A Generative Model for Molecular Distance Geometry

Gregor N. C. Simm * 1 José Miguel Hernández-Lobato 1

Abstract
Computing equilibrium states for many-body systems, such as molecules, is a long-standing challenge. In the absence of methods for generating statistically independent samples, great computational effort is invested in simulating these systems using, for example, Markov chain Monte Carlo. We present a probabilistic model that generates such samples for molecules from their graph representations. Our model learns a low-dimensional manifold that preserves the geometry of local atomic neighborhoods through a principle learning representation that is based on Euclidean distance geometry. In a new benchmark for molecular conformation generation, we show experimentally that our generative model achieves state-of-the-art accuracy. Finally, we show how to use our model as a proposal distribution in an importance sampling scheme to compute molecular properties.

1. Introduction
Over the last few years, many highly-effective deep learning methods generating small molecules with desired properties (e.g., novel drugs) have emerged (Gómez-Bombarelli et al., 2018; Segler et al., 2018; Dai et al., 2018; Jin et al., 2018; Bradshaw et al., 2019a; Liu et al., 2018; You et al., 2018; Bradshaw et al., 2019b). These methods operate using graph representations of molecules in which nodes and edges represent atoms and bonds, respectively. A representation that is closer to the physical system is one in which a molecule is described by its geometry or conformation. A conformation \( x \) of a molecule is defined by a set atoms \( \{ (\epsilon_i, \mathbf{r}_i) \}_{i=1}^{N_v} \), where \( N_v \) is the number of atoms in the molecule, \( \epsilon_i \in \{ \text{H, C, O, ...} \} \) is the chemical element of the atom \( i \), and \( \mathbf{r}_i \in \mathbb{R}^3 \) is its position in Cartesian coordinates. Importantly, the relative positions of the atoms are restricted by the bonds in the molecule and the angles between them. Due to thermal fluctuations resulting in stretching of and rotations around bonds, there exist infinitely many conformations of a molecule. A molecule’s graph representation and a set of its conformations are shown in Fig. 1. Under a wide range of conditions, the probability \( p(x) \) of a conformation \( x \), is governed by the Boltzmann distribution and is proportional to \( \exp\{-E(x)/k_BT\} \), where \( E(x) \) is the conformation’s energy, \( k_B \) is the Boltzmann constant, and \( T \) is the temperature.

To compute a molecular property for a molecule, one must sample from \( p(x) \). The main approach is to start with one conformation and make small changes to it over time, e.g., by using Markov chain Monte Carlo (MCMC) or molecular dynamics (MD). These methods can be used to accurately sample equilibrium states of molecules, but they become computationally expensive for larger ones (Shim & MacKerell, 2011; Ballard et al., 2015; De Vivo et al., 2016). Other heuristic approaches exist in which distances between atoms are set to fixed idealized values (Havel, 2002; Blaney & Dixon, 2007). Several methods based on statistical learning have also recently been developed to tackle the issue of conformation generation. However, they are mainly geared towards studying proteins and their folding dynamics (AlQuraishi, 2019). Some of these models are

Figure 1. Standard graph representation \( G \) of a molecule (left) with a set of possible conformations \( \{x_i\} \) (right). It is the goal of this work to generate such conformations from the graph representation of a molecule. Conformations feature the same atom types and bonds but the atoms are arranged differently in space. These differences arise from rotations around and stretching of bonds in the molecule. Hydrogen (H), carbon (C), and oxygen (O) atoms are colored white, gray, and red, respectively.
not targeting a distribution over conformations but the most stable folded configuration, e.g. AlphaFold (Senior et al., 2020), while others are not transferable between different molecules (Lemke & Peter, 2019; Noé et al., 2019).

This work includes the following key contributions:

- We introduce a novel probabilistic model for learning conformational distributions of molecules with graph neural networks.
- We create a new, challenging benchmark for conformation generation, which is made publicly available. To the best of our knowledge, this is the first benchmark of this kind.
- By combining a conditional variational autoencoder (CVAE) with an Euclidean distance geometry (EDG) algorithm we present a state-of-the-art approach for generating one-shot samples of molecular conformations for unseen molecules that is independent of their size and shape.
- We develop a rigorous experimental approach for evaluating and comparing the accuracy of conformation generation methods based on the mean maximum deviation distance metric.
- We show how this generative model can be used as a proposal distribution in an importance sampling (IS) scheme to estimate molecular properties.

2. Method

Our goal is to build a statistical model that generates molecular conformations in a one-shot fashion from a molecule’s graph representation. First, we describe how a molecule’s conformation can be represented by a set of pairwise distances between atoms and why this presentation is advantageous over one in Cartesian coordinates (Section 2.1). Second, we present a generative model in Section 2.2 that will generate sets of atomic distances for a given molecular graph. Third, we explain in Section 2.3 how a set of predicted distances can be transformed into a molecular conformation and why this transformation is necessary. Finally, we detail in Section 2.4 how our generative model can be used as a proposal distribution in an IS scheme to estimate molecular properties.

2.1. Extended Molecular Graphs and Distance Geometry

In this study, a molecule is represented by an undirected graph which is defined as a tuple $G = (V, E)$. $V = \{v_i\}_{i=1}^{N}$ is the set of nodes representing atoms, where each $v_i \in \mathbb{R}^{K_u}$ holds atomic attributes (e.g., the element type $\epsilon_i$). $E = \{(e_k, r_k, s_k)\}_{k=1}^{N_e}$ is the set of edges, where each $e_k \in \mathbb{R}^{F_e}$ holds an edge’s attributes (e.g., the bond type), and $r_k$ and $s_k$ are the nodes an edge is connecting. Here, $E$ represents the molecular bonds in the molecule.

We assume that, given a molecular graph $G$, one can represent one of its conformations $x$ by a set of atomic distances $d = \{d_k\}_{k=1}^{N_e}$, where $d_k = |r_k - s_k|$ is the Euclidean distance between the positions of the atoms $r_k$ and $s_k$ in this conformation. As the set of edges between the bonded atoms ($E_{\text{bond}}$) alone would not suffice to describe a conformation, we expand the traditional graph representation of a molecule by adding auxiliary edges to obtain an extended graph $\mathcal{G}$. Auxiliary edges between atoms that are second neighbors in the original graph $G$ fix angles between atoms, and those between third neighbors fix dihedral angles (denoted $E_{\text{angle}}$ and $E_{\text{dihedral}}$, respectively). In this work, $E_{\text{angle}}$ are added between nodes in $\mathcal{G}$ which are second neighbors in $G$. After all $E_{\text{angle}}$ have been added, additional edges are added to $\mathcal{G}$ from a node $v$ to a randomly chosen third neighbor of $v$ in $G$ if $v$ has less than three neighbors in $\mathcal{G}$. Therefore, a graph $G$ can give rise to multiple different extend graphs $\mathcal{G}$. In Fig. 2, the process of extending the molecular graph and the extraction of $d$ from $x$ and $\mathcal{G}$ are illustrated.

A key advantage of a representation in terms of distances is its invariance to rotation and translation; by contrast, Cartesian coordinates depend on the (arbitrary) choice of origin, for example. In addition, it reflects pair-wise physical interactions and their generally local nature. Auxiliary edges can be placed between higher-order neighbors depending on how far the physical interactions dominating the potential energy of the system reach.

We have a set of $N_G$ pairs, $\{G_i, x_i\}_{i=1}^{N_G}$, consisting of a molecular graph and a conformation. With the protocol described above, we convert each pair into a pair of an extended molecular graph together with a set of distances $d$ to obtain $\{G_i, d_i\}_{i=1}^{N_G}$. With this data, we will train a generative model which we detail in the following section.

2.2. Generative Model

We employ a CVAE (Kingma & Welling, 2014; Pagnoni et al., 2018) to model the distribution over distances $d$ given a molecular graph $G$. A CVAE first encodes $G$ together with $d$ into a latent space $z \in \mathbb{R}^{kN_r}$, where $k \in \mathbb{N}$, with an encoder $q_\phi(z|d, G)$. Subsequently, the decoder $p_\theta(d|z, G)$ decodes $z$ back into a set of distances. A graphical model is shown in Fig. 2, C.

A conformation has, in general, $3N_v - 6$ spatial degrees of freedom (dofs): one dof per spacial dimension per atom minus three translational and three rotational dofs. Therefore, the latent space should be proportional to the number of
Figure 2. A) The structural formula of a molecule $G$ is converted to an extended molecular graph $\mathcal{G}$ consisting of nodes representing atoms (circles, e.g., $v_1$) and edges representing molecular bonds (solid lines, e.g., $e_1 \in E_{\text{bnd}}$) and auxiliary edges (dotted lines, e.g., $e_2 \in E_{\text{angle}}$ and $e_3 \in E_{\text{dihedral}}$). B) The distances $d$ are extracted from a conformation $x$ based on the edges $E$. C) Graphical model of the variational autoencoder: generative model $p_{\theta}(d|z, \mathcal{G})p_{\phi}(z|\mathcal{G})$ (solid lines) and variational approximation $q_{\phi}(z|d, \mathcal{G})$ (dashed lines).

atoms in the molecule. In addition, the latent space should be smaller than $3N_v$, as it is the role of the encoder to project the conformation into a lower-dimensional space. As a result, we set $k = 1$.\footnote{Experiments showed that our model performs similarly well with a latent space of $\mathbb{R}^{3N_v}$ and $\mathbb{R}^{3N_v}$. We chose to use $k = 1$ for simplicity.}

Here, $q_{\phi}(z|d, \mathcal{G})$ and $p_{\theta}(d|z, \mathcal{G})$ are Gaussian distributions, the mean and variance of which are modeled by two artificial neural networks. At the center of this model are message-passing neural networks (MPNNs) (Gilmer et al., 2017). In short, an MPNN is a convolutional neural network that allows end-to-end learning of prediction pipelines whose inputs are graphs of arbitrary size and shape. In a convolution, neighboring nodes exchange so-called messages between neighbors to update their attributes. Edges update their attributes with the features of the nodes they are connecting. The MPNN is a well-studied technique that achieves state-of-the-art performance in representation learning for molecules (Kipf & Welling, 2017; Duvenaud et al., 2015; Kearnes et al., 2016; Schütt et al., 2017b; Gilmer et al., 2017; Kusner et al., 2017; Bradshaw et al., 2019a).

In the following, we describe the details of the mode which is illustrated in Fig. 3.\footnote{The model is available online https://github.com/gnns/graphdg} In the encoder $q_{\phi}(z|d, \mathcal{G})$, each $d_k$ is concatenated with the respective edge feature $e_k$ to give $e_k' \in \mathbb{R}^{F_{\text{e}} + 1}$. Then, each $v_i$ and $e_i'$ are passed to $F_{\text{enc}, v}$ and $F_{\text{enc}, e}$ (two multilayer perceptrons, MLPs), respectively, to give $G_{\text{enc}}^{(0)}$, where $G_{\text{enc}} = \{v_{i, \text{enc}}(0), (e_{k, \text{enc}}, r_k, s_k)\}_{k=1}^{N_e}, v_{i, \text{enc}} \in \mathbb{R}^{L_v},$ and $e_{k, \text{enc}} \in \mathbb{R}^{L_e}$. Then, $T$ MPNNs of depth 1, $\{M_{\text{enc}}^{(i)}\}_{i=1}^T$, are consecutively applied to obtain $G_{\text{enc}}^{(T)}$. Finally, the readout function $R_{\text{enc}}$ (an MLP) takes each $e_{k, \text{enc}}^{(T)}$ to predict the mean $\mu_{e_k} \in \mathbb{R}$ and the variance $\sigma_{e_k}^2 \in \mathbb{R}$ of the Gaussian distribution for $e_k$. In summary,

$\bar{v}_{i, \text{enc}}^{(0)} = F_{\text{enc}, v}(v_i), \quad \bar{e}_{k, \text{enc}}^{(0)} = F_{\text{enc}, e}(e_i')$,  
(1)  
$G_{\text{enc}}^{(i+1)} = M_{\text{enc}}^{(i)}(G_{\text{enc}}^{(i)}), \quad \mu_{e_k} = R_{\text{enc}}(e_{k, \text{enc}}^{(i)}), \quad \sigma_{e_k}^2 = R_{\text{dec}}(e_{k, \text{enc}}^{(i)}), \quad i = 1, \ldots, T$.  
(2)  
(3)

In the decoder $p_{\theta}(d|z, \mathcal{G})$, each $z_i$ is concatenated with the respective node feature $v_i$ to give $v_i' \in \mathbb{R}^{F_v+1}$. Each $v_i'$ and each $e_k$ are passed to $F_{\text{dec}, v}$ and $F_{\text{dec}, e}$ (two MLPs), respectively, to give $G_{\text{dec}}^{(0)}$, where $G_{\text{dec}} = \{v_{i, \text{dec}}(0), (e_{k, \text{dec}}, r_k, s_k)\}_{k=1}^{N_e}, v_{i, \text{dec}} \in \mathbb{R}^{L_v},$ and $e_{k, \text{dec}} \in \mathbb{R}^{L_e}$. Then, $T$ MPNNs of depth 1, $\{M_{\text{dec}}^{(i)}\}_{i=1}^T$, are consecutively applied to obtain $G_{\text{dec}}^{(T)}$. Finally, the readout function $R_{\text{dec}}$ (an MLP) takes each $e_{k, \text{dec}}^{(T)}$ to predict the mean $\mu_{d_k} \in \mathbb{R}$ and the variance $\sigma_{d_k}^2 \in \mathbb{R}$ of the Gaussian distribution for $d_k$. In summary,

$\bar{v}_{i, \text{dec}}^{(0)} = F_{\text{dec}, v}(v_i'), \quad \bar{e}_{k, \text{dec}}^{(0)} = F_{\text{dec}, e}(e_i)$,  
(4)  
$G_{\text{dec}}^{(i+1)} = M_{\text{dec}}^{(i)}(G_{\text{dec}}^{(i)}), \quad \mu_{d_k} = R_{\text{dec}}(e_{k, \text{dec}}^{(i)}), \quad \sigma_{d_k}^2 = R_{\text{dec}}(e_{k, \text{dec}}^{(i)}), \quad i = 1, \ldots, T$.  
(5)  
(6)

The sets of parameters in the encoder and decoder, $\phi$ and $\theta$ (i.e., parameters in $F_{\text{enc}, v}, F_{\text{enc}, e}, \{M_{\text{enc}}^{(i)}\}_{i=1}^T, R_{\text{enc}}, F_{\text{dec}, v}, F_{\text{dec}, e}, \{M_{\text{dec}}^{(i)}\}_{i=1}^T, R_{\text{dec}}$), respectively, are optimized by minimizing the evidence lower bound (ELBO):

$$L = \mathbb{E}_{z \sim q_{\phi}(z|d, \mathcal{G})}[\log p_{\theta}(d|z, \mathcal{G})] - D_{\text{KL}}[q_{\phi}(z|d, \mathcal{G})||p_{\theta}(z|\mathcal{G})],$$  
(7)

where the prior $p_{\theta}(z|\mathcal{G})$ consists of factorized standard Gaussians. The optimal values for the hyperparameters for the network dimensions, number of message passes, batch size, and learning rate of the Adam optimizer (Kingma & Ba, 2014) were manually tuned by maximizing the validation performance (ELBO) and are reported in the Appendix.
2.3. Conformation Generation through Euclidean Distance Geometry

To compute molecular properties, quantum-chemical methods need to be employed which require the input, i.e., the molecule, to be in Cartesian coordinates. Therefore, we use an EDG algorithm to translate the set of distances \( \{d_k\}_{k=1}^{N_v} \) to a set of atomic coordinates \( \{r_i\}_{i=1}^{N_v} \).

EDG is the mathematical basis for a geometric theory of molecular conformation. In the field of machine learning, Weinberger & Saul (2006) used it for learning image manifolds, Tenenbaum et al. (2000) for image understanding and handwriting recognition, Jain & Saul (2004) for speech and music, and Demaine et al. (2009) for music and musical rhythms. An EDG description of a molecular system consists of a list of lower and upper bounds on the distances between pairs of atoms \( \{(d_{k_{\min}}, d_{k_{\max}})\}_{k=1}^{N_v} \). Here, \( p_0(\mathbf{d}|\mathbf{z}, \mathcal{G}) \) is used to model these bounds, namely, we set the bounds to \( \{ (\mu_{d_k} - \sigma_{d_k}, \mu_{d_k} + \sigma_{d_k}) \} \), where \( \mu_{d_k} \) and \( \sigma_{d_k} \) are the mean and standard deviation for each distance \( d_k \) given by the CVAE. Then, an EDG algorithm determines a set of Cartesian coordinates \( \{r_i\}_{i=1}^{N_v} \) so that these bounds are fulfilled (see the Appendix for details). Together with the corresponding chemical elements \( \{\epsilon_i\}_{i=1}^{N_v} \), we obtain a conformation \( \mathbf{x} \).

2.4. Calculation of Molecular Properties

We can get an MC estimate of the expectation \( E[O] \) of a property \( O \) (e.g., the dipole moment) for a molecule represented by \( \mathcal{G} \) by generating an extended graph \( \mathcal{G} \), drawing conformational samples \( x_i \sim \hat{p}(\mathbf{x}|\mathcal{G}) \), and computing \( O(x_i) \in \mathbb{R} \) with a quantum-chemical method (e.g., density functional theory). Since we cannot draw samples from \( p(\mathbf{x}|\mathcal{G}) \) directly, we employ an IS integration scheme (Bishop, 2009) with our CVAE as the proposal distribution. We assume that we can readily evaluate the unnormalized probability of a conformation \( \hat{p}(\mathbf{x}|\mathcal{G}) = \exp\{-E(\mathbf{x})/k_BT\} \), where \( \mathbf{x} \) must be a conformation of the molecule and the energy \( E(\mathbf{x}) \) is determined with a quantum-chemical method. Since the EDG algorithm is mapping the distribution \( p_0(\mathbf{d}|\mathbf{z}, \mathcal{G}) \) to a point mass in \( \mathbb{R}^{3N_v} \), the MC estimate for the resulting distribution \( p_{\text{prop}}(\mathbf{x}|\mathcal{G}) \) is approximated by a mixture of delta functions, each of which is centered at the \( x_i \) resulting from mapping \( p_0(\mathbf{d}|\mathbf{z}_i, \mathcal{G}) \) to \( \mathbb{R}^{3N_v} \), where \( z_i \sim p_0(\mathbf{z}|\mathcal{G}) \), that is, \( p_{\text{prop}}(\mathbf{x}|\mathcal{G}) \approx \frac{1}{N} \sum_{i=1}^{N} \delta(\mathbf{x} - x_i) \). The IS estimator for the expectation of \( \mathcal{O} \) w.r.t. \( \hat{p}(\mathbf{x}|\mathcal{G}) \) then reads

\[
\tilde{E}_\mathcal{O}[\mathcal{O}] \approx \frac{1}{N} \sum_{i=1}^{N} \mathcal{O}(x_i) \approx \frac{1}{N} \sum_{i=1}^{N} \mathcal{O}(x'_i) \frac{\hat{p}(x'_i|\mathcal{G})}{p_{\text{prop}}(x'_i|\mathcal{G})},
\]

where \( x_i \sim \hat{p}(x_i|\mathcal{G}) \) and \( x'_i \sim p_{\text{prop}}(x'_i|\mathcal{G}) \), so that the expectation of \( \mathcal{O} \) w.r.t. the normalized version of \( \hat{p}(\mathbf{x}) \) is then

\[
\mathbb{E}_\mathcal{O}[\mathcal{O}] = \frac{\tilde{E}_\mathcal{O}[\mathcal{O}]}{\tilde{E}_\mathcal{O}[1]} \approx \frac{1}{Z} \sum_{i=1}^{N} \mathcal{O}(x_i) \hat{p}(x'_i|\mathcal{G}),
\]

where \( \tilde{E}_\mathcal{O}[1] \) is the expectation of an operator that returns 1 for every conformation \( \mathbf{x} \), \( Z \approx \sum_{i=1}^{N} \hat{p}(x'_i|\mathcal{G}) \), and \( N \) is
the number of samples. When dividing two delta functions we have assumed that they take some arbitrarily large finite value.

3. Related Works

The standard approach for generating molecular conformations is to start with one, and make small changes to it over time, e.g., by using MCMC or MD. These methods are considered the gold standard for sampling equilibrium states, but they are computationally expensive, especially if the molecule is large and the Hamiltonian is based on quantum-mechanical principles (Shim & MacKerell, 2011; Ballard et al., 2015; De Vivo et al., 2016).

A much faster but more approximate approach for conformation generation is EDG (Havel, 2002; Blaney & Dixon, 2007; Lagorce et al., 2009; Riniker & Landrum, 2015). Lower and upper distance bounds for pairs of atoms in a molecule are fixed values based on ideal bond lengths, bond angles, and torsional angles. These values are often extracted from crystal structure databases (Allen, 2002). These methods aim to produce a low-energy conformation, not to generate unbiased samples from the underlying distribution at a certain temperature.

There exist several machine learning approaches as well, however, they are mostly tailored towards studying protein dynamics. For example, Noé et al. (2019) trained Boltzmann generators on the energy function of proteins to provide unbiased, one-shot samples from their equilibrium states. This is achieved by training an invertible neural network to learn a coordinate transformation from a system’s configurations to a latent space representation. Further, Lemke & Peter (2019) proposed a dimensionality reduction algorithm that is based on a neural network autoencoder in combination with a nonlinear distance metric to generate samples for protein structures. Both models learn protein-specific coordinate transformations that cannot be transferred to other molecules.

AlQuraishi (2019) introduced an end-to-end differentiable recurrent geometric network for protein structure learning based on amino acid sequences. Also, Ingraham et al. (2019) proposed a neural energy simulator model for protein structure that makes use of protein sequence information. Recently, Senior et al. (2020) significantly advanced the field of protein-structure prediction with a new model called AlphaFold. In contrast to amino acid sequences, molecular graphs are, in general, not linear but highly branched and often contain cycles. This makes these approaches unsuitable for general molecules.

Finally, Mansimov et al. (2019) presented a conditional deep generative graph neural network to generate molecular conformations given a molecular graph. Their goal is to predict the most likely conformation and not a distribution over conformations. Instead of encoding molecular environments in atomic distances, they work directly in Cartesian coordinates. As a result, the generated conformations showed significant structural differences compared to the ground-truth and required refinement through a force field, which is often employed in MD simulations.

We argue that our model has several advantages over the approaches reviewed above:

- It is a fast alternative to resource-intensive approaches based on MCMC or MD.
- Our principled representation based on pair-wise distances does not restrict our approach to any particular molecular structure.
- Our model is, in principle, transferable to unseen molecules.

4. The CONF17 Benchmark

The CONF17 benchmark is the first benchmark for molecular conformation sampling. It is based on the ISO17 dataset (Schütt et al., 2017a) which consists of conformations of various molecules with the atomic composition C2H10O2 drawn from the QM9 dataset (Ramakrishnan et al., 2014). These conformations were generated by ab initio molecular dynamics simulations at 500 Kelvin. From the ISO17 dataset, 430692 valid molecular graph-conformation pairs could be extracted and 197 unique molecular graphs could be identified. We split the dataset into training and test sets such that no molecular graph in the training set can be found in the test or vice versa. Training and test splits consist of 176 and 30 unique molecular graphs, respectively (see Appendix A for details).

In Fig. 4, A, the structural formulae of a random selection of molecules from this benchmark are shown. Most molecules feature highly-strained, complex 3D structures such as rings which are typical of drug-like molecules. It is thus the structural complexity of the molecules, not their number of degrees of freedom, that makes this benchmark challenging. In Fig. 4, B–D, the frequency of distances (in Å) in the conformations are shown for each edge type. It can be seen that the marginal distributions of the edge distances are multimodal and highly context-dependent.

5. Experiments

We assess the performance of our method, named Graph Distance Geometry (GRAPHIDG), by comparing it with two
state-of-the-art methods for molecular conformation generation: RDKIT (Riniker & Landrum, 2015), a classical EDG approach, and DL4CHEM (Mansimov et al., 2019), a machine learning approach. We trained GRAPHDG and DL4CHEM on three different training and test splits of the Conf17 benchmark using Adam (Kingma & Ba, 2014). We generated 100 conformations with each method for molecular graphs in a test set.

5.1. Distributions Over Distances

We assessed the accuracy of the distance distributions of RDKIT, DL4CHEM, and GRAPHDG by calculating the maximum mean discrepancy (MMD) (Gretton et al., 2012) to the ground-truth distribution. In particular, we compute the MMD using a Gaussian kernel, where we set the standard deviation to be the median distance between distances \(d\) in the aggregate sample. For this, we determined the distances in the conformations from the ground-truth and those generated by RDKIT, DL4CHEM, and GRAPHDG. For each train-test split and each \(G\) in a test set, we compute the MMD of the joint distribution of distances between \(C\) and \(O\) atoms \(p(\{d_k\}|G)\) (H atoms are usually ignored), the MMDs of pair-wise distances \(p(d_i, d_j|G)\), and the MMDs between the marginals of individual distances \(p(d_i|G)\). We aggregate the results of three train-test splits, and, finally, compute the median MMDs and average rankings. The results are summarized in Table 1. It can be seen that the samples from GRAPHDG are significantly closer to the ground-truth distribution than the other methods. RDKIT is slightly worse than GRAPHDG while DL4CHEM seems to struggle with the complexity of the molecules and the small number of graphs in the training set.

In Fig. 5, we showcase the accuracy of our model by plotting the marginal distributions \(p(d_i|G)\) for distances between \(C\) and \(O\) atoms, given a molecular graph from a test set. It can be seen that RDKIT consistently underestimates the marginal variances. This is because this method aims to predict the most stable conformation, i.e., the distribution’s mode. In contrast, DL4CHEM often fails to predict the correct mean. For this molecule, GRAPHDG is the most accurate, predicting the right mean and variance in most cases. Additional figures can be found in the Appendix, where we also show plots for the marginal distributions \(p(d_i, d_j|G)\).

5.2. Generation of Conformations

We passed the distances from our generative model to an EDG algorithm to obtain conformations. For 99.9% of the sets of distances, all triangle inequalities held. For 83% of the molecular graphs, the algorithm succeeded which is 7 pp higher than the success rate we observed for RDKIT. For each molecular graph in a test set, we generated 50 conformations with each method. This took DL4CHEM, RDKIT, and GRAPHDG on average around hundreds of milliseconds per molecule.\(^7\) In contrast, a single conformation in the ISO17 dataset takes around a minute to compute.

To assess the approximations made in the IS scheme, we studied the overlap between \(p(d|z, G)\) for a given \(G\) and different samples of \(z\). We found experimentally that for 50 samples the overlap between the distributions is small. This finding can be explained by the high dimensionality of \(d\) which is on average \(\approx 60\).

In Fig. 6, an overlay of these conformations of six molecules generated by the different methods is shown. It can be seen that RDKIT’s conformations show too little variance, while DL4CHEM’s structures are mostly invalid, which is due in part to its failure to predict the correct interatomic angles. Our method slightly overestimates the structural variance (see, for example, Fig. 6, top row, second column), but produces conformations that are the closest to the ground-truth.

\(^7\)All simulations were carried out on a computer equipped with an i7-3820 CPU and a GeForce GTX 1080 Ti GPU.
Table 1. Assessment of the accuracy of the distributions over conformations generated by three models compared to the ground-truth. We compare the distributions with respect to the marginals $p(d_k|G)$, $p(d_k, d_l|G)$, and the distribution over all edges between C and O atoms $p(\{d_k\}|G)$. Two different metrics are used: median MMD between ground-truth conformations and generated ones, and mean ranking (1 to 3) based on the MMD. Reported are the results for molecular graphs in a test set from three train-test splits. Standard deviations are given in brackets.

<table>
<thead>
<tr>
<th></th>
<th>RDKit Median MMD</th>
<th>DL4Chem GraphDG</th>
<th>RDKit Mean Ranking</th>
<th>DL4Chem GraphDG</th>
</tr>
</thead>
<tbody>
<tr>
<td>$p(d_k</td>
<td>G)$</td>
<td>0.37 (0.23)</td>
<td>1.11 (0.25)</td>
<td>0.13 (0.13)</td>
</tr>
<tr>
<td>$p(d_k, d_l</td>
<td>G)$</td>
<td>0.47 (0.18)</td>
<td>1.12 (0.15)</td>
<td>0.14 (0.11)</td>
</tr>
<tr>
<td>$p({d_k}</td>
<td>G)$</td>
<td>0.57 (0.11)</td>
<td>1.03 (0.13)</td>
<td>0.19 (0.08)</td>
</tr>
</tbody>
</table>

Figure 5. Marginal distributions $p(d_k|G)$ of ground-truth and predicted distances (in Å) between C and O atoms given a molecular graph from a test set. The atoms connected by each edge $d_k$ are indicated in each subplot ($s_k$–$r_k$). In the 3D structure of the molecule, carbon and oxygen atoms are colored gray and red, respectively. H atoms are omitted for clarity.

Figure 6. Overlay of 50 conformations from the ground-truth and three models based on six random molecular graphs from the test set. C, O, and H atoms are colored gray, red, and white, respectively.
Table 2. Median difference in average properties between ground-truth and RDKit and GraphDG: total electronic energy \( E_{\text{elec}} \) (in kJ/mol), the energy of the HOMO and the LUMO \( \epsilon_{\text{HOMO}} \) and \( \epsilon_{\text{LUMO}} \) respectively (in eV), and the dipole moment \( \mu \) (in debye). Reported are the results for molecular graphs from the test set, averaged over three train-test splits. Standard errors are given in brackets.

<table>
<thead>
<tr>
<th></th>
<th>RDKit</th>
<th>GraphDG</th>
</tr>
</thead>
<tbody>
<tr>
<td>( E_{\text{elec}} )</td>
<td>42.7 (4.3)</td>
<td>58.0 (21.0)</td>
</tr>
<tr>
<td>( \epsilon_{\text{HOMO}} )</td>
<td>0.08 (0.04)</td>
<td>0.10 (0.05)</td>
</tr>
<tr>
<td>( \epsilon_{\text{LUMO}} )</td>
<td>0.15 (0.03)</td>
<td>0.09 (0.05)</td>
</tr>
<tr>
<td>( \mu )</td>
<td>0.29 (0.05)</td>
<td>0.33 (0.09)</td>
</tr>
</tbody>
</table>

5.3. Calculation of Molecular Properties

We estimate expected molecular properties for molecular graphs from the test set with \( N = 50 \) conformational samples each. Due to their poor quality, we could not compute properties \( O(x) \), including the energy \( E(x) \), for conformations generated with DL4Chem, and thus, this method is excluded from this analysis. In Table 2, it can be seen that RDKit and GraphDG perform similarly well (computational details can be found in the Appendix). However, both methods are still highly inaccurate for \( E_{\text{elec}} \) (in practice, an accuracy of less than 5 kJ/mol is required). Close inspection of the conformations shows that, even though GraphDG predicts the most accurate distances overall, the variances of certain strongly constrained distances (e.g., triple bonds) are overestimated so that the energies of the conformations increase drastically.

6. Limitations

The first limitation of this work is that the CVAE can sample invalid sets of distances for which there exists no 3D structure. Second, the CONF17 benchmark covers only a small portion of chemical space. Finally, a large set of auxiliary edges would be required to capture long-range correlations (e.g., in proteins). Future work will address these points.

7. Conclusions

We presented GraphDG, a transferable, generative model that allows sampling from a distribution over molecular conformations. We developed a principled learning representation of conformations that is based on distances between atoms. Then, we proposed a challenging benchmark for comparing molecular conformation generators. With this benchmark, we show experimentally that conformations generated by GraphDG are closer to the ground-truth than those generated by other methods. Finally, we employ our model as a proposal distribution in an IS integration scheme to estimate molecular properties. While orbital energies and the dipole moments were predicted well, a larger and more diverse dataset will be necessary for meaningful estimates of electronic energies. Further, methods have to be devised to estimate how many conformations need to be generated to ensure all important conformations have been sampled. Finally, our model could be trained on conformational distributions at different temperatures in a transfer learning-type setting.

Acknowledgments

We would like to thank the anonymous reviewers for their valuable feedback. We further thank Robert Perharz and Hannes Harbrecht for useful discussions and feedback. GNCS acknowledges funding through an Early Postdoc.Mobility fellowship by the Swiss National Science Foundation (P2EZP2_181616).

References


A Generative Model for Molecular Distance Geometry


