

Reusable density control for inverse materials design with autoregressive models

Can Polat¹ ■ Mustafa Kurban^{2,3,✉} ■ Erchin Serpedin¹ ■ Hasan Kurban^{4,✉}

¹ Department of Computer and Electrical Engineering, Texas A&M University, College Station, TX, USA

² Department of Electrical and Computer Engineering, Texas A&M University at Qatar, Doha, Qatar

³ Department of Prosthetics and Orthotics, Ankara University, Ankara, Turkey

⁴ College of Science and Engineering, Hamad Bin Khalifa University, Doha, Qatar

✉ Correspondence: kurbanm@ankara.edu.tr, hkurban@hbku.edu.qa

ABSTRACT

Inverse materials design requires generating property-targeted candidate structures without repeatedly retraining generative models or sacrificing compositional variety. Here, we present a cell-first mass-budgeting framework that applies local minimum-Kullback–Leibler moment projections during autoregressive decoding, dynamically guiding element choices toward requested target densities using pre-emitted cell volumes. Retargeting a single trained model across ten density requests demonstrates robust checkpoint reusability without parameter updates. Across candidate selection pools of size sixteen, local density steering reduces average candidate density error by 70.5%. Selecting among sixteen steered candidates achieves nearly identical density accuracy to sixty-four unsteered candidates while expanding the unique composition fraction from 22.8% to 37.1%. Analytical enumeration over an exact sequence space further demonstrates that per-step moment control concentrates total additive-feature outcomes while clarifying global variance behavior. These results demonstrate that reusable, representation-aligned decoding control significantly enhances candidate selection efficiency while balancing property precision and compositional diversity.

KEYWORDS materials generation; density control; constrained decoding; autoregressive models; differentiable optimization

1 Introduction

Inverse materials design aims to discover novel atomic structures tailored to specific functional property targets [1]. Generative models accelerate this paradigm by learning structural priors directly from crystallographic data, expanding candidate discovery beyond static databases [2]. Property-conditioned generation further enables targeted sampling toward numeric requirements, including multi-objective property targets [3]. However, aggressive reward-driven optimization often causes mode collapse and severe loss of candidate diversity, restricting the variety of distinct chemical compositions available for downstream evaluation [4]. Effective inverse design therefore

demands generative frameworks capable of accurately steering outputs toward property requests while maintaining diverse candidate pools for post-generation selection.

Existing strategies introduce controllable steering at distinct stages of the generative pipeline. Pretrained crystal diffusion models incorporate property and symmetry constraints through fine-tuning [5], whereas autoregressive transformers learn discrete sequence distributions over crystallographic representations [6] or combine structural priors with property likelihoods for conditional sampling [7]. Fine-tuning-free alternatives apply adaptive constraint guidance during diffusion trajectories [8] or leverage reinforcement learning with reasoning-focused rewards [9]. Despite these advances, adapting pretrained models to arbitrary target values often requires repeated retrainings or expensive sampling procedures, highlighting the need for reusable, training-efficient steering mechanisms during discrete sequence decoding.

For discrete sequence generation, enforcing strict support rules and directing continuous property expectations represent distinct operations. Restricting token emissions to valid syntax via support-only constraints minimizes Kullback–Leibler (KL) divergence from an unconstrained model over admissible token sets [10]. Incorporating expected-feature constraints under KL minimization provides a framework for distributional control [11], which aligns with differentiable convex optimization layers in neural architectures [12]. These foundations raise two essential questions: how can a physical property computed from a partially emitted structural prefix dynamically guide remaining discrete choices, and how does such local guidance affect both candidate accuracy and compositional uniqueness across selection budgets?

Here, we address these challenges through density-directed autoregressive generation of composition–cell pairs. Our main contributions are:

1. **Cell-First Mass Budgeting.** *Formulate* a cell-first density control mechanism that projects local token distributions onto convex mass targets via minimum-KL optimization during autoregressive decoding, enabling zero-shot checkpoint retargeting across diverse density requests.
2. **Reusable Candidate Selection.** *Demonstrate* multi-request checkpoint reusability on the MP-20 crystal dataset [13], establishing that local density steering significantly reduces candidate selection error while expanding compositional uniqueness compared to unsteered baselines.
3. **Analytical Moment Bounds.** *Derive* a closed-form additive variance identity and perform exact sequence-space enumeration to show how per-step conditional moment enforcement bounds total output variance and isolates global expectation matching from local target hits.

2 Results

2.1 Cell-first generation exposes a mass budget

Emitting the unit cell geometry prior to chemical composition transforms a global density target into a local mass steering constraint during autoregressive decoding. For a requested target density ρ^* and an emitted unit cell volume V , the required total formula mass c in atomic mass units follows directly:

$$c = \frac{\rho^* V}{1.66053906660}, \quad (1)$$

where V is measured in $\text{\AA}^3$ and ρ^* in g/cm^3 . As the decoder sequentially emits element–count pairs, each completed pair contributes its mass to a running ledger. The remaining mass budget is

updated dynamically based on an estimated remaining atom count (configured at an expected total of 10.4 atoms), directly coupling continuous property requests to discrete token decisions without altering the pre-emitted unit cell (Fig. 1).

For any interior feasible mass request, applying a minimum KL projection produces an exponential tilt of the masked model distribution via a single scalar multiplier. Direct evaluation of 2,048 generated outputs from a density-5 model shows a high valid-decoding rate: 2,031 outputs completed token syntax with valid cell volumes and recognized atomic masses—achieving a 99.17% decoding yield without assigning explicit atomic positions. Detailed chemical screens and reference baseline evaluations are provided in Supplementary Section S1.

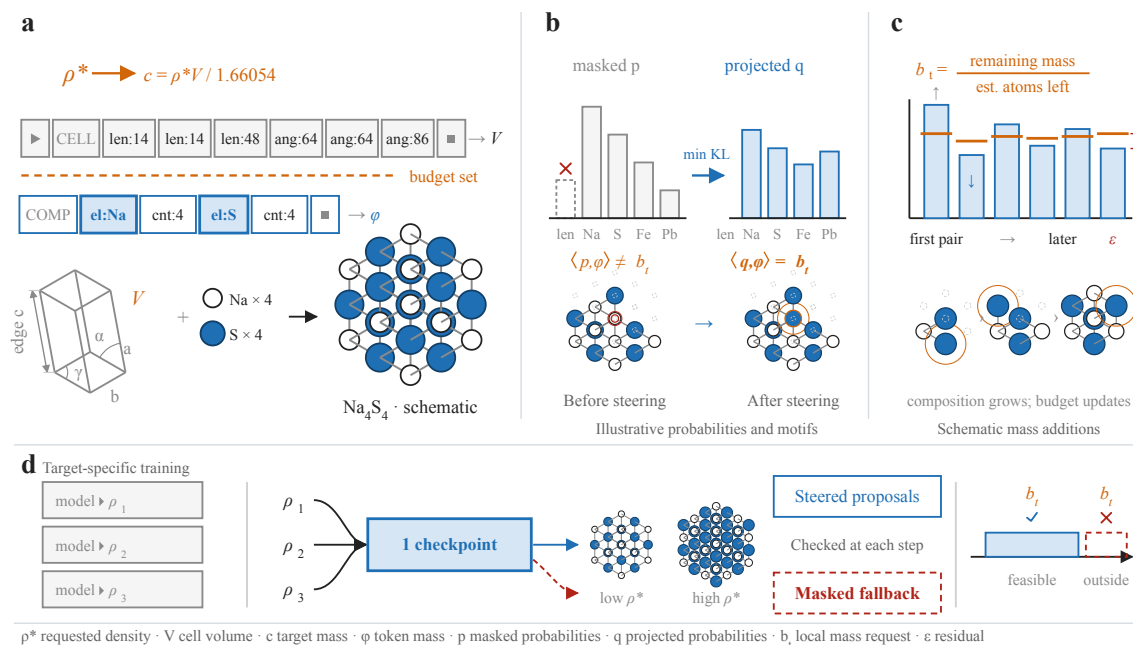


Figure 1 | Cell-first generation supports reusable density steering. **a**, Cell closure fixes volume V , converting requested density ρ^* into target mass c in atomic mass units. Length and angle token suffixes are bin identifiers, not physical dimensions. The Na₄S₄ sequence and cell motif illustrate the representation. **b**, A minimum-KL projection transforms masked probabilities p into q satisfying the feasible interior expected-mass request b_t ; ϕ denotes token mass. The crossed-out length-token family is grammar-excluded. Bar heights illustrate a normalized distribution and its exponential tilt. **c**, The remaining mass and estimated remaining atom count update after each completed element-count pair. Blue bars denote schematic mass additions, orange marks local requests and ϵ a residual; these are not measured trajectories or an exact-final-density guarantee. **d**, Multiple requests reuse one checkpoint. Local feasibility and solver convergence determine whether projected probabilities are used; otherwise decoding uses the masked fallback. The interval depicts feasibility of a local mass request, not global density reachability. All atomic motifs are schematic and do not represent generated atomic positions.

2.2 Trained weights support changed density requests

Loaded model weights can be redirected across varying density targets without requiring parameter fine-tuning. Evaluating three checkpoints trained at baseline densities of 3, 5, and 10 g/cm³

across ten target requests spanning 1.5–20 g/cm³ confirms zero-shot retargeting across all thirty response curves. Every evaluation trial maintained grammar completion while shifting output mass distributions (Fig. 2).

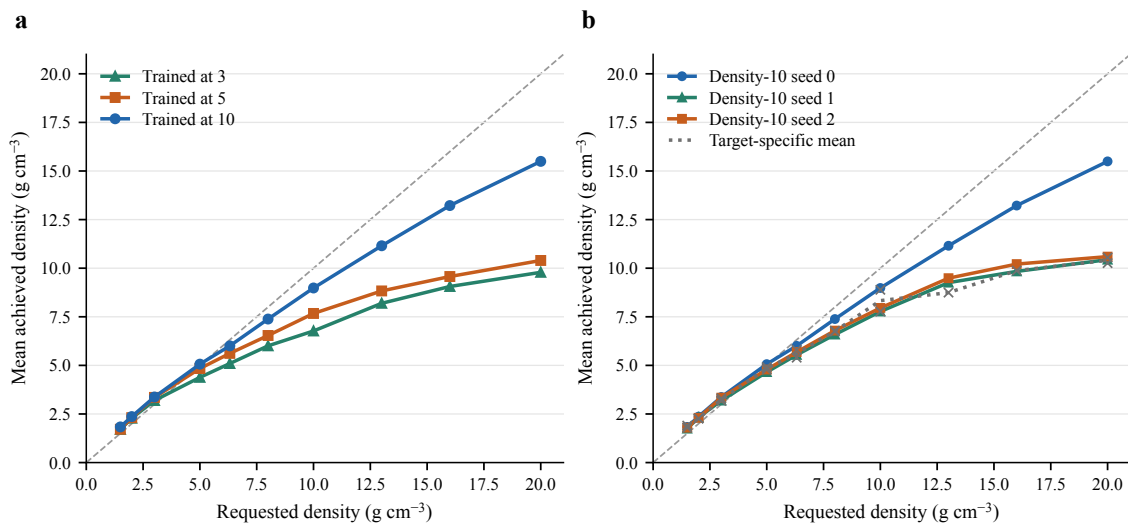


Figure 2 | Checkpoint reuse across the full density range. **a**, Mean achieved density for checkpoints trained at 3, 5 and 10 g cm⁻³ and deployed at all ten requests, with one training seed per curve. **b**, Three training seeds at density 10 are shown individually, alongside the mean response of separately trained target-specific models. Small grey markers show each completed target-specific run; no uncertainty is inferred for single-run targets. Dashed lines denote identity. Each retargeting point summarizes 4,096 attempted outputs; density means condition on computable values in (0, 100) g cm⁻³. The target-specific sweep contains eighteen completed runs: two at eight targets and one each at 13 and 16. Lines connect the fixed evaluated requests and do not estimate responses between them.

Response magnitude remains constrained by early autoregressive decisions and checkpoint-specific priors. For instance, seed 0 of the density-10 model reaches a mean density of 15.5 g/cm³ at the highest request, whereas seeds 1 and 2 plateau near 10.4 and 10.6 g/cm³, illustrating how prefix selection bounds local steering capacity. Across the seven target requests at or below 10 g/cm³, three density-10 model seeds achieved target-averaged mean density absolute errors of 0.4406, 0.7889, and 0.7197 g/cm³. Their three-seed average (0.6497 g/cm³) closely matches the performance of target-specific retrained baselines (0.6620 g/cm³), showing that, across these seven targets, zero-shot steering achieves target accuracy comparable to dedicated retraining (Supplementary Section S2).

2.3 Projection improves candidate selection across ten targets

Local moment projections concentrate generated distributions around property targets, substantially improving post-generation candidate selection efficiency. Evaluating best-of- n candidate selection from saved pools (1,021 unprojected and 511 projected outputs per target from a recorded density-5 model over 200 independent trials) reveals systematic improvements in target accuracy (Fig. 3).

In single-candidate sampling ($n = 1$), steering reduces target-averaged candidate error from

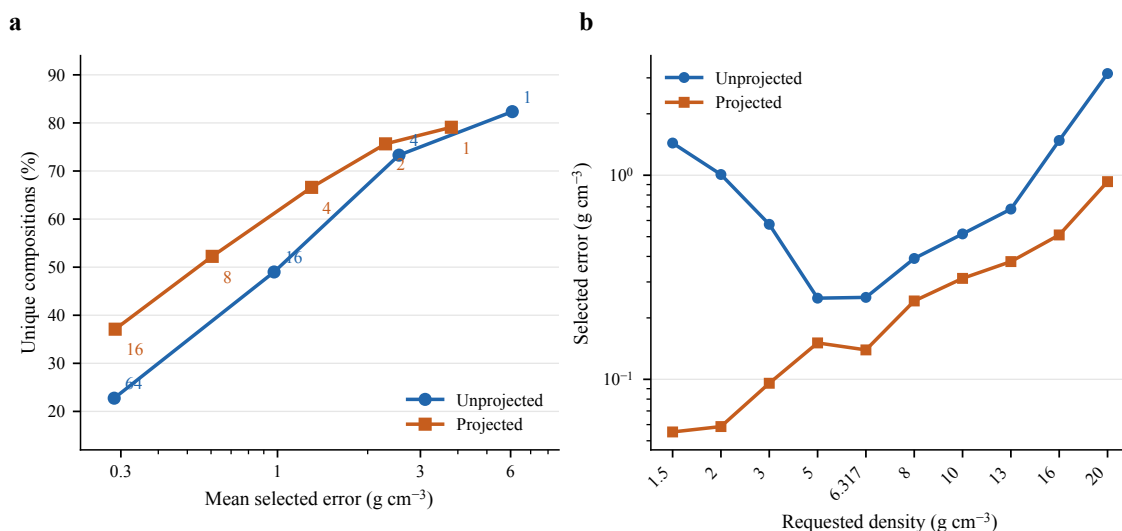


Figure 3 | Projection changes the selection trade-off. **a**, Mean unique-composition fraction against mean selected-candidate absolute density error, giving equal weight to all ten targets. Each point is labelled by the number n of candidates considered per selection. **b**, Selected-candidate error at $n = 16$ for each target. Each error value averages 200 trials conditional on its saved pool; uniqueness is the fraction of distinct element-count tuples among 200 outputs from a separate set of trials. The comparison has one recorded checkpoint and sampling seed. No training-seed uncertainty or replicate uniqueness error bars are inferred. Retained pool sizes are 1,021 unprojected and 511 projected per target.

6.078 to 3.805 g/cm^3 (a 37.4% reduction), confirming that projected proposals are centered closer to target values before best-of- n ranking. At $n = 16$, local projection reduces target-averaged selection error from 0.974 to 0.287 g/cm^3 —a 70.5% reduction across all ten targets, with target-specific error reductions reaching up to 96.2%.

Furthermore, local projection optimizes the operational balance between candidate accuracy and compositional variety. Unprojected best-of-64 candidate selection achieves a mean error of 0.285 g/cm^3 with a 22.75% unique composition fraction. By contrast, projected best-of-16 achieves nearly identical accuracy (0.287 g/cm^3) while using one-fourth the selection budget and preserving a 37.10% unique composition fraction—a 14.35 percentage point increase in diversity. Comparisons against target-filtered retraining and sequential Monte Carlo (SMC) baselines are documented in Supplementary Section S1.

2.4 Closing-step moments connect local and total control

Evaluating an exact enumerable task (65,536 length-eight sequences across four charged symbols) elucidates how local moment control concentrates global outcome distributions without requiring exact target attainment on every individual string. At target charge zero, the deployed projection model achieves a mean charge of 0.0170, variance of 1.547, and an exact zero-charge hit rate of 38.45% across three training seeds (Fig. 4). By comparison, an exact sequence-level KL tilt exhibits a variance of 10.452 and a 12.45% exact hit rate. Local steering thus concentrates probability density tightly around the target expectation, reducing overall outcome variance compared to

global sequence tilting.

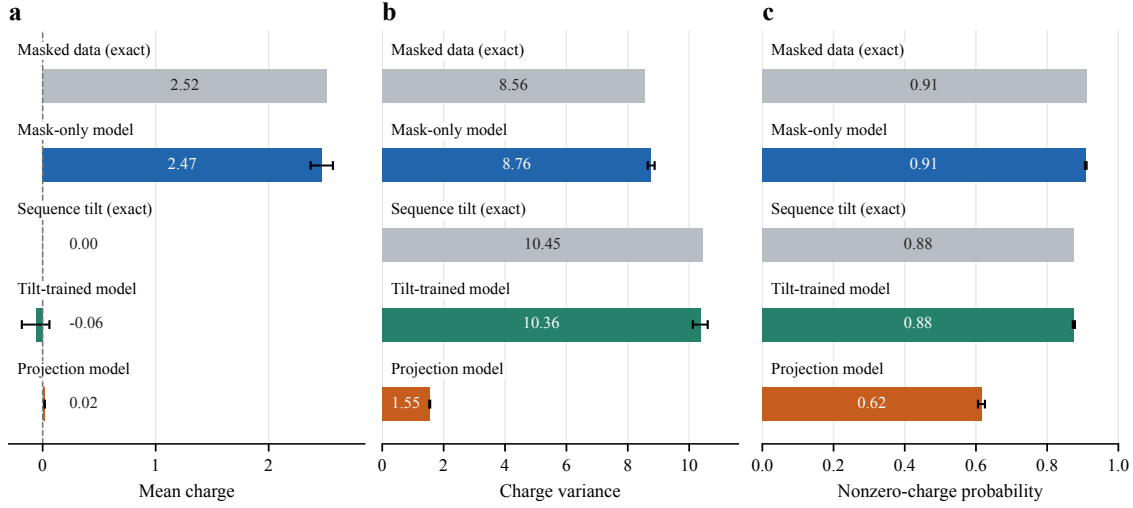


Figure 4 | An expected target does not specify the outcome distribution. Mean total charge (a), total-charge variance (b) and nonzero-charge probability (c) at target zero. The mask-conditioned data distribution and sequence-level KL tilt are exact references. Trained-model bars are means over three seeds; error bars show the sample standard deviation (SD) across these seeds. Each run’s distribution is evaluated by enumeration, with no sampling uncertainty. The projection model is read with its trained decoder active. Numbers report bar values rounded for display; near-zero values appear beside their bars for legibility. The dashed zero line in a marks the requested expectation.

The mathematical connection between step-wise conditional steering and total additive features is formalized at the closing generation step. For a sequence length L , target c , and final conditional residual $\delta(H) = \mathbb{E}[\phi(x_L) | H] - b_L$ given prefix H , the overall bias and variance resolve to:

$$\mathbb{E}[S] - c = \mathbb{E}[\delta], \quad \text{Var}(S) = \mathbb{E}[\text{Var}(\phi(x_L) | H)] + \text{Var}(\delta). \quad (2)$$

Equation (2) proves that enforcing an exact final conditional expectation ($\delta = 0$) guarantees an unbiased global outcome whose variance equals the expected final conditional variance. At target charge zero, analytical evaluation identifies locally infeasible requests at 5,460 of 11,760 final-prefix nodes (46.4%); these are unweighted node counts, not probability-weighted frequencies. The local construction yields a total mean of 0.103 and variance of 3.103 (Supplementary Section S4). These bounds confirm that sequential steering stabilizes global property expectations while preserving distribution variance.

3 Discussion

Integrating reusable density control into autoregressive generation redefines candidate selection as an explicit component of the inverse design objective. Decoupling per-step logit projection from downstream candidate ranking establishes that generative interventions should be evaluated by the property precision and compositional diversity of candidates that survive selection, rather than relying solely on raw proposal distribution statistics. Because projection modifies token probabilities

before sequence completion while selection acts on fully emitted objects, this two-stage workflow delivers significant accuracy gains even when mean output densities do not perfectly track extreme property requests. In complementary materials prediction and inverse-retrieval tasks, SCALAR evaluates numerical accuracy alongside physical consistency and output validity [14].

The closing-step moment analysis provides a rigorous diagnostic framework for extending running budget constraints. While early autoregressive decisions determine prefix feasibility, evaluating final conditional residuals isolates instances where the generator fails to enforce the target total. In materials sequence decoding, tracking the running mass budget exposes these local failures directly, offering explicit mathematical conditions for global moment claims even when expected atom counts vary.

Aligning steering mechanisms with a model’s intrinsic sequence representation avoids the computational overhead of training auxiliary surrogate predictors. Rather than fitting a separate density estimator, our approach leverages the explicit additive relation exposed by cell-first serialization. Contextualizing these findings alongside broader evaluations of crystal generators [15] demonstrates that optimal steering strategies depend strongly on available structural information and downstream selection goals. Related benchmarking with RADII identifies architecture-dependent losses of structural fidelity under size extrapolation [16]. Rather than treating a single aggregate benchmark as a universal ranking of materials generators, property controls should be selected based on representation structure and selection budgets.

The empirical evidence established here defines specific operational boundaries. Composition-cell representations omit explicit 3D atomic coordinates, meaning subsequent chemical screens do not guarantee thermodynamic stability, relaxed structural equilibrium, or synthesizability. Furthermore, selection evaluations reflect recorded single-checkpoint pools and specific sampling seeds; independent model retraining across high-density regimes illustrates seed-dependent variation. Similarly, finite-iteration Newton solvers and expected atom count estimates introduce numerical approximations that distinguish implemented backpropagation from ideal projection Jacobians. These boundaries delineate the scope of current claims while establishing the exact validation protocol required for broader deployments.

Connecting cell-first composition controllers to full 3D atomistic models represents a promising direction for future research, testing whether near-target, compositionally distinct candidates yield viable crystal structures upon structural relaxation. Beyond density retargeting, formulating other additive physical or chemical attributes as running sequential budgets offers a scalable path toward multi-property inverse design. Each extension will require rigorous feasibility validation to evaluate how local steering affects both target precision and candidate diversity in downstream discovery pipelines.

4 Methods

4.1 Data, representation and training

Empirical evaluations utilize the MP-20 dataset splits distributed with the Crystal Diffusion Variational Autoencoder (CDVAE) [13], originating from the Materials Project [17]. The canonical splits contain 27,136 training, 9,047 validation, and 9,046 test records. Preprocessing filters the dataset to 26,910 valid training records containing explicit cell parameters, recognized chemical elements, and

integer multiplicities, verifying that crystallographic information file (CIF) strings do not duplicate across splits.

The custom serializer converts periodic crystals into token sequences comprising six structural markers, three cell lengths, three cell angles, and alphabetically ordered element–count pairs. Over an 884-token vocabulary (6 markers, 94 elements, 16 counts, 512 length bins, and 256 angle bins quantized over $[2, 46]$ Å and $[5, 175]^\circ$), the longest sequence spans 26 tokens. Multiplicities exceeding sixteen are clipped to 16, affecting 39 training records, while atomic coordinates are deliberately omitted. Triclinic cell volume is computed as:

$$V = abc \sqrt{1 - \cos^2 \alpha - \cos^2 \beta - \cos^2 \gamma + 2 \cos \alpha \cos \beta \cos \gamma}. \quad (3)$$

Sequences producing nonpositive squared volumes in Eq. (3) are rejected. Formula mass $M = \sum_j k_j m_j$ yields crystal density $\rho = 1.66053906660M/V$ in g/cm³.

The autoregressive decoder consists of a 6-layer transformer with 8 attention heads, hidden dimension 256, feed-forward dimension 1,024, learned positional embeddings, causal masking, and zero dropout (5,199,220 parameters for the 884-token vocabulary). Models are trained on an NVIDIA RTX 5090 GPU for 6,000 steps using AdamW [18] (batch size 128, peak learning rate 3×10^{-4} with 300 linear warmup steps, gradient norm clipping at 1.0, $\beta_1 = 0.9$, $\beta_2 = 0.999$, $\epsilon = 10^{-8}$, and weight decay 0.01). Training examples are drawn with replacement, optimizing teacher-forced cross-entropy over padded sequence batches where configured target requests define target mass independent of true density.

4.2 Mass budgeting and local projection

Following unit cell emission, the committed mass $M_{<t}$ and atom count $N_{<t}$ update dynamically upon completing each element–count pair. The running mass budget R_t , estimated remaining atom count $\bar{n}_t$, and per-step local mass request b_t follow:

$$R_t = \max(c - M_{<t}, 0), \quad \bar{n}_t = \max(10.4 - N_{<t}, 1), \quad b_t = R_t / \bar{n}_t. \quad (4)$$

Equation (4) uses a fixed total-atom estimate of 10.4, consistent with the mean of 10.4233 atoms across the original 27,136 MP-20 training records before preprocessing, rounded to one decimal place. The retained 26,910 training records have a mean of 10.4799 atoms. The configured value is a budgeting approximation, not an enforced composition size.

Local projection intervenes at element decisions after cell closure. Let A_t denote the finite grammar-allowed token set on which the baseline model places positive mass, and let $\bar{p}_t$ be its normalized distribution over A_t . For atomic mass feature vector ϕ and local target b_t , minimum-KL constrained optimization seeks a distribution q_t :

$$q_t = \arg \min_{q \in \Delta(A_t)} \text{KL}(q \| \bar{p}_t) \quad \text{subject to} \quad \sum_i q_i \phi_i = b_t. \quad (5)$$

For any interior request within attainable feature limits, the convex dual solution yields an exponential logit tilt parameterized by scalar multiplier λ_t :

$$q_{t,i} = \frac{\bar{p}_{t,i} \exp(\lambda_t \phi_i)}{Z_t(\lambda_t)}, \quad Z_t(\lambda) = \sum_{j \in A_t} \bar{p}_{t,j} \exp(\lambda \phi_j). \quad (6)$$

Because the derivative $dE_q[\phi]/d\lambda = \text{Var}_{q_t}(\phi) > 0$ is strictly positive for nonconstant features, every interior request possesses a unique finite multiplier λ_t . For any feasible distribution q , the identity:

$$\text{KL}(q\|\bar{p}_t) - \text{KL}(q_t\|\bar{p}_t) = \text{KL}(q\|q_t) \geq 0 \quad (7)$$

guarantees global optimality and uniqueness under local KL divergence. Removing the moment equality constraint recovers normalized grammar masking [10], whereas global sequence-level constraints regulate full trajectory distributions [11].

4.3 Numerical implementation and differentiation

Finding the multiplier λ_t employs a float64 damped Newton solver (maximum 50 iterations, convergence tolerance 10^{-10} , Hessian damping 10^{-10} , and at most 30 backtracking step-halvings). Projections accept solver solutions achieving absolute residuals below 10^{-8} . Training clips local requests b_t to the reachable feature interval $[\min_{A_t} \phi, \max_{A_t} \phi]$, applying projected softmax at active positions and falling back to normalized masking upon solver non-convergence.

During inference, local requests are solved on clipped bounds, applying projected probabilities when the raw request is feasible within 10^{-6} and solver residuals remain below 10^{-8} ; otherwise, normalized masked softmax is retained. Before sampling, nonfinite probability weights are set to zero and negative weights are clipped to zero. Rows with no positive mass fall back to similarly sanitized masked-softmax weights; if these also contain no positive mass, the end-of-sequence token is emitted. Sampling executes multinomial draws at unit temperature without top- k or nucleus filtering. Algorithm 1 summarizes the operational decoding pipeline.

Algorithm 1 | Materials decoding at a supplied prefix

Input: logits z , grammar support A , emitted cell, committed mass/count

- 1 $\bar{p} \leftarrow \text{softmax}(z \text{ masked to } A)$; $q \leftarrow \bar{p}$
 - 2 **if** next decision is an element position and the cell budget is defined **then**
 - 3 Compute $b = \max(c - M, 0) / \max(10.4 - N, 1)$
 - 4 $\ell \leftarrow \min_A \phi$; $h \leftarrow \max_A \phi$
 - 5 Solve the clipped request $\min(h, \max(\ell, b))$ by damped Newton
 - 6 **if** $\ell - 10^{-6} \leq b \leq h + 10^{-6}$ and the final residual is below 10^{-8} **then**
 - 7 $q \leftarrow \text{solver output}$
 - 8 **end if**
 - 9 **end if**
 - 10 Apply the finite-probability safeguards described in the text
 - 11 Sample from q ; update mass and count when a multiplicity is committed
-

For ideal interior solutions, differentiating the constrained projection with respect to input logits z while holding features and targets fixed yields the Jacobian J [19]:

$$J = \text{diag}(q) - qq^\top - \frac{uu^\top}{\text{Var}_q(\phi)}, \quad u = q \odot (\phi - b). \quad (8)$$

Differentiating $dq = (\text{diag}(q) - qq^\top) dz + u d\lambda$ alongside the target identity $\phi^\top q = b$ confirms $d\lambda = -u^\top dz / \text{Var}_q(\phi)$, establishing Eq. (8). Training backpropagates directly through unrolled Newton

steps rather than implicit Jacobian wrappers, maintaining numerical stability during optimization.

4.4 Retargeting and density metrics

Zero-shot retargeting evaluates ten density requests (1.5, 2, 3, 5, 6.317, 8, 10, 13, 16, and 20 g/cm³, where 6.317 represents the training set median). Each evaluated cell attempts 4,096 generative sampling runs. Of twenty planned target-specific runs, eighteen completed; target 13, seed 0 and target 16, seed 1 failed during multinomial sampling before the safeguard repair. These runs remain unavailable and were not rerun with the repaired safeguard; no missing values are imputed. Across evaluation runs, mean density error Δ_{mean} and candidate mean absolute error MAE_{cand} evaluate aggregate and individual performance:

$$\Delta_{\text{mean}} = \left| N^{-1} \sum_i \rho_i - \rho^* \right|, \quad \text{MAE}_{\text{cand}} = N^{-1} \sum_i |\rho_i - \rho^*|, \quad (9)$$

conditioning metrics on grammar-valid outputs with computed densities in (0, 100) g/cm³.

4.5 Candidate-selection protocol

Candidate selection evaluations utilize checkpoint A3_d5.0.pt (density-5, seed 0). Saved pools contain 1,021 unprojected and 511 projected retained outputs per target. Best-of- n selection evaluates candidate pools ($n \in \{1, 4, 16, 64\}$ unprojected; $n \in \{1, 2, 4, 8, 16\}$ projected) over 200 independent trials, selecting the candidate minimizing absolute density error. Composition uniqueness measures the fraction of distinct, sorted element-count tuples across 200 independently selected outputs.

4.6 Alternative controls and chemical screens

Target-filtered training fits unprojected models to training subsets restricted within ± 0.5 g/cm³ of target densities (3, 5, 8, and 10 g/cm³). Sequential Monte Carlo (SMC) inference introduces a Gaussian density potential during autoregressive sampling [20], executing particle filtering with width $\sigma = 0.5$ g/cm³ and $K = 16$ particles.

Chemical validity screens evaluate 2,048 generated density-5 outputs using SMACT [21] for neutral oxidation state charge balance, Pauling electronegativity ordering, and volume-per-atom packing constraints derived from training set percentiles (8.919–46.302 Å³). Reference benchmarks evaluate 800 randomly sampled training structures under identical criteria.

4.7 Enumerable experiment and closing-step analysis

The finite target task evaluates sequence length 8 over 4 charged symbols (2, 1, −1, −2) with symbol multiplicity constraints, defining 36,120 admissible strings among 65,536 total sequences. Baseline Markov chains, exact sequence-level KL tilts, and trained transformer models (4 layers, 4 heads, width 128, feed-forward width 512) are evaluated via exact space enumeration across three independent random seeds.

Proposition 1 | Closing-step control. *Let $L \geq 1$ be fixed, ϕ a bounded additive feature, c a fixed target, and $H = x_{<L}$ the sequence prefix preceding the final emission. Define total sum $S = \sum_{t=1}^L \phi(x_t)$, remaining budget $b_L = c - \sum_{t<L} \phi(x_t)$, and final conditional residual $\delta(H) = \mathbb{E}[\phi(x_L) \mid H] - b_L$.*

Then:

$$\mathbb{E}[S] - c = \mathbb{E}[\delta], \quad (10)$$

$$\text{Var}(S) = \mathbb{E}[\text{Var}(\phi(x_L) | H)] + \text{Var}(\delta), \quad (11)$$

$$\mathbb{E}[(S - c)^2] = \mathbb{E}[\text{Var}(\phi(x_L) | H)] + \mathbb{E}[\delta^2]. \quad (12)$$

If the final conditional moment is enforced almost surely ($\delta = 0$), the generated total is unbiased and its total variance equals the average final conditional variance.

Proof. By sequence bookkeeping, $S - c = \phi(x_L) - b_L$, giving conditional expectation $\mathbb{E}[S - c | H] = \delta(H)$ and conditional variance $\text{Var}(S | H) = \text{Var}(\phi(x_L) | H)$. Taking expected values yields Eq. (10), while applying the law of total variance proves Eq. (11). Summing squared bias and variance establishes Eq. (12). ■

Proposition 1 demonstrates that local conditional feasibility bounds total sequence variance even when individual target outcomes vary.

4.8 Use of large language models

Large language models assisted drafting, copy editing, formatting, code development, literature organization, and verification. The authors remain responsible for reviewing and verifying all scientific statements, proofs, analyses, code, figures, tables, references, and results, and for the manuscript’s content.

DATA AVAILABILITY

The MP-20 dataset used in this study is publicly available through the CDVAE repository and originates from the Materials Project. The numerical data supporting the findings of this study are available in the project repository at <https://github.com/KurbanIntelligenceLab/cpt-ms>. Additional results and analyses are provided in the Supplementary Information.

CODE AVAILABILITY

The code used to generate and analyze the results reported in this study is available at <https://github.com/KurbanIntelligenceLab/cpt-ms>.

AUTHOR CONTRIBUTIONS

C.P. developed and implemented the methodology and software, performed the computational analyses, prepared the visualizations, and wrote the initial draft of the manuscript. M.K. supervised the research, evaluated and validated the materials-science aspects of the study, contributed to the interpretation of the data and results, and reviewed and edited the manuscript. E.S. provided resources, contributed to the supervision of the research, and reviewed and edited the manuscript. H.K. contributed to the supervision of the research, interpretation of the results, and review and editing of the manuscript. All authors reviewed and approved the final manuscript.

COMPETING INTERESTS

The authors declare no competing interests.

ACKNOWLEDGMENTS

This research received no specific grant from any funding agency.

References

- [1] Alex Zunger. Inverse design in search of materials with target functionalities. *Nature Reviews Chemistry*, 2:0121, 2018. doi: 10.1038/s41570-018-0121.
- [2] 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. In *International Conference on Learning Representations (ICLR)*, 2024.
- [3] Yan Chen, Xueru Wang, Xiaobin Deng, Yilun Liu, Xi Chen, Yunwei Zhang, Lei Wang, and Hang Xiao. MatterGPT: A generative transformer for multi-property inverse design of solid-state materials. arXiv:2408.07608, 2024.
- [4] Andy Xu, Rohan Desai, Larry Wang, Ethan T. Ritz, and Gabriel Hope. PLaID++: A preference aligned language model for targeted inorganic materials design. In *Proceedings of the 43rd International Conference on Machine Learning*, volume 306 of *Proceedings of Machine Learning Research*, pages 140903–140922. PMLR, 2026.
- [5] Claudio Zeni, Robert Pinsler, Daniel Zügner, Andrew Fowler, Matthew Horton, Xiang Fu, Zilong Wang, Aliaksandra Shysheya, Jonathan Crabbé, Shoko Ueda, Roberto Sordillo, Lixin Sun, Jake Smith, Bichlien Nguyen, Hannes Schulz, Sarah Lewis, Chin-Wei Huang, Ziheng Lu, Yichi Zhou, Han Yang, Hongxia Hao, Jielan Li, Chunlei Yang, Wenjie Li, Ryota Tomioka, and Tian Xie. A generative model for inorganic materials design. *Nature*, 639:624–632, 2025. doi: 10.1038/s41586-025-08628-5.
- [6] Luis M. Antunes, Keith T. Butler, and Ricardo Grau-Crespo. Crystal structure generation with autoregressive large language modeling. *Nature Communications*, 15:10570, 2024. doi: 10.1038/s41467-024-54639-7.
- [7] Zhendong Cao, Xiaoshan Luo, Jian Lv, and Lei Wang. Space group informed transformer for crystalline materials generation. *Science Bulletin*, 70(21):3522–3533, 2025. doi: 10.1016/j.scib.2025.09.035.
- [8] Auguste de Lambilly, Vladimir Baturin, David Portehault, Guillaume Lambard, Nataliya Sokolovska, Florence d’Alché Buc, and Jean-Claude Crivello. Fine-tuning-free diffusion model with adaptive constraint guidance for inorganic crystal structure generation. *arXiv preprint arXiv:2604.13354v3*, 2026.
- [9] Yuyang Wu, Stefano Falletta, Delia McGrath, and Sherry Yang. CrystalReasoner: Reasoning and RL for property-conditioned crystal structure generation. *arXiv preprint arXiv:2605.14344v2*, 2026. doi: 10.48550/arXiv.2605.14344.
- [10] Hanwool Lee. (G)I-DLE: Generative inference via distribution-preserving logit exclusion with KL divergence minimization for constrained decoding, 2025.
- [11] Muhammad Khalifa, Hady Elsahar, and Marc Dymetman. A distributional approach to controlled text generation. In *International Conference on Learning Representations (ICLR)*, 2021.
- [12] Akshay Agrawal, Brandon Amos, Shane Barratt, Stephen Boyd, Steven Diamond, and J. Zico Kolter. Differentiable convex optimization layers. In *Advances in Neural Information Processing Systems (NeurIPS)*, 2019.
- [13] Tian Xie, Xiang Fu, Octavian-Eugen Ganea, Regina Barzilay, and Tommi S. Jaakkola. Crystal diffusion variational autoencoder for periodic material generation. In *International Conference on Learning Representations (ICLR)*, 2022.
- [14] Can Polat, Mustafa Kurban, Erchin Serpedin, and Hasan Kurban. SCALAR: Quantifying structural hallucination, consistency, and reasoning gaps in materials foundation models. *Digital Discovery*, 10 2026. ISSN 2635-098X. doi: 10.1039/D6DD00393A. URL <https://doi.org/10.1039/D6DD00393A>.
- [15] Alexander New, Michael Pekala, Elizabeth A. Pogue, Nam Q. Le, Janna Domenico, Christine D. Piatko, and Christopher D. Stiles. Evaluating the diversity and utility of materials proposed by generative models. arXiv:2309.12323, 2023.

- [16] Can Polat, Erchin Serpedin, Mustafa Kurban, and Hasan Kurban. How far can you grow? characterizing the extrapolation frontier of graph generative models for materials science. In *Proceedings of the 32nd ACM SIGKDD Conference on Knowledge Discovery and Data Mining V.2*, pages 11821–11832, 2026. doi: 10.1145/3770855.3818872. URL <https://doi.org/10.1145/3770855.3818872>.
- [17] Anubhav Jain, Shyue Ping Ong, Geoffroy Hautier, Wei Chen, William Davidson Richards, Stephen Dacek, Shreyas Cholia, Dan Gunter, David Skinner, Gerbrand Ceder, and Kristin A. Persson. Commentary: The Materials Project: A materials genome approach to accelerating materials innovation. *APL Materials*, 1(1):011002, 2013. doi: 10.1063/1.4812323.
- [18] Ilya Loshchilov and Frank Hutter. Decoupled weight decay regularization. In *International Conference on Learning Representations*, 2019. URL <https://openreview.net/forum?id=Bkg6RiCqY7>.
- [19] Brandon Amos and J. Zico Kolter. OptNet: Differentiable optimization as a layer in neural networks. In *International Conference on Machine Learning (ICML)*, 2017.
- [20] Stephen Zhao, Rob Brekelmans, Alireza Makhzani, and Roger Baker Grosse. Probabilistic inference in language models via twisted sequential Monte Carlo. In *International Conference on Machine Learning (ICML)*, 2024.
- [21] Daniel W. Davies, Keith T. Butler, Adam J. Jackson, Jonathan M. Skelton, Kazuki Morita, and Aron Walsh. SMOCT: Semiconducting materials by analogy and chemical theory. *Journal of Open Source Software*, 4(38):1361, 2019. doi: 10.21105/joss.01361.

Supplementary Information

Reusable density control for inverse materials design with autoregressive models

Can Polat, Mustafa Kurban, Erchin Serpedin, and Hasan Kurban

S1 Materials screens and alternative controls

S1.1 Chemical screens characterize the generated composition–cell pairs

The density-5 audit separates grammar completion, composition–cell decoding and chemical screening. All 2,048 generated outputs complete the grammar (100%), while 2,031 decode to composition–cell pairs (99.17%). Chemical screening rates are evaluated among these decoded pairs and compared with 800 reference structures (Table S1); no atomic positions or relaxed structures are generated.

Because marginal screening rates evaluate distinct chemical and geometric criteria independently, individual pass fractions cannot be multiplied to infer a joint passing yield. Furthermore, because this audit utilizes a separate generative output set, it does not estimate chemically screened best-of- n candidate selection yields. Establishing thermodynamic stability and synthesizability requires full 3D atomic position generation and structural relaxation, as outlined in main Methods.

Table S1 | Generated and reference screening. The density-5 audit is separate from the selection pools. Chemical rates are marginal proportions among 2,031 decoded generated pairs or 800 references. They are not a joint pass rate. No atomic positions are generated.

Check	Generated	Reference
Grammar complete / attempted	2,048 / 2,048	—
Decoded / attempted	2,031 / 2,048	—
Charge balance	48.40%	71.50%
Electronegativity order	45.49%	64.00%
Packing	79.42%	97.75%

S1.2 Target-filtered training concentrates the output distribution

Comparing zero-shot local projection against dedicated target-filtered models highlights the fundamental trade-off between population mean accuracy and compositional coverage (Table S2). Restricting training examples to narrow density windows (± 0.5 g/cm³ around target requests) concentrates output distributions tightly, achieving a target-averaged mean-response error of 0.0563 g/cm³ compared to 0.8253 g/cm³ for projection.

However, target-filtered training causes severe dataset collapse, reducing compositional uniqueness (43.98% vs. 78.58%) and training-set novelty (5.87% vs. 94.06%). Because target-filtered models lack explicit numerical conditioning inputs and operate without decoding projections, their population mean statistics reflect a distinct generative regime from post-generation candidate selection pools.

Table S2 | Target-filtered training comparison. Each entry averages two training seeds. Error is absolute deviation of mean density (g cm^{-3}), not selected-candidate MAE. Unique and novel fractions use grammar-complete element-count tuples. Novelty is relative to the full retained training split. The filtered model has no numerical conditioning input.

Target	Arm	Mean-response error	Unique (%)	Novel (%)
3	Projection	0.2402	64.04	92.70
3	Target-filtered	0.1376	42.22	7.35
5	Projection	0.1511	78.22	94.58
5	Target-filtered	0.0328	51.86	5.93
8	Projection	1.2403	84.48	94.14
8	Target-filtered	0.0469	50.01	7.26
10	Projection	1.6697	87.57	94.81
10	Target-filtered	0.0078	31.85	2.94

S1.3 Particle selection provides a separate inference-time comparison

Sequential Monte Carlo (SMC) particle filtering provides an inference-time steering baseline by evaluating continuous Gaussian density potentials across partial particle completions. Operating with $K = 16$ particles and width parameter $\sigma = 0.5 \text{ g/cm}^3$ across 100 runs per target, particle selection achieves a seven-target mean selected error of 0.7280 g/cm^3 and a uniqueness fraction of 67.71% (Table S3).

A broader hyperparameter sweep over nine bandwidths and ten targets identifies $\sigma = 0.5 \text{ g/cm}^3$ as optimal (0.9376 g/cm^3 aggregate error across 16 runs per cell). Because SMC updates particles sequentially through importance resampling rather than drawing candidates from a static saved pool, equal particle counts ($K = 16$) and selection pool allowances ($n = 16$) reflect different computational budgets and candidate sampling dynamics.

Table S3 | SMC evaluation at fixed width. $K = 16$, $\sigma = 0.5 \text{ g cm}^{-3}$, 100 particle runs per target. Each run returns one output. Error is selected-candidate absolute density error; SD is across these runs. Uniqueness counts element-count tuples among the returned outputs. This is distinct from finite-pool resampling.

Target	Valid outputs	Error	Run SD	Unique (%)
1.5	100	0.8744	0.6716	99.0
2	100	0.7805	0.8765	95.0
3	100	0.5761	0.6683	78.0
5	100	0.3729	0.2958	51.0
6.317	100	0.5062	0.4210	46.0
8	100	0.8133	0.6220	41.0
10	100	1.1728	0.7974	64.0

S2 Checkpoint reuse across density requests

S2.1 A deployment request changes the response of a fixed checkpoint

Varying deployment density requests across fixed model parameters alters generated output distributions without requiring retraining. Reusing seed-0 checkpoints trained at densities 3, 5, and 10 g/cm³ across ten evaluation requests (1.5–20 g/cm³) demonstrates systematic, continuous response redirection (Table S4).

These population averages reflect mean densities over computable outputs, illustrating how local moment steering shifts generative trajectories across target requests. However, population response curves reflect global mean behavior and do not directly predict candidate selection accuracy in post-generation filtering.

Table S4 | Full training-target deployment matrix. Entries are mean achieved density (g cm⁻³). Each column is one seed-0 checkpoint, reused across all requests with 4,096 attempted outputs per cell. Computability and grammar completion are retained in the source-data file.

Request	Trained at 3	Trained at 5	Trained at 10
1.5	1.7079	1.7133	1.8346
2	2.2809	2.3115	2.3657
3	3.1931	3.3527	3.3733
5	4.3836	4.8388	5.0640
6.317	5.0956	5.6170	6.0045
8	6.0054	6.5377	7.3841
10	6.7703	7.6713	8.9819
13	8.1933	8.8298	11.1536
16	9.0585	9.5715	13.2221
20	9.7871	10.3973	15.4951

S2.2 Seed replication places reuse beside target-specific training

Evaluating three independent training seeds for the density-10 model alongside target-specifically trained baselines quantifies the magnitude of seed-dependent variation across retargeting regimes (Table S5). Across requests at or below 10 g/cm³, zero-shot retargeting achieves mean density errors comparable to target-specific retraining (0.6497 vs. 0.6620 g/cm³).

At higher density requests (13–20 g/cm³), response curves flatten as local budgets encounter structural limits imposed by pre-emitted prefixes. Excluding seed 0 yields a secondary lower-target error average of 0.7543 g/cm³, highlighting seed sensitivity without altering the core conclusion that zero-shot retargeting matches dedicated retraining performance.

Table S5 | All density-10 seeds and completed target-specific runs. Entries are mean achieved density (g cm^{-3}). The three seed columns reuse density-10 training. The last two columns are separately trained runs at the row request; a dash denotes a failed, unavailable run. No value is imputed.

Request	Reuse s0	Reuse s1	Reuse s2	Specific s0	Specific s1
1.5	1.8346	1.7462	1.7958	1.9021	1.7935
2	2.3657	2.3217	2.3011	2.2642	2.2677
3	3.3733	3.1886	3.3398	3.1755	3.3050
5	5.0640	4.6552	4.7855	4.9174	4.7805
6.317	6.0045	5.5385	5.6864	5.8008	5.3959
8	7.3841	6.5856	6.7891	6.7374	6.7819
10	8.9819	7.7716	7.9547	8.8884	7.7721
13	11.1536	9.2566	9.4866	—	8.7441
16	13.2221	9.8350	10.2112	9.8748	—
20	15.4951	10.4387	10.5958	10.5487	10.2607

S3 Candidate selection from projected proposals

S3.1 Projection changes the accuracy–uniqueness trade-off

Averaging selection performance across all ten density requests demonstrates how local moment projection shifts the Pareto frontier governing target accuracy and compositional uniqueness (Table S6). Under equal target weighting, projected decoding consistently achieves lower candidate errors across matching candidate allowances n .

Furthermore, local steering achieves high target accuracy at reduced selection budgets. Projected best-of-16 candidate selection (0.2872 g/cm^3 error, 37.10% uniqueness) matches the density precision of unprojected best-of-64 (0.2850 g/cm^3 error, 22.75% uniqueness) while preserving a 14.35 percentage point higher unique composition fraction.

Table S6 | All-ten-target selection aggregates. Error is mean selected-candidate absolute density error in g cm^{-3} ; uniqueness is averaged equally over the same ten targets. Protocols and uncertainty definitions are in main Methods.

Protocol	n	Error	Unique (%)
Unprojected	1	6.0781	82.35
Unprojected	4	2.5444	73.30
Unprojected	16	0.9742	49.00
Unprojected	64	0.2850	22.75
Projected	1	3.8047	79.10
Projected	2	2.2944	75.65
Projected	4	1.2998	66.60
Projected	8	0.6053	52.25
Projected	16	0.2872	37.10

S3.2 The complete target grid resolves the aggregate result

Expanding aggregate selection metrics into all 90 individual settings (40 unprojected and 50 projected target/count combinations) reveals detailed performance trends across saved candidate pools (Table S7). Across individual requests, increasing candidate pool size n systematically reduces selection error while generally decreasing the unique composition yield.

Trial standard deviations quantify within-pool sampling variation across 200 independent selection trials. Compositional uniqueness is calculated independently across 200 separate selected outputs. Reporting the complete disaggregated grid ensures full transparency across all evaluated requests without restricting analysis to summary averages.

Table S7 | Complete selection grid. Every recorded target/count setting is retained. Error and its trial SD have units g cm^{-3} . SD describes 200 trials conditional on the pool, not training uncertainty. Uniqueness uses a separate set of 200 selected outputs. U: unprojected; P: projected.

Arm	Target	n	Error	Trial SD	Unique (%)
U	1.5	1	6.9049	4.1986	84.0
U	1.5	4	3.0446	1.7858	72.5
U	1.5	16	1.4378	0.8610	52.5
U	1.5	64	0.4842	0.4093	22.5
U	2	1	7.1273	5.5590	85.0
U	2	4	2.3476	1.6773	72.0
U	2	16	1.0082	0.7077	50.5
U	2	64	0.2898	0.2845	22.0
U	3	1	5.0899	4.7368	82.5
U	3	4	1.9098	1.6034	70.0
U	3	16	0.5747	0.4497	48.0
U	3	64	0.1379	0.1491	22.5
U	5	1	4.3031	4.8121	80.5
U	5	4	1.0063	0.9897	72.0
U	5	16	0.2498	0.2435	44.5
U	5	64	0.0735	0.0588	20.5
U	6.317	1	3.9103	4.2481	81.5
U	6.317	4	0.8433	0.7296	73.0
U	6.317	16	0.2521	0.2348	45.5
U	6.317	64	0.0624	0.0661	21.5
U	8	1	3.7289	3.8089	83.5
U	8	4	1.1389	0.9265	71.5
U	8	16	0.3910	0.3577	48.0
U	8	64	0.0969	0.0891	22.0
U	10	1	4.1119	2.7798	84.0
U	10	4	1.5714	1.2083	77.0
U	10	16	0.5157	0.4214	51.5
U	10	64	0.1300	0.1304	23.0
U	13	1	5.8443	3.1853	83.5
U	13	4	2.3416	1.9292	74.5
U	13	16	0.6826	0.7193	51.5
U	13	64	0.1770	0.2156	24.0
U	16	1	7.9064	3.6335	80.5
U	16	4	4.2691	2.7853	76.0
U	16	16	1.4803	1.4109	49.5

Table S7 (continued)

Arm	Target	<i>n</i>	Error	Trial SD	Unique (%)
U	16	64	0.4484	0.4648	24.5
U	20	1	11.8541	4.2103	78.5
U	20	4	6.9718	3.7631	74.5
U	20	16	3.1494	2.5220	48.5
U	20	64	0.9502	0.9183	25.0
P	1.5	1	1.0054	1.6002	76.5
P	1.5	2	0.4781	0.4138	73.5
P	1.5	4	0.2159	0.2060	64.0
P	1.5	8	0.1060	0.1080	50.0
P	1.5	16	0.0552	0.0488	32.5
P	2	1	1.3551	2.0494	77.5
P	2	2	0.6107	0.7106	74.0
P	2	4	0.2642	0.2715	63.0
P	2	8	0.1370	0.1468	52.5
P	2	16	0.0587	0.0677	39.5
P	3	1	1.6162	1.7857	79.5
P	3	2	1.0680	1.0066	78.0
P	3	4	0.5074	0.5632	68.0
P	3	8	0.2267	0.2373	53.5
P	3	16	0.0957	0.1056	37.0
P	5	1	2.2780	2.9014	77.0
P	5	2	1.0321	1.0168	73.5
P	5	4	0.5997	0.6715	66.0
P	5	8	0.2701	0.2268	51.5
P	5	16	0.1508	0.1509	38.5
P	6.317	1	2.7138	3.1820	80.5
P	6.317	2	1.2390	1.1963	76.0
P	6.317	4	0.7736	0.7636	64.5
P	6.317	8	0.3390	0.3673	56.0
P	6.317	16	0.1394	0.1498	36.5
P	8	1	3.3141	2.6854	80.0
P	8	2	1.8798	1.5947	71.5
P	8	4	0.9231	0.8825	62.5
P	8	8	0.4685	0.4208	50.5
P	8	16	0.2421	0.1909	38.5
P	10	1	3.6959	2.7487	82.0
P	10	2	2.2690	1.9907	81.0
P	10	4	1.2412	1.1635	70.0
P	10	8	0.6616	0.6450	54.0
P	10	16	0.3123	0.2603	41.5
P	13	1	5.2461	3.6047	82.5
P	13	2	3.1458	2.5680	77.5
P	13	4	1.6425	1.6647	68.5
P	13	8	0.8380	0.9398	53.5
P	13	16	0.3774	0.3999	38.0
P	16	1	6.4890	4.2170	76.0
P	16	2	4.2946	3.5390	77.0
P	16	4	2.3911	2.1927	68.5
P	16	8	1.1078	1.0244	50.0

Table S7 (continued)

Arm	Target	n	Error	Trial SD	Unique (%)
P	16	16	0.5098	0.5065	35.0
P	20	1	10.3332	5.4648	79.5
P	20	2	6.9269	4.8369	74.5
P	20	4	4.4396	3.6944	71.0
P	20	8	1.8985	1.8470	51.0
P	20	16	0.9302	0.9889	34.0

S4 Expected-property control and individual outcomes

S4.1 Matched expectations can accompany different charge distributions

Exact enumeration of the 65,536 length-eight sequence space demonstrates that identical target expectations can emerge from fundamentally different probability distributions (Table S8). At target charge zero, the deployed projection model achieves a low total variance (1.5471) and a 38.45% exact target hit rate.

By contrast, the exact sequence-level KL tilt exhibits higher total variance (10.4520) and a lower exact zero-charge probability (12.45%, corresponding to an 87.55% violation rate). A tilt-trained transformer closely matches the exact sequence tilt (10.3619 variance, 87.51% violation rate), confirming that local step-wise moment projection concentrates probability density around target expectations more effectively than global trajectory tilting.

Table S8 | Finite-task moments at charge target zero. Exact references have no training variation. Other entries are mean $\pm$ sample SD across three saved training runs, evaluated by enumeration. Sample SD is consistently recomputed from run records. Violation is probability of nonzero total charge.

Distribution	Mean charge	Variance	Violation
Masked data (exact)	2.5164	8.5636	0.9109
Mask-only model	2.4694 ± 0.0988	8.7620 ± 0.1118	0.9083 ± 0.0023
Sequence tilt (exact)	0.0000	10.4520	0.8755
Tilt-trained model	-0.0619 ± 0.1220	10.3619 ± 0.2417	0.8751 ± 0.0029
Projection model	0.0170 ± 0.0017	1.5471 ± 0.0137	0.6155 ± 0.0098

S4.2 The target sweep connects concentration to final-step feasibility

Sweeping target charge requests across seven values on the mask-conditioned base confirms that local conditional steering consistently increases exact target hit probabilities compared to sequence-level tilts (Table S9). Across the seven target requests, the local construction achieves higher individual hit probabilities (36.17%–54.93% vs. 10.26%–28.59%). At target charge zero, the feasibility audit identifies 5,460 infeasible requests among 11,760 final-prefix nodes (46.43%). These are unweighted node counts, not probability-weighted frequencies. At target charge zero, the local analytical construction has a total mean of 0.103 and a total variance of 3.103.

These exact analytical counts demonstrate that local moment projection concentrates probability distributions around target expectations even when final conditional residuals $\delta(H)$ remain non-zero. These properties explain why local sequential steering stabilizes property expectations across autoregressive sequence generation.

Table S9 | Complete analytical charge-target sweep. The local construction and sequence-level tilt act on the same mask-conditioned base. Local mean error is absolute deviation from the requested expectation. Hit probability is probability of the exact requested charge; the tilt satisfies the expected target but need not attain it on every sequence.

Target	Local mean	Local mean error	Local hit	Tilt hit
-6	-4.9623	1.0377	0.5061	0.1651
-4	-3.3227	0.6773	0.3953	0.1848
-2	-1.6909	0.3091	0.4074	0.1026
0	0.1028	0.1028	0.3800	0.1245
2	1.8785	0.1215	0.4125	0.1116
4	3.5658	0.4342	0.3617	0.1626
6	5.3441	0.6559	0.5493	0.2859