from Robocrystallographer (Ganose & Jain, 2019). By training the LLM to mimic these structural descriptors, including coordination motifs, bond length distributions, and connectivity, the model learns to transform abstract symmetry into a coherent topological representation. These generated descriptors serve as intermediate physical constraints that assist the LLM in resolving the precise spatial coordinates.

Finally, the model predicts the expected physical properties, such as the band gap and the thermodynamic stability. These predictions commit the model to macroscopic quantities that the coordinates it is about to write will have to reproduce, acting as further constraints on the structure.

C EXAMPLE OF THINKING TRACES

Table 4 shows an example prompt and the corresponding model output. Appendix D documents the trace-construction program in full, including the exact templates, the libraries and tolerances used for every field, and the corpus statistics of the resulting supervision.

Freedoms and Constraints. In this example, Rb_2SnCl_6 has $N = 9$ atoms in its primitive cell and hence $3N + 6 = 33$ continuous degrees of freedom. Its space group $Fm\bar{3}m$ (No. 225) forces $a = b = c$ and $\alpha = \beta = \gamma$, leaving a single lattice parameter, and it places Sn on the Wyckoff orbit $4a$, Rb on $8c$ and Cl on $24e$; of these only $24e$ carries a free coordinate. The structure is therefore pinned down by *two* numbers, and every one of the remaining 31 directions is a direction in which an unconstrained generator can go wrong.

D CONSTRUCTION OF THE THINKING TRACES

The information in the thinking traces is collected from the ground-truth Materials Project record, with MLIP calculations overwriting some of the properties. Because no human annotation is involved, every statement in a trace is correct with respect to its source record; the only systematic error is the gap between the MLIP predictions that overwrite some fields and the DFT values they approximate.

Source database. We obtain 200,290 materials from the Materials Project, together with the following properties: `material_id`, `structure`, `energy_above_hull`, `formation_energy_per_atom`, `is_stable`, `decomposes_to`, `band_gap`, `is_metal`, `efermi`, `bulk_modulus`, `shear_modulus`, `homogeneous_poisson`, `formula_pretty`, and the natural-language structural `description` produced by Robocrystallographer (Ganose & Jain, 2019) from Materials Project.

MLIP-derived property targets. To align with MLIP calculation during training and evaluation, `bulk_modulus` and `shear_modulus` are overwritten with the MatterSim-derived Voigt–Reuss–Hill moduli, and `thermal_expansion_300k` is set from the quasi-harmonic workflow of Appendix A. The Poisson ratio is left at its DFT value. On the SFT split, 97.8% of the materials that carry a bulk modulus carry the MLIP scalar; the remainder retain the original Materials Project VRH dictionary, and the trace renders them with the $K_{\mathrm{VRH}}/G_{\mathrm{VRH}}$ wording plus an anisotropy sentence. Property-conditioned generation is therefore MLIP-self-consistent by construction, which we regard as a feature for reward design and a limitation for absolute physical accuracy.

Trace template. For a record e with ground-truth structure S, the response is assembled in a fixed order. Let $\textsc{Sym}(S)$ denote the symmetry-and-sites block, $\textsc{Desc}(e)$ the Robocrystallographer description, and $\textsc{Val}(S)$ the validity sentence. The trace is

```
Let's generate a material report first, according to the
given information.
## Material Report:
### Crystal Structure
First, consider space groups and atom numbers.  SYM(S)
Second, consider band gaps.  DESC(e) Third, consider
structure validity.  VAL(S)
## CIF File
<CIF>...</CIF>
```

Inside the validity part $\textsc{Val}(S)$, three optional sections `### Electronic Properties`, `### Stability` and `### Mechanical Properties` may appear, whenever the underlying field is available for the structure. Their relative order is resampled uniformly for every training example, in order to increase the diversity of the traces.

Symmetry, sites and oxidation states. The space group is obtained with `spglib` (Togo et al., 2024) through `pymatgen`'s (Ong et al., 2013) `SpacegroupAnalyzer` at `symprec` = 0.01,

Table 4: A partially specified prompt and the corresponding `CrysReas` output. The instruction fixes only the chemical formula and the space group; the formation energy, the hull distance and the band gap are *not* given and must be predicted by the model. Colors mark the trace segments: symmetry, site multiplicities and electroneutrality, local coordination environment, self-consistency check and predicted functional properties, the emitted structure. Verification of the emitted structure against the instruction: the reduced formula matches, `spglib` assigns space group No. 225 as requested, and the structure passes the structural and chemical validity checkers of Appendix E. The predicted properties are also close to the Materials Project reference values ($E_g = 2.55\,\text{eV}$, $E_f = -1.89\,\text{eV/atom}$, on the hull). All 10 samples drawn for this prompt satisfied the composition and space group constraints. Note that the order of the trailing property sections is resampled per generation, as described in Appendix D.

User Prompt
Below is a description of a bulk material. The chemical formula is Rb2SnCl6. The space group number is 225. Generate a description of the lengths and angles of the lattice vectors and then the element type and coordinates for each atom within the lattice:

Assistant Response
Let's generate a material report first, according to the given information.

Material Report:

Crystal Structure
First, consider space groups and atom numbers. This material Rb2SnCl6 should have the space group Fm-3m (id 225). Since for Rb, 2=2, for Sn, 1=1, for Cl, 6=6, 2*(+1)+1*(+4)+6*(-1)=0, the structure is like this: Rb has 1 sites: one site has 2 atoms, oxidation state +1. Sn has 1 sites: one site has 1 atoms, oxidation state +4. Cl has 1 sites: one site has 6 atoms, oxidation state -1.
Second, consider band gaps. Rb_2SnCl_6 crystallizes in the cubic $Fm\bar{3}m$ space group. Rb^{1+} is bonded to twelve equivalent Cl^{1-} atoms to form $RbCl_{12}$ cuboctahedra that share corners with twelve equivalent $RbCl_{12}$ cuboctahedra, faces with six equivalent $RbCl_{12}$ cuboctahedra, and faces with four equivalent $SnCl_6$ octahedra. All Rb-Cl bond lengths are 3.64 Å. Sn^{4+} is bonded to six equivalent Cl^{1-} atoms to form $SnCl_6$ octahedra that share faces with eight equivalent $RbCl_{12}$ cuboctahedra. All Sn-Cl bond lengths are 2.45 Å. Cl^{1-} is bonded in a distorted single-bond geometry to four equivalent Rb^{1+} and one Sn^{4+} atom. Third, consider structure validity. The structure is reasonable, because the bond lengths are all greater than 0.5, and the structure's volume 273.68 is larger than 1.

Stability
Thermodynamic Status: It is predicted to be thermodynamically stable (on the hull). The formation energy per atom is -1.938 eV/atom.

Electronic Properties
Classification: It is a semiconductor (narrow/moderate band gap) with a calculated band gap (E_g) of 2.237 eV.

CIF File
P1
7.27478708 7.27478685 7.27478659
60 60 60
Rb 1 0.25000000 0.25000000 0.25000000
Rb 1 0.75000000 0.75000000 0.75000000
Sn 1 -0.00000000 -0.00000000 0.00000000
Cl 1 0.76195631 0.23804369 0.23804369
Cl 1 0.23804369 0.76195631 0.76195631
Cl 1 0.23804369 0.76195631 0.23804369
Cl 1 0.76195631 0.23804369 0.76195631
Cl 1 0.23804369 0.23804369 0.76195631
Cl 1 0.76195631 0.76195631 0.23804369

falling back to symprec $= 0.1$ when the tighter tolerance returns no dataset. Sites are grouped into symmetry orbits with get_symmetrized_structure(); the reported multiplicity is the orbit size *within the deposited cell*, not the Wyckoff multiplicity of the conventional cell. Wyckoff letters are computed and used to order the orbits, but are not verbalized. Oxidation states come from pymatgen's bond-valence analyser (BVAnalyzer, default settings: symm_tol $= 0.1$, max_radius $= 4$ Å, charge-neutrality tolerance 10^{-5}). When the analyser fails, the oxidation states and the electroneutrality clause are dropped from the trace; this happens for about 40% of the training materials. The clause itself has the form

```
Since for Li, 4=4, for C, 4=4, for S, 4=4, for N, 4=4,
4*(+1)+4*(+4)+4*(-2)+4*(-3)=0, the structure is like this:
Li has 1 sites:  one site has 4 atoms, oxidation state +1.
...
```

where the per-element equations check that the assigned orbit sizes reproduce the requested stoichiometry and the final equation checks charge balance. Because the bond-valence analyser only returns an assignment that is already charge neutral, the charge equation always evaluates to zero when it is printed: it is an exposition device that teaches the model the form of the check, not a filter that rejects data.

Validity sentence. The trace closes the structural block with a fixed sentence asserting that all bond lengths exceed 0.5 Å and reporting the cell volume. This part acts as a final confirmation after previous inference.

Electronic and stability sections. A material is verbalized as a metal when the Materials Project is_metal flag is set and a Fermi energy is available; otherwise a positive band gap is rendered as an insulator when $E_g > 3.0$ eV and as a semiconductor otherwise. On the SFT split this yields 62.8% metals, 27.6% semiconductors and 9.6% insulators. Stability has exactly two verbalizations: on-hull, keyed to the Materials Project is_stable flag, and "metastable, lying x eV/atom above the convex hull". There is no third, "unstable" class and no numerical threshold in the trace; the E_{hull} threshold used for the S.U.N. metric enters only at evaluation time. Band gaps, energies and moduli are rendered to three decimal places and the Fermi energy to two.

Structure serialization. The final block is a simplified CIF, delimited by the special tokens <CIF> and </CIF> which are added to the tokenizer vocabulary. Its format is a space group token, a lattice-length line, a lattice-angle line, and one line per atom:

```
P1
3.85878300 5.34539600 12.34309100
90 90 90
Li 1 0.25000000 0.43864100 0.90123400
...
```

Angles, lengths and fractional coordinates all carry eight decimal places. The leading P1 records that no symmetry is applied and that every atom of the cell is listed explicitly, and the per-atom integer column is a multiplicity field that is always 1 in this setting; it is retained for compatibility with symmetric variants of the format used in prior work such as PLAID++.

Instruction construction and partial conditioning. The user instruction always begins "Below is a description of a bulk material.", always states the chemical formula, and then appends a *random subset* of the remaining conditions before the generation request. For the main conditional task the candidate set is {formation energy per atom, band gap, energy above the convex hull, space group number}; the number of conditions k is drawn uniformly from $\{0, 1, 2, 3, 4\}$ and then a uniformly random k-subset is chosen in a uniformly random order. Each individual property therefore appears in exactly half of the prompts, and 20% of prompts specify nothing beyond the chemical formula. This is the reason a single model must be able to *predict* the properties it is not told, and it is what the thinking trace is for; Table 4 shows a partially specified example. The property-conditioned tasks are deterministic instead: spacegroup always adds the space group number, elastic adds bulk- and shear-modulus intervals, and cte adds a thermal-expansion interval together with a "fewer than ten atoms" instruction. Continuous targets are never given as point values; they are quantized to the

Table 5: Terms of the reward of Equation 1. "Gated" marks the two indicators that multiply other terms instead of being added to them. All continuous terms are bounded, which keeps a single outlier from dominating a GRPO group; $R_{\text{stability}}$ exceeds 1 only for structures predicted *below* the hull, which we do not clip. Definitions are in Appendix E.

Term	Measures	Source	Range	Weight
$\mathbf{1}_{\text{format}}$	delimiters and stop tokens well formed	string match	$\{0, 1\}$	gate
$\mathbf{1}_{\text{validity}}$	structural and chemical checks passed	checkers	$\{0, 1\}$	gate
$\mathbf{1}_{\text{parse}}$	the string can be decoded to a structure	parser	$\{0, 1\}$	$w_p = 0.1$
$R_{\text{structural}}$	contacts, volume, cell angles	CDVAE checker	$\{0, 1\}$	$w_s = 0.3$
R_{chemical}	charge-neutral oxidation assignment	SMACT	$\{0, 1\}$	$w_c = 0.3$
$R_{\text{instruction}}$	reduced formula matches the request	string match	$\{0, 1\}$	$w_i = 0.3$
$R_{\text{stability}}$	hull distance E_{hull}	MatterSim + MatterGen hull	$(0, 1]$ on $E_{\text{hull}} \geq 0$	$\alpha_{\text{stability}} = 10$
$R_{\text{range}}(P)$	property P inside the requested interval	MLIP workflow	$(-1, 1]$	specialists only

intervals listed in Appendix E and rendered as "in $[L, R]$", "less or equal than R" or "greater or equal than L", so that the instruction is satisfiable and matches the reward exactly.

Determinism. At generation time the condition subset is seeded from the numeric part of the `mp-id`, so every evaluation prompt is reproducible and identical across models. During supervised fine-tuning the subset is *not* seeded, which acts as a data augmentation: the same structure is seen under different partial specifications across epochs.

Corpus statistics. Over the supervised split, responses average 1088 tokens, of which 641 are thinking tokens and 447 are the CIF block; user prompts average 69 tokens. Within the thinking prefix the symmetry block averages 129 tokens, the Robocrystallographer description 301, the validity sentence 38, and the property sections 133. The Electronic and Stability sections are present for essentially every material, the Mechanical section for 92%, and the oxidation-state clause for 60%. Thinking-token count correlates with cell size at Pearson $r = 0.57$ in the supervision itself, rising from 430 tokens for cells of one to four atoms to 943 tokens for cells of seventeen to twenty atoms. This is by construction, since larger cells have more orbits to enumerate.

Representation baselines. The same pipeline renders the two prior-work baselines, so that the only difference between CrysReas and the baselines is the intermediate representation. The CrystalTextLLM (Gruver et al., 2024) serialization applies a random rigid translation to all sites. The PLAID++ (Xu et al., 2025) Wyckoff serialization prints the reduced formula, the Hermann–Mauguin symbol, lattice parameters, and one line per site annotated with its Wyckoff symbol. Both baselines are trained on a 66/34 mixture of generation and element-infilling examples, following the original works. Both baselines are using 8-digit decimal numbers, consistent with CrysReas.

E Metrics and Rewards

This appendix specifies every quantity that is either reported as a metric or consumed as a reward. Evaluation and online RL share a single implementation: each quantity is a node in a dependency graph over a table of generated structures, so a number that appears in a results table is produced by exactly the same code path that produced the corresponding training signal. We first describe the shared machinery, then the individual terms, and finally the composite objectives used by each experiment.

Shared evaluation graph. A generation run yields one row per (prompt, sample) pair. Metrics are registered with their dependencies and a topological rank, and requesting a metric pulls in its ancestors automatically: for instance `energy_above_hull` requires `relaxed_structures`, which in turn requires `simple_structure`, `structure_validity` and `smact_validity`. Expensive MLIP stages are therefore never executed for rows that already failed a cheap check, and that is what makes online reward computation affordable. A failed prerequisite propagates as a missing value and not as a zero, and missing values are mapped to a reward of 0 at the last moment; a structure that cannot be parsed is therefore neither rewarded nor explicitly punished. Lightweight

symbolic checks are sharded across CPU workers and MLIP stages across GPU workers, as described in Appendix A.

E.1 Validity and Instruction Following

Output format. Before any physical quantity is computed, a response must be well formed: it must contain <CIF> and </CIF> followed by the end-of-turn and end-of-text tokens in that order, with at most four characters of slack between the closing tag and the end-of-turn token and between the end-of-turn and end-of-text tokens. This indicator multiplies the entire reward, so a malformed response receives exactly zero regardless of the structure it may contain. It is a gate and is never itself rewarded, which prevents the policy from collecting reward for emitting the delimiters alone.

Parseability. A separate, weakly weighted indicator records whether the delimited block parses into a structure at all. It is distinct from structural validity: a response can parse into a geometrically absurd structure, and the two terms then disagree. Keeping them separate gives the policy a dense early signal at the beginning of RL, when a non-trivial fraction of rollouts still emit unparseable coordinate blocks.

Structural validity. We follow the checker introduced by CDVAE (Xie et al., 2021), extended with three lattice sanity conditions:

$$R_{\text{structural}} = \mathbf{1}\left[\min_{i\neq j} d_{ij}^{\text{PBC}} \geq 0.5\,\text{Å} \;\wedge\; V \geq 1.0\,\text{Å}^3 \;\wedge\; \min(a,b,c) \geq 1.1\,\text{Å} \;\wedge\; \alpha,\beta,\gamma \in [20^\circ, 160^\circ]\right], \tag{2}$$

where d_{ij}^{PBC} is the minimum-image interatomic distance, so periodic images are included. The 0.5 Å contact threshold is deliberately permissive: it rejects atoms placed on top of one another without rejecting short but physical bonds such as those in hydrides.

Chemical validity. We use SMACT (Davies et al., 2019) to test whether the generated composition admits an assignment of oxidation states that is charge neutral and consistent with electronegativity ordering.

Composition consistency. The reduced formula of the generated structure must match the requested one exactly, as a canonical string comparison; there is no tolerance. This is the only instruction-following term entering the validity reward:

$$R_{\text{instruction}} = \mathbf{1}_{\{\text{reduced formula matches}\}}. \tag{3}$$

E.2 Energy and Thermodynamic Stability

Relaxation and hull distance. Generated structures are relaxed with MatterSim (Yang et al., 2024b) using the FIRE optimizer under a variable-cell filter, so both atomic positions and lattice parameters respond. The relaxation is capped at 20 steps during reward computation, which is the main cost control in the online loop; a 300-step variant is used when exporting candidates for DFT. Relaxed structures and their MatterSim energies are then passed through MatterGen's evaluation stack (Zeni et al., 2023): each structure is wrapped so that the Materials Project 2020 compatibility corrections apply to it exactly as they would to a GGA(+U) calculation, and the hull distance E_{hull} in

Table 6: Reward configuration of each RL run. All runs are gated on output format. "Validity block" denotes the four terms $(0.1, 0.3, 0.3, 0.3)$ on parseability, structural validity, chemical validity and composition consistency.

Model	Validity block	Property / stability terms	Conditioning
`CrysReas` (Mixed)	yes	$10\,R_{\text{stability}}$	composition, partial properties
`CrysReas-RL`	yes	$10\,R_{\text{stability}}$	composition, partial properties
Validity Only	yes	—	composition, partial properties
Energy Only	no	$10\,R_{\text{stability}}$	composition, partial properties
`CrysReas-SpaceGroup`	yes	$1 \cdot R_{\text{spacegroup}}$	formula + space group
`CrysReas-ElasticProperties`	no	$5\big(R_{\text{range}}(K) + R_{\text{range}}(G)\big) + 1\,R_{\text{stability}}$	formula + K, G intervals
`CrysReas-ThermalExpansion`	no	$5\,R_{\text{range}}(\alpha_V) + 1\,R_{\text{stability}}$	formula + α_V interval

eV/atom is computed against the `ReferenceMP2020Correction` phase diagram, which combines Materials Project and Alexandria entries. Negative hull distances are kept as they are and never clipped, so a structure predicted to lie below the current hull is reported as such. Structures the potential cannot handle yield a missing value, which becomes a reward of zero.

Stability reward. A raw $-E_{\text{hull}}$ reward has three defects: it provides no signal when the MLIP fails, its range is unbounded so a single outlier dominates a GRPO group, and its gradient is constant, so it is no more sensitive near the hull than far from it. We therefore use

$$R_{\text{stability}}(E_{\text{hull}}) = \begin{cases} 0, & E_{\text{hull}} \text{ undefined}, \\ 1 - \dfrac{E_{\text{hull}}}{2E_0}, & E_{\text{hull}} \le E_0, \\ \dfrac{E_0}{2E_{\text{hull}}}, & E_{\text{hull}} > E_0, \end{cases} \qquad E_0 = 1 \text{ eV/atom}. \tag{4}$$

The two branches agree in value and slope at $E_{\text{hull}} = E_0$, so the reward is continuously differentiable. On the physically relevant range $E_{\text{hull}} \ge 0$ it takes values in $(0, 1]$, decays like $1/E_{\text{hull}}$ for grossly unstable structures so that outliers cannot dominate a group, and is steepest near the hull, which is where additional accuracy is worth the most. For the small fraction of structures predicted below the hull the reward exceeds 1; we do not clip this, because a below-hull prediction is exactly the outcome the objective is meant to encourage. The scale $E_0 = 1$ eV/atom is chosen to match the typical hull distance of the SFT policy at the start of RL, so that the initial rollouts populate the steep part of the curve.

E.3 Range-Constraint Reward for Property Conditioning

Continuous property targets are specified as intervals and not as points, both in the instruction and in the reward, because the underlying MLIP predictions carry uncertainty that a point target would not respect. Given a predicted value P and a requested interval $[L, R]$, let $z = \big(P - \frac{L+R}{2}\big)/(R-L)$ and $y = 1 - 2z^2$. The reward is

$$R_{\text{range}}(P; L, R) = \begin{cases} y, & y \ge 0 \iff |z| \le \frac{1}{\sqrt{2}}, \\ e^{\,y} - 1, & \text{otherwise}, \end{cases} \qquad R_{\text{range}} \in (-1, 1]. \tag{5}$$

It attains its maximum of 1 at the interval midpoint, which gives a single unambiguous optimum instead of a flat plateau over $[L, R]$ that would provide no gradient once the constraint is nominally satisfied. At the interval endpoints the reward equals 0.5, and it remains positive slightly outside the interval, out to $|z| = 1/\sqrt{2} \approx 0.707$; beyond that the exponential tail decays smoothly to -1, so the reward keeps responding even for predictions far from the target. If the prediction or either bound is unavailable the reward is 0. Requested intervals are the quantization bins listed in Table 8; instruction and reward therefore reference the same interval by construction, and a model cannot satisfy the text while violating the reward.

Space group reward. The space group specialist is trained on the indicator $R_{\text{spacegroup}} = \mathbf{1}[\,\text{the detected space group of the generated structure carries at least the requested symmetry}\,]$, using the same matcher and `symprec` schedule as the space group consistency metric of Section 4.1.

Table 7: Agreement between the thinking trace and the structure the model then writes. For each generated sample we compare the number of symmetry-distinct sites, the cell volume and the bond lengths asserted in the trace against those measured on the emitted structure, broken down by requested space group and pooled over all of them. Agreement is close in the two cubic groups, which are nearly determined by their symmetry, and much weaker once all space groups are pooled, so the trace acts as a prior on the structure rather than a specification the decoder always meets.

Metric	Fm-3m	Fd-3m	P3m1	All structures
Site Match (↑)	99.75%	100.00%	81.85%	75.19%
Volume rel. diff. (↓)	0.71%	2.46%	3.81%	5.05%
Bond Length rel. diff. (↓)	1.08%	2.02%	21.91%	23.25%

Table 8: Quantization bins used both to verbalize a property constraint in the instruction and to define the reward target. Bins are half-open intervals between consecutive edges.

Property	**Unit**	**Bin edges**
Bulk modulus K_{VRH}	GPa	$-\infty,\ 0,\ 10,\ 50,\ 150,\ 300,\ \infty$
Shear modulus G_{VRH}	GPa	$-\infty,\ 0,\ 20,\ 80,\ 200,\ \infty$
Volumetric CTE $\alpha_V(300\,\mathrm{K})$	$10^{-6}\,\mathrm{K}^{-1}$	$-\infty,\ -10,\ 0,\ 5,\ 15,\ 40,\ 100,\ 200,\ 300,\ \infty$

Elastic and thermal-expansion rewards. The elastic specialist is trained on $R_{\text{range}}(K_{\mathrm{VRH}}) + R_{\text{range}}(G_{\mathrm{VRH}})$, where the moduli are obtained by relaxing each candidate to a tight force threshold, computing the stress response to symmetry-allowed strains, assembling the 6×6 tensor in Voigt notation and reducing it by the Voigt–Reuss–Hill average. The thermal-expansion specialist is trained on $R_{\text{range}}(\alpha_V(300\,\mathrm{K}))$, with α_V from the quasi-harmonic workflow of Appendix A; because that workflow is far more expensive than a relaxation, it is applied only to the lowest-energy candidates within each rollout batch, and the remaining rollouts receive zero for this term. Both specialists additionally carry a stability term, so that conditioning does not come at the price of producing unstable structures.

E.4 Uniqueness, Novelty, S.U.N., and S.S.U.N.

These quantities are reported but never used as rewards. Structure matching uses MatterGen's disordered structure matcher with length tolerance 0.2, site tolerance 0.3 and angle tolerance $5°$, comparing primitive cells after scaling, and candidates are only compared within a reduced-formula group.

Uniqueness is computed greedily inside a single seed: a structure counts as unique if it matches no previously accepted structure of that seed, so the first member of each duplicate class is counted and the rest are not. It is a property of the whole generated set and not of an individual sample, and it is sensitive to set size; all uniqueness numbers in this paper are computed over sets of the same size, namely the $1{,}024$ structures of one seed, and are then averaged over the 10 seeds. Because candidates are only compared within a reduced-formula group and a seed contributes exactly one structure per composition, a duplicate inside a seed means the model returned the same structure for two different requested compositions. This concept is different from that in prior works, which evaluates all generation over different seeds, because scoring uniqueness over the 10 seeds pooled together would instead confine every comparison to the 10 samples drawn for a single composition, and would therefore charge a model for answering that composition consistently, which, however, the conditioned task is asking it to do.

Novelty is measured against the SFT training split, the $24{,}231$ Materials Project entries the model was fine-tuned on, so a structure counts as novel when the matcher finds nothing like it among the structures the model was trained on. The evaluation compositions share no Materials Project id with the training split, and 21.5% of them share a reduced formula with a training entry, so that is the share on which novelty can fail at all. We also computed the stricter reading, novelty against the same large Materials Project and Alexandria reference set that defines the convex hull. On the same generated sets it gives 1.37% for CrysReas, 1.16% for PLAID++ Wyckoff and 0.77% for CrystalTextLLM, the same ordering as Table 1 over 140, 119 and 79 structures. This check holds stability at the 0.1 eV/atom window so that only the novelty reference set changes; at the $E_{\text{hull}} \leq 0$ threshold of S.U.N. the counts fall to 17, 9 and 2 structures, which is too few to order the models. Those counts are small enough that we report the training-split reading in the main text and keep this one as a check.

S.U.N. is the conjunction of stable, unique and novel, and is the quantity most directly related to the yield of a screening campaign: it counts how many of the generated candidates are simultaneously worth computing, not already known, and not duplicates of each other. **S.S.U.N.** requires space group consistency on top of those three, so it counts the candidates that are also built in the symmetry the instruction asked for. Following Xu et al. (2025), the two use different stability thresholds: S.U.N. takes stable to mean $E_{\text{hull}} \leq 0$ eV/atom, the same threshold as the Stability column, while S.S.U.N. takes it to mean $E_{\text{hull}} \leq 0.1$ eV/atom, the metastability window. This is why S.S.U.N. is

Table 9: Elasticity-conditioned generation: trade-off between target adherence and structural validity. The specialist `CrysReas-ElasticProperties` improves follow-elasticity rate at a small cost to structural validity compared to the baseline `CrysReas`.

Metric	CrysReas	CrysReas-ElasticProperties
Structural Validity (↑)	**95.29%** ± 0.21%	94.20% ± 0.23%
Chemical Validity (↑)	**99.85%** ± 0.04%	99.82% ± 0.04%
Follow Elasticity Requirement (↑)	56.85% ± 0.37%	**61.22%** ± 0.35%

the larger of the two despite carrying one extra factor. Rows whose relaxation failed count as neither unique nor novel, so failures are charged against the denominator.

E.5 Composite Objectives

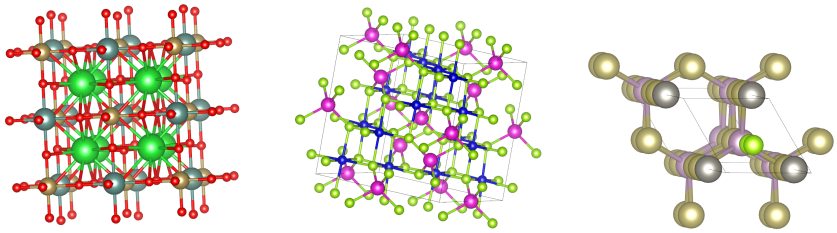
The reward presented to GRPO is a format-gated weighted sum,

$$R_{\text{target}} = \mathbf{1}_{\text{format}} \Big[\underbrace{w_p \mathbf{1}_{\text{parse}} + w_s R_{\text{structural}} + w_c R_{\text{chemical}} + w_i R_{\text{instruction}}}_{\alpha_{\text{validity}} R_{\text{validity}}} + \alpha_{\text{stability}} \mathbf{1}_{\text{validity}} R_{\text{stability}} \Big], \tag{6}$$

with $(w_p, w_s, w_c, w_i) = (0.1, 0.3, 0.3, 0.3)$, so the validity block contributes at most 1, and $\alpha_{\text{stability}} = 10$ for the stability-optimized models. The gate $\mathbf{1}_{\text{validity}}$ is not an explicit multiplier but an emergent property of the dependency graph: hull distance is only computed for structures that pass the structural and chemical checkers, and an undefined hull distance maps to $R_{\text{stability}} = 0$. This gives the behavior we want. Early in training, when few rollouts are valid, the validity block supplies almost all of the signal and its terms are dense and easy to improve. Once validity saturates the block becomes nearly constant across a GRPO group and, because group-relative advantages subtract the group mean, it stops contributing gradient; the stability term, which is continuous and far from saturated, then dominates.

Table 6 lists the objective of every RL run reported in the paper. The **Validity Only** and **Energy Only** ablations of Section 4.3 differ only in the presence of the validity block; the stability weight is unchanged between **Energy Only** and **Mixed**, so the comparison isolates the regularizing effect of the validity terms and not a change of scale. And the two property specialists drop the validity block entirely in favor of their property term plus a stability term, which is the most likely explanation for the small loss of structural validity reported in Table 9.

Why these rewards admit online RL. Online RL requires the reward to be computable on every rollout within the time budget of a policy step, and to be informative enough that a group of eight rollouts contains a usable ranking. The design above satisfies both. Every term is either a symbolic check costing microseconds or a bounded MLIP quantity computed on a subset that has already passed the cheap checks, and the expensive stages are overlapped with log-probability computation



(a) Fm-3m (b) Fd-3m (c) P3m1

Figure 5: Structures generated by CrysReas for three requested space groups, one example each, rendered from the emitted CIF. They illustrate the regime of Table 7: the two cubic cases (a), (b) are nearly determined by their symmetry and are where the thinking trace and the generated structure agree most closely, whereas the trigonal case (c) leaves free coordinates and agrees least.

as described in Appendix A. Every continuous term is bounded and smooth, so no single rollout can dominate the group-relative advantage, and no term saturates at the point where the previous term stops providing gradient. This is what distinguishes the setup from offline preference learning on precomputed candidates, where the reward may be arbitrarily expensive but the policy never sees feedback on its own current distribution.

Reward hacking and its mitigation. Because the stability signal comes from a surrogate potential, the policy could in principle improve its reward by exploiting regions where the potential underestimates energy, without generating genuinely stable crystals. Three properties of the design limit this. The reward is gated on symbolic validity checks that the potential cannot influence; the hull distance is evaluated against a fixed, large reference set and not against a learned critic; and the decaying tail of $R_{\text{stability}}$ removes most of the incentive to produce extreme outliers. We also recompute hull distances with DFT on a held-out set of generated structures in Section 4.3, and they give the same ordering between model variants that MatterSim gives.

F Property Prediction Accuracy of Thinking Traces

Physical Properties Are Predicted Before Generation. To understand the relation between thinking traces and the final atomic coordinates, we compare the difference between the predicted physical properties in thinking tokens (bond length and volume) and the actual physical properties of the generated structure in Table 7. In the two high-symmetry space groups the agreement is close: site counts match on almost every structure and bond lengths agree to within 1–2%. It degrades as symmetry is lowered, and over all space groups the site match is 75.19% and the mean relative bond-length deviation 23.25%. The pattern follows the number of free parameters the symmetry leaves open: where the space group nearly determines the structure, trace and structure agree; where it does not, the trace constrains the structure without determining it. Figure 5 shows one generated structure per space group.