

# Trustworthy Inverse Molecular Design via Alignment with Molecular Dynamics

Kevin Tirta Wijaya,\* Navid Ansari, Hans-Peter Seidel, and Vahid Babaei\*

Data-driven inverse molecular design (IMD) has attracted significant attention in recent years. Despite the remarkable progress, existing IMD methods lag behind in terms of *trustworthiness*, as indicated by their misalignment to the ground-truth function that models the molecular dynamics. Here, TrustMol, an IMD method built to be trustworthy is proposed by inverting a reliable molecular property predictor. TrustMol first constructs a latent space with a novel variational autoencoder (VAE) and trains an ensemble of property predictors to learn the mapping from the latent space to the property space. The training samples for the ensemble are obtained from a new reacquisition method to ensure that the samples are representative of the latent space. To generate a desired molecule, TrustMol optimizes a latent design by minimizing both the predictive error and the uncertainty quantified by the ensemble. As a result, TrustMol achieves state-of-the-art performance in terms of IMD accuracy, and more importantly, it is aligned with the ground-truth function that indicates trustworthiness.

based on the molecular dynamics. However, devising an inversion approach for the NFP is extremely challenging. The common approach, fueled by deep learning breakthroughs, is to approximate the NFP using a surrogate model, often implemented as a neural network, trained on data sampled from the NFP.

Surrogate-based IMD has emerged as a popular IMD approach. For example,<sup>[6–8]</sup> have introduced neural network-based IMD methods that generate molecules in an autoregressive manner. <sup>[9]</sup> uses an equivariant normalizing flows method that integrates equivariant graph neural network to obtain an invertible function. Diffusion-based generative models have also been proposed, operating either in the molecule space<sup>[10]</sup> or latent space.<sup>[11]</sup> These prior works have progressively improved the state-of-the-art IMD accuracy,

but have largely overlooked an equally critical aspect of IMD: *trustworthiness*.

The trustworthiness of a surrogate-based IMD method can be defined as how well it aligns with the NFP. In methods such as,<sup>[12,13]</sup> this alignment can be quantified by measuring the distance between surrogate predictions and NFP calculations. Misalignments between the surrogate and the NFP can then be categorized into two classes (**Figure 1**). First, the surrogate model identifies a molecule as a good match for the target property, but the NFP proves it to be a poor match or invalid. Second, the surrogate considers a molecule as a poor match, while the NFP would consider it as a good match. An IMD method that lacks alignment with the NFP is not effective for discovering new molecules, as the NFP serves as the ground-truth representation of the molecular dynamics of the real world.

The cause of misalignments in a surrogate-based IMD can be traced back into two root issues: **(I1)** the surrogate fails to correctly model the forward process (e.g., the mapping from a molecular design space to the property space) on the training set, and **(I2)** the surrogate becomes unreliable when operating on molecules that are completely different from the training set, a scenario that often occurs during the inversion step.

Here, we propose TrustMol, a surrogate-based IMD method that is designed to be trustworthy by addressing the two issues. To improve the forward modeling **(I1)**, TrustMol first constructs an intermediate molecular latent space using a novel VAE designed to encourage similar latents to exhibit similar properties. A neural surrogate model is then trained to map this latent space to the

## 1. Introduction

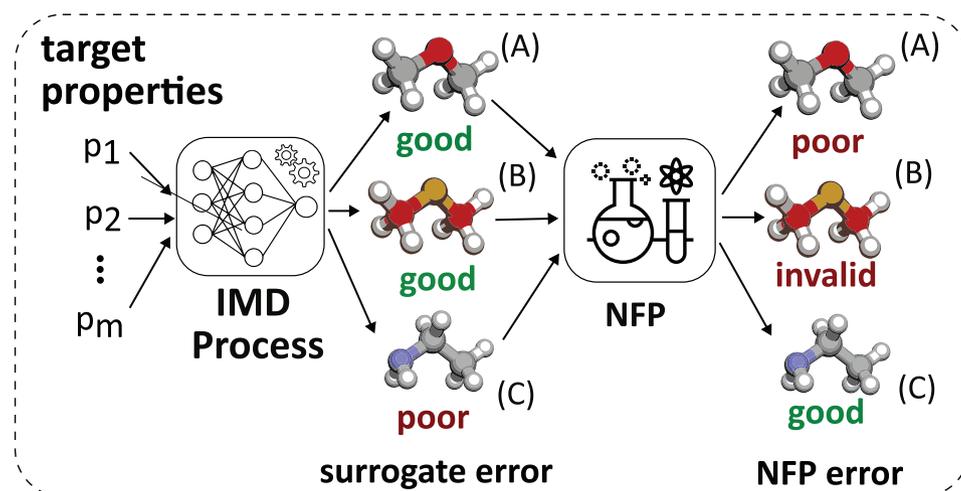
The discovery of new molecules with desired properties is the central goal of materials science and chemistry. Recent molecular discoveries have improved the performance of grid-scale<sup>[1]</sup> and high-voltage<sup>[2]</sup> batteries, and inhibit tumour drivers<sup>[3]</sup> as well as the viral replication of SARS-CoV-2,<sup>[4]</sup> to name a few. Despite its importance, traditional methods for molecular discovery are often slow and complex, relying on experts to propose, synthesize, and evaluate newly-designed molecules in an iterative manner. A promising approach to speed up the molecular discovery process is through data-driven IMD. In IMD, molecules are designed to exhibit a target property. This should ideally be carried out through inverting the *native forward process* (NFP)<sup>[5]</sup>—the ground-truth function that maps molecules to their properties

K. T. Wijaya, N. Ansari, H.-P. Seidel, V. Babaei  
Max Planck Institute for Informatics  
Saarland Informatic Campus  
66123 Saarbruecken, Germany  
E-mail: kwijaya@mpi-inf.mpg.de; vbabaei@mpi-inf.mpg.de

 The ORCID identification number(s) for the author(s) of this article can be found under <https://doi.org/10.1002/advs.202416356>

© 2025 The Author(s). Advanced Science published by Wiley-VCH GmbH. This is an open access article under the terms of the [Creative Commons Attribution](#) License, which permits use, distribution and reproduction in any medium, provided the original work is properly cited.

DOI: 10.1002/advs.202416356



**Figure 1.** Surrogate-NFP misalignment. The IMD's surrogate predicts that A and B are good matches (low surrogate error) for the given target properties, while C is a poor match. However, when put through the NFP, A and B are in fact poor matches (high NFP error), while C is a good match.

property space. This approach allows the neural network to better model the forward process as the latent-to-property mapping is inherently smoother and lower-frequency than the molecule-to-property mapping.

We also introduce a latent-property pairs *reacquisition* method to improve the quality of the training stage of the surrogate model. This reacquisition method ensures that the selected latent-property pairs accurately represent the latent space, thereby enhancing the ability of the surrogate to correctly model the forward process (I1).

After obtaining an accurate model of the forward process, TrustMol can generate molecules with desired properties by inverting the surrogate model: a randomly initialized molecular latent is iteratively adjusted to minimize the distance between predicted and target properties. However, an unconstrained optimization may yield a molecular latent that is completely different to the training samples (I2), compromising the reliability of the surrogate model predictions. To address this issue, we integrate *epistemic* uncertainty quantification into the optimization, guiding the surrogate to explore novel molecular latents that remain close enough to the training data.

We evaluate TrustMol against several state-of-the-art IMD baselines using two key metrics. The first metric, NFP error, measures the distance between the target properties and the actual properties of the molecule as computed by the NFP. The second, NFP-surrogate misalignment, quantifies the gap between the property predictions of the surrogate and those of the NFP. TrustMol consistently outperforms the baselines across both metrics, achieving state-of-the-art performance in both single-objective and multi-objective IMD tasks.

The source code for TrustMol, which will be made public, is available at <https://github.com/ktirta/TrustMol>. We also provide an interactive tool (Appendix C) with precomputed results to demonstrate the potential of the IMD process of TrustMol. We encourage the readers to experiment with this tool at: <https://repo012424.streamlit.app/>.

### 1.1. TrustMol Framework

TrustMol adopts the surrogate-based IMD approach to generate a molecule with a desired property, denoted by  $p$ . A straightforward implementation of this approach involves using a property prediction surrogate,  $\Phi$ , to optimize a randomly-initialized molecular design,  $x$ , by solving:

$$x^* = \arg \min_x |p - \Phi(x)| \quad (1)$$

where  $x^*$  is the optimal molecular design. Unfortunately, this approach is often misaligned to the NFP, where the optimal molecules found by the surrogate are deemed as poor matches or invalids by the NFP.

As mentioned in Section 1, the root issues of misalignments lie on the forward modeling (I1) and the inversion (I2). Consider the IMD approach in Equation (1). The mapping of molecular structures to their corresponding properties is inherently high-frequency, where small changes in structures can lead to significant changes in properties. This presents a challenge for neural networks, which tend to struggle to model high-frequency functions,<sup>[14,15]</sup> resulting in issue I1. Furthermore, not all molecular designs are valid; many molecular configurations are unstable and therefore invalid. Since the surrogate model operates over continuous input-output domains, it may generate molecules that appear valid to the surrogate but are, in fact, chemically invalid, further exacerbating issue I1. Finally, the optimization in Equation (1) is unconstrained and can result in a molecular design that differ significantly from molecules in the training set, which ultimately leads to issue I2.

To address these challenges, we introduce three novel components: SGP-VAE (I1), latent-property pair reacquisition (I1), and uncertainty-aware molecular latent optimization (I2), described in detail in the following subsections.

### 1.1.1. Molecular Latent Optimization with SGP-VAE

We propose to perform the optimization in a latent space (Figure 2a (right) and 2b) to tackle the high-frequency and discontinuous nature of the molecule space, thus improving the forward modeling capability of the surrogate and addressing issue I1. The latent space is learned by TrustMol through a VAE<sup>[16]</sup> that is trained to reconstruct molecular representations from latent vectors. Our distinct contribution here is to incorporate three sources of information, molecular strings, molecular 3D structures, and molecular properties information, into our novel SELFIES-Graph-Property (SGP) VAE.

We employ SELFIES<sup>[17]</sup> as the primary representation of the VAE to ensure that any latent vector can be decoded into a valid molecule, effectively addressing the discontinuity issue. However, similarities between molecular strings do not always correspond to similar properties. For instance, the SMILES strings 'C\1 = C\CC / 1' and 'C\1 = C(\C)C/1' have a low Levenshtein Distance, yet exhibit a significant difference in dipole moment (0.708 D,<sup>[18]</sup>). Therefore, solely relying on SELFIES reconstruction might be insufficient to address the high-frequency problem.

We augment the VAE training with two auxiliary tasks: predicting properties directly from the latent vectors and reconstructing 3D molecular graphs. Learning latent-to-property predictions can organize the latent space with respect to property values,<sup>[12]</sup> while 3D structural information is a useful indicator of similarity in property space.<sup>[19]</sup> With the three training objectives, our SGP-VAE can learn a latent space in which similar latents are more likely to correspond to molecules with similar properties. As a result of the smoother mapping, the quality of the forward modeling is improved.

### 1.1.2. Latent-Property Pairs Reacquisition

While the function that maps latents to properties is smoother, existing work<sup>[13]</sup> suggests that the quality of a latent-to-property surrogate is lower than a molecule-to-property surrogate, indicating that the latent-to-property surrogate fails to accurately model the forward process (I1). This phenomenon arises from how the latent-to-property surrogate is trained. Given a VAE encoder  $\Psi^{\text{enc}}$  that has been pretrained on a dataset  $\mathbb{D} = \{(m_i, p_i^{\text{gt}})\}$  where  $m_i$  is the  $i$ -th molecule and  $p_i^{\text{gt}}$  is its corresponding property, the common approach to train a latent-to-property surrogate  $\Phi$  parameterized by  $\phi$  is,

$$\mathbb{Z} = \{\Psi^{\text{enc}}(m_i)\} = \{z_i\} \quad (2)$$

$$\phi^* = \arg \min_{\phi} |p_i^{\text{gt}} - \Phi(z_i)| \quad (3)$$

where  $i = 1, \dots, |\mathbb{D}|$ ,  $\phi^*$  is the optimal parameter of  $\Phi$  and  $z_i$  is the latent representation of  $m_i$ . One problem in this approach is that there are molecules in  $\mathbb{D}$  that cannot be well-represented by the latent vectors. Encoding such molecules with  $\Psi^{\text{enc}}$  will produce valid latent vectors, but decoding them back with a pretrained VAE decoder  $\Psi^{\text{dec}}$  will result in incorrect molecules due to non-zero reconstruction errors associated with training any VAE. Training  $\Phi$  to predict the properties of the latents of these molecules would result in an unreliable surrogate, as the decod-

ing process produces incorrect molecules that could have significantly different properties. Another problem is that  $\mathbb{Z}$  only contains latents of molecules that is part of  $\mathbb{D}$ . There are other latents that are not in  $\mathbb{D}$ , but can nonetheless be decoded back into valid molecules. Limiting the training samples to  $\mathbb{Z}$  will therefore limit the diversity of samples seen by the surrogate model.

Here, we propose a latent-property pairs reacquisition method to collect representative training samples for the surrogate. Utilizing a trained decoder  $\Psi^{\text{dec}}$  alongside a conformer generator  $h$  (RDKit,<sup>[20]</sup>) and the NFP  $f$  (Psi4<sup>[21]</sup>), we generate the new dataset  $\mathbb{D}_{\text{new}}$  of latent-property pairs for training the surrogate  $\Phi$  according to the following steps. First, latent representations  $z$  are randomly sampled from a Gaussian  $\mathcal{N}(\mu, \sigma)$  with mean  $\mu$  and variance  $\sigma$ ,

$$\mathbb{Z}_{\text{new}} = \{z_{\text{new},i} \mid z_{\text{new},i} \sim \mathcal{N}(0, 1)\}_{i=1}^N \quad (4)$$

The properties of the molecules represented by the sampled latents are then calculated by decoding the latents back into molecules using  $\Psi^{\text{dec}}$ , generating the corresponding 3D conformations using  $h$ , and passing the conformations to  $f$ , before collecting the latent-property pairs into one dataset,

$$\mathbb{P}_{\text{new}} = \{f(h(\Psi^{\text{dec}}(z_{\text{new},i}))) \mid \forall z_{\text{new},i} \in \mathbb{Z}_{\text{new}}\} \quad (5)$$

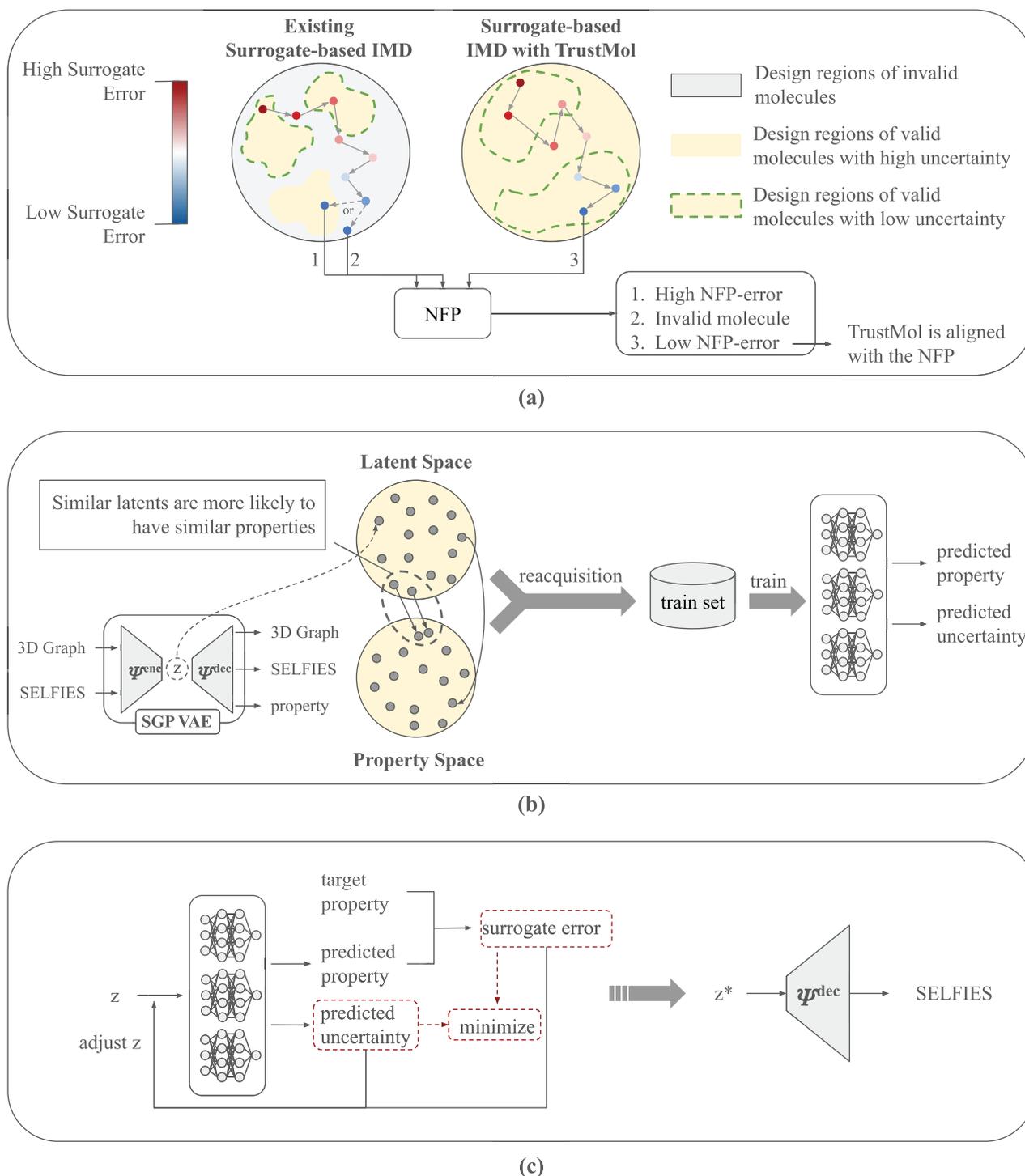
$$\mathbb{D}_{\text{new}} = \{(z_{\text{new},i}, p_{\text{new},i}) \mid \forall z_{\text{new},i} \in \mathbb{Z}_{\text{new}} \text{ and } \forall p_{\text{new},i} \in \mathbb{P}_{\text{new}}\} \quad (6)$$

This simple latent-property pairs reacquisition method, illustrated in Figure 2b, proves to be highly effective in improving the inverse design performance of TrustMol, as we discuss in Appendix B.1.

It is important to note that, in terms of training data size, comparing TrustMol with other baselines in Section 3 remains fair even without applying the reacquisition method to those baselines. This is because we only reacquire around 10K latent-property pairs to train the surrogate model. In contrast, other baselines leverage large datasets, such as QM9, which provides around 100K training samples for training their surrogate models. Given that the number of training samples for our surrogate model is considerably smaller than those available in large datasets, the observed performance improvements are not due to reacquisition acting as a data augmentation method.

### 1.1.3. Uncertainty-Aware Molecular Latent Optimization

A neural network is most reliable when performing prediction on samples from regions that are well-represented during training. Guiding the molecular latent optimization to find novel molecules in these familiar regions would then be the key to address issue I2. In TrustMol, we incorporate *epistemic* uncertainty into the molecular latent optimization<sup>[5]</sup> to guide the optimization into regions where the surrogate is most reliable. Epistemic uncertainty can be seen as the measure of data sparsity within a region that is available during training. Therefore, minimizing epistemic uncertainty is equivalent to guiding the optimization toward molecular latents that are novel, but not completely different from latents that are available in the training dataset.



**Figure 2.** The framework of TrustMol. a) Existing surrogate-based IMD often finds solutions in high-uncertainty regions that are far away from the training distribution, in which the surrogate predictions are most unreliable. This could lead to molecules that are invalid or have high NFP-error. TrustMol directs the IMD process into low-uncertainty regions where the surrogate can be trusted. b) Improvement in the forward modeling comes from the SGP-VAE, which encourages similar latents to exhibit similar properties. Moreover, the surrogate model is trained with latent-property pairs that are representative of the learned latent space. c) During inversion, TrustMol optimizes a latent design by minimizing the predicted surrogate error and the epistemic uncertainty. The optimal latent design will then be decoded back into SELFIES by the pretrained SGP-VAE decoder.

The epistemic uncertainty can be quantified by measuring the predictive disagreement between accurate and diverse neural networks.<sup>[22]</sup> Here, we define our surrogate model as an ensemble of  $n$  multilayer perceptrons (MLPs) with identical number of layers but different activation functions. The surrogate model is trained to fit the NFP, i.e.,  $\{\Phi_j | \Phi_j : z \mapsto \hat{p}\}, j = 1, \dots, n$ . Given the mean prediction  $\Phi^{\text{avg}}(z) = \frac{1}{N} \sum_{j=1}^N \Phi_j(z)$ , the epistemic uncertainty ( $U$ ) can be defined as,

$$U(z) = \frac{1}{N} \sum_{j=1}^N (\Phi_j(z))^2 - (\Phi^{\text{avg}}(z))^2 \quad (7)$$

The final uncertainty-aware IMD process of TrustMol (Figure 2c) obtains the optimal molecular latent  $z^*$  through gradient descent,

$$z^* = \arg \min_z |\Phi^{\text{avg}}(z) - p| + U(z) \quad (8)$$

## 2. Conclusion

We introduced TrustMol, a molecular latent optimization method that focuses on aligning with the NFP for a trustworthy IMD. TrustMol not only demonstrates superior performance over existing IMD methods in accuracy, but also excels in trustworthiness, as indicated by the low disagreement with the NFP. Another key benefit of TrustMol is the ease of incorporating additional constraints into the IMD process, such as favoring molecules with lower mass (Appendix B.3). This flexibility presents an opportunity to tune the IMD process to better meet the practical requirements of the end users, e.g., chemists. The effectiveness of TrustMol, however, is limited by the expressiveness of the latent space and the reliability of the surrogate model. Therefore, improving the latent space construction and the surrogation is crucial for a highly performant IMD. A promising path toward this goal is to explore the latent space further with active learning.<sup>[23]</sup> We note that our uncertainty-aware molecular latent optimization is closely related to Bayesian optimization (BO).<sup>[24]</sup> However, TrustMol follows an offline model-based optimization approach<sup>[25]</sup> and does not assume access to the NFP during the optimization, whereas BO requires frequent back and forth with the NFP (i.e., density functional theory (DFT)).

## 3. Experimental Section

**Single-Objective Inverse Molecular Design:** In single-objective IMD, the property of interest was set as either HOMO, LUMO, or dipole moment, as these three properties could be calculated using the calculated using the DFT as the NFP with relatively high accuracy.<sup>[26,27]</sup> The *target* property values were defined as a set of  $n = 2000$  evenly-spaced values within a specified range  $[a, b]$  that covers both property values present in and absent from the training dataset. The ranges were set to  $[-10, 0]$  for HOMO,  $[-4, 2]$  for LUMO, and  $[0, 4]$  for dipole moment. Each IMD method had a budget of  $k = 10$  tries to generate a molecule for each target property value, and only the molecule exhibiting the lowest absolute error was retained.

Due to compute limitation,  $n = 20$  was set for JANUS and omit its novelty and uniqueness metrics to ensure fairness with other methods that generate significantly more molecules. When using 2,000 CPU threads on AMD EPYC 7702 processors, the DFT-

**Table 1.** Experimental results for single-objective IMD (HOMO, LUMO, or Dipole Moment). We report the mean and standard deviation over three runs. The unit of measurements is electronvolt (eV) for HOMO and LUMO, and Debye (D) for Dipole Moment. We also report the novelty, uniqueness, and latency metrics for all methods, where the batch-latency is evaluated when generating 2000 molecules in parallel. Bolded values indicate the best performance on the column.

Model	NFP Error			Nov. (%)	Uni. (%)	Latency (s)	
	H (eV)	L (eV)	D (D)			single	batch
JANUS <sup>[28]</sup>	3.29	0.80	0.90	—	—	7113	—
GeoLDM <sup>[11]</sup>	1.16 $\pm$ 0.03	0.39 $\pm$ 0.02	0.56 $\pm$ 0.03	81.06	<b>94.26</b>	8.67	1617
SELFIES LDM	0.97 $\pm$ 0.01	0.33 $\pm$ 0.04	0.95 $\pm$ 0.02	82.28	48.20	0.64	<b>0.77</b>
MGCVAE <sup>[29]</sup>	1.65 $\pm$ 0.03	0.30 $\pm$ 0.01	0.44 $\pm$ 0.02	<b>90.17</b>	85.97	<b>0.33</b>	6.55
SELFIES VAE <sup>[12]</sup>	3.75 $\pm$ 0.29	1.99 $\pm$ 0.20	4.98 $\pm$ 0.04	21.26	7.82	8.57	—
LIMO <sup>[13]</sup>	1.23 $\pm$ 0.18	0.35 $\pm$ 0.14	0.59 $\pm$ 0.08	87.80	21.30	4.12	7.80
LIMO on $z$ <sup>[13]</sup>	1.31 $\pm$ 0.21	0.49 $\pm$ 0.17	0.82 $\pm$ 0.12	81.87	43.26	3.98	6.75
TrustMol (ours)	<b>0.95<math>\pm</math>0.06</b>	<b>0.25<math>\pm</math>0.01</b>	<b>0.40<math>\pm</math>0.02</b>	87.70	88.0	7.62	11.53

based molecular property calculation of 20K (i.e.,  $n \cdot k$ ) molecules takes around 6 h to complete.

Four metrics were employed to evaluate the methods. The **NFP Error** was the mean absolute errors between the NFP-calculated properties of the generated molecules and the target properties. The NFP error helps evaluating the designed molecules in a more real-world setup. This contrasts previous works<sup>[10,11]</sup> that use the surrogate error by letting neural networks to predict the properties of the generated molecules. **Novelty** and **uniqueness** were used to measure the diversity of the generated molecular designs, with novelty representing the number of designs not present in the QM9 dataset,<sup>[30]</sup> and uniqueness representing the number of unique designs generated. **Latency** was measured in two ways: *single*, the time to generate one molecule individually, and *batch*, the total time to generate multiple molecules in parallel. Molecule stability metric was omitted because all methods except GeoLDM successfully generate at least one valid molecule for all targets.

As shown in **Table 1**, TrustMol outperforms all methods by a substantial margin in all three target property categories. These results demonstrated that improving explainability through a neural surrogate-based latent optimization approach did not compromise IMD accuracy. All methods also displayed high novelty, indicating the effectiveness of both denoising and property prediction networks for discovering novel molecules. However, existing optimization-based IMD methods tend to produce identical molecules, as reflected by their uniqueness. In contrast, TrustMol attains a high score for uniqueness that was competitive with state-of-the-art diffusion model, GeoLDM. The high uniqueness score could be attributed to the improved surrogate model of TrustMol, which, due to the latent-property pairs reacquisition, had been trained on a more diverse set of latent vectors, enabling it to navigate toward more diverse latent solutions during optimization. Similar to other optimization-based approaches, TrustMol could generate molecules within reasonable time frame, especially when compared to GeoLDM in batch generation setup where the latency of TrustMol was two orders of magnitude smaller.

**Multi-Objective Inverse Molecular Design:** While single-objective IMD had been commonly used in previous studies,<sup>[10,11]</sup>

**Table 2.** NFP-surrogate error misalignment comparison between TrustMol and other models, and experimental results for multi-objective IMD. The NFP-surrogate misalignment is defined as the absolute difference between the NFP error and the surrogate error. Note that some methods cannot predict the surrogate errors.

Model	NFP-Surrogate Misalignment			Multi-Objective NFP Error		
	H (eV)	L (eV)	D (D)	H (eV)	L (eV)	D (D)
JANUS	3.32	1.11	1.56	2.46	1.33	1.07
LIMO	1.01 $\pm$ 0.07	0.54 $\pm$ 0.06	1.36 $\pm$ 0.32	0.85 $\pm$ 0.05	1.02 $\pm$ 0.05	1.17 $\pm$ 0.11
MGCVAE	—	—	—	2.26 $\pm$ 0.02	0.71 $\pm$ 0.01	3.76 $\pm$ 0.01
SELFIES VAE	3.75 $\pm$ 0.29	1.99 $\pm$ 0.20	4.98 $\pm$ 0.04	3.26 $\pm$ 0.26	1.70 $\pm$ 0.17	1.96 $\pm$ 0.02
TrustMol (ours)	<b>0.89<math>\pm</math>0.13</b>	<b>0.25<math>\pm</math>0.01</b>	<b>0.40<math>\pm</math>0.02</b>	<b>0.62<math>\pm</math>0.03</b>	<b>0.63<math>\pm</math>0.02</b>	<b>0.79<math>\pm</math>0.03</b>

real-world applications often were more interested in multi-objective IMD. Therefore, an analysis of multi-objective IMD performance of TrustMol and other IMD methods were provided. In this comparison, the IMD methods were tasked with generating molecular designs that simultaneously exhibit specific values of HOMO, LUMO, and dipole moment. The target ranges of HOMO to [-8, -3] and of LUMO to [-3, 2] were set to avoid scenarios where the target HOMO value was lower than the target LUMO value.

The multi-objective NFP error of each property in **Table 2** was shown. The hypervolume metric of the Pareto front in Appendix B.4 was also provided and visualized as an aggregate metric for multi-objective IMD. As shown in **Table 2**, simultaneously optimizing for multiple properties tends to reduce the accuracy of IMD methods. Nevertheless, TrustMol manages to minimize the deterioration of its IMD accuracy, significantly outperforming others in all property categories. The superior performance of TrustMol could be attributed to the synergy of the uncertainty-aware optimization and latent-property pairs reacquisition for training the surrogate model.

**Measuring Surrogate-NFP Alignment:** For a neural surrogate-based IMD method to be considered reliable, it should demonstrate a reasonable alignment between its surrogate and the NFP. This alignment could be evaluated by comparing the IMD errors as predicted by the surrogate (surrogate error) and those calculated by the NFP (NFP error). In the unlikely event when a surrogate-based IMD method was perfectly aligned with the NFP, the gap between the NFP and surrogate errors, i.e., the NFP-surrogate misalignment, would be zero.

**Table 2** shows the NFP-surrogate misalignment of several IMD methods. It could be seen that the misalignments of other surrogate-based IMD methods were relatively high. On the other hand, TrustMol achieved lower NFP-surrogate misalignment across all three property categories. These results validate the hypothesis that incorporating epistemic uncertainty into the optimization process could effectively reduce the NFP-surrogate misalignment, resulting in a more trustworthy IMD method. An additional analysis on epistemic uncertainty quantification was provided in Appendix B.5.

**Verifying High-Frequency and Discontinuous Nature of the Molecule Space:** In earlier sections, the high-frequency and discontinuous nature of the mapping from molecular space to property space had been discussed, which had motivated to choose molecular latents as the design representation. To validate the

**Table 3.** Effects of small perturbations on stability and property values. We randomly add  $\mathcal{N}(0, 0.1)$  noise to an atom coordinate or a latent's component, and randomly change an atom type or a SELFIES' alphabet. We show the NFP errors between the original and perturbed molecules' properties.

Perturbation	Stable (%)	NFP Error		
		H (eV)	L (eV)	D (D)
On				
Graph - 3D coord.	38.5	1.59	1.79	0.53
Graph - atom type	38.0	1.48	1.44	0.41
SELFIES	60.0	0.86	1.16	0.47
Latent	<b>67.2</b>	<b>0.42</b>	<b>0.47</b>	<b>0.24</b>

design choices, the impact of minimal noise injections on various molecular design representations was analyzed with respect to their molecular properties.

**Table 3** shows the mean absolute error (MAE) between properties of the original and the noise-perturbed molecular designs. When noise from a  $\mathcal{N}(0, 0.1)$  distribution was injected into a randomly-chosen atom coordinate of a 3D graph, the proportion of stable molecules drastically decreases to 38.5%. Additionally, the properties of the remaining stable molecules changes significantly, as indicated by the relatively high MAE values. The same trend could be seen when the perturbation targets atom types of the 3D graphs, in which a single atom type into another was randomly changed. Interestingly, utilizing SELFIES as molecular representations could improve robustness to such perturbations. For instance, replacing a randomly-selected alphabet in a SELFIES string with another valid alphabet only reduces the stability to 60.0%, while the MAEs between the original and perturbed molecular designs show improvements. Note that while SELFIES strings could always be translated into a stable molecule, the NFP that was used to generate the corresponding 3D conformation might not always converge due to the complexity of the molecule, which flags the molecule as unstable in our evaluation.

Finally, it could be seen that latent representations of molecules exhibit the greatest robustness toward perturbations. When a  $\mathcal{N}(0, 0.1)$  noise was injected into the latents, the proportion of stable molecules remains high at 67.2%, and the MAE between the properties of the original and perturbed molecules was approximately 45% lower in average than that observed with SELFIES strings. These results validate the explanations regarding the high-frequency and discontinuous nature of the molecule-property mapping, and support our strategy of developing a custom latent space to smooth this mapping.

## 4. Methods

### 4.1. Dataset and Molecular Properties

We use the QM9 dataset<sup>[30]</sup> as our initial training dataset  $\mathbb{D}$  for the SGP-VAE. QM9 is a quantum chemistry dataset that consists of around 130K small molecules. Each molecule is represented at atomic-level, i.e., atom types and their corresponding 3D coordinates. The molecules contains up to nine heavy atoms (C, N, O, F), and up to 29 atoms when including the Hydrogens. QM9 also provides various molecular properties including dipole moment, isotropic polarizability, Highest Occupied Molecular Orbital (HOMO), Lowest Unoccupied Molecular Orbital (LUMO), thermal capacity, among others.

In our experiments, we use HOMO, LUMO, and dipole moment as the potential target properties of the inversion. The gap between HOMO and LUMO can be used to predict the stability of a compound. Dipole moment, on the other hand, is a measure of a molecule's polarity, which in turn can be used to predict various physical properties such as solubility in water and boiling point.

## 4.2. Implementation Details

We implement all neural networks with PyTorch.<sup>[31]</sup> AdamW optimizer<sup>[32]</sup> and cosine annealing learning rate scheduler<sup>[33]</sup> are used in the optimization process for all models. We train the SGP-VAE for 50 epochs and the ensemble surrogate model for 300 epochs, with a batch size of 32. To improve diversity of the ensemble surrogate model, at each iteration, a subnetwork  $\Phi_i$  in the ensemble has a probability of only  $q = 0.3$  to perform a gradient descent step. This is equivalent to independently training each subnetwork for 90 epochs with different random seeds.

We use RDKit<sup>[20]</sup> and Psi4<sup>[21]</sup> as the NFP, the ground truth functions that model the behavior of molecules in real-world. RDKit is an open-source cheminformatics and machine learning software that can perform analysis on chemical structures. We use RDKit to generate the molecular conformation, i.e., the spatial arrangement of atoms in a molecule, of the SELFIES strings generated by LIMO<sup>[13]</sup> and TrustMol. Psi4 is an open-source quantum chemistry software that is capable of accurately predicting the properties of a molecular conformation using DFT. We use Psi4 to calculate the HOMO, LUMO, and dipole moment values of molecular conformations generated by the IMD methods.

## 4.3. Loss Function of the SGP-VAE

Our SGP-VAE architecture features an encoder  $\Psi^{\text{enc}}$  that takes as inputs the multiview representations of a molecule,  $\mathbf{x}_{\text{selfies}}$  and  $\mathbf{x}_{\text{graph}}$ . The graph representation is processed with a graph neural network (EGNN,<sup>[34]</sup>) before being fused with features from the SELFIES representation into a latent vector  $\mathbf{z}$ . During training, the VAE's decoder  $\Psi^{\text{dec}}$  reconstructs both SELFIES and graph representations and predict the properties of the molecule directly from its latent. The loss is calculated as follows,

$$\mathcal{L} = |p_x - \hat{p}_x| + \|\mathbf{x}_{\text{graph}} - \hat{\mathbf{x}}_{\text{graph}}\|_2^2 + \text{CE}(\mathbf{x}_{\text{selfies}}, \hat{\mathbf{x}}_{\text{selfies}}) + \text{KL}(\mathbf{z} || \mathcal{N}(0, 1)) \quad (9)$$

where CE and KL are cross-entropy and KL-divergence<sup>[35]</sup> loss functions, respectively. We verify the benefits of the multi-objective VAE pretraining in an ablation study in Appendix B. Note that the graph reconstruction in our VAE pretraining is non-equivariant, and we provide a detailed explanation and experiments on this design choice in Appendix B.2.

## Appendix A: Additional Details

### A.1. Related Work

One of the earliest computational methods for molecular design<sup>[36]</sup> proposed the Quantitative Structure-Activity Relationships (QSAR) method

for analyzing the biochemical structure-activity problems based on a regression model. Since then, numerous methods have been proposed, ranging from improvements of the QSAR method<sup>[37,38]</sup> to genetic algorithms.<sup>[39–41]</sup> In recent years, deep learning has started to influence the field. For instance, prior works have proposed models that can generate molecules in an autoregressive manner<sup>[6–8]</sup> E-NF<sup>[9]</sup> proposes an equivariant normalizing flows method that integrate an equivariant graph neural network as a differential equation to obtain an invertible function. Generative diffusion models have also been proposed, operating in either molecule space (EDM,<sup>[10]</sup>) or a learned latent space (GeoLDM,<sup>[11]</sup>).

To generate molecules with desired properties, a pretrained molecular generative model can be finetuned on a subset of the dataset in which the molecules exhibit the desired properties<sup>[6,8]</sup> While straightforward, this strategy comes at a cost of the controllability of the IMD process. For example, it is challenging to use this strategy to generate a molecule that exhibits a HOMO-LUMO gap of exactly  $x$  eV. Another approach to conditionally generate molecules is by using the property values of the molecules as additional training inputs to the denoising network of the EDM<sup>[10]</sup> or GeoLDM model.<sup>[11]</sup> Similarly, MGCVAE<sup>[29]</sup> utilizes a conditional VAE<sup>[16]</sup> to learn latent representations of 2D molecular graphs and molecular properties, which can be used to decode new latents to generate novel molecules.

Another angle to our IMD problem is from the lens of multi-objective optimization (MOO), which aims to optimize a molecular design toward the Pareto Front, e.g., maximizing property A while minimizing property B. Recently introduced methods in this area include GFlowNet,<sup>[42]</sup> a policy-based generative method based on flow networks, JANUS,<sup>[28]</sup> a parallel tempering-like genetic algorithm augmented with deep neural network, and REINVENT4,<sup>[43]</sup> an improvement of REINVENT<sup>[44]</sup> with combined reinforcement/curriculum learning and transformer models.

Closest to our approach are the SMILES VAE<sup>[12]</sup> and LIMO<sup>[13]</sup> (LIMO), which invert a surrogate model to directly optimize a randomly-initialized molecular latent. SMILES VAE utilizes a Gaussian process as the surrogate model to predict the properties of the latents, while LIMO opts for a more scalable neural network surrogate. Our distinct contributions with respect to these methods are improving latent space construction method, accounting for the VAE reconstruction errors when training the surrogate, and integrating uncertainty-awareness during the inversion.

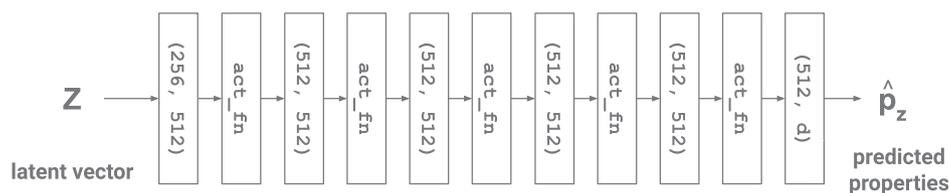
### A.2. Baselines

We compare TrustMol against a wide range of IMD methods with publicly available source codes: GeoLDM,<sup>[11]</sup> LIMO,<sup>[13]</sup> MGCVAE,<sup>[29]</sup> JANUS,<sup>[28]</sup> and SELFIES VAE, an adaptation of SMILES VAE.<sup>[12]</sup> GeoLDM utilizes an equivariant latent diffusion model to conditionally generate molecules based on the target properties. LIMO and SELFIES VAE invert property prediction surrogate models to directly optimize a randomly-initialized molecular latent. MGCVAE utilizes a conditional VAE<sup>[16]</sup> to learn latent representations of 2D molecular graphs and molecular properties, which can be used to decode new latents to generate novel molecules. Meanwhile, JANUS employs a parallel tempering-like genetic algorithm augmented with deep neural network

Additionally, we compare TrustMol with LIMO on  $\mathbf{z}$ , a variant of LIMO where its surrogate model is trained to predict properties directly from the latent vectors. This diverse set of comparisons enables us to demonstrate the advantages of TrustMol not only over other surrogate-based molecular optimization methods, but also against deep generative models and traditional methods such as genetic algorithm.

### A.3. Architecture Details for SGP-VAE

The encoder of SGP-VAE is realized with an EGNN<sup>[34]</sup> block to process the molecular 3D graph and an MLP block to process the SELFIES string. The EGNN block consists of 3 EGNN layers, each with 192 hidden features. The MLP block for SELFIES is constructed with 4 nn.Linear layers of {64, 128, 256, 256} hidden features and SILU activation function. We encode



**Figure A1.** The architecture for the latent-to-property subnetwork. A  $(x, y)$  block represents an nn.Linear layer with an input dimensionality of  $x$  and an output dimensionality of  $y$ .

the features extracted by the EGNN block for 3D graph and the MLP block for SELFIES into a latent vector with another MLP block similar to the one used for SELFIES. We set the size of latent vector to 256. The decoders of SGP-VAE are realized with MLPs. There are three decoders in total: one for reconstructing 3D graph with 5 nn.Linear layers of  $\{512, 512, 1024, 1024, m \times c\}$  hidden features, one for reconstructing SELFIES string with 7 nn.Linear layers of  $\{512, 512, 512, 512, 512, 512, n \times o\}$  hidden features, and one for predicting the molecular properties of a latent with 5 nn.Linear layers of  $\{512, 512, 256, 256, d\}$  hidden properties.

- $m$ : maximum number of atoms in a 3D graph
- $c$ : number of atom types + 3 (for 3D coordinates)
- $n$ : maximum number of alphabet in a SELFIES string
- $o$ : number of alphabet types
- $d$ : number of properties to be predicted

#### A.4. Architecture Details for Latent-to-Property Surrogate Model

The latent-to-property surrogate model is an ensemble of ten similar multi-layer perceptrons (MLPs). Specifically, each subnetwork consists of a sequence of nn.Linear layers with an activation function  $act\_fn$  in between, as illustrated in Figure A1. To promote diversity within the ensemble surrogate model, we use a different  $act\_fn$  obtained from the set {Hardswish, LeakyReLU, ReLU, SiLU, Softplus} for every two subnetworks.

## Appendix B: Additional Experimental Results

### B.1. Ablation Study for SGP-VAE and Latent-Property Pairs Reacquisition

We analyze the effects of the VAE representation and latent-property pairs reacquisition. We use HOMO as the target property of the IMD process, and calculate the mean absolute error with respect to the target property values only over molecules that are valid.

As shown in Table B1, using 3D graph as both VAE and design representations leads to the worst result. Because of the non-zero VAE reconstruction error, an optimal latent solution could be translated back into an

**Table B1.** Ablation study on the choice of VAE representation and latent-property pairs reacquisition. VAE Rep. denotes the molecular representations that are used to construct the latent space of the VAE, while Design Rep. denotes the molecular design representation that we optimize to obtain the desired molecules.

VAE Rep.	Design	Latent-Property	NFP Error	
3D Graph	SELFIES	Rep.	Pairs Reacquisition	HOMO (eV)
✓	✗	3D Graph	✗	1.893
✗	✓	SELFIES	✗	1.398
✓	✓	SELFIES	✗	1.295
✓	✓	SELFIES	✓	0.901

incorrect 3D graph. This VAE reconstruction error is exacerbated by the high-frequency and discontinuous characteristics of the molecule space, resulting in a poor IMD performance. Simply changing the representation into SELFIES helps in improving the IMD performance, an evidence that the mapping from SELFIES to property is relatively smoother. Note that 3D graph molecular designs are not always valid, as we have seen in Table 1, while SELFIES molecular designs are guaranteed to be always valid.

Motivated by the fact that structurally similar molecules tend to exhibit similar properties,<sup>[19]</sup> we incorporate both 3D Graph and SELFIES for constructing the latent space, but keep the molecular design representation to SELFIES only. In this way, we get the best of both worlds, i.e., structural similarity, smoother mapping, and always-valid molecular designs. Finally, we can eliminate the negative effects of the VAE reconstruction error by performing a latent-property pairs reacquisition and creating a new dataset to train the latent-to-property surrogate model. The utilization of this reacquisition method results in a significant jump in IMD performance, enabling TrustMol to outperform other IMD methods.

We also validate the effects of each of SGP-VAE's training objective to the overall IMD performance of TrustMol. Table B2 shows the NFP errors of TrustMol when its VAE is trained with different combinations of the three training objectives. We can see that while the NFP error with SELFIES reconstruction-only VAE is relatively low, augmenting the pretraining with property prediction and 3D graph reconstruction can further improve the IMD performance. The results confirm the necessity of a well-structured latent space for a reliable IMD process.

### B.2. Non-equivariant 3D Reconstruction Loss of the SGP-VAE

The 3D reconstruction for the VAE pretraining is not equivariant because of the projection of the EGNN's equivariant features into a 1D molecular latent vector. However, this non-equivariance is not an issue in our specific case. In QM9 dataset, one molecule is associated with only one conformation, therefore, the non-equivariant loss function in Equation (9) will never result in two different values for the same input–output pairs. Although this could limit the generalization capability of the 3D graph decoder, our objective in the VAE pretraining is not to obtain the perfect 3D graph decoder. Instead, we want to incorporate 3D structural information to organize a smooth latent space, for which the non-equivariant loss function is sufficient.

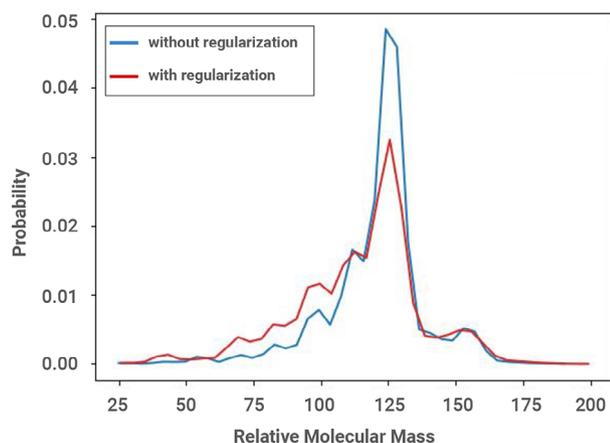
To verify this, we train another SGP-VAE with an equivariant loss function in which the ground-truth and reconstructed 3D molecular graphs are aligned through Kabsch algorithm.<sup>[45]</sup> As shown in Table B3, the additional 3D alignment with Kabsch algorithm improves the quality of 3D graph reconstruction, but at the cost of SELFIES reconstruction and prop-

**Table B2.** Ablation on VAE Training Objectives.

VAE Training Objectives			NFP Error
SELFIES Rec.	Property Prediction	3D Rec.	HOMO (eV)
✓	✗	✗	1.126
✓	✓	✗	1.020
✓	✓	✓	0.901

**Table B3.** Ablation on non-equivariant loss function for SGP-VAE pretraining. Equivariant loss helps the 3D graph reconstruction, but at the cost of SELFIES reconstruction and property prediction, resulting in lower overall IMD performance.

Model	VAE Loss at Convergence			NFP Error		
	3D Rec.	SELFIES Rec.	Prop. Pred.	H [eV]	L [eV]	D [D]
with Kabsch	4.33 $\pm$ 0.31	2.15 $\pm$ 0.05	0.61 $\pm$ 0.17	1.93 $\pm$ 0.09	2.13 $\pm$ 0.18	0.86 $\pm$ 0.02
without Kabsch	6.92 $\pm$ 0.18	0.35 $\pm$ 0.06	0.11 $\pm$ 0.03	0.95 $\pm$ 0.06	0.25 $\pm$ 0.01	0.40 $\pm$ 0.02



**Figure B1.** Additional regularizations can be easily incorporated into TrustMol. Here, we add molecular mass to the optimization objectives, penalizing molecular designs with high masses. We can see that the distribution of the generated molecular designs shifts toward molecules with lower molecular mass.

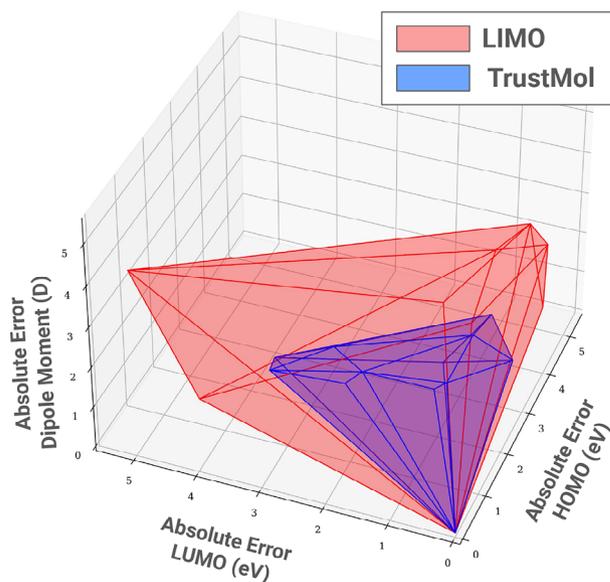
erty prediction. As a result, the overall IMD performance drops, as indicated by the increased NFP errors. This is because our molecule generation pipeline (Figure 2C) depends on the SELFIES strings instead of 3D graphs, meaning that a good SELFIES reconstruction is preferable than a good 3D graph reconstruction.

### B.3. Regularization of Inverse Molecular Design

Another advantage of TrustMol is the simplicity of adding more regularization into the optimization process. For instance, suppose that we want to find molecules with not only specific HOMO, LUMO, or Dipole Moment values, but also small molecular mass. To incorporate the new molecular mass regularization, we can train another neural network to fit the mapping from latent space to the molecular mass space, and minimizing the predicted molecular mass of the latent vectors should result in molecular designs with smaller molecular mass. Figure B1 shows the distribution of molecular designs generated with and without an additional molecular mass regularization. As we can see, the addition of the molecular mass regularization shifts the distribution toward molecular design with smaller molecular mass.

### B.4. Hypervolume of Pareto Front as an Aggregate Metric for Multi-Objective IMD

While Table 2 provides a comprehensive information regarding the multi-objective IMD performance on individual property, we are also interested in assessing the multi-objective IMD as a whole with a single metric. One candidate for such a metric is the hypervolume metric,<sup>[46]</sup> which represent the size of the space covered by the Pareto frontier.



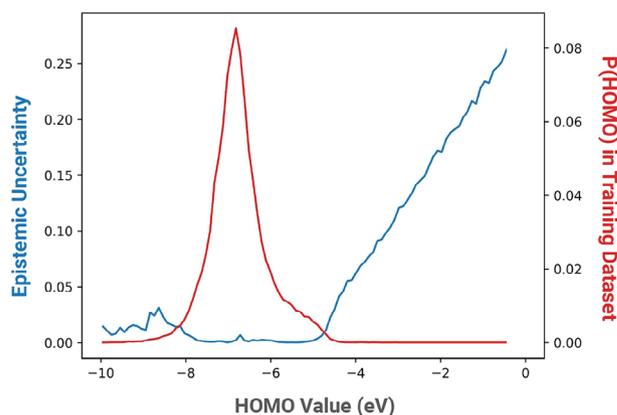
**Figure B2.** Visualization of the hypervolume of MAE for LIMO and TrustMol. We can clearly see the smaller space covered by the hypervolume of TrustMol.

We calculate the hypervolume metrics of the mean absolute errors of LIMO and TrustMol with Pymoo.<sup>[47]</sup> LIMO has a hypervolume metric of 37.117, while TrustMol has a hypervolume metric of 16.851. Since the size of the hypervolume of MAE grows with the worst-possible errors of the IMD method, we can easily conclude that TrustMol outperforms LIMO in multi-objective IMD. For clarity, we visualize the hypervolumes of LIMO and TrustMol in Figure B2.

### B.5. The Epistemic Uncertainty for Molecular Dataset

We have discussed in Section 1.1.3 how the epistemic uncertainty quantification helps with finding molecules that are novel but not completely different from those that are present in the training dataset. As a supplementary analysis, we plot the epistemic uncertainty predicted by a pretrained surrogate model along with the distribution of HOMO values available in the training dataset in Figure B3.

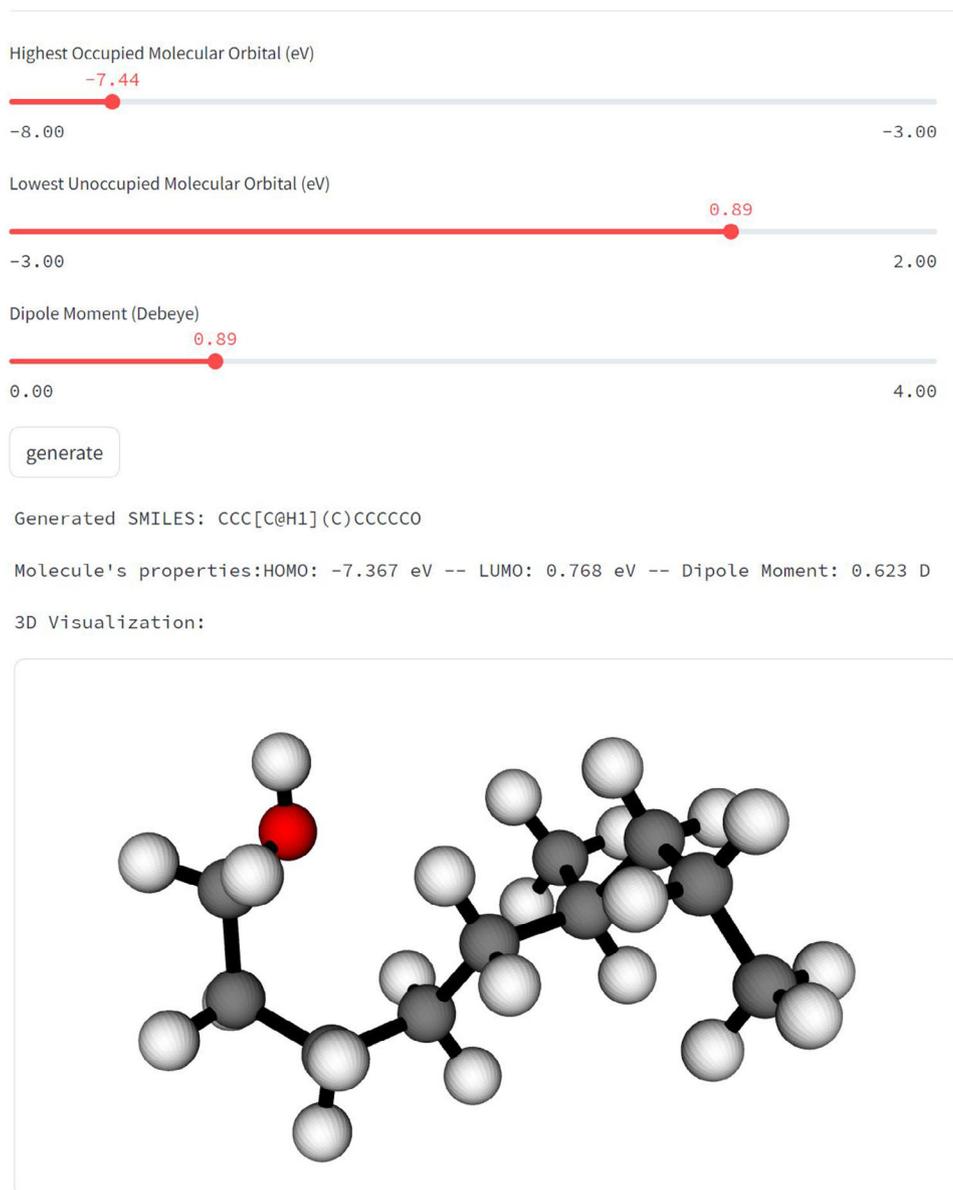
As we can see in Figure B3, the surrogate model predicts low epistemic uncertainty when the HOMO values are densely available in the training



**Figure B3.** Plot of epistemic uncertainty values predicted by a surrogate model and the distribution of HOMO values in the training dataset.

# TrustMol

## Trustworthy Inverse Molecular Design



**Figure C1.** An interactive tool for generating molecules with TrustMol.

dataset. The reverse is true when there are few-to-no samples available in the training dataset. Therefore, we can verify that epistemic uncertainty is a measure of data sparsity within a region that is available during training. Adding epistemic uncertainty as an optimization objective (Equation (8)) is then equivalent to keeping the molecular design similar to the designs found in the training dataset, which are queries of the NFP. Since a neural network typically shows an excellent performance for in-distribution data, incorporating epistemic uncertainty ultimately leads to a better alignment between the surrogate model and the NFP.

## Appendix C: Interactive Tool

To improve the accessibility of TrustMol for users from various background, we offer a web-based interactive tool that showcases the IMD process of TrustMol. The user interface is shown in **Figure C1**, and can be accessed via <https://repo012424.streamlit.app/>. It can also be run on a local device by following the instructions provided in <https://github.com/ktirta/TrustMol>. We visualize the atoms in a molecule with the following

colors: hydrogen - white, carbon - grey, nitrogen - blue, oxygen - red, fluorine - green.

Currently we use the precomputed results of the multi-objective IMD shown in Table 2 since the NFP-based evaluations are too computationally demanding. However, our ultimate objective is to fully integrate all technical components of TrustMol within this interactive tool, in line with our vision to bring IMD to its end users, i.e., chemistry and materials science practitioners.

## Acknowledgements

Open access funding enabled and organized by Projekt DEAL.

## Conflict of Interest

The authors declare no conflict of interest.

## Data Availability Statement

The data that support the findings of this study are openly available in TrustMol at <https://github.com/ktirta/TrustMol>, reference number 0. These data were derived from the following resources available in the public domain: QM9 dataset, <https://www.QM9.com> dataset; <https://doi.org/10.6084/m9.figshare.c.978904.v5>; ...;

## Keywords

alignment, inverse molecular design, trustworthiness

Received: December 6, 2024

Revised: March 4, 2025

Published online:

- [1] W. Chen, G. Li, A. Pei, Y. Li, L. Liao, H. Wang, J. Wan, Z. Liang, G. Chen, H. Zhang, J. Wang, Y. Cui, *Nat. Energy* **2018**, *3*, 428.
- [2] Q. Zheng, Y. Yamada, R. Shang, S. Ko, Y.-Y. Lee, K. Kim, E. Nakamura, A. Yamada, *Nat. Energy* **2020**, *5*, 291.
- [3] C. Yan, D. Liu, L. Li, M. F. Wempe, S. Guin, M. Khanna, J. Meier, B. Hoffman, C. Owens, C. L. Wysocky, M. D. Nitz, W. E. Knabe, M. Ahmed, D. L. Brautigam, B. M. Paschal, M. A. Schwartz, D. N. M. Jones, D. Ross, S. O. Meroueh, D. Theodorescu, *Nature* **2014**, *515*, 443.
- [4] L. Riva, S. Yuan, X. Yin, L. Martin-Sancho, N. Matsunaga, L. Pache, S. Burgstaller-Muehlbacher, P. D. De Jesus, P. Teriete, M. V. Hull, M. W. Chang, J. F. Chan, J. Cao, V. K. Poon, K. M. Herbert, K. Cheng, T. H. Nguyen, A. Rubanov, Y. Pu, C. Nguyen, A. Choi, R. Rathnasinghe, M. Schotsaert, L. Miorin, M. Dejoze, T. P. Zwaka, K. Sit, L. Martinez-Sobrido, W. Liu, K. M. White, et al., *Nature* **2020**, *586*, 113.
- [5] N. Ansari, H.-P. Seidel, N. Vahidi Ferdowsi, V. Babaei, *Adv. Neural Inf. Process. Syst.* **2022**, *35*, 8675.
- [6] S. Luo, J. Guan, J. Ma, J. Peng, *Adv. Neural Inf. Process. Syst.* **2021**, *34*, 6229.
- [7] Y. Luo, S. Ji, in *International Conference on Learning Representations (ICLR)*. **2022**.
- [8] N. Gebauer, M. Gastegger, K. Schütt, *Adv. Neural Inf. Process. Syst.* **2019**, *32*.
- [9] V. Garcia Satorras, E. Hoogeboom, F. Fuchs, I. Posner, M. Welling, *Adv. Neural Inf. Process. Syst.* **2021**, *34*, 4181.
- [10] E. Hoogeboom, V. G. Satorras, C. Vignac, M. Welling, in *International conference on machine learning*. PMLR, New York **2022**, pp. 8867–8887.
- [11] M. Xu, A. S. Powers, R. O. Dror, S. Ermon, J. Leskovec, in *International Conference on Machine Learning*. PMLR, New York **2023**, pp. 38592–38610.
- [12] R. Gómez-Bombarelli, J. N. Wei, D. Duvenaud, J. M. Hernández-Lobato, B. Sánchez-Lengeling, D. Sheberla, J. Aguilera-Iparraguirre, T. D. Hirzel, R. P. Adams, A. Aspuru-Guzik, *ACS Cent. Sci.* **2018**, *4*, 268.
- [13] P. Eckmann, K. Sun, B. Zhao, M. Feng, M. Gilson, R. Yu, in *International Conference on Machine Learning*. PMLR, New York **2022**, pp. 5777–5792.
- [14] Z.-Q. J. Xu, Y. Zhang, T. Luo, Y. Xiao, Z. Ma, *arXiv preprint arXiv:1901.06523* **2019**.
- [15] N. Rahaman, A. Baratin, D. Arpit, F. Draxler, M. Lin, F. Hamprecht, Y. Bengio, A. Courville, in *International Conference on Machine Learning*. PMLR, New York **2019**, pp. 5301–5310.
- [16] D. P. Kingma, M. Welling, *arXiv preprint arXiv:1312.6114* **2013**.
- [17] M. Krenn, F. Häse, A. Nigam, P. Friederich, A. Aspuru-Guzik, *Mach. Learn.: Sci. Technol.* **2020**, *1*, 045024.
- [18] NIST, Computational chemistry comparison and benchmark database, <https://cccbdb.nist.gov/diplistx.asp#NISTHydrocarbon> (accessed: January **2024**).
- [19] Y. C. Martin, J. L. Kofron, L. M. Traphagen, *J. Med. Chem.* **2002**, *45*, 4350.
- [20] RDKit Open-source cheminformatics, **2023**, <https://www.rdkit.org>.
- [21] D. G. Smith, L. A. Burns, A. C. Simmonett, R. M. Parrish, M. C. Schieber, R. Galvelis, P. Kraus, H. Kruse, R. Di Remigio, A. Alenaizan, A. M. James, S. Lehtola, J. P. Misiewicz, M. Scheurer, R. A. Shaw, J. B. Schriber, Y. Xie, Z. L. Glick, D. A. Sirianni, J. S. O'Brien, J. M. Waldrop, A. Kumar, E. G. Hohenstein, B. P. Pritchard, B. R. Brooks, H. F. 3rd Schaefer, A. Y. Sokolov, K. Patkowski, A. E. 3rd DePrince, U. Bozkaya, et al., *J. Chem. Phys.* **2020**, *152*, 18.
- [22] B. Lakshminarayanan, A. Pritzel, C. Blundell, *Adv. Neural Inf. Process. Syst.* **2017**, *30*.
- [23] B. Settles **2009**.
- [24] P. I. Frazier, *arXiv preprint arXiv:1807.02811* **2018**.
- [25] B. Trabucco, X. Geng, A. Kumar, S. Levine, in *International Conference on Machine Learning*. PMLR, New York **2022**, pp. 21658–21676.
- [26] F. A. Faber, L. Hutchison, B. Huang, J. Gilmer, S. S. Schoenholz, G. E. Dahl, O. Vinyals, S. Kearnes, P. F. Riley, O. A. Von Lilienfeld, *J. Chem. Theory Comput.* **2017**, *13*, 5255.
- [27] A. M. Matuszek, J. Reynisson, *Mol. Inf.* **2016**, *35*, 46.
- [28] A. Nigam, R. Pollice, A. Aspuru-Guzik, *Digital Discov.* **2022**, *1*, 390.
- [29] M. Lee, K. Min, *J. Chem. Inf. Model.* **2022**, *62*, 2943.
- [30] R. Ramakrishnan, P. O. Dral, M. Rupp, O. A. Von Lilienfeld, *Sci. Data* **2014**, *1*, 1.
- [31] A. Paszke, S. Gross, F. Massa, A. Lerer, J. Bradbury, G. Chanan, T. Killeen, Z. Lin, N. Gimelshein, L. Antiga, et al., *Adv. Neural Inf. Process. Syst.* **2019**, *32*.
- [32] I. Loshchilov, F. Hutter, *arXiv preprint arXiv:1711.05101* **2017**.
- [33] I. Loshchilov, F. Hutter, *arXiv preprint arXiv:1608.03983* **2016**.
- [34] V. G. Satorras, E. Hoogeboom, M. Welling, in *International conference on machine learning*. PMLR, New York **2021**, pp. 9323–9332.
- [35] S. Kullback, R. A. Leibler, *Annals Math. stat.* **1951**, *22*, 79.
- [36] C. Hansch, *Acc. Chem. Res.* **1969**, *2*, 232.
- [37] H. Kubinyi, *3D QSAR in drug design: volume 1: theory methods and applications*, vol. 1, Springer Science & Business Media, Berlin **1993**.
- [38] A. Vedani, P. Zbinden, *Pharm. Acta Helv.* **1998**, *73*, 11.
- [39] R. C. Glen, A. Payne, *J. Comput.-Aided Mol. Des.* **1995**, *9*, 181.

- [40] V. Venkatasubramanian, K. Chan, J. M. Caruthers, *J. Chem. Inf. Comput. Sci.* **1995**, 35, 188.
- [41] A. Sundaram, V. Venkatasubramanian, *J. Chem. Inf. Comput. Sci.* **1998**, 38, 1177.
- [42] E. Bengio, M. Jain, M. Korablyov, D. Precup, Y. Bengio, *Adv. Neural Inf. Process. Syst.* **2021**, 34, 27381.
- [43] H. H. Loeffler, J. He, A. Tibo, J. P. Janet, A. Voronov, L. H. Mervin, O. Engkvist, *J. Cheminf.* **2024**, 16, 20.
- [44] M. Olivecrona, T. Blaschke, O. Engkvist, H. Chen, *J. Cheminf.* **2017**, 9, 1.
- [45] W. Kabsch, *Acta Crystallograph. Sec. A: Cryst. Phys., Diffr., Theoret. Gen. Crystallogr.* **1976**, 32, 922.
- [46] E. Zitzler, L. Thiele, in *International conference on parallel problem solving from nature*, Springer, Berlin **1998**, pp. 292–301.
- [47] J. Blank, K. Deb, *IEEE Access* **2020**, 8, 89497.