GENERATING VALID EUCLIDEAN DISTANCE MATRICES

Anonymous authors
Paper under double-blind review

ABSTRACT

Generating point clouds, e.g., molecular structures, in arbitrary rotations, translations, and enumerations remains a challenging task. Meanwhile, neural networks utilizing symmetry invariant layers have been shown to be able to optimize their training objective in a data-efficient way. In this spirit, we present an architecture which allows to produce valid Euclidean distance matrices, which by construction are already invariant under rotation and translation of the described object. Motivated by the goal to generate molecular structures in Cartesian space, we use this architecture to construct a Wasserstein GAN utilizing a permutation invariant critic network. This makes it possible to generate molecular structures in a one-shot fashion by producing Euclidean distance matrices which have a three-dimensional embedding.

1 INTRODUCTION

Recently there has been great interest in deep learning based on graph structures (Defferrard et al., 2016; Kipf & Welling, 2016; Gilmer et al., 2017) and point clouds (Qi et al., 2017; Li et al., 2018b; Yang et al., 2019). A prominent application example is that of molecules, for which both inference based on the chemical compound, i.e., the molecular graph structure (Kearnes et al., 2016; Janet & Kulik, 2017; Winter et al., 2019a), and based on the geometry, i.e. the positions of atoms in 3D space (Behler & Parrinello, 2007; Rupp et al., 2012; Schütt et al., 2017b; Smith et al., 2017) are active areas of research.

A particularly interesting branch of machine learning for molecules is the reverse problem of generating molecular structures, as it opens the door for designing molecules, e.g., obtain new materials (Sanchez-Lengeling & Aspuru-Guzik, 2018; Barnes et al., 2018; Elton et al., 2018; Li et al., 2018a), design or discover pharmacological molecules such as inhibitors or antibodies (Popova et al., 2018; Griffen et al., 2018), optimize biotechnological processes (Guimaraes et al., 2017). While this area of research has exploded in the past few years, the vast body of work has been done on the generation of new molecular compounds, i.e. the search for new molecular graphs, based on string encodings of that graph structure or other representations (Gómez-Bombarelli et al., 2018; Winter et al., 2019b). On the other hand, exploring the geometry space of the individual chemical compound is equally important, as the molecular geometries and their probabilities determine all equilibrium properties, such as binding affinity, solubility etc. Sampling different geometric structures is, however, still largely left to molecular dynamics (MD) simulation that suffers from the rare event sampling problem, although recently machine learning has been used to speed up MD simulation (Ribeiro et al., 2018; Bonati et al., 2019; Zhang et al., 2019; Plattner et al., 2017; Doerr & Fabritiis, 2014) or to perform sampling of the equilibrium distribution directly, without MD (Noé et al., 2019). All of these techniques only sample one single chemical compound in geometry space.

Here we explore—to our best knowledge for the first time in depth—the simultaneous generation of chemical compounds and geometries. The only related work we are aware of (Gebauer et al., 2018; 2019) demonstrates the generation of chemical compounds, placing atom by atom with an autoregressive model. It was shown that the model can recover compounds contained in the QM9 database of small molecules (Ruddigkeit et al., 2012; Ramakrishnan et al., 2014) when trained on a subset, but different configurations of the same molecule were not analyzed.

While autoregressive models seem to work well in the case of small (< 9 heavy atoms) molecules like the ones in the QM9 database, they can be tricky for larger structures as the probability to completely form complex structures, such as rings, decays with the number of involved steps.

To avoid this limitation, in our method we investigate in one shot models for point clouds which have no absolute orientation, i.e., the point cloud structure is considered to be the same independent of its rotation, translation, and of the permutation of points.

A natural representation, which is independent of rotation and translation is the Euclidean distance matrix, which is the matrix of all squared pairwise Euclidean distances. Furthermore, Euclidean distance matrices are useful determinants of valid molecular structures.

While a symmetric and non-negative matrix with a zero diagonal can easily be parameterized by, e.g., a neural network, it is not immediately clear that this matrix indeed belongs to a set of n points in Euclidean space and even then, that this space has the right dimension.

Here we develop a new method to parameterize and generate valid Euclidean distance matrices without placing coordinates directly, hereby taking away a lot of the ambiguity.

We furthermore propose a Wasserstein GAN architecture for learning distributions of pointclouds, e.g., molecular structures invariant to rotation, translation, and permutation. To this end the data distribution as well as the generator distribution are represented in terms of Euclidean distance matrices.

In summary, our main contributions are as follows:

- We introduce a new method of training neural networks so that their output are Euclidean distance matrices with a predefined embedding dimension.
- We propose a GAN architecture, which can learn distributions of Euclidean distance matrices, while treating the structures described by the distance matrices as set, i.e., invariant under their permutations.
- We apply the proposed architecture to a set of $\mathrm{C_7O_2H_{10}}$ isomers contained in the QM9 database and show that it can recover parts of the training set as well as generalize out of it.

2 Generating Euclidean distance matrices

We describe a way to generate Euclidean distance matrices $D \in \mathbb{EDM}^n \subset \mathbb{R}^{n\times n}$ without placing coordinates in Cartesian space. This means in particular that the parameterized output is invariant to translation and rotation.

A matrix D is in $\mathbb{EDM}^n$ by definition if there exist points $\boldsymbol{x}_1, \ldots, \boldsymbol{x}_n \in \mathbb{R}^d$ such that $D_{ij} = \|\boldsymbol{x}_i - \boldsymbol{x}_j\|_2^2$ for all $i, j = 1, \ldots, n$. Such a matrix D defines a structure in Euclidean space $\mathbb{R}^d$ up to a combination of translation, rotation, and mirroring. The smallest integer $d > 0$ for which a set of n points in $\mathbb{R}^d$ exists that reproduces the matrix D is called the embedding dimension.

The general idea of the generation process is to produce a hollow (i.e., zeros on the diagonal) symmetric matrix $\tilde{D}$ and then weakly enforce $\tilde{D} \in \mathbb{EDM}^n$ through a term in the loss. It can be shown that

$$\tilde{D} \in \mathbb{EDM}^n \Leftrightarrow -\frac{1}{2} J\tilde{D}J \text{ positive semi-definite,} \tag{1}$$

where $J = \mathbb{I} - \frac{1}{n}\mathbf{1}\mathbf{1}^\top$ and $\mathbf{1} = (1, \ldots, 1)^\top \in \mathbb{R}^n$ (Schoenberg, 1935; Gower, 1982). However trying to use this relationship directly in the context of deep learning by parameterizing the matrix $\tilde{D}$ poses a problem, as the set of EDMs forms a cone (Dattorro (2010)) and not a vector space, which is the underlying assumption of the standard optimizers in common deep learning frameworks. One can either turn to optimization techniques on Riemannian manifolds (Zhang et al. (2016)) or find a reparameterization in which the network's output behaves like a vector space and that can be transformed into an EDM.

Here, we leverage a connection between EDMs and positive semi-definite matrices Alfakih et al. (1999); Krislock & Wolkowicz (2012) in order to parameterize the problem in a space that behaves like a vector space. In particular, for $D \in \mathbb{EDM}^n$ by definition there exist points $\boldsymbol{x}_1, \ldots, \boldsymbol{x}_n \in \mathbb{R}^d$ generating D. The EDM D has a corresponding Gram matrix $M \in \mathbb{R}^{n\times n}$ by the relationship

$$M_{ij} = \langle \boldsymbol{y}_i, \boldsymbol{y}_j \rangle_2 = \frac{1}{2}(D_{1j} + D_{i1} - D_{ij}) \tag{2}$$

with $\boldsymbol{y}_k = \boldsymbol{x}_k - \boldsymbol{x}_1,\ k = 1, \ldots, n$ and vice versa

$$D_{ij} = M_{ii} + M_{jj} - 2M_{ij}. \tag{3}$$

The matrix M furthermore has a specific structure

$$M = \begin{pmatrix} 0 & \mathbf{0}^\top \\ \mathbf{0} & L \end{pmatrix} \tag{4}$$

with $L \in \mathbb{R}^{n-1 \times n-1}$ and is symmetric and positive semi-definite. It therefore admits an eigenvalue decomposition $M = USU^\top = (U\sqrt{S})(U\sqrt{S})^\top = YY^\top$ which, assuming that $S = \operatorname{diag}(\lambda_1, \ldots, \lambda_n)$ with $\lambda_1 \geq \lambda_2 \geq \ldots \geq \lambda_n \geq 0$, reveals a composition of coordinates Y in the first d rows where d is the embedding dimension and the number of non-zero eigenvalues of M associated to D.

Therefore, the embedding dimension d of D is given by the rank of M or equivalently the number of positive eigenvalues. In principle it would be sufficient to parameterize a symmetric positive semi-definite matrix $L \in \mathbb{R}^{n-1 \times n-1}$, as it then automatically is also a Gram matrix for some set of vectors. However, also the set of symmetric positive semi-definite matrices behaves like a cone, which precludes the use of standard optimization techniques.

Instead, we propose to parameterize an arbitrary symmetric matrix $\tilde{L} \in \mathbb{R}^{n-1 \times n-1}$, as the set of symmetric matrices behaves like a vector space. This symmetric matrix can be transformed into a symmetric positive semi-definite matrix

$$L = g(\tilde{L}) = g\left(U \begin{pmatrix} \lambda_1 & & \\ & \ddots & \\ & & \lambda_{n-1} \end{pmatrix} U^\top\right) = U \begin{pmatrix} g(\lambda_1) & & \\ & \ddots & \\ & & g(\lambda_{n-1}) \end{pmatrix} U^\top \tag{5}$$

by any non-negative function $g(\cdot)$ and then used to reconstruct D via (3) and (4).

This approach is shown in Algorithm 1 for the context of neural networks and the particular choice of $g = \mathrm{sp}$, the softplus activation function. A symmetric matrix $\tilde{L}$ is parameterized and transformed into a Gram matrix M and a matrix D. For M there is a loss in place that drives it towards a specific rank and for D we introduce a penalty on negative eigenvalues of (1).

Algorithm 1 Algorithm to train a generative neural network to produce Euclidean distance matrices, where N_z is the dimension of the input vector, m the batch size, and n the number of points to place relative to one another.

Sample $\mathbf{z} \sim \mathcal{N}(0,1)^{m \times N_z}$
Generate $X = G(\mathbf{z}) \in \mathbb{R}^{m \times (n-1) \times (n-1)}$
for $i = 1$ to m **do**
    Symmetrize $\tilde{L} \leftarrow \frac{1}{2}\left(X_i + X_i^\top\right)$
    Make positive semi-definite $L \leftarrow \mathrm{sp}(\tilde{L})$
    Assemble $M = M(L)$ with (4)
    Assemble $D = D(M)$ with (3)
    Compute eigenvalues $\mu_1, \ldots, \mu_n$ of (1) for D
    $L_{\mathrm{edm}}^{(i)} \leftarrow \sum_{k=1}^{n} \mathrm{ReLU}(-\mu_k)^2$
    Compute eigenvalues $\lambda_1, \ldots, \lambda_n$ of M such that $\lambda_1 \geq \lambda_2 \geq \ldots \lambda_n$
    $L_{\mathrm{rank}}^{(i)} \leftarrow \sum_{k=d+1}^{n} \lambda_k^2$
end for
$L \leftarrow \eta_1 \frac{1}{m} \sum_{i=1}^{m} L_{\mathrm{edm}}^{(i)} + \eta_2 \frac{1}{m} \sum_{i=1}^{m} L_{\mathrm{rank}}^{(i)}$
Optimize weights of G with respect to ∇L.

3 EUCLIDEAN DISTANCE MATRIX WGAN

We consider the class of generative adversarial networks (Goodfellow et al. (2014)) (GANs) and in particular Wasserstein GANs (WGANs), i.e., the ones that minimize the Wasserstein-1 distance

in contrast to the original formulation, where the former can be related to minimizing the Jensen-Shannon divergence Arjovsky et al. (2017). WGANs consist of two networks, one generator network $G(\cdot)$, which transforms a prior distribution into a target distribution $\mathbb{P}_g$ which should match the data's underlying distribution $\mathbb{P}_r$ as closely as possible. The other network is a so-called critic network $C(\cdot) \in \mathbb{R}$, which assigns scalar values to individual samples from either distribution. The overall optimization objective reads

$$\min_G \max_{C \in \mathcal{D}} \mathbb{E}_{\boldsymbol{x} \sim \mathbb{P}_r} \left[C(\boldsymbol{x}) \right] - \mathbb{E}_{\boldsymbol{x} \sim \mathbb{P}_g} \left[C(\boldsymbol{x}) \right], \tag{6}$$

where $\mathcal{D}$ is the set of all Lipschitz continuous functions with a Lipschitz constant $L \leq 1$. We enforce the Lipschitz constant using a gradient penalty (WGAN-GP) introduced in Gulrajani et al. (2017). One can observe that the maximum in Eq. (6) is attained when as large as possible values are assigned to samples from $\mathbb{P}_r$ and as small as possible values to samples from $\mathbb{P}_g$. Meanwhile the minimum over the generator network G tries to minimize that difference, which turns out to be exactly the Wasserstein-1 distance according to the Kantorovich–Rubinstein duality (Villani (2008)). Since the Wasserstein-1 distance is a proper metric of distributions, the generated distribution $\mathbb{P}_g$ is exactly the data distribution $\mathbb{P}_r$ if and only if the maximum in Eq. (6) is zero. The networks G and C are trained in an alternating fashion.

We choose for the critic network the message-passing neural network SchNet (Schütt et al., 2017c;a; 2018) $C_{\mathrm{SchNet}}(\cdot)$, which was originally designed to compute energies of molecules.

It operates on the pairwise distances $(\sqrt{D_{ij}})_{i,j=1}^n$, $D \in \mathbb{EDM}^n$ in a structure and the atom types $\mathcal{T}^n$. If there is no atom type information present, these can be just constant vectors that initially carry no information. These atom types are then embedded into a state vector and transformed with variable sharing across all atoms. Furthermore there are layers in which continuous convolutions are performed based on the relative distances between the atoms. In a physical sense this corresponds to learning energy contributions of, e.g., bonds and angles. Finally all states are mapped to a scalar and then pooled in a sum.

Due to the pooling and the use of only relative distances but never absolute coordinates, the output is invariant under translation, rotation, and permutation.

The generator network G employs the construction of Section 2 to produce approximately EDMs with a fixed embedding dimension. Therefore this architecture is able to learn distributions of Euclidean distance matrices.

4 Application and results

The WGAN-GP introduced in Sec. 3 is applied to a subset of the QM9 dataset consisting of 6095 isomers with the chemical formula $\mathrm{C_7O_2H_{10}}$. To this end the distribution not only consists of the Euclidean distance matrices describing the molecular structure but also of the atom types. The generator produces an additional type vector in a multi-task fashion which is checked against a constant type reference with a cross-entropy loss. Furthermore the prior of a minimal distance between atoms is applied, i.e., we have a loss penalizing distances that are too small. Altogether we optimize the losses

$$\begin{aligned}
L_{\mathrm{critic}} &= \mathbb{E}_{(D,\mathbf{t}) \sim \mathbb{P}_g} \left[C(D, \mathbf{t}) \right] - \mathbb{E}_{(D,\mathbf{t}) \sim \mathbb{P}_r} \left[C(D, \mathbf{t}) \right] && \text{(original WGAN loss)} && (7)\\
&+ \lambda L_{\mathrm{GP}} && \text{(gradient penalty of WGAN-GP)} && (8)\\
&+ \varepsilon_{\text{drift}} \mathbb{E}_{(D,\mathbf{t}) \sim \mathbb{P}_r} \left[C(D, \mathbf{t})^2 \right] && \text{(drift term Karras et al. (2017))} && (9)\\
L_{\mathrm{gen}} &= -\mathbb{E}_{(D,\mathbf{t}) \sim \mathbb{P}_g} \left[C(D, \mathbf{t}) \right] && \text{(original WGAN loss)} && (10)\\
&+ -\mathbb{E}_{(D,\mathbf{t}) \sim \mathbb{P}_g} \left[H(\mathbf{t}, \mathbf{t}_{\mathrm{ref}}) \right] && \text{(cross entropy for types)} && (11)\\
&+ k \cdot \mathbb{E}_{(D,\mathbf{t}) \sim \mathbb{P}_g} \left[\frac{1}{2} \sum_{i \neq j} (\sqrt{D_{ij}} - r)^2 \right] && \text{(harmonic repulsion)} && (12)\\
&+ L_{\text{edm}} && \text{(for EDMs, see Alg. 1)} && (13)
\end{aligned}$$

with $C(\cdot)$ being a SchNet critic, $\mathbf{t}_{\mathrm{ref}}$ a reference type order, $\lambda = 10$, $\varepsilon_{\text{drift}} = 10^{-3}$, $k = 10$, and r being the minimal pairwise distance achieved in the considered QM9 subset. Although in principle

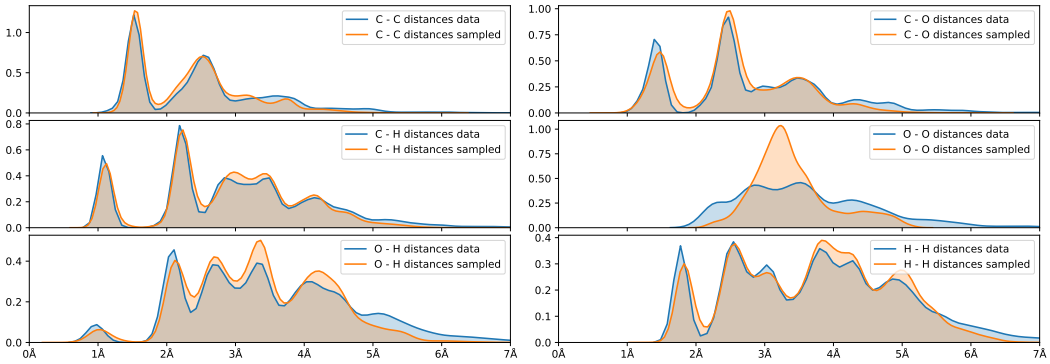

Figure 1: Distribution of pairwise distances between different kinds of atom type after training a Euclidean distance matrix WGAN-GP (Sec. 3) on the $C_7O_2H_{10}$ isomer subset of QM9.

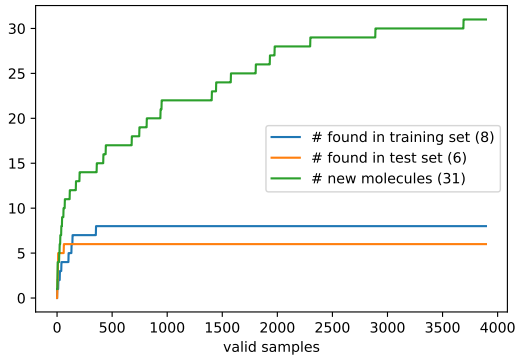

Figure 2: Number of unique molecular structures in terms of their topology for roughly 4000 valid generated samples and whether they could be found in the training set (blue), the test set (orange), or had a new topology altogether (green).

the cross entropy loss (11) is not required we found in our experiments that it qualitatively helps convergence. The generator network $G(\cdot)$ uses a combination of deconvolution and dense layers.

The function $g(\cdot)$ ensuring positive semi-definiteness (5) was chosen to be the softplus activation $g = \mathrm{sp}$ for the largest three eigenvalues and we explicitly set all other eigenvalues to zero. This leads to a Gram matrix with exactly the right rank and the constraint does not need to be weakly enforced anymore in the generator's loss.

Prior to training the data was split into 50% training and 50% test data. After training on the training data set we evaluate the distribution of pairwise distances between different types of atoms, see Fig. 1. The overall shape of the distributions is picked up and only the distance between pairs of oxygen atoms are not completely correctly distributed.

After generation we perform a computationally cheap validity test by inferring bonds and bond orders with Open Babel O'Boyle et al. (2011). On the inferred bonding graph we check for connectivity and valency, i.e., if for each atom the number of inferred bonds add up to its respective valency. This leaves us with roughly 7.5% of the generated samples.

For the valid samples we infer canonical SMILES representations which are a fingerprint of the molecule's topology in order to determine how many different molecule types can be produced using the trained generator. Fig. 2 shows the cumulative number of unique SMILES fingerprints when producing roughly 4000 valid samples. It can be observed that the network is able to generalize out of the training set and is able to generate not only topologies which can be found in the test set but also entirely new ones. Nevertheless, the current performance with respect to the number of found topologies is not optimal and can likely be improved by a better hyperparameter selection.

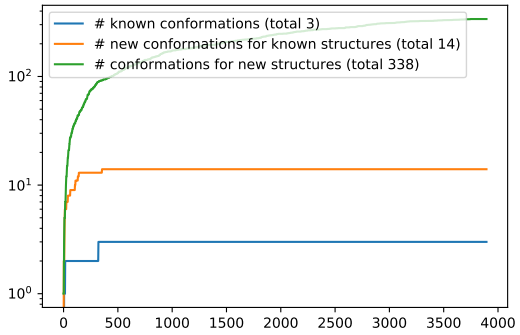

Figure 3: Unique generated conformations up to a maximal heavy atom distance cutoff of $d_{\text{cutoff}} = 0.6\,\text{Å}$ after assignment and superposition. We distinguish the categories of known conformations in the considered subset of the QM9 database (blue), new conformations for contained molecular structures (orange), and distinct conformations for molecular structures that are not contained.

While SMILES can be used to get an idea about the different bonding structures that were generated, it contains no information about different possible conformations in these bonding structures. To analyze the number of unique conformations that were generated, we compared each generated structure against all structures in the considered QM9 subset. Since the architecture is designed in such a way that it is permutation invariant, i.e., applying the critic onto a matrix $D = (D_{i,j})_{i,j=1}^n$ and $D_\sigma = (D_{\sigma(i),\sigma(j)})_{i,j=1}^n$ for some permutation σ yields the same result, one first has to find the best possible assignment of atoms.

To this end, we apply the Hungarian algorithm Kuhn (1955) onto a cost matrix $C \in \mathbb{R}^{n\times n}$ for EDMs D_1, D_2 and type vectors $\mathbf{t}_1, \mathbf{t}_2 \in \mathbb{R}^n$ with

$$C_{i,j} = \begin{cases} \left|\frac{1}{n}\sum_{k=1}^n (D_1)_{i,k} - \frac{1}{n}\sum_{k=1}^n (D_2)_{j,k}\right| & \text{, if } (\mathbf{t}_1)_i = (\mathbf{t}_2)_j, \\ \infty & \text{, otherwise.} \end{cases} \tag{14}$$

Intuitively this means that the cost of assigning atom i in the first structure to atom j in the second structure depends on whether their atom types match, in which case we compare the mean distance from the i-th atom to all other atoms in its structure to the same quantity for the j-th atom in the second structure. If the atom types do not match, we assign a very high number so that this particular mapping is not considered. After we have found an assignment between the atoms, we superpose the structures using functionality from the software package MDTraj (McGibbon et al. (2015)) and evaluate the maximal atomic distance between all heavy atoms (i.e., carbons and oxygens) after alignment. The cutoff at which we consider a structure to be a distinct conformation is a maximal atomic distance between heavy atoms of more than $d_{\text{cutoff}} = 0.6\,\text{Å}$, i.e., more than half a carbon–carbon bond length.

The results of this analysis are depicted in Fig. 3. One can observe that although the reported number of unique molecular structures via SMILES is rather low, under our similarity criterion a lot of different valid conformations are discovered; in particular also some new conformations of structures that were already contained in the QM9 database.

Finally, we also check for the approximate total energies of the generated molecules compared to the database's. To this end, we use the semi-empirical method provided by the software package MOPAC Stewart (1990) to relax all structures in the considered QM9 subset as well as all valid generated valid structures, see Fig. 4. It can be observed that after relaxation all energies are contained within the same range of roughly $-1586\,\text{eV}$ to $-1581\,\text{eV}$.

In Figure 5 we show examples of generated molecules in the top row (A–D) and the closest respective matches in the QM9 database in the bottom row (A'–D'). The closeness of a match was determined by the maximal atomic distance after assignment of atom identities and superposition. Configurations A and B could be matched with a maximal atomic distance of less than d_{cutoff}.

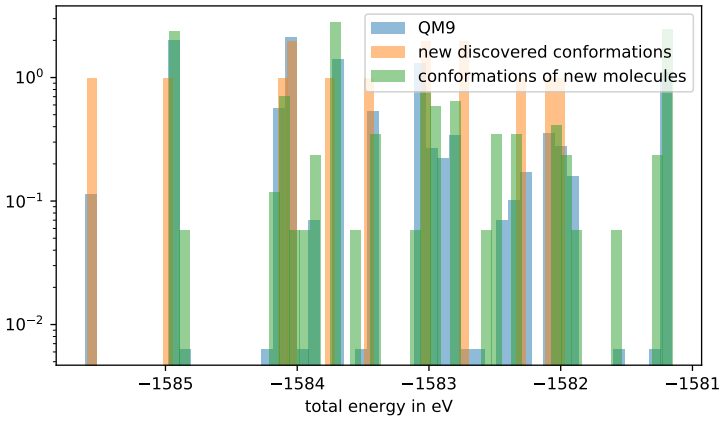

Figure 4: Total energies of structures that were relaxed with the semi-empirical method implemented by MOPAC, in particular for molecules contained in the considered QM9 subset (blue), structures that correspond to new conformations for contained molecules (orange), and unique conformations that belong to new molecules.

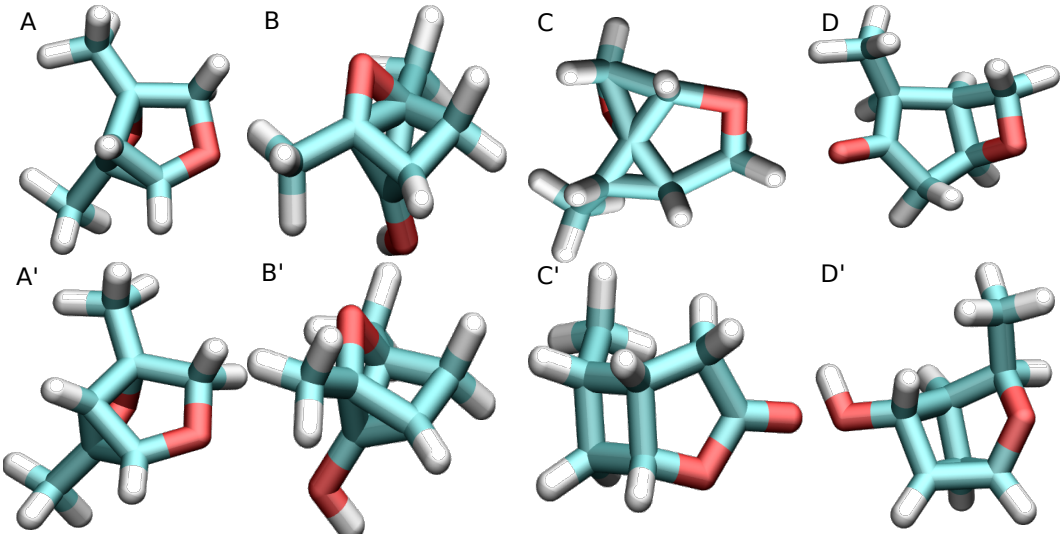

Figure 5: Sampled structures with the Euclidean distance matrix WGAN. Top row A to D are generated samples, bottom row A' to D' are closest matches from the QM9 database. Generated molecules A and B could be matched with A' and B' up to a maximum atom distance of $0.6\,\text{Å}$. Generated molecules C and D are new molecular structures with their closest matches C' and D', respectively.

5 CONCLUSION AND DISCUSSION

We have developed a way to parameterize the output of a neural network so that it produces valid Euclidean distance matrices with a predefined embedding dimension without placing coordinates in Cartesian space directly. This enables us to be naturally invariant under rotation and translation of the described object. Given a network that is able to produce valid Euclidean distance matrices we introduce a Wasserstein GAN that can learn to one-shot generate distributions of point clouds irrespective of their orientation, translation, or permutation. The permutation invariance is achieved by incorporating the message passing neural network SchNet as critic.

We applied the introduced WGAN to the $\mathrm{C_7O_2H_{10}}$ isomer subset of the QM9 molecules database and could generalize out of the training set as well as achieve a good representation of the distribution of pairwise distances in this set of molecules.

In future work we want to improve on the performance of the network on the isomer subset as well as extend it to molecules of varying size and chemical composition. We expect the ideas of this work to be applicable for, e.g., generating, transforming, coarse graining, or upsampling point clouds.

REFERENCES

Abdo Y Alfakih, Amir Khandani, and Henry Wolkowicz. Solving euclidean distance matrix completion problems via semidefinite programming. *Computational optimization and applications*, 12(1-3):13–30, 1999.

Martin Arjovsky, Soumith Chintala, and Léon Bottou. Wasserstein generative adversarial networks. In *International conference on machine learning*, pp. 214–223, 2017.

Brian C Barnes, Daniel C Elton, Zois Boukouvalas, DeCarlos E Taylor, William D Mattson, Mark D Fuge, and Peter W Chung. Machine learning of energetic material properties. *arXiv preprint arXiv:1807.06156*, 2018.

J. Behler and M. Parrinello. Generalized neural-network representation of high-dimensional potential-energy surfaces. *Phys. Rev. Lett.*, 98:146401, 2007.

L. Bonati, Y.-Y.Zhang, and M. Parrinello. Neural networks-based variationally enhanced sampling. *Proc. Natl. Acad. Sci. USA, DOI: 10.1073/pnas.1907975116*, 2019.

Jon Dattorro. *Convex optimization & Euclidean distance geometry*. Lulu.com, 2010.

Michaël Defferrard, Xavier Bresson, and Pierre Vandergheynst. Convolutional neural networks on graphs with fast localized spectral filtering. In *Advances in neural information processing systems*, pp. 3844–3852, 2016.

S. Doerr and G. De Fabritiis. On-the-fly learning and sampling of ligand binding by high-throughput molecular simulations. *J. Chem. Theory Comput.*, 10:2064–2069, 2014.

Daniel C Elton, Zois Boukouvalas, Mark S Butrico, Mark D Fuge, and Peter W Chung. Applying machine learning techniques to predict the properties of energetic materials. *Scientific reports*, 8 (1):9059, 2018.

Niklas WA Gebauer, Michael Gastegger, and Kristof T Schütt. Generating equilibrium molecules with deep neural networks. *arXiv preprint arXiv:1810.11347*, 2018.

Niklas WA Gebauer, Michael Gastegger, and Kristof T Schütt. Symmetry-adapted generation of 3d point sets for the targeted discovery of molecules. *arXiv preprint arXiv:1906.00957*, 2019.

Justin Gilmer, Samuel S Schoenholz, Patrick F Riley, Oriol Vinyals, and George E Dahl. Neural message passing for quantum chemistry. In *Proceedings of the 34th International Conference on Machine Learning-Volume 70*, pp. 1263–1272. JMLR. org, 2017.

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, and A. Aspuru-Guzik. Automatic chemical design using a data-driven continuous representation of molecules. *ACS Cent. Sci.*, 4: 268–276, 2018.

Ian Goodfellow, Jean Pouget-Abadie, Mehdi Mirza, Bing Xu, David Warde-Farley, Sherjil Ozair, Aaron Courville, and Yoshua Bengio. Generative adversarial nets. In *Advances in neural information processing systems*, pp. 2672–2680, 2014.

John Clifford Gower. Euclidean distance geometry. *Math. Sci*, 7(1):1–14, 1982.

Edward J Griffen, Alexander G Dossetter, Andrew G Leach, and Shane Montague. Can we accelerate medicinal chemistry by augmenting the chemist with big data and artificial intelligence? *Drug discovery today*, 2018.

Gabriel Lima Guimaraes, Benjamin Sanchez-Lengeling, Carlos Outeiral, Pedro Luis Cunha Farias, and Alán Aspuru-Guzik. Objective-reinforced generative adversarial networks (organ) for sequence generation models. *arXiv preprint arXiv:1705.10843*, 2017.

Ishaan Gulrajani, Faruk Ahmed, Martin Arjovsky, Vincent Dumoulin, and Aaron C Courville. Improved training of wasserstein gans. In *Advances in Neural Information Processing Systems*, pp. 5767–5777, 2017.

Jon Paul Janet and Heather J Kulik. Predicting electronic structure properties of transition metal complexes with neural networks. *Chemical science*, 8(7):5137–5152, 2017.

Tero Karras, Timo Aila, Samuli Laine, and Jaakko Lehtinen. Progressive growing of gans for improved quality, stability, and variation. *arXiv preprint arXiv:1710.10196*, 2017.

Steven Kearnes, Kevin McCloskey, Marc Berndl, Vijay Pande, and Patrick Riley. Molecular graph convolutions: moving beyond fingerprints. *Journal of computer-aided molecular design*, 30(8): 595–608, 2016.

Thomas N Kipf and Max Welling. Semi-supervised classification with graph convolutional networks. *arXiv preprint arXiv:1609.02907*, 2016.

Nathan Krislock and Henry Wolkowicz. Euclidean distance matrices and applications. In *Handbook on semidefinite, conic and polynomial optimization*, pp. 879–914. Springer, 2012.

H. W. Kuhn. The hungarian method for the assignment problem. *Nav. Res. Logist. Quart.*, 2:83–97, 1955.

Haichen Li, Christopher R Collins, Thomas G Ribelli, Krzysztof Matyjaszewski, Geoffrey J Gordon, Tomasz Kowalewski, and David J Yaron. Tuning the molecular weight distribution from atom transfer radical polymerization using deep reinforcement learning. *Molecular Systems Design & Engineering*, 3(3):496–508, 2018a.

Yangyan Li, Rui Bu, Mingchao Sun, Wei Wu, Xinhan Di, and Baoquan Chen. Pointcnn: Convolution on x-transformed points. In *Advances in Neural Information Processing Systems*, pp. 820–830, 2018b.

R. T. McGibbon, K. A. Beauchamp, M. P. Harrigan, C. Klein, J. M. Swails, C. X. Hernández, C. R. Schwantes, L. P. Wang, T. J. Lane, and V. S. Pande. Mdtraj: A modern open library for the analysis of molecular dynamics trajectories. *Biophys J.*, 109:1528–1532, 2015.

Frank Noé, Simon Olsson, Jonas Köhler, and Hao Wu. Boltzmann generators - sampling equilibrium states of many-body systems with deep learning. *arXiv:1812.01729*, 2019.

Noel M O'Boyle, Michael Banck, Craig A James, Chris Morley, Tim Vandermeersch, and Geoffrey R Hutchison. Open babel: An open chemical toolbox. *Journal of cheminformatics*, 3(1):33, 2011.

N. Plattner, S. Doerr, G. De Fabritiis, and F. Noé. Protein-protein association and binding mechanism resolved in atomic detail. *Nat. Chem.*, 9:1005–1011, 2017.

Mariya Popova, Olexandr Isayev, and Alexander Tropsha. Deep reinforcement learning for de novo drug design. *Science advances*, 4(7):eaap7885, 2018.

Charles R Qi, Hao Su, Kaichun Mo, and Leonidas J Guibas. Pointnet: Deep learning on point sets for 3d classification and segmentation. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pp. 652–660, 2017.

Raghunathan Ramakrishnan, Pavlo O Dral, Matthias Rupp, and O Anatole von Lilienfeld. Quantum chemistry structures and properties of 134 kilo molecules. *Scientific Data*, 1, 2014.

João Marcelo Lamim Ribeiro, Pablo Bravo, Yihang Wang, and Pratyush Tiwary. Reweighted autoencoded variational bayes for enhanced sampling (rave). *J. Chem. Phys.*, 149:072301, 2018.

Lars Ruddigkeit, Ruud Van Deursen, Lorenz C Blum, and Jean-Louis Reymond. Enumeration of 166 billion organic small molecules in the chemical universe database gdb-17. *Journal of chemical information and modeling*, 52(11):2864–2875, 2012.

M. Rupp, A. Tkatchenko, K.-R. Müller, and O. A. Von Lilienfeld. Fast and accurate modeling of molecular atomization energies with machine learning. *Phys. Rev. Lett.*, 108:058301, 2012.

Benjamin Sanchez-Lengeling and Alán Aspuru-Guzik. Inverse molecular design using machine learning: Generative models for matter engineering. *Science*, 361(6400):360–365, 2018.

Isaac J Schoenberg. Remarks to maurice frechet's article"sur la definition axiomatique d'une classe d'espace distances vectoriellement applicable sur l'espace de hilbert. *Annals of Mathematics*, pp. 724–732, 1935.

Kristof Schütt, Pieter-Jan Kindermans, Huziel Enoc Sauceda Felix, Stefan Chmiela, Alexandre Tkatchenko, and Klaus-Robert Müller. Schnet: A continuous-filter convolutional neural network for modeling quantum interactions. In *Advances in Neural Information Processing Systems*, pp. 991–1001, 2017a.

Kristof Schütt, Pieter-Jan Kindermans, Huziel Enoc Sauceda, Stefan Chmiela, Alexandre Tkatchenko, and Klaus-Robert Müller. Schnet: A continuous-filter convolutional neural network for modeling quantum interactions. *Adv. Neural Inf. Process. Syst.*, pp. 991–1001, 2017b.

Kristof T Schütt, Farhad Arbabzadah, Stefan Chmiela, Klaus R Müller, and Alexandre Tkatchenko. Quantum-chemical insights from deep tensor neural networks. *Nature communications*, 8:13890, 2017c.

Kristof T Schütt, Huziel E Sauceda, P-J Kindermans, Alexandre Tkatchenko, and K-R Müller. Schnet–a deep learning architecture for molecules and materials. *The Journal of Chemical Physics*, 148(24):241722, 2018.

J. S. Smith, O. Isayev, and A. E. Roitberg. Ani-1: an extensible neural network potential with dft accuracy at force field computational cost. *Chem. Sci.*, 8:3192–3203, 2017.

James JP Stewart. Mopac: a semiempirical molecular orbital program. *Journal of computer-aided molecular design*, 4(1):1–103, 1990.

Cédric Villani. *Optimal transport: old and new*, volume 338. Springer Science & Business Media, 2008.

R. Winter, F. Montanari, F. Noé, and D.-A. Clevert. Learning continuous and data-driven molecular descriptors by translating equivalent chemical representations. *Chem. Sci.*, 10:1692–1701, 2019a.

R. Winter, F. Montanari, A. Steffen, H. Briem, F. Noé, and D. A. Clevert. Efficient multi-objective molecular optimization in a continuous latent space. *https://doi.org/10.26434/chemrxiv.7971101.v1*, 2019b.

Guandao Yang, Xun Huang, Zekun Hao, Ming-Yu Liu, Serge Belongie, and Bharath Hariharan. Pointflow: 3d point cloud generation with continuous normalizing flows. *arXiv preprint arXiv:1906.12320*, 2019.

Hongyi Zhang, Sashank J Reddi, and Suvrit Sra. Riemannian svrg: Fast stochastic optimization on riemannian manifolds. In *Advances in Neural Information Processing Systems*, pp. 4592–4600, 2016.

J. Zhang, Y. I. Yang, and F. Noé. Targeted adversarial learning optimized sampling. *ChemRxiv. DOI: 10.26434/chemrxiv.7932371*, 2019.