JCIM JOURNAL OF CHEMICAL INFORMATION AND MODELING

pubs.acs.org/jcim Article

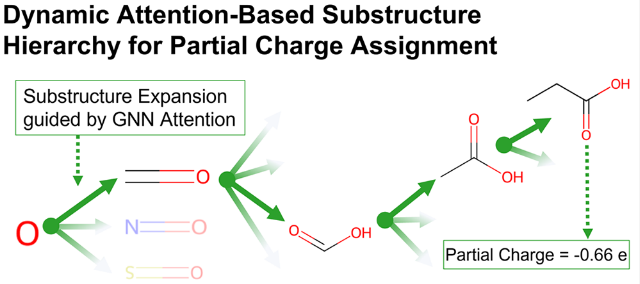
DASH: Dynamic Attention-Based Substructure Hierarchy for Partial Charge Assignment

Marc T. Lehner,[‡] Paul Katzberger,[‡] Niels Maeder, Carl C.G. Schiebroek, Jakob Teetz, Gregory A. Landrum, and Sereina Riniker*

Cite This: J. Chem. Inf. Model. 2023, 63, 6014−6028 Read Online

ACCESS | Metrics & More | Article Recommendations | Supporting Information

ABSTRACT: We present a robust and computationally efficient approach for assigning partial charges of atoms in molecules. The method is based on a hierarchical tree constructed from attention values extracted from a graph neural network (GNN), which was trained to predict atomic partial charges from accurate quantum-mechanical (QM) calculations. The resulting dynamic attention-based substructure hierarchy (DASH) approach provides fast assignment of partial charges with the same accuracy as the GNN itself, is software-independent, and can easily be integrated in existing parametrization pipelines, as shown for the Open force field (OpenFF). The implementation of the DASH workflow, the final DASH tree, and the training set are available as open source/open data from public repositories.


Dynamic Attention-Based Substructure Hierarchy for Partial Charge Assignment
Substructure Expansion guided by GNN Attention
O
O
N
O
S
O
O
OH
O
OH
OH
O
Partial Charge = -0.66 e

INTRODUCTION

Molecular dynamics (MD) simulations enable the time-resolved study of molecular systems and are, therefore, widely used in biology, chemistry, and material science. The physical interactions between the particles in the system are thereby approximated by a set of potential-energy functions (i.e., the force field).[1−6] Applying Newton's equation of motion allows the propagation of the system through time. The quality of the MD simulations is determined by the approximations made in the functional form of the force-field terms as well as their parameters. For biomolecular simulations, fixed-charge atomistic force fields are predominantly used due to their reasonable accuracy and low computational cost.[6] In such force fields, the contributions are split into bonded terms (i.e., bond stretching, bond-angle bending, and dihedral-angle torsion) and nonbonded terms (i.e., electrostatic and van der Waals).[6] The nonbonded terms describe the intermolecular interactions, which are directly related to experimental observables such as the density or the heat of vaporization of a compound. The calculation of the nonbonded interactions between the atoms in the system constitutes the computationally most expensive part of every classical MD simulation. While the van der Waals forces decay quickly with increasing distance between the atoms, the long-range contribution of the electrostatic forces is non-negligible, and many schemes have been developed for their efficient treatment (e.g., Ewald summation[7] based methods such as smooth particle mesh Ewald[8] or reaction-field (RF) correction[9]). The slow decay of the electrostatic forces also means that small changes in the parameters (i.e., partial charges, dielectric constant) can lead to large changes in the potential energy.

Molecules can only have integer charges, but even in a simple Lewis representation, the assignment of atomic formal charges can be ambiguous since they are not experimentally measurable and resonance structures can exist. Nevertheless, many techniques have been developed over the past decades to determine atomic partial charges, which can be used to predict chemical reactivities or perform MD simulations, among other applications. Early examples of such models include Gasteiger charges,[10] Hirshfeld-type charges,[11] Merck molecular force field (MMFF) charges,[12] and Mulliken-type charges.[13] The partial charges are extracted from a quantum-mechanical (QM) calculation (e.g., Hartree−Fock (HF), density functional theory (DFT), or semiempirical methods) and/or fitted to reproduce experimental properties. Mulliken-type charges, for instance, are calculated by integrating the electron density over the volume of the atoms. One of the most commonly used representatives from this family is AM1 population charges, which employs the semiempirical method AM1[14] for the QM calculation. Additional bond charge corrections (BCCs) are then applied to better reproduce the electrostatic potential (ESP) calculated with the more accurate HF

Received: May 25, 2023
Published: September 22, 2023

https://doi.org/10.1021/acs.jcim.3c00800

method.[14] The resulting AM1-BCC model is a reasonably fast and reliable method, which is used in classical force fields such as the general AMBER force field (GAFF)[15] and OpenFF.[16,17] Recently, progress has been made with atoms-in-molecule (AIM) charges such as DDEC[18] and MBIS,[19] showing that they are more accurate in reproducing ESP surfaces than AM1-BCC charges. However, the higher accuracy comes with increased computational costs. Additionally, the accuracy depends on the level of theory and basis set used in the underlying QM calculation, introducing hard limits on the feasibility for larger molecules like proteins. The computational time needed to extract partial charges also matters if the number and/or size of molecules is large, like in enzyme screens, where nonbonded interactions are used as features for substrate prediction,[20] or for large virtual screening runs in drug discovery.[21]

Since accurate AIM charges are computationally expensive and scale poorly with the number of atoms, alternatives based on machine learning (ML) have been explored in recent years. These attempts range from simpler regression[22] or random forest models[23] to more complex graph neural networks.[24−30] Bleiziffer et al.[23] showed that a random forest model trained on DDEC partial charges (TPSSh/def2-TZVP level of theory with an implicit solvent) from 130,000 molecules could predict partial charges of unseen molecules reasonably well with a root-mean-square error (RMSE) of 0.03 e. More recent approaches have explored the usage of different charge models as well as other ML techniques.[26−28,30] All of these ML approaches predict partial charges with good accuracy while offering a drastic increase in speed relative to performing a separate QM calculation for each new molecule and, in the case of ESPALOMA,[29,30] even the integration into a classical force field. However, the ML models are generally not interpretable; there is a risk of overfitting, and most models do not provide uncertainties with their predictions. In addition, these models are highly dependent on the correct featurizers and library versions, which often do not have long-term stability in the rapidly evolving field of machine learning.

In an attempt to peer into the black box of ML models, explainable artificial intelligence (AI) tools have been developed (e.g., LIME[31] or SHAP[32]) to explain the predictions of a model in a retrospective manner. These tools are often based on the idea of local linear approximations. A recent addition is the GNNExplainer[33] for graph neural networks (GNNs), where each neighbor of a certain atom is assigned an attention value, representing the importance that this neighbor has in the prediction of the value for the given atom. There are many different ways to get such an attention score. GNNExplainer is a stochastic explainer, randomly generating subgraphs and comparing the predictions of the model on these subgraphs to the predictions on the full graph. This provides the advantage that the approach is agnostic to the architecture of the model. While such explanation-based methods are able to explain a specific prediction and assign a measurement of importance to each neighboring atom, they are also computationally expensive due to the iterative and stochastic learning of the method, presenting a challenge for large data sets.

In this work, we train a GNN on a substantially increased data set of QM reference partial charges compared to ref 23. We demonstrate that the attention values extracted from this GNN model are in agreement with common chemical knowledge. Unlike chemical intuition, however, the attention values are quantitative, enabling us to rank certain atoms and functional groups over others. Thus, we can not only extract the important features as patterns but also use the attention values to construct a dynamic hierarchical tree structure to assign partial charges without the GNN model. The resulting dynamic attention-based substructure hierarchy (DASH) is independent of the ML software library with which the model was built and provides accuracy similar to that of the underlying GNN. Moreover, the DASH is human-readable and provides confidence values for each result.

■ METHODS

Data Set Generation. A generally applicable force field for organic molecules needs to cover a large chemical space including the combination of functional groups. In ref 23, we generated a data set with a large coverage of chemical space while focusing on lead-like compounds (molecular weight in the range 250−350 g/mol), such that the molecules were small enough for high-level QM methods. We used the unique bits of Morgan fingerprints with a radius of 2 (MFP2) of all lead-like compounds in ChEMBL[34] and ZINC[35] to select a diverse subset of 130,000 molecules that represented all MFP2 bits. At the time, we considered only one conformer per molecule since the conformational dependence of DDEC partial charges was found to be low overall. However, this can introduce noise for molecules for which the conformational dependency of the charge assignment is above average. This is, for instance, the case for molecules that are symmetric in the two-dimensional (2D) graph but are asymmetric in the three-dimensional (3D) conformation. In the recently published QMugs data set[36] three conformers were included for each of the 200,000 molecules, using a semiempirical method for geometry optimization and DFT for the calculation of the QM properties.

Selection of Molecules. In this work, we generated an extended data set by collecting and filtering molecules from four different sources: (i) the QMugs data set,[36] (ii) the training set from ref 23, (iii) lead-like molecules from ChEMBL version 30 (filtered as in ref 23), and (iv) organic liquids from refs 37−40. The goal was to have the minimal number of molecules that represent all unique MFP2 atom environments found in the lead-like compounds of ChEMBL at least five times.

First, the QMugs data set was filtered by removing larger molecules (molecular weight >500 g/mol), which are impractical for high-level QM calculations, and by iteratively removing molecules for which the bits of their MFP2 fingerprint were already represented at least five times by the other molecules in the data set. Note that the molecules in the QMugs data set have a neutral formal charge, but molecules can be zwitterions, i.e., contain a positively and negatively charged functional group. With the QMugs subset at hand, molecules from other sources were added iteratively if their MFP2 fingerprints contained new bits. For this, the molecules were sorted by the number of "unseen bits" in their MFP2 fingerprint, and the list was updated after each addition of a molecule. Finally, we observed that the data set did not contain a diverse enough set of charged nitrogen environments (e.g., protonated amines), thus 21 manually selected molecules with charged nitrogen-containing functional groups (but a net zero charge) were added.

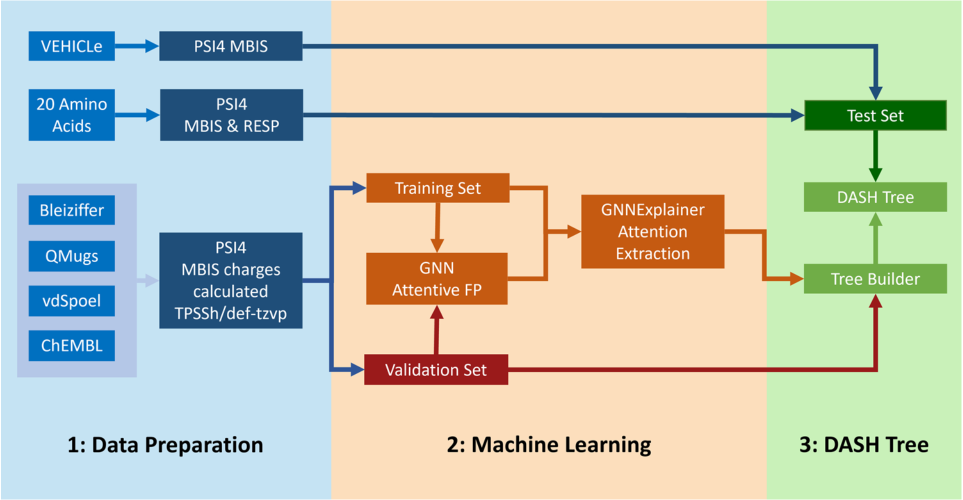


Figure 1. Schematic depiction of the workflow to construct the DASH tree structure: (1: Data Preparation) Reference charges were calculated for molecules from multiple data sets (blue). (2: Machine Learning) Molecules were split into a training set (orange) and a validation set (red). The training set was the input for training an Attentive FP GNN model to learn partial charges. The GNN was in turn the input for the GNNExplainer to extract attention values on the training set molecules. (3: DASH Tree) These data were subsequently used to construct the DASH tree structure. The GNN and DASH tree employed the same validation set. The test set (dark green) was used only for verification of the final DASH tree.

The final data set contains 348,935 molecules with elements from the organic set (C, H, N, O, P, S, Cl, Br, I, F, B) and a molecular weight of up to 500 g/mol.

Conformer Generation and Extraction of Atomic Partial Charges. For the molecules originating from the QMugs data set, all three semiempirically optimized conformers were considered. For the other molecules, three conformers were generated with a similar workflow as in ref 36. The ETKDG conformer generator[41] as implemented in the RDKit[42] was used to generate three diverse conformers. The three conformers of a compound were treated as separate molecules in the workflow, except when splitting the data set into training and test sets; i.e., all conformers of a given molecule were always assigned to the same split. The conformers were first optimized with the MMFF94 force field[12] as implemented in RDKit[43] for an initial relaxation. These conformers were further optimized with the semiempirical method xTB-GFN2[44] for 100 cycles with an implicit solvent (dielectric permittivity $\epsilon = 4.9$) using the software package PSI4.[45] The choice of this implicit solvent was based on ref 23, where we showed that this dielectric permittivity leads to partial charges most compatible with the van der Waals parameters of existing force fields (compared to $\epsilon = 1$ (vacuum) or $\epsilon = 78$ (water)). A single point DFT calculation was performed for each optimized conformer with the TPSSh functional[46] and a def2-TZVP basis set[47,48] in PSI4. The choice of functional and basis set is the same as in ref 23, where a small benchmarking of functionals and basis sets was performed. The PCM implicit solvent model was used with chloroform as an implicit solvent. MBIS charges were calculated with the `oeprop` function in PSI4 with the wave function from the single point TPSSh calculation, with at most 300 iterations, 10^{-4} a.u. as convergence value, 75 radial points, and 302 spherical points. Conformers with nonphysical partial charges or charges that disagreed with the cheminformatics expectation were filtered out. The filter was defined by selecting partial-charge ranges for each element type and discarding conformers with atoms with partial charges outside the range. In a second step, we used the difference between the partial charge of the same atom in the three conformers to discard conformers with differences larger than 0.4 e.

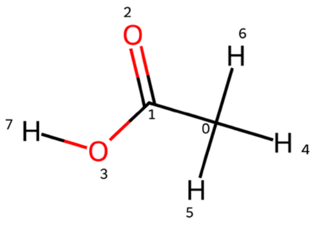
Training of the Graph Neural Network. *Model Architecture.* The model architecture was based on the first two layer types of the Attentive FP network developed by Xiong et al.[49] (i.e., the input layer and the attention layer for atom embedding) and a three-layer multilayer perceptron (MLP)[50] with ReLu activation functions.[51] The atomic features are first passed through the input layer followed by five layers of the attention layer for the atom embedding type and are then decoded by the MLP. In a final step, the predicted partial charges q_i are normalized such that the sum of all partial charges is an integer (i.e., the formal charge q_{formal} of the molecule). This was achieved by subtracting the average predicted partial charge of a molecule from each partial charge and adding the formal charge normalized to the number of atoms (eq 1). The latter term is necessary for formal charges ≠ 0. A size of 200 was chosen for all hidden layers.

$$q_i' = q_i + \frac{q_{\text{formal}}}{N_{\text{atoms}}} - \frac{1}{N_{\text{atoms}}} \sum_i^{N_{\text{atoms}}} q_i \quad (1)$$

where N_{atoms} is the number of atoms (partial charges) in the molecule.

For the atom and bond embedding, an adapted version of the features proposed by Kearnes et al.[52] was used. Atoms were encoded by creating a feature vector of length 23 containing element type (i.e., C, N, O, F, P, S, Cl, Br, I, B, or H), formal charge, hybridization (i.e., SP, SP2, or SP3), aromaticity, and degree (i.e., 0, 1, 2, 3, 4, 5, or other). Bonds were encoded by creating a feature vector of length 11 containing the bond type (i.e., single, double, triple, or aromatic), whether the bond is in a ring, whether the bond is conjugated, and stereo code following the RDKit[42] definition (i.e., STEREONONE, STEREOANY, STEREOE, STEREOZ, or other).

Training Procedure. The GNN was trained on all available conformers of a randomly selected 90% subset (976081 3D structures) using the Adam optimizer[53] for 100 epochs. The mean square error was chosen as the loss function. The effects of different learning rates (i.e., 0.0001, 0.00001, and 0.000001)



Level	Index in Molecule	Atom type	(Atom type) Key	Key with Connectivity
0	2	O 1 0 T 0	80	[80, -1, -1]
1	1	C 3 0 T 0	26	[26, 0, 4]
2	3	O 2 0 T 1	86	[86, 1, 4]
2	0	C 4 0 F 3	34	[34, 1, 1]
...	...	...	...	...

Figure 2. Example of the feature vectors/atom types of acetic acid. The oxygen atom with index 2 is selected as the root of the subgraph (level 0). Therefore, its connection information is nonexistent and set to −1. The second atom to be added is the carbon with index 1 and an atom type 26, which is connected to atom 0 in the subgraph with a conjugated double bond (type 4). Next, a choice has to be made between the oxygen with index 3 and the carbon with index 0 based on the attention values.

and batch sizes (i.e., 32, 64, 128, 256, and 512) were studied in a hyperparameter optimization. The remaining 10% of the data set (100,171 3D structures) was used as a validation set during training of the GNN. The same split was later used for the DASH tree construction (see Figure 1).

Extraction of the Attention Values. The attention values of the trained GNN were extracted with GNNExplainer from PyTorch Geometric.[33] GNNExplainer takes as input the trained graph-based model, the data for which the attention should be extracted, and the number of epochs on which the GNNExplainer model should be run in order to generate attention values. The number of epochs was set to 500, the learning rate was set to 0.01, and the return type was set to the default value of log_prob. These values were found to give a good performance of the model, and no systematic parameter search was performed. The attention values were then extracted for all atoms in all molecules in the training set. Note that the attention values are not directly normalized per molecule. To enable a comparison of values between molecules, we divided the attention value of each atom in a molecule by the sum of all attention values in the molecule.

For a given atom, the neighboring atoms contributing most to the prediction of its partial charge can be identified using either an attention threshold or a fixed number of atoms (environment size). The subgraphs (or substructures) of a molecule extracted in this manner can be compared to chemical intuition and can be processed further to generate a substructure-based table (i.e., using SMARTS or SMILES) to assign the atomic partial charges of a molecule. The choice of the metaparameter (attention threshold or the number of atoms) determines the performance of such an assignment table (accuracy vs speed of assignment).

Dynamic Attention-Based Substructure Hierarchy (DASH). To circumvent the issues associated with a static cutoff (in either the number of atoms or the attention), we propose a dynamic attention-based substructure hierarchy (DASH), where each node corresponds to a certain atom type, the neighbors of a node are ordered by attention, and branches can have different depths (dynamic). This way, the attention values can be used to linearize the search through the exponentially growing number of possible patterns. To parametrize a particular atom in a molecule, a subgraph (substructure) of the molecule is grown starting from this atom by iteratively adding the neighboring atom with the highest attention value to the subgraph until a user-defined depth is reached. Note that no attention values have to be calculated for new molecules; the partial-charge assignment occurs by looking up the environments of each atom in the DASH tree.

Atom Features. To build the subgraphs, we define a feature vector for each atom that contains the necessary information to identify the atom type. While the same features as in the GNN training could be used, the feature vector for DASH should be as small as possible to reduce the number of possible patterns and improve human readability. We wanted to be able to calculate the atom features easily and rapidly from an RDKit molecule. Thus, we decided for an atomic feature vector with the following information:

- Element type (H, C, N, O, S, F, Cl, Br, P, I, B)
- Number of bonds (1, 2, 3, 4, 5)
- Formal charge (−1, 0, 1)
- Is conjugated (True or False)
- Number of attached hydrogens (0, 1, 2, 3)

The conjugated flag is set to true for an atom if at least one of its connecting bonds is conjugated. This definition leads to 122 possible initial atom types, since many combinations of these properties are not physical or not present in our data set of organic molecules. For example, a hydrogen atom with one bond and a formal charge of zero has the atom type "H 1 0 False 0". These feature vectors are further translated to simple integers (keys) and stored in a dictionary object.

When the subsection is extended by one atom during the DASH construction, the atom with the highest attention value is added. The feature vector for the new atom contains additional information about how it is attached to the current subgraph (i.e., relative index in the subgraph and bond type). The bond type is an integer with values 1 (single), 2 (double), 3 (triple), or 4 (conjugated, independent of whether single or double bond). For the hydrogen dimer H_2 as example, the second hydrogen atom has the feature vector [37, 0, 1] (atom-type key with connectivity), where the first element is the key of the atom type in the dictionary, the second element is the level of the atom in the subgraph (the root of the subgraph has level 0), and the third element is the bond type. The key with connectivity information can be used as an identifier for a node (atom) chained together to generate a chemical pattern. For a more complex example, the atom feature vectors of acetic acid are shown in Figure 2.

DASH Implementation as Tree Structure. Level 0 of the tree consists of the 122 nodes, one for each atom type, which branch out from the root. Every node stores the key of the atom type together with the GNN attention value and computes the partial charge of every atom with that type in the training set. Level 0 could already be a simple lookup table for partial charges for a force field by simply averaging the partial charges at each node. This simple approach would, however, ignore most of the information about the environment of the atom and result in fairly crude partial charges.

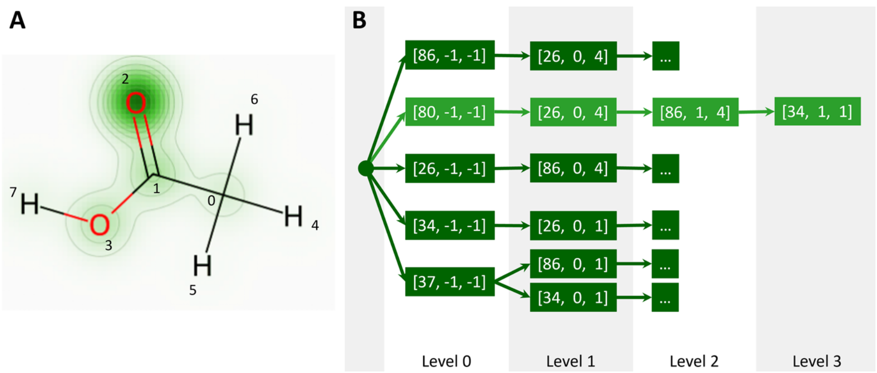


Figure 3. Example of acetic acid. (**A**) Attention values for the carbonyl oxygen atom (index 2, level 0) are shown as a heatmap overlaid with the molecule. (**B**) DASH tree if it were constructed based only on acetic acid. In light green, the subgraph starting at the oxygen atom (index 2) is highlighted, with the atom types and connectivity information from Figure 2. The remaining nodes of the DASH tree are colored dark green. For clarity, nodes in higher levels are omitted, as indicated with "⋯". In this simple example, all heavy atoms could be uniquely identified on level 0. The branching of the hydrogens (atom type 37) into the two different subgraphs occurs in level 1.

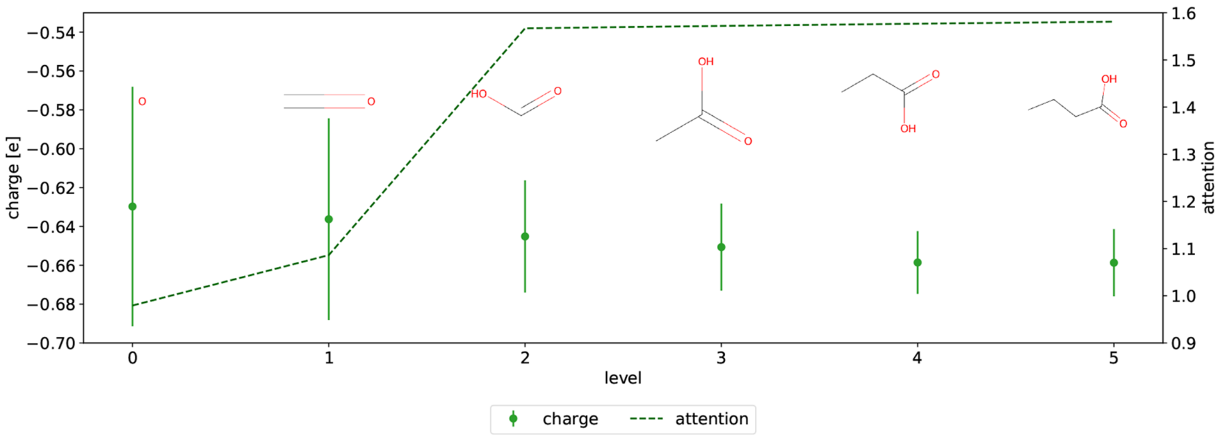


Figure 4. Example of the DASH charge assignment for matching "═O" in the molecule CCCC(═O)O. By traversing the levels of the DASH tree, a fragment for the charge assignment is built up. The partial charge at each level is denoted by a green dot with a standard deviation (left *y*-axis). The attention value is shown with a green dashed line (right *y*-axis).

The accuracy can be improved by taking larger substructures into account, thus adding more information about the atomic environment. In the DASH approach, this is done by adding neighboring atoms in the order of decreasing attention values until the maximum graph depth is reached or all of the atoms in the training molecule have been added. Since each node in the tree stores not only a list of partial charges (from which an average charge per node can be calculated) but also the attention values, the attention could be used as an early stopping condition during the construction procedure to avoid overfitting of the tree to the training data set (not done here). A pseudocode implementation of the algorithm is provided in the Supporting Information. Figure 3 shows an example DASH tree if it were constructed based on only one molecule (acetic acid). The final DASH tree was built with the same training data set as used for the GNN training to avoid any mixing of the training and validation sets.

Normalizing DASH Partial Charges. As each atom is considered individually in the DASH assignment process, the resulting partial charges do not necessarily sum up exactly to the formal charge on the molecule. We explored two normalization schemes to address this issue. The first scheme calculates the difference between the sum of the partial charges and the formal charge on the molecule (eq 2), divides this by the number of atoms, and adds the result to the partial charge of each atom (eq 3). The second normalization scheme makes use of the standard deviation of the partial charges assigned to each atom (similar to the idea used in refs 22 and 23). Here, the difference between the sum of the partial charges and the formal charge is distributed across the atoms using weights derived from the standard deviations of the partial charges assigned by DASH (eq 4).

$$\Delta Q = \sum_{i=0}^{N} Q_i - Q_{\text{formal}} \tag{2}$$

$$Q_i' = Q_i + \frac{\Delta Q}{N} \tag{3}$$

$$Q_i' = Q_i + \frac{\Delta Q \cdot \sigma_i}{\sum_{j=0}^{N} \sigma_j} \tag{4}$$

Here, Q_i is the partial charge assigned to atom i by the tree, Q_{formal} is the formal charge on the molecule with N atoms, and σ_i is the standard deviation of the partial charges in the leaf of the DASH tree corresponding to atom i from the tree. Both methods were tested on the validation set and compared to both the QM reference charges and the raw DASH partial charges.

Symmetrizing DASH Partial Charges. The QM reference charges are, per definition, dependent on the 3D conformation of the molecule. This means that topologically equivalent atoms can have different computed partial charges. This conformational dependency is, in principle, removed in the GNN due to the 2D input (topology); i.e., partial charges of topologically equivalent atoms will be averaged in the GNN predictions. However, asymmetries may be (re)introduced in the DASH assignment process because subgraphs are matched using a greedy approach (always adding the node with the highest attention). The degree of asymmetry can be tested using the RDKit CanonicalRankAtoms function to find atoms with the same rank (topologically equivalent) and comparing the partial charges from the QM reference calculation, the GNN, and DASH. Note that the DASH partial charges can be simply symmetrized by averaging the partial charges of the atoms with the same rank.

Assigning DASH Partial Charges for New Molecules. DASH partial charges of a new molecule are assigned by first matching each atom separately in the DASH tree structure. For each atom, the tree is traversed until either the maximal depth or the attention threshold is reached or the subgraph is equal to the size of the molecule. In Figure 4, the assignment process is shown for the double-bonded oxygen in butyric acid as an example. The subgraph is built up over six levels, where the nodes at each level contain partial charges with standard deviations and attention values (which are used as the stopping criterion). After all atoms have partial charges assigned individually, the atomic charges are normalized and symmetrized.

Note that the DASH tree is by design applicable only to molecules with a valid RDKit representation (Lewis structure) and with atom types present in the training set. For molecules outside of this applicability domain, an error is returned. If a specific atom type is missing and should be included, the MBIS calculation could be carried out for representative molecules, and the values could be added to the DASH tree. This behavior is comparable to other rule-based partial charge models, like the Gasteiger model[10] or classical force fields like MMFF94.[54]

OpenFF Plug-In. To integrate the DASH partial charges with the rest of the force-field assignment, the DASH tree structure was implemented as a NonBondedHandler in the OpenFF toolkit software[55] and can be installed as a plug-in. The plug-in, the stand-alone DASH tree, and the DASH tree constructor functions as well as the GNN are available as open source source code on GitHub (https://github.com/rinikerlab/DASH-tree).

Performance Assessment. The prediction accuracy with DASH was assessed with the same validation set as used for the GNN as well as two external test sets. The first external data set with the 20 canonical amino acids shows the potential applicability of DASH for biomolecular force fields. Two different meta-parameters were compared: the maximal depth of the DASH tree structure and the attention threshold when constructing DASH. In addition, simple pruning by the maximal depth or attention threshold was compared to a pruning scheme based on the standard deviation of the partial charges in the nodes. If the change in the average partial charge from parent to child was smaller than the standard deviation of all partial charges in the child nodes multiplied by a scaling factor, the node was pruned (eq 5).

$$\left(\frac{1}{N_{\text{parent}}}\sum_{i}^{N_{\text{parent}}} q_i - \frac{1}{N_{\text{child}}}\sum_{j}^{N_{\text{child}}} q_j\right) < s\cdot\sigma(q_i^{\text{parent}}) \tag{5}$$

Different scaling factors were tested.

The performance was assessed using the mean absolute error (MAE), the root-mean-squared error (RMSE), and the Pearson correlation coefficient R^2 compared to the QM reference partial charges. In addition, the computing time needed to assign the DASH charges and the size of the DASH tree itself was monitored.

In addition to the amino acid test set, MBIS charges were calculated for 24,657 molecules of the VEHICLe data set (virtual exploratory heterocyclic library),[56] which were not in the DASH data set. The MBIS charges were calculated with the same procedure as described above and matched with the DASH tree.

Other Partial-Charge Models. The DASH partial charges were compared with semiempirical Mulliken-type charges,[57] AM1-BCC charges,[58] 2D Gasteiger charges,[10] and MMFF94 partial charges.[54] The Mulliken-type charges were taken from the XTB-GFN2[44] conformer optimization step during data preparation. AM1-BCC charges were calculated with the OpenFF toolkit (version 0.10.0),[55] using the Amber toolkit (version 22.0).[59] The 2D Gasteiger and MMFF94 partial charges were obtained with RDKit[42] (version 2022.9.1). For the amino acid test set, restrained electrostatic potential (RESP) charges[60] calculated with PSI4 and PsiRESP (B3LYP/STO-3G, RESP2, and TPSSh/def2-TZVP) were also compared.

Liquid Properties: MD Simulations. MD simulations were performed for a set of 123 organic liquids with experimental values for density and heat of vaporization available.[37] The molecules were parametrized using the OpenFF toolkit[55] with OpenFF version 2.0.0 (Sage)[17] and the DASH plug-in. To evaluate the density and heat of vaporization, the openFF-evaluator[61] package was used, with the default schemes to estimate the two properties in the OpenMM[62] engine (version 8.0.0). The default scheme uses a box of 1000 molecules and consists of an energy minimization, an NPT equilibration (100,000 steps with 2 fs), and up to 100 NPT production runs (1,000,000 steps with 2 fs) until a convergence criteria is met, followed by a decorrelation step. All simulations in this scheme were performed with the Langevin integrator (298.15 K) and a Monte Carlo barostat (101.325 kPa) for the NPT simulations. The results were compared to simulations using AM1-BCC charges.

■ RESULTS AND DISCUSSION

Overview of the Data Set. The final data set contained 398,935 unique molecules with up to three conformers per molecule. These molecules were selected to represent the substructures (as measured by unique bits in MFP2) found in molecules with a maximum molecular weight of 500 g/mol in ChEMBL with a minimal subset of molecules. Figure 5 shows the number of data points per element. While the goal was that each bit is represented at least five times in the data set, a few MFP2 bits are only present once. These belong to very small molecules for which radius 2 describes the entire molecule (i.e., there exists exactly one molecule that can have this bit).

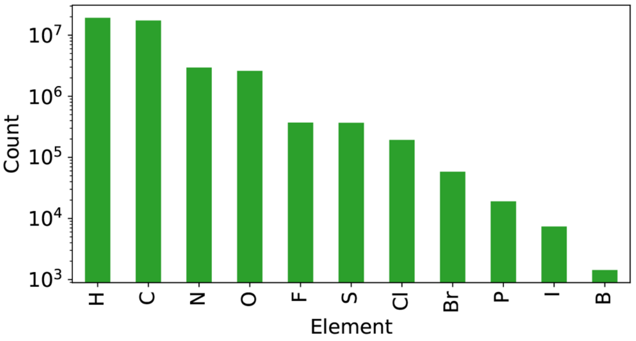


Figure 5. Atom counts per element in the full data set (398,935 unique molecules).

To estimate the conformational variation of the partial charges, we compared the differences in the MBIS reference charges between the three conformers of the same molecule. The RMSE of the individual conformer to the median over the three conformers is 0.0125 e, which presents a lower bound on the accuracy that can be reached by an ML model that uses the 2D topology of the molecule as input. A histogram of the absolute difference between the conformers can be found in Figure S1 in the Supporting Information.

The data set of 398,935 unique molecules (three conformers each, i.e., 1,029,785 3D structures in total) was split randomly into a 90% subset for training of the GNN and DASH, while the remaining 10% (100,171 3D structures) served as validation set. Note that the three conformers of a molecule were always kept together, i.e., either in the training set or in the test set.

GNN Performance. The architecture of the GNN was already optimized in ref 49; therefore it was kept constant in this study. The hyperparameters (learning rate from 0.000001 to 0.01 with 10-fold increments and batch size ranging from 64 to 512 with 2-fold increments) were screened to identify optimal values for the data set (Table S1 in the Supporting Information). A learning rate of 0.0001 and a batch size of 64 yielded the GNN model with the smallest RMSE on the validation set (left panel of Figure 6). The distribution of the absolute differences between the GNN prediction and the MBIS reference reaches an accuracy similar to the conformational variation limit. The direct comparison is provided in the right panel of Figure 6.

DASH Performance. After the GNN was successfully trained and tested, the attention values were extracted, and the DASH tree structure was constructed using the training set of the GNN. The validation set was used to evaluate the performance of DASH and to tune its hyperparameters: maximal depth and attention threshold. A tree with a maximal depth of 16 layers and an attention threshold of 5.23 was found to perform well on the validation set (Figure 7). The RMSE as a function of the maximal depth is provided in Figure S2 in the Supporting Information. The same figure also shows that the time to assign partial charges increases roughly linearly with the maximal depth. The choice of this hyperparameter is thus a trade-off between accuracy and speed of assignment.

The right panel of Figure 7 shows the RMSE values of the DASH partial charges with respect to the MBIS reference charges on the validation set for each element. The RMSE values are generally very small. The largest RMSE values are observed for phosphorus, which is particularly difficult for charge assignment because it shows a large range of partial charges and is under-represented in the data set (and generally in ChEMBL). Normalizing the DASH charges with eq 4 reduces the errors slightly. Using eq 3 instead gives very similar results (Figure S3 in the Supporting Information). As integer values for the total charge of a molecule are important for MD simulations, we used normalization with eq 4 in the remainder of this work.

The effect of the depth of the DASH tree structure can also be seen in Figure 8, which shows the distribution of the nodes over the range of partial charges at different levels of the DASH tree. Note that level 0 consists of the 122 initial atom types, which can be seen as discrete bars in the histogram. At level 1, the possible partial charges have already a much larger set of possible but still clearly separated values. This strong discretization becomes further refined with an increasing depth. At the maximum depth of 16, there is a nearly

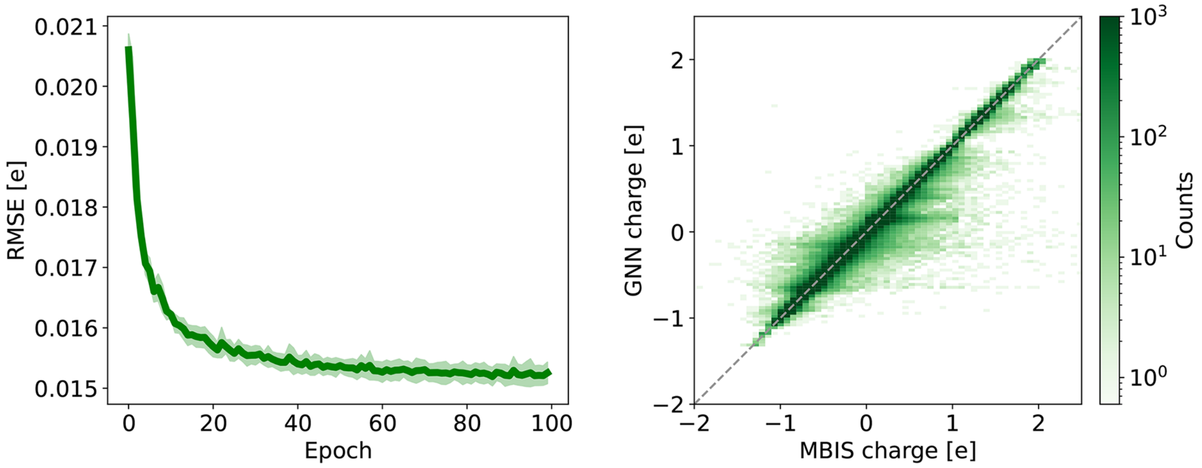


Figure 6. (Left) RMSE of the GNN predicted charges with respect to the MBIS reference charges on the validation set (100,171 3D structures) as a function of the training epoch. The GNN was trained with a learning rate of 0.0001 and a batch size of 64 (final RMSE = 0.0153 ± 0.0002 e). The shaded area indicates the statistical uncertainty. (Right) Comparison between the GNN predicted charges (100 epochs) and the MBIS reference charges on the validation set.

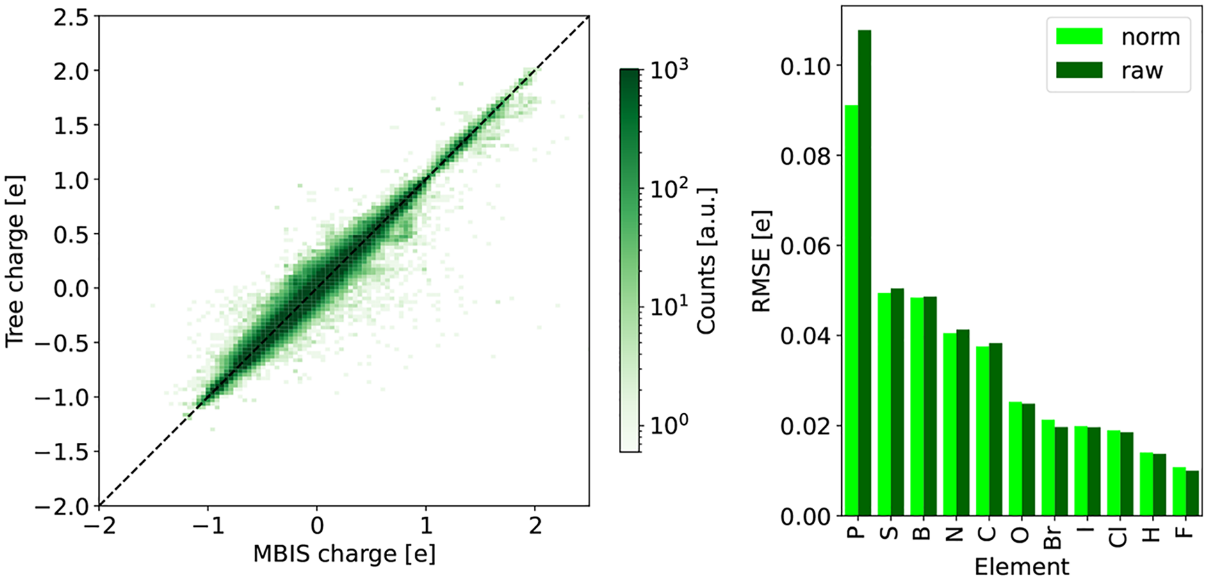


Figure 7. (Left) Comparison between the normalized DASH partial charges and the MBIS reference charges on the validation set (100,171 3D structures). A maximal depth of 16 layers and an attention threshold of 5.23 were used to construct the DASH tree structure. The colors of the points indicate the number of atoms in a pixel. (Right) RMSE of the DASH partial charges with respect to the MBIS reference charges for each element in the validation set. The "raw" DASH charges are shown in dark green, and the normalized ones (eq 4) are in light green.

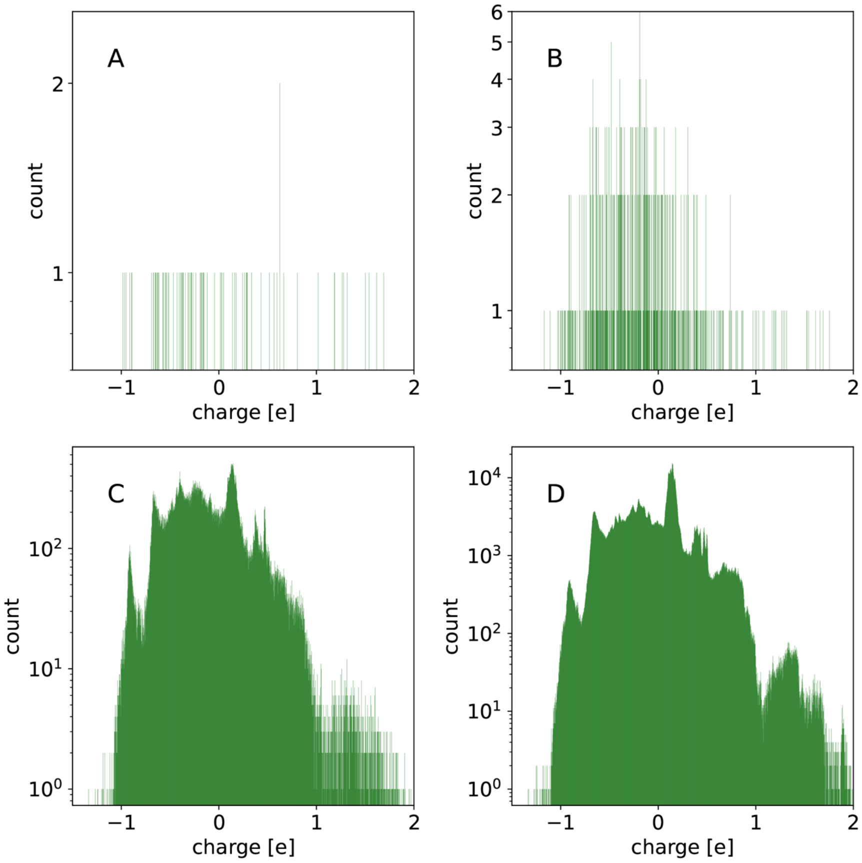


Figure 8. Distribution of the nodes in the DASH tree structure over the range of partial charges for the tree level = 0 (A), 1 (B), 8 (C), and 16 (D). A bin size of 1/2000 e was used.

continuous distribution of the 4 × 10^6 nodes over the full range of possible partial charges.

A similar trend from a more discrete to a more continuous prediction of the partial charge can be seen in Figure 9 as a function of the attention threshold. Note that the attention in the nodes is not strictly confined to the range of 0 to 1 because nodes can contain information from multiple molecules, and the attention is normalized in the GNN over all atoms in a

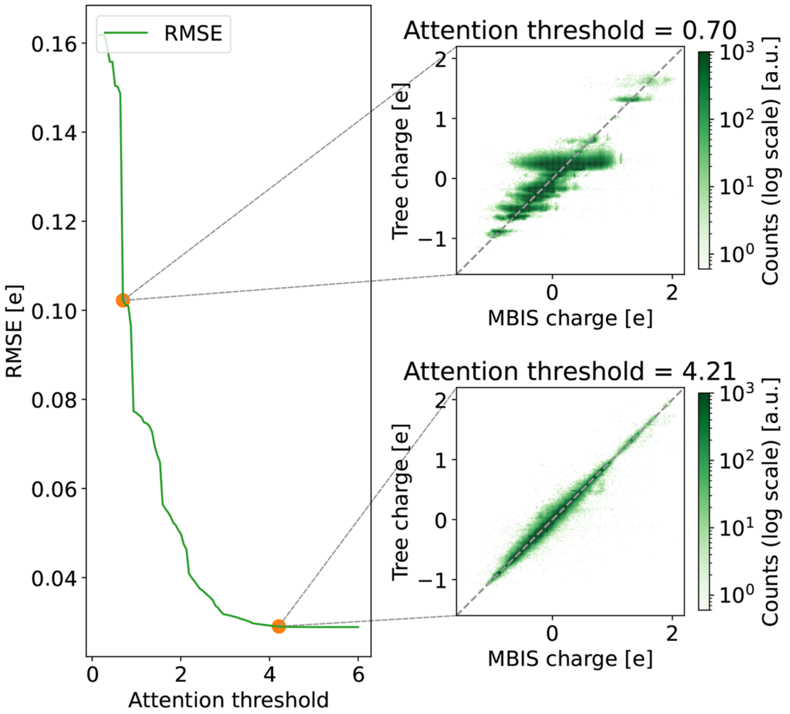


Figure 9. RMSE of the DASH partial charges with respect to the MBIS reference charges on the validation set as a function of the attention threshold. The minimum RMSE is at an attention threshold of 5.2. For two attention thresholds (0.7 and 4.21), a comparison between the DASH partial charges and the MBIS reference charges is shown on the right. A maximal depth of 16 was used for the DASH tree.

molecule. The RMSE of the DASH partial charges with respect to the MBIS reference charges on the validation set decreases with increasing attention threshold, reaching a minimum at around 5.2 (the exact value will depend on the training set). Interestingly, the RMSE converges to a value slightly above the minimum when the attention threshold is increased further. The fact that additional information does not improve the accuracy anymore is an indication that the tree is already able to capture all relevant information at this point and that additional information leads to overfitting.

The initial attention values of the 122 atom types (nodes at level 0 in the DASH tree) are shown in Figure S4 in the Supporting Information. Our assumption is that atom types with a high initial attention need little additional information from the environment for a good prediction of the partial charge, while atom types with low attention values require larger subgraphs for a precise prediction. It is possible to observe some chemical trends by comparing the selected atom types. For example, the attention of the four atom types for the halogens show that fluorine atoms have the steepest increase in attention with increasing depth of the DASH tree, while iodine atoms have the slowest initial increase in attention (Figure 10). This slower increase in attention may be explained by the lower hardness of iodine compared to fluorine and therefore the stronger influence of the environment on the partial charge.

When assigning the partial charge of an atom with the DASH tree structure, a greedy approach is followed; i.e., at each step the neighboring atom with the highest attention value is added to the subgraph. This can mean that topologically symmetric atoms may have different partial charges assigned. However, such asymmetries are rare and small (Figure S5 in the Supporting Information). To resolve this issue, a symmetrization step (i.e., averaging the partial charges of the symmetric atoms) was added after the normalization. This does not decrease the RMSE significantly (reduced by 0.12%) on the validation set. It should be noted here that the symmetry is mostly conserved for atoms that are equivalent on a QM level of theory, even if they are not equivalent in their Lewis representation. For example, the oxygen atoms of a nitro group will have the same partial charge (inside the confidence interval) even though their representation and therefore matched subgraph is different.

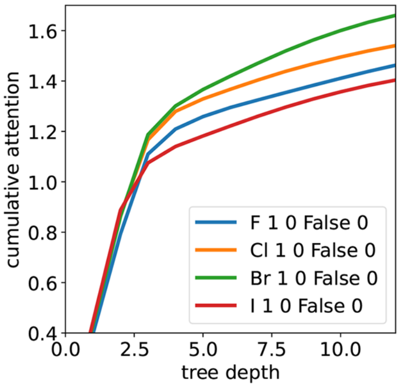


Figure 10. Cumulative attention of the nodes as a function of the DASH tree depth for the four halogen atom types.

Comparison with Other Partial-Charge Models. First, we compared the accuracy of the DASH partial charges on the validation set to the performance of the AM1-BCC, Gasteiger, and MMFF94 methods (Figure 11). Unsurprisingly, the Gasteiger partial charges have the lowest accuracy and are clearly not suited for MD simulations of condensed-phase systems. The weaker polarization results in a narrower range of

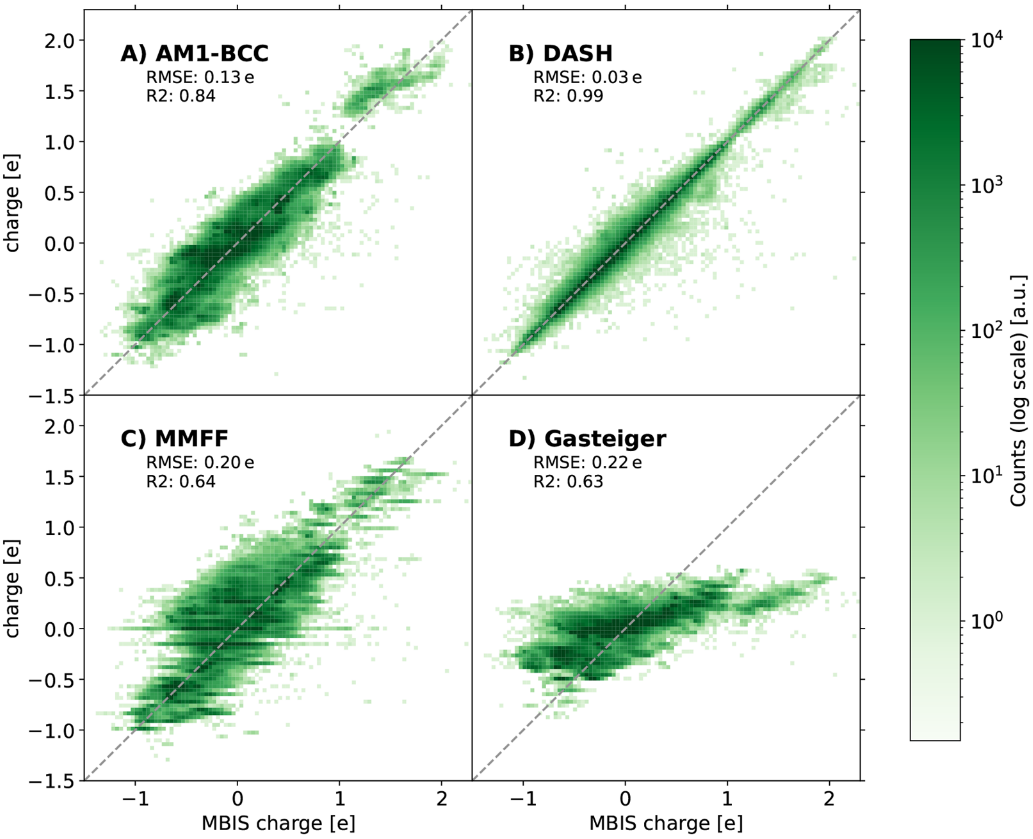


Figure 11. Comparison between the estimated partial charges with AM1-BCC (**A**), DASH (**B**), MMFF94 (**C**), and Gasteiger (**D**) and the MBIS reference charges on the validation set. The corresponding RMSE and R^2 values are given in the subplots.

partial charges (approximately between −0.5 e and +0.5 e), while AM1-BCC and MMFF94 charges are reasonably close to the MBIS reference charges. Both methods show larger deviations than the DASH partial charges (part of this may be because they were fitted to other reference charges), especially for slightly charged carbons in large conjugated systems where the partial charge is influenced by far away atoms. Interestingly, some discretization effects can be observed for the MMFF94 charges (visible as horizontal lines in the figure) due to the limited number of atom types in MMFF94.

Next, we compared the different partial-charge models for an external test set that consists of 20 canonical amino acids. The motivation behind this data set is the use of the DASH partial charges in protein simulations in the future. For this smaller data set, we calculated also RESP charges and Mulliken-type charges from the XTB-GFN2 optimization. The results are shown in Figure 12. The Mulliken-type charges from the XTB-GFN2 optimization show a similarly narrow range as the Gasteiger charges due to smaller polarization, which indicates that they are also not suited for fixed-charge MD simulations of condensed-phase systems. RESP charges show overall a similar behavior to the MBIS charges, although deviations can be observed for some atoms. Reasons for this could be the different functional and basis set typically used for RESP (B3LYP/STO-3G) compared to the MBIS charges extracted in this work and/or the stronger conformational dependency of RESP compared to MBIS.[19] To exclude the first reason, we calculated RESP charges also with TPSSh/def2-TZVP as used for the MBIS charges. The differences were minimal (Figure S6 in the Supporting Information) compared with the conformational dependency.

As a second external test set, we considered the VEHICLe set,[56] which contains 24,657 small heterocyclic molecules with challenging atom environments for partial-charge assignment. In Figure 13, the different charge models are shown against the MBIS reference charges. The DASH charges still perform well with an RMSE of 0.09 e. A more detailed analysis of the charges per element type can be found in Figure S7 in the Supporting Information. Also for this test set, sulfur atoms are some of the largest outliers, which could potentially be improved by including more diverse sulfur environments in the DASH training set. The findings for the other charge models are the same as those discussed above for the other test sets.

Liquid Properties. Finally, we tested the combination of DASH partial charges with the OpenFF-2.0.0 force field by calculating liquid properties (density and heat of vaporization) of 123 organic liquids with experimental data available.[37] To construct the topologies, the DASH plug-in was used in the OpenFF workflow. All force-field parameters were taken from OpenFF 2.0.0 (Sage),[17] and only the partial charges were calculated differently. The comparison between the calculated values (using either DASH or AM1-BCC charges) and the experimental properties is shown in Figure 14. Given that the Lennard-Jones parameters were not intended for DASH partial charges (i.e., no refitting was performed), the performance is similar with only a small increase in the RMSE values. For the heat of vaporization, there seems to be a slight shift toward larger values (overestimation). Nevertheless, the overall good agreement indicates that the DASH partial charges can be

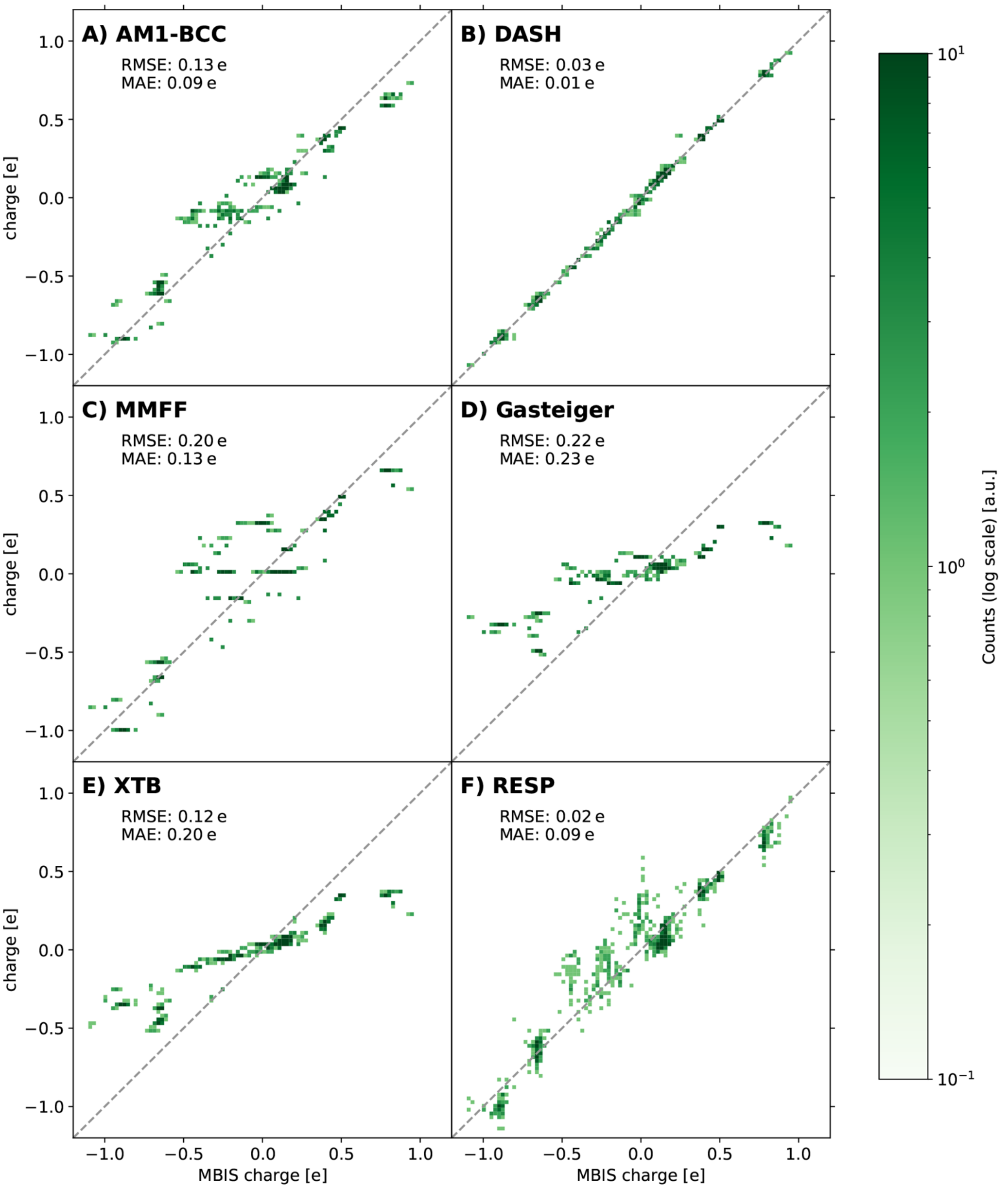


Figure 12. Comparison between the estimated partial charges with AM1-BCC (**A**), DASH (**B**), MMFF94 (**C**), Gasteiger (**D**), Mulliken-type from XTB-GNF2 (**E**), and RESP (**F**) and the MBIS reference charges on the external test set with the 20 canonical amino acids. Note that a different functional is used for the standard RESP partial charges than for the MBIS charges.

combined with the OpenFF-2.0.0 force field for condensed-phase simulations, with a substantially reduced computing time for the charge assignment (see below).

Timings. In addition to a high accuracy with respect to the MBIS reference, the DASH partial charges can be assigned much faster (Table 1) than with commonly used methods, such as AM1-BCC, and even 4 orders of magnitude faster than MBIS. While the assignment with Gasteiger and MMFF94 is even faster than DASH, the resulting partial charges are not well-suited for MD simulations (as discussed above). Note that the assignment with DASH was carried out as a sequential single-thread program. The matching of different atoms in a molecule could, in principle, be done in parallel, potentially decreasing the computation time.

The required storage space to save all required data is slightly larger for DASH (about ≈ 500MB if saved as CSV files per branch and zipped, or ≈ 150MB stored as compressed pickle files for each branch) compared to the PyTorch state-dict of the GNN (7MB). However, on most modern machines with TBs of storage, these differences should not affect the performance. Note that the state-dict is a compressed machine format, while the CSV is a human readable file and contains extra information that is redundant to machines. Furthermore, GNN requires PyTorch and PyTorch-Geometric libraries, while DASH relies only on RDKit to interpret the molecules and assign the simple feature vectors required by the tree. This basic RDKit functionality is reasonably stable between versions of the toolkit; therefore, we anticipate that it should not be necessary to regenerate the DASH model for each new RDKit release.

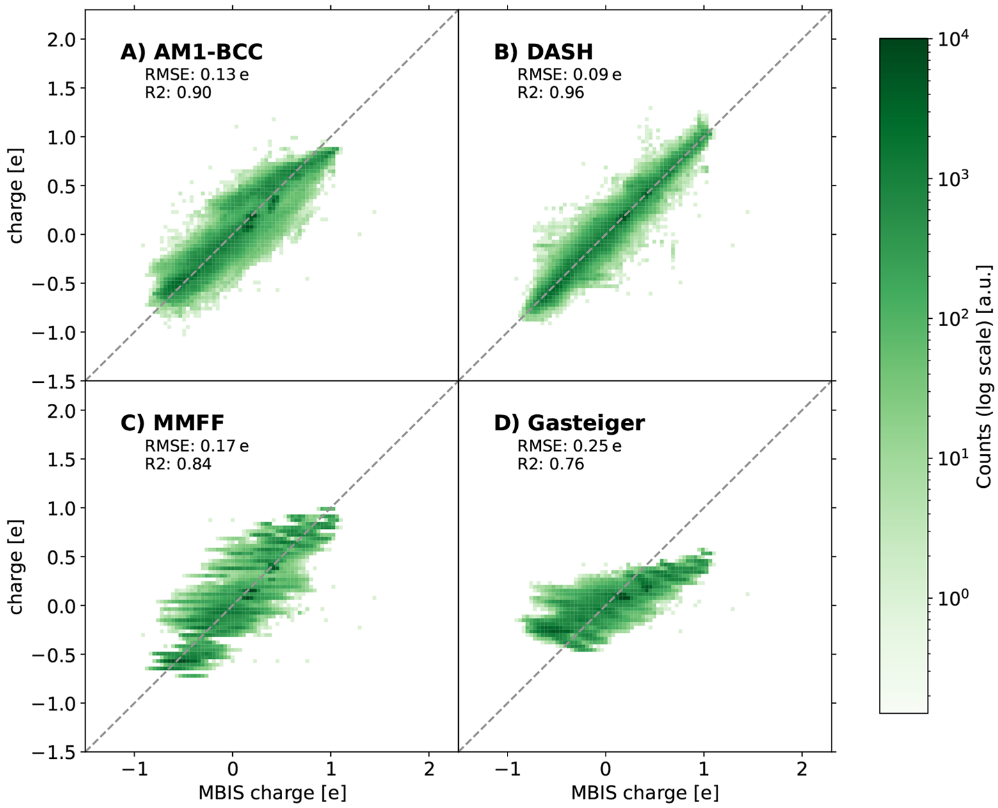


Figure 13. Comparison between the estimated partial charges with AM1-BCC (**A**), DASH (**B**), MMFF94 (**C**), Gasteiger (**D**), and the MBIS reference charges on the external VEHICLe[56] test set with 24,657 small heterocyclic molecules.

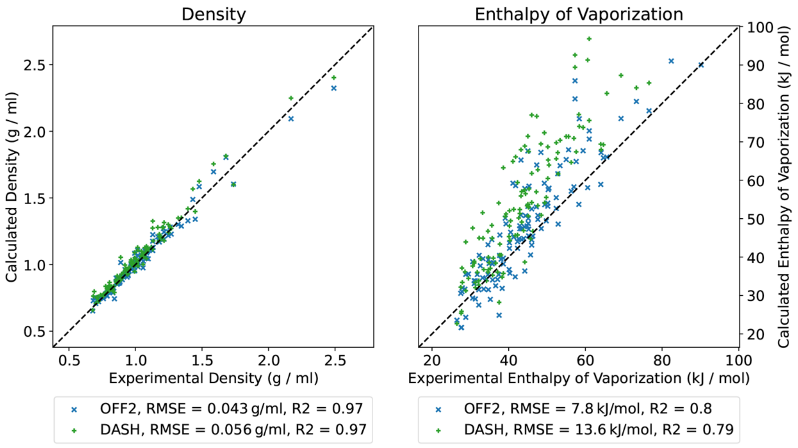


Figure 14. Comparison of the experimental density (left) and heat of vaporization (right) values of 123 organic liquids[37] with the calculated values using OpenFF-2.0.0 with the default AM1-BCC charges (blue) and the DASH partial charges (green).

■ CONCLUSIONS

In this work, a new approach to assign atomic partial charges in molecules was developed by using a dynamic attention-based substructure hierarchy (DASH) in a tree structure, where the attention values are extracted from a GNN trained on high-quality QM reference charges. DASH was found to provide a prediction accuracy that is comparable to the GNN but is independent of fast-changing and quickly deprecated ML libraries (the only requirement is basic functionality in the RDKit), directly human interpretable, and allows the retrieval of meaningful error bars on the predicted partial charges. Furthermore, assignments can be changed by the user if needed for a specific application.

To train the model, a data set was built from four different sources with a total of 393,692 unique molecules and up to three conformers per molecule. The molecules were selected to represent the substructures (as defined by MFP2) of the lead-like molecules in ChEMBL. The QM reference partial charges were calculated with TPSSh/def2-TZVP in an implicit solvent (dielectric permittivity ϵ of 4.9) and extracted with the MBIS method. The attention values from the GNN that was trained on this data set were used to order the atom types and construct the DASH tree structure, where the maximal depth

Table 1. Computation Time Required for Different Charge Assignment Methods Measured on a 16 Core Intel Xeon(R) W-1270P CPU for the 60 Molecules in the Amino Acid Test Set[a]

Method	Time [s]
MMFF	1.0×10^{-02}
Gasteiger	2.5×10^{-02}
GNN	$1.0 \times 10^{+00}$
DASH	$3.9 \times 10^{+00}$
AM1-BCC	$1.9 \times 10^{+02}$
MBIS	$8.5 \times 10^{+03}$
RESP	$1.1 \times 10^{+04}$

[a] Note that MMFF, Gasteiger, and DASH do not make use of multiple cores. Models and data are preloaded into memory for all comparisons. Note that the computational cost of MBIS and RESP charges is dominated by the time needed for the DFT calculation, and the slight overhead for RESP stems from the different choice of Python package.

of the tree and the attention threshold were optimized hyperparameters. Postassignment normalization and symmetrization ensure physically reasonable partial charges. The DASH approach outperforms commonly used methods for classical force fields such as AM1-BCC or RESP in assignment speed by two or more orders of magnitude, while predicting partial charges close to the MBIS reference.

In this work, the DASH tree was built with MBIS as the reference charge extraction method due to its high accuracy and low conformational dependency. However, the same procedure could be applied with any other type of partial charge as reference or also for other atomic properties as target.

In conclusion, DASH is a robust, fast, and accurate method for partial charge assignments, where all assignments can be visualized as fragments of a molecule for full human readability of each partial charge assignment. The DASH tree structure and underlying source code as well as an OpenFF plug-in are freely available.

■ ASSOCIATED CONTENT

Data Availability Statement

The implementation of the DASH tree and all software used for this work is open source and available on GitHub (https://github.com/rinikerlab/DASH-tree). The full data set of 1,076,252 3D structures is freely available in the ETH Research Collection (https://www.research-collection.ethz.ch/handle/20.500.11850/613415).

Supporting Information

The Supporting Information is available free of charge at https://pubs.acs.org/doi/10.1021/acs.jcim.3c00800.

Pseudocode implementations, and additional figures and tables mentioned in the text (PDF)

■ AUTHOR INFORMATION

Corresponding Author

Sereina Riniker − *Department of Chemistry and Applied Biosciences, ETH Zurich, 8093 Zurich, Switzerland;* orcid.org/0000-0003-1893-4031; Email: sriniker@ethz.ch

Authors

Marc T. Lehner − *Department of Chemistry and Applied Biosciences, ETH Zurich, 8093 Zurich, Switzerland;* orcid.org/0000-0003-4963-4824

Paul Katzberger − *Department of Chemistry and Applied Biosciences, ETH Zurich, 8093 Zurich, Switzerland;* orcid.org/0000-0003-4937-4911

Niels Maeder − *Department of Chemistry and Applied Biosciences, ETH Zurich, 8093 Zurich, Switzerland*

Carl C.G. Schiebroek − *Department of Chemistry and Applied Biosciences, ETH Zurich, 8093 Zurich, Switzerland;* orcid.org/0009-0007-3516-1508

Jakob Teetz − *Department of Chemistry and Applied Biosciences, ETH Zurich, 8093 Zurich, Switzerland*

Gregory A. Landrum − *Department of Chemistry and Applied Biosciences, ETH Zurich, 8093 Zurich, Switzerland;* orcid.org/0000-0001-6279-4481

Complete contact information is available at: https://pubs.acs.org/10.1021/acs.jcim.3c00800

Author Contributions

‡These authors contributed equally.

Notes

The authors declare no competing financial interest.

■ ACKNOWLEDGMENTS

The numerical simulations were performed on the Euler cluster operated by the High Performance Computing group at ETH Zurich. The authors gratefully acknowledge financial support by the Swiss National Science Foundation (Grant No. 200021-212732) and ETH Zurich (Grant No. ETH-50 21-1).

■ REFERENCES

(1) van Gunsteren, W. F.; Weiner, P. K.; Wilkinson, A. J. *Computer Simulation of Biomolecular Systems: Theoretical and Experimental Applications;* Springer, 2013; Vol. *3*.

(2) Ponder, J. W.; Case, D. A. Force Fields for Protein Simulations. *Adv. Protein. Chem.* **2003**, *66*, 27−85.

(3) MacKerell, A. D., Jr Empirical Force Fields for Biological Macromolecules: Overview and Issues. *J. Comput. Chem.* **2004**, *25*, 1584−1604.

(4) Jorgensen, W. L.; Tirado-Rives, J. Potential Energy Functions for Atomic-Level Simulations of Water and Organic and Biomolecular Systems. *Proc. National. Acad. Sci.* **2005**, *102*, 6665−6670.

(5) van Gunsteren, W. F.; Dolenc, J. Thirty-Five Years of Biomolecular Simulation: Development of Methodology, Force Fields and Software. *Mol. Simul.* **2012**, *38*, 1271−1281.

(6) Riniker, S. Fixed-Charge Atomistic Force Fields for Molecular Dynamics Simulations in the Condensed Phase: An Overview. *J. Chem. Inf. Model.* **2018**, *58*, 565−578.

(7) Ewald, P. P. Die Berechnung optischer und elektrostatischer Gitterpotentiale. *Ann. Phys.* **1921**, *369*, 253−287.

(8) Essmann, U.; Perera, L.; Berkowitz, M. L.; Darden, T.; Lee, H.; Pedersen, L. G. A Smooth Particle Mesh Ewald Method. *J. Chem. Phys.* **1995**, *103*, 8577−8593.

(9) Barker, J. A.; Watts, R. O. Monte Carlo Studies of the Dielectric Properties of Water-Like Models. *Mol. Phys.* **1973**, *26*, 789−792.

(10) Gasteiger, J.; Marsili, M. A New Model for Calculating Atomic Charges in Molecules. *Tetrahedron. Lett.* **1978**, *19*, 3181−3184.

(11) Hirshfeld, F. L. Bonded-Atom Fragments for Describing Molecular Charge Densities. *Theor. Chim. Acta.* **1977**, *44*, 129−138.

(12) Halgren, T. A. Merck Molecular Force Field. I. Basis, Form, Scope, Parameterization, and Performance of MMFF94. *J. Comput. Chem.* **1996**, *17*, 490−519.

(13) Jakalian, A.; Jack, D. B.; Bayly, C. I. Fast, Efficient Generation of High-Quality Atomic Charges. AM1-BCC Model: II. Parameterization and Validation. *J. Comput. Chem.* **2002**, *23*, 1623−1641.

(14) Dewar, M. J. S.; Zoebisch, E. G.; Healy, E. F.; Stewart, J. J. P. AM1: ANew General Purpose Quantum Mechanical Molecular Model. *J. Am. Chem. Soc.* **1985**, *107*, 3902−3909.

(15) Wang, J.; Wolf, R. M.; Caldwell, J. W.; Kollman, P. A.; Case, D. A. Development and Testing of a General Amber Force Field. *J. Comput. Chem.* **2004**, *25*, 1157−1174.

(16) Qiu, Y.; Smith, D. G. A.; Boothroyd, S.; Jang, H.; Hahn, D. F.; Wagner, J.; Bannan, C. C.; Gokey, T.; Lim, V. T.; Stern, C. D.; Rizzi, A.; Tjanaka, B.; Tresadern, G.; Lucas, X.; Shirts, M. R.; Gilson, M. K.; Chodera, J. D.; Bayly, C. I.; Mobley, D. L.; Wang, L.-P. Development and Benchmarking of Open Force Field V1.0.0 - The Parsley Small-Molecule Force Field. *J. Chem. Theory Comput.* **2021**, *17*, 6262−6280.

(17) Boothroyd, S.; Behara, P. K.; Madin, O. C.; Hahn, D. F.; Jang, H.; Gapsys, V.; Wagner, J. R.; Horton, J. T.; Dotson, D. L.; Thompson, M. W.; Maat, J.; Gokey, T.; Wang, L.-P.; Cole, D. J.; Gilson, M. K.; Chodera, J. D.; Bayly, C. I.; Shirts, M. R.; Mobley, D. L. Development and Benchmarking of Open Force Field 2.0.0: The Sage Small Molecule Force Field. *J. Chem. Theory Comput.* **2023**, *19*, 3251−3275.

(18) Manz, T. A.; Limas, N. G. Introducing DDEC6 Atomic Population Analysis: Part 1. Charge Partitioning Theory and Methodology. *RSC Adv.* **2016**, *6*, 47771−47801.

(19) Verstraelen, T.; Vandenbrande, S.; Heidar-Zadeh, F.; Vanduyfhuys, L.; Van Speybroeck, V.; Waroquier, M.; Ayers, P. W. Minimal Basis Iterative Stockholder: Atoms in Molecules for Force-Field Development. *J. Chem. Theory Comput.* **2016**, *12*, 3894−3912.

(20) Mou, Z.; Eakes, J.; Cooper, C. J.; Foster, C. M.; Standaert, R. F.; Podar, M.; Doktycz, M. J.; Parks, J. M. Machine Learning-Based Prediction of Enzyme Substrate Scope: Application to Bacterial Nitrilases. *Proteins Struct. Funct. Bioinform.* **2021**, *89*, 336−347.

(21) Wan, J.; Zhang, L.; Yang, G.; Zhan, C. G. Quantitative Structure-Activity Relationship for Cyclic Imide Derivatives of Protoporphyrinogen Oxidase Inhibitors: A Study of Quantum Chemical Descriptors From Density Functional Theory. *J. Chem. Inf. Comput. Sci.* **2004**, *44*, 2099−2105.

(22) Rai, B. K.; Bakken, G. A. Fast and Accurate Generation of Ab Initio Quality Atomic Charges Using Nonparametric Statistical Regression. *J. Comput. Chem.* **2013**, *34*, 1661−1671.

(23) Bleiziffer, P.; Schaller, K.; Riniker, S. Machine Learning of Partial Charges Derived From High-Quality Quantum-Mechanical Calculations. *J. Chem. Inf. Model.* **2018**, *58*, 579−590.

(24) Martin, R.; Heider, D. ContraDRG: Automatic Partial Charge Prediction by Machine Learning. *Frontier Genet.* **2019**, *10*, 990.

(25) Kato, K.; Masuda, T.; Watanabe, C.; Miyagawa, N.; Mizouchi, H.; Nagase, S.; Kamisaka, K.; Oshima, K.; Ono, S.; Ueda, H.; Tokuhisa, A.; Kanada, R.; Ohta, M.; Ikeguchi, M.; Okuno, Y.; Fukuzawa, K.; Honma, T. High-Precision Atomic Charge Prediction for Protein Systems Using Fragment Molecular Orbital Calculation and Machine Learning. *J. Chem. Inf. Model.* **2020**, *60*, 3361−3368.

(26) Behler, J. Four Generations of High-Dimensional Neural Network Potentials. *Chem. Rev.* **2021**, *121*, 10037−10072.

(27) Jiang, D.; Sun, H.; Wang, J.; Hsieh, C.-Y.; Li, Y.; Wu, Z.; Cao, D.; Wu, J.; Hou, T. Out-of-the-Box Deep Learning Prediction of Quantum-Mechanical Partial Charges by Graph Representation and Transfer Learning. *Brief. Bioinform.* **2022**, *23*, 1.

(28) Gallegos, M.; Guevara-Vela, J. M.; Pendás, A. M. NNAIMQ: A Neural Network Model for Predicting QTAIM Charges. *J. Chem. Phys.* **2022**, *156*, 014112.

(29) Wang, Y.; Fass, J.; Kaminow, B.; Herr, J. E.; Rufa, D.; Zhang, I.; Pulido, I.; Henry, M.; Bruce Macdonald, H. E.; Takaba, K.; Chodera, J. D. End-to-end differentiable construction of molecular mechanics force fields. *Chem. Sci.* **2022**, *13*, 12016−12033.

(30) Wang, Y.; Pulido, I.; Takaba, K.; Kaminow, B.; Scheen, J.; Wang, L.; Chodera, J. D. EspalomaCharge: Machine Learning-Enabled Ultra-Fast Partial Charge Assignment. *arXiv*, 2023. DOI: 10.48550/arXiv.2302.06758

(31) Ribeiro, M. T.; Singh, S.; Guestrin, C. Why Should I Trust You?": Explaining the Predictions of Any Classifier. *Conf. North. Am. Chapter. Assoc. Comput. Linguistics. Hum. Lang. Technol. (NAACL-HLT 2016)* **2016**, 97−101.

(32) Lundberg, S. M.; Lee, S. I. A Unified Approach to Interpreting Model Predictions. *Adv. Neural. Inf. Process. Syst.* **2017**, 4766−4775.

(33) Ying, R.; Bourgeois, D.; You, J.; Zitnik, M.; Leskovec, J. GNNExplainer: Generating Explanations for Graph Neural Networks. *arXiv* 2017. DOI: 10.48550/arXiv.1903.03894

(34) Bento, A. P.; Gaulton, A.; Hersey, A.; Bellis, L. J.; Chambers, J.; Davies, M.; Krüger, F. A.; Light, Y.; Mak, L.; McGlinchey, S.; Nowotka, M.; Papadatos, G.; Santos, R.; Overington, J. P. The ChEMBL Bioactivity Database: An Update. *Nucleic Acid. Res.* **2014**, *42*, D1083−D1090.

(35) Sterling, T.; Irwin, J. J. ZINC 15 − Ligand Discovery for Everyone. *J. Chem. Inf. Model.* **2015**, *55*, 2324−2337.

(36) Isert, C.; Atz, K.; Jiménez-Luna, J.; Schneider, G. QMugs, Quantum Mechanical Properties of Drug-Like Molecules. *Sci. Data.* **2022**, *9*, 1−11.

(37) Caleman, C.; van Maaren, P. J.; Hong, M.; Hub, J. S.; Costa, L. T.; van der Spoel, D. Force Field Benchmark of Organic Liquids: Density, Enthalpy of Vaporization, Heat Capacities, Surface Tension, Isothermal Compressibility, Volumetric Expansion Coefficient, and Dielectric Constant. *J. Chem. Theory. Comput.* **2012**, *8*, 61−74.

(38) Horta, B. A.; Merz, P. T.; Fuchs, P. F.; Dolenc, J.; Riniker, S.; Hünenberger, P. H. A GROMOS-compatible Force Field for Small Organic Molecules in the Condensed Phase: The 2016H66 Parameter Set. *J. Chem. Theory Comput.* **2016**, *12*, 3825−3850.

(39) Schuler, L. D.; van Gunsteren, W. F. On the Choice of Dihedral Angle Potential Energy Functions for n-Alkanes. *Mol. Simul.* **2000**, *25*, 301−319.

(40) Moine, E.; Privat, R.; Sirjean, B.; Jaubert, J.-N. Estimation of Solvation Quantities From Experimental Thermodynamic Data: Development of the Comprehensive CompSol Databank for Pure and Mixed Solutes. *J. Phys. Chem. Ref. Data.* **2017**, *46*, 033102.

(41) Riniker, S.; Landrum, G. A. Better Informed Distance Geometry: {U}sing What We Know to Improve Conformation Generation. *J. Chem. Inf. Model.* **2015**, *55*, 2562−2574.

(42) Landrum, G.; Tosco, P.; Kelley, B.; Ric; sriniker; gedeck; Cosgrove, D.; Vianello, R.; NadineSchneider; Kawashima, E.; N, D.; Dalke, A.; Jones, G.; Cole, B.; Swain, M.; Turk, S.; AlexanderSavelyev; Vaucher, A.; Wójcikowski, M.; Take, I.; Probst, D.; Scalfani, V. F.; Ujihara, K.; guillaume godin; Pahl, A.; Berenger, F.; JLVarjo. RDKit (Q3 2022) Release. *Zenodo* **2023**, *7671152*, 1.

(43) Tosco, P.; Stiefl, N.; Landrum, G. Bringing the MMFF Force Field to the RDKit: Implementation and Validation. *J. Cheminform.* **2014**, *6*, 37.

(44) Bannwarth, C.; Ehlert, S.; Grimme, S. GFN2-xTB—An Accurate and Broadly Parametrized Self-Consistent Tight-Binding Quantum Chemical Method With Multipole Electrostatics and Density-Dependent Dispersion Contributions. *J. Chem. Theory Comput.* **2019**, *15*, 1652−1671.

(45) Turney, J. M.; Simmonett, A. C.; Parrish, R. M.; Hohenstein, E. G.; Evangelista, F. A.; Fermann, J. T.; Mintz, B. J.; Burns, L. A.; Wilke, J. J.; Abrams, M. L.; Russ, N. J.; Leininger, M. L.; Janssen, C. L.; Seidl, E. T.; Allen, W. D.; Schaefer, H. F.; King, R. A.; Valeev, E. F.; Sherrill, C. D.; Crawford, T. D. Psi4: An Open-Source Ab Initio Electronic Structure Program. *WIREs Comput. Mol. Sci.* **2012**, *2*, 556−565.

(46) Staroverov, V. N.; Scuseria, G. E.; Tao, J.; Perdew, J. P. Comparative Assessment of a New Nonempirical Density Functional: Molecules and Hydrogen-Bonded Complexes. *J. Chem. Phys.* **2003**, *119*, 12129−12137.

(47) Weigend, F.; Ahlrichs, R. Balanced Basis Sets of Split Valence, Triple Zeta Valence and Quadruple Zeta Valence Quality for H to Rn: Design and Assessment of Accuracy. *Phys. Chem. Chem. Phys.* **2005**, *7*, 3297−3305.

(48) Weigend, F. Accurate Coulomb-Fitting Basis Sets for H to Rn. *Phys. Chem. Chem. Phys.* **2006**, *8*, 1057−1065.

(49) Xiong, Z.; Wang, D.; Liu, X.; Zhong, F.; Wan, X.; Li, X.; Li, Z.; Luo, X.; Chen, K.; Jiang, H.; Zheng, M. Pushing the Boundaries of Molecular Representation for Drug Discovery With the Graph Attention Mechanism. *J. Med. Chem.* **2020**, *63*, 8749−8760.

(50) White, B. W.; Rosenblatt, F. Principles of Neurodynamics. Perceptrons and the Theory of Brain Mechanisms. *Am. J. Psychol.* **1963**, *76*, 705.

(51) Fukushima, K. Visual Feature Extraction by a Multilayered Network of Analog Threshold Elements. *IEEE. Trans. Syst. Sci. Cybern.* **1969**, *5*, 322−333.

(52) Kearnes, S.; McCloskey, K.; Berndl, M.; Pande, V.; Riley, P. Molecular Graph Convolutions: Moving Beyond Fingerprints. *J. Comput. Mol.* **2016**, *30*, 595−608.

(53) Kingma, D. P.; Ba, J. L. Adam: A Method for Stochastic Optimization. Poster at 3rd International Conference for Learning Representations (ICLR), San Diego, CA, 2015; *arXiv*; 2015. DOI: 10.48550/arXiv.1412.6980

(54) Halgren, T. A. Merck Molecular Force Field. II. MMFF94 Van Der Waals and Electrostatic Parameters for Intermolecular Interactions. *J. Comput. Chem.* **1996**, *17*, 520−552.

(55) Mobley, D. L.; Bannan, C. C.; Rizzi, A.; Bayly, C. I.; Chodera, J. D.; Lim, V. T.; Lim, N. M.; Beauchamp, K. A.; Slochower, D. R.; Shirts, M. R.; Gilson, M. K.; Eastman, P. K. Escaping Atom Types in Force Fields Using Direct Chemical Perception. *J. Chem. Theory. Comput.* **2018**, *14*, 6076−6092.

(56) Pitt, W. R.; Parry, D. M.; Perry, B. G.; Groom, C. R. Heteroaromatic Rings of the Future. *J. Med. Chem.* **2009**, *52*, 2952−2963.

(57) Mulliken, R. S. Electronic Population Analysis on LCAO−MO Molecular Wave Functions. I. *J. Chem. Phys.* **1955**, *23*, 1833−1840.

(58) Jakalian, A.; Jack, D. B.; Bayly, C. I. Fast, Efficient Generation of High-Quality Atomic Charges. AM1-BCC Model: II. Parameterization and Validation. *J. Comput. Chem.* **2002**, *23*, 1623−1641.

(59) Case, D. A.; Aktulga, H. M.; Belfon, K.; Ben-Shalom, I. Y.; Berryman, J. T.; Brozell, S. R.; Cerutti, D. S.; Cheatham, T. E.; Cisneros, G. A.; Cruzeiro, V. W. D.; Darden, T. A.; Duke, R. E.; Giambasu, G.; Gilson, M. K.; Gohlke, H.; Goetz, A. W.; Harris, R.; Izadi, S.; Izmailov, S. A.; Kasavajhala, K.; Kaymak, M. C.; King, E.; Kovalenko, A.; Kurtzman, T.; Lee, T. S.; LeGrand, S.; Li, P.; Lin, C.; Liu, J.; Luchko, T.; Luo, R.; Machado, M.; Man, V.; Manathunga, M.; Merz, K. M.; Miao, Y.; Mikhailovskii, O.; Monard, G.; Nguyen, H.; O'Hearn, K. A.; Onufriev, A.; Pan, F.; Pantano, S.; Qi, R.; Rahnamoun, A.; Roe, D. R.; Roitberg, A.; Sagui, C.; Schott-Verdugo, S.; Shajan, A.; Shen, J.; Simmerling, C. L.; Skrynnikov, N. R.; Smith, J.; Swails, J.; Walker, R. C.; Wang, J.; Wang, J.; Wei, H.; Wolf, R. M.; Wu, X.; Xiong, Y.; Xue, Y.; York, D. M.; Zhao, S.; Kollman, P. A. *AMBER 2022;* University of California: San Francisco, CA, 2022.

(60) Bayly, C. I.; Cieplak, P.; Cornell, W.; Kollman, P. A. A Well-Behaved Electrostatic Potential Based Method Using Charge Restraints for Deriving Atomic Charges: The RESP Model. *J. Phys. Chem.* **1993**, *97*, 10269−10280.

(61) Boothroyd, S.; Wang, L.-P.; Mobley, D. L.; Chodera, J. D.; Shirts, M. R. Open Force Field Evaluator: An Automated, Efficient, and Scalable Framework for the Estimation of Physical Properties From Molecular Simulation. *J. Chem. Theory Comput.* **2022**, *18*, 3566−3576.

(62) Eastman, P.; Swails, J.; Chodera, J. D.; McGibbon, R. T.; Zhao, Y.; Beauchamp, K. A.; Wang, L. P.; Simmonett, A. C.; Harrigan, M. P.; Stern, C. D.; et al. OpenMM 7: Rapid Development of High Performance Algorithms for Molecular Dynamics. *PLoS Comput. Biology.* **2017**, *13*, e1005659.