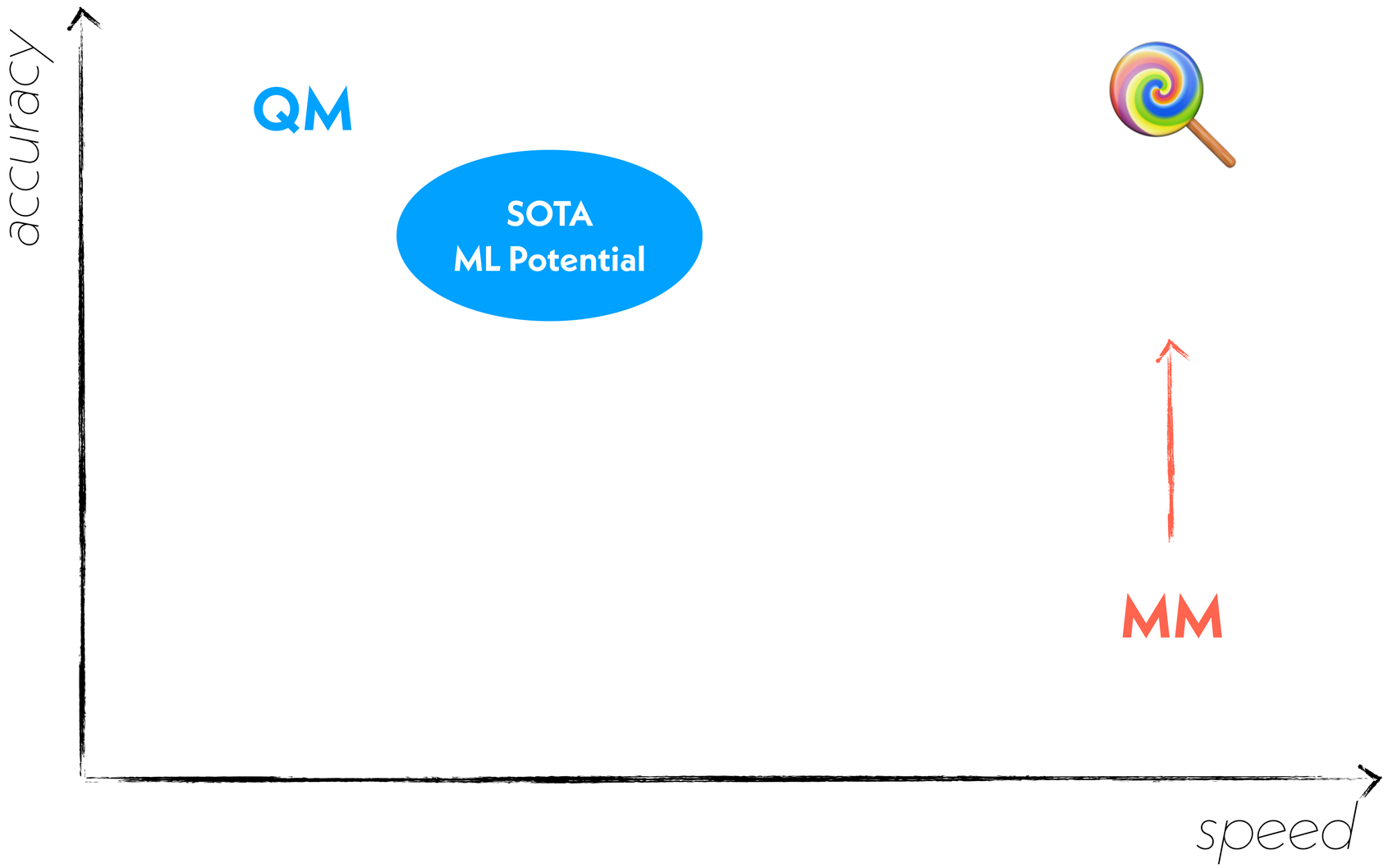
from Espaloma 🍹 to SAKE 🍶: to brew, distill, and mix force fields with balanced briskness, smoothness, and intricacy

yuanqing wang
new york university

16 mar 2023 santa fe, n. méx.

@YuanqingWang wangyq.net wangyq@wangyq.net


accuracy
QM
SOTA
ML Potential
MM
speed

$$U_{\texttt{bond}}(r; K_r, r_0) = \frac{1}{2}(K_r r - r_0)^2$$

$$U_{\texttt{angle}}(\theta; K_\theta, \theta_0) = \frac{1}{2}(K_\theta \theta - \theta_0)^2$$

$$U_{\texttt{torsion}}(\phi; \{K_{\phi,n}\}, \phi_0) = \sum_{0}^{n_{\texttt{max}}} K_{\phi,n}[1 + \cos(n\phi - \phi_{0,n})]$$

x **Φ_{FF}**

$$\begin{aligned}
U_{\text{MM}}(\mathbf{x}; \mathcal{G}, \Phi_{\text{FF}}) \quad = & \sum_{(v_i,v_j)\in\mathcal{G}_{\text{bond}}} U_{\text{bond}}\left(r(\mathbf{x}; v_i, v_j); K_r(\Phi_{\text{FF}}; v_i, v_j), r_0(\Phi_{\text{FF}}; v_i, v_j)\right) \\
& + \sum_{(v_i,v_j,v_k)\in\mathcal{G}_{\text{angle}}} U_{\text{angle}}\left(\theta(\mathbf{x}; v_i, v_j, v_k); K_\theta(\Phi_{\text{FF}}; v_i, v_j, v_k), \theta_0(\Phi_{\text{FF}}; v_i, v_j, v_k)\right) \\
& + \sum_{(v_i,v_j,v_k,v_l)\in\mathcal{G}_{\text{torsion}}} U_{\text{torsion}}\left(\phi(\mathbf{x}; v_i, v_j, v_k, v_l); \{K_{\phi,n}(\Phi_{\text{FF}}; v_i, v_j, v_k, v_l)\}_{n=1}^{n_{\text{max}}}, \phi_0(\Phi_{\text{FF}}; v_i, v_j, v_k, v_l)\right) \\
& + \sum_{(v_i,v_j)\in\mathcal{G}_{\text{Coulomb}}} U_{\text{Coulomb}}\left(r(\mathbf{x}; v_i, v_j); q(\Phi_{\text{FF}}; v_i), q(\Phi_{\text{FF}}; v_j)\right) \\
& + \sum_{(v_i,v_j)\in\mathcal{G}_{\text{van der Waals}}} U_{\text{van der Waals}}\left(r(\mathbf{x}; v_i, v_j); \sigma(\Phi_{\text{FF}}; v_i, v_j), \epsilon(\Phi_{\text{FF}}; v_i, v_j)\right) \qquad (1)
\end{aligned}$$

H-C≡C-H

$$U_{\texttt{bond}}(r; K_r, r_0) = \frac{1}{2} K_r (r - r_0)^2$$

H-C≡C-H

$$U_{\texttt{angle}}(\theta; K_\theta, \theta_0) = \frac{1}{2} K_\theta (\theta - \theta_0)^2$$

H-C≡C-H

$$U_{\texttt{torsion}}(\phi; \{K_{\phi,n}\}, \phi_0) = \sum_{0}^{n_{\texttt{max}}} K_{\phi,n} [1 + \cos(n\phi - \phi_{0,n})]$$

legacy atom typing schemes are labor-intensive and poorly extensible

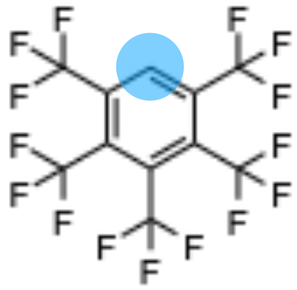
```
<HarmonicBondForce>
 <Bond type1="ow" type2="hw" length="0.09572" k="462750.4"/>
 <Bond type1="hw" type2="hw" length="0.15136" k="462750.4"/>
 <Bond type1="br" type2="br" length="0.2542" k="103093.76"/>
```

```
<HarmonicAngleForce>
 <Angle type1="hw" type2="ow" type3="hw" angle="1.82421813418" k="836.8"/>
 <Angle type1="hw" type2="hw" type3="ow" angle="2.2294835865" k="0.0"/>
 <Angle type1="br" type2="c1" type3="br" angle="3.14159265359" k="483.33568"/>
```

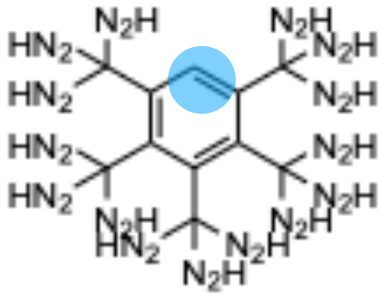
Table 1. Atom Types and Their Definitions in GAFF. Wang et al. (2014) doi:10.1002/jcc.20035

No.	Atom type	Description	No.	Atom type	Description
1	c	sp^2 carbon in C═O, C═S	2	c1	sp^1 carbon
3	c2	sp^2 carbon, aliphatic	4	c3	sp^3 carbon
5	ca	sp^2 carbon, aromatic	6	n	sp^2 nitrogen in amides
7	n1	sp^1 nitrogen	8	n2	sp^2 nitrogen with 2 subst., real double bonds
9	n3	sp^3 nitrogen with 3 subst.	10	n4	sp^3 nitrogen with 4 subst.
11	na	sp^2 nitrogen with 3 subst.	12	nh	amine nitrogen connected to aromatic rings

type: ca



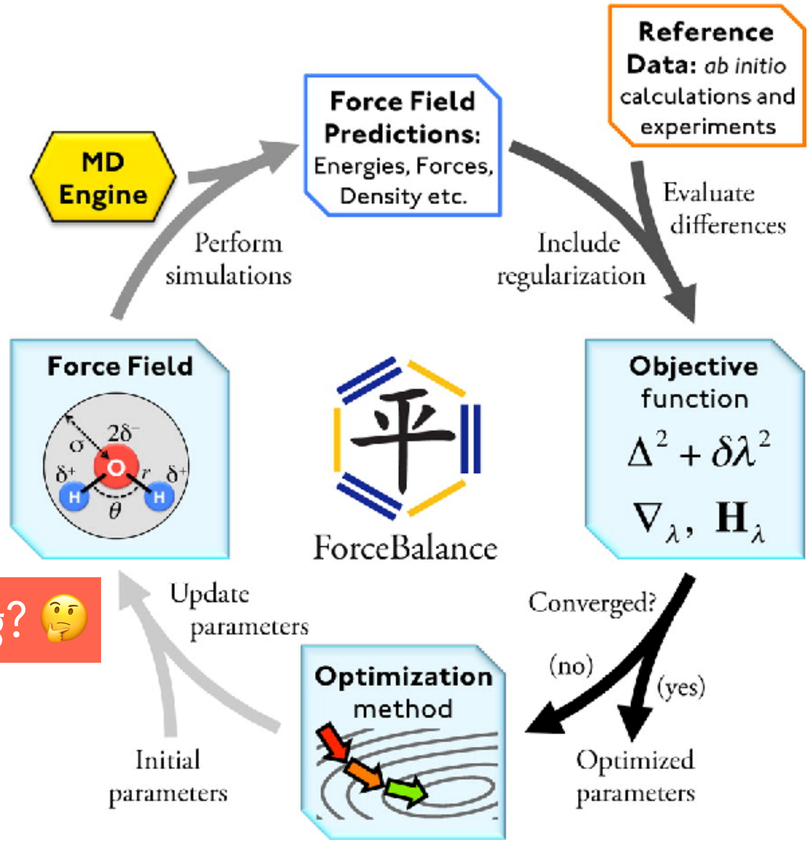
type: ca



same type! 😱😱😱

optimizing FF *parameters* alone has been feasible

$$\Phi^*_{\mathtt{MM}} = \underset{\Phi_{\mathtt{MM}}}{\operatorname{argmin}}\, \mathscr{L}(U_{\mathtt{MM}}(\mathbf{x}; \mathscr{G}, \Phi_{\mathtt{MM}}), U_{\mathtt{qm}}(\mathbf{x}))$$



but not typing? 🤔

RMSE ≈ 3 kcal/mol w.r.t. QM

Wang et al. (2014) doi://10.1021/jz500737m

legacy atom typing schemes are labor-intensive and poorly extensible

```
<HarmonicBondForce>
 <Bond type1="ow" type2="hw" length="0.09572" k="462750.4"/>
 <Bond type1="hw" type2="hw" length="0.15136" k="462750.4"/>
 <Bond type1="br" type2="br" length="0.2542" k="103093.76"/>
```

```
<HarmonicAngleForce>
 <Angle type1="hw" type2="ow" type3="hw" angle="1.82421813418" k="836.8"/>
 <Angle type1="hw" type2="hw" type3="ow" angle="2.2294835865" k="0.0"/>
 <Angle type1="br" type2="c1" type3="br" angle="3.14159265359" k="483.33568"/>
```

Table 1. Atom Types and Their Definitions in GAFF. Wang et al. (2014) doi:10.1002/jcc.20035

No.	Atom type	Description	No.	Atom type	Description
1	c	sp^2 carbon in C=O, C=S	2	c1	sp^1 carbon
3	c2	sp^2 carbon, aliphatic	4	c3	sp^3 carbon
5	ca	sp^2 carbon, aromatic	6	n	sp^2 nitrogen in amides
7	n1	sp^1 nitrogen	8	n2	sp^2 nitrogen with 2 subst., real double bonds
9	n3	sp^3 nitrogen with 3 subst.	10	n4	sp^3 nitrogen with 4 subst.
11	na	sp^2 nitrogen with 3 subst.	12	nh	amine nitrogen connected to aromatic rings

we need:
an optimizable function
from node attribute and neighborhood multiset
to node representation

$$h_v = f(h_v^{(0)}, \rho(\mathcal{N}(v)))$$

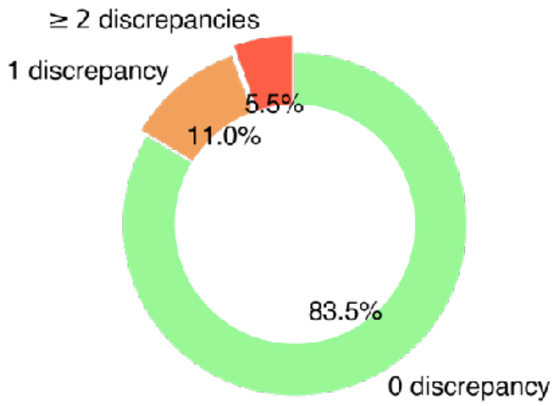
node attribute
number of neighbors
neighbor attributes
neighborhood multisite

hypothesis:
GNNs are at least as expressive as legacy atom typing schemes

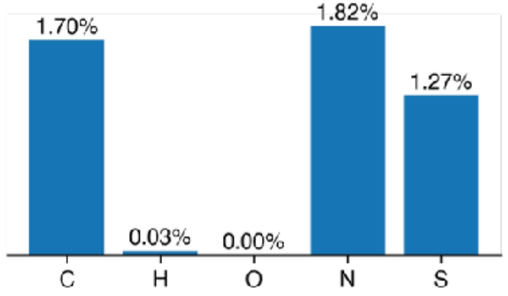
stage 1: graph net continuous atom typing replaces discrete ones

Overall agreement: 99.07%

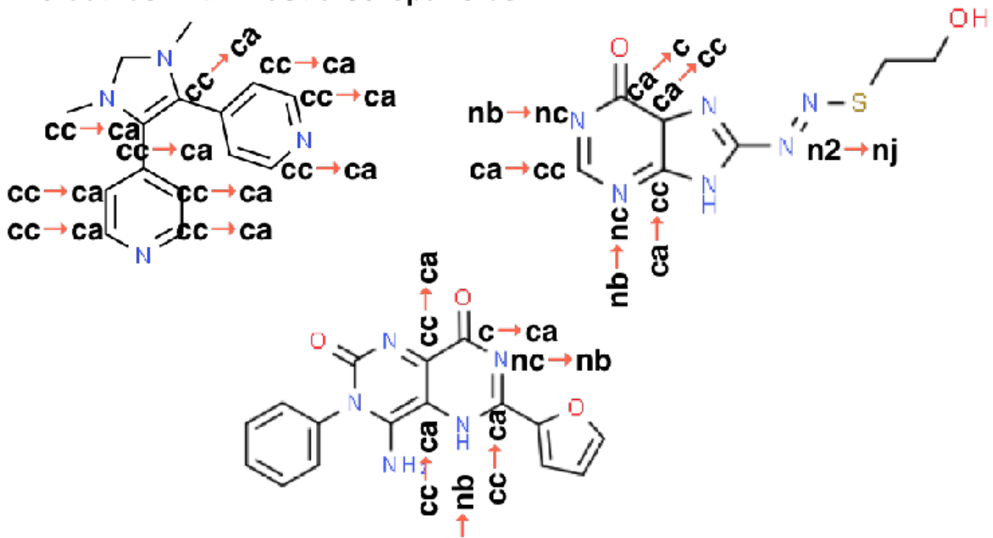
(a) # discrepancies per molecule



(b) error rate per element



(c) molecules with most discrepancies

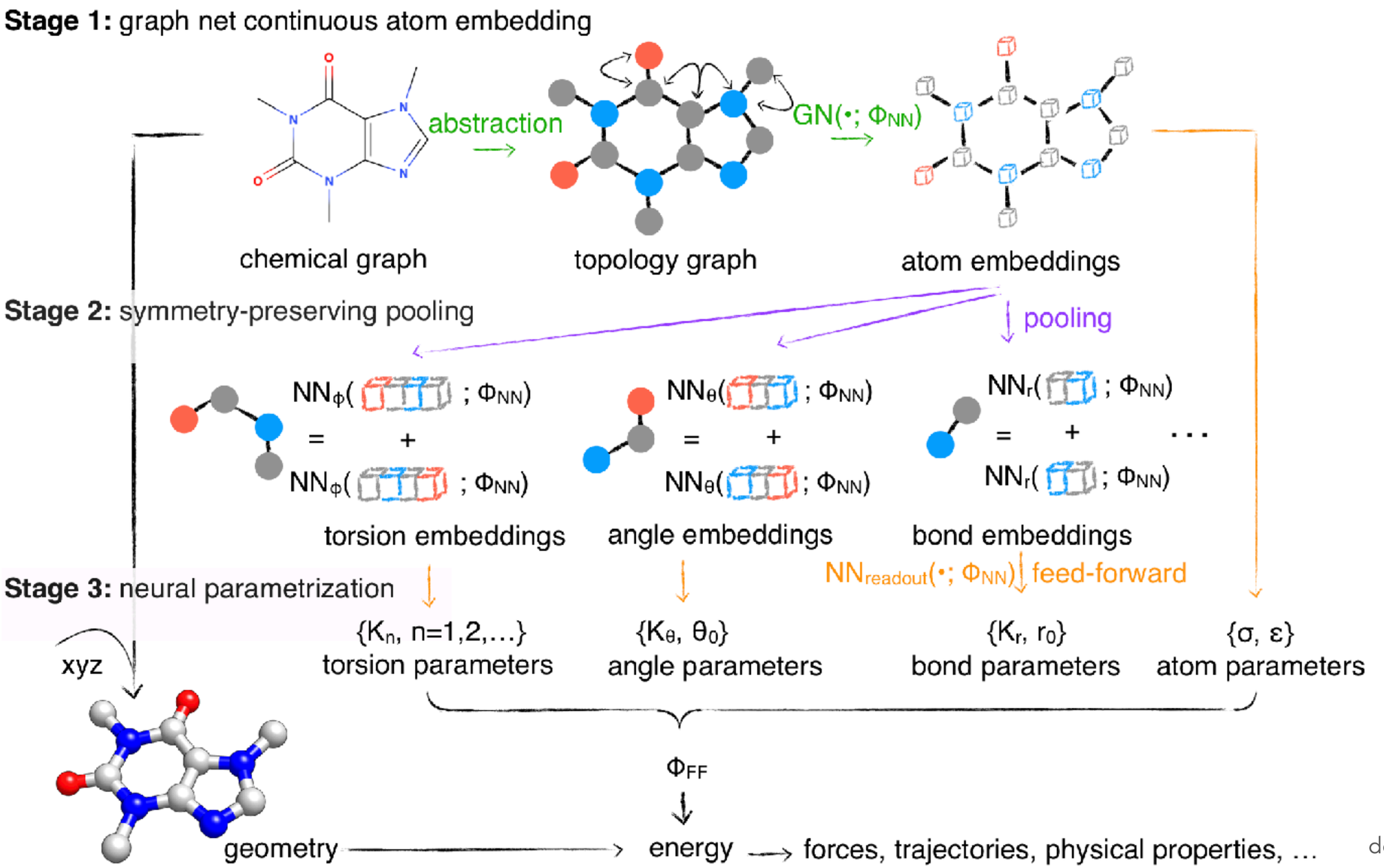


(d) confusion matrix of learned vs reference atom type

	c	c1	c2	c3	ca	cp	ce	cg	ch
c	98.1		0.4		0.0		0.5		
c1		100.0							
c2	1.1		98.2				0.5		
c3				100.0					
ca	0.2				99.1	39.4			
cp					0.8	60.6			
ce	0.6		1.3				99.0		
cg								67.7	36.8
ch								32.3	63.2
	7.5	0.2	3.2	30.9	53.9	0.7	2.9	0.4	0.3

Wang et. al. (2022)
doi: 10.1039/D2SC02739A

extensible surrogate potential optimized by message-passing

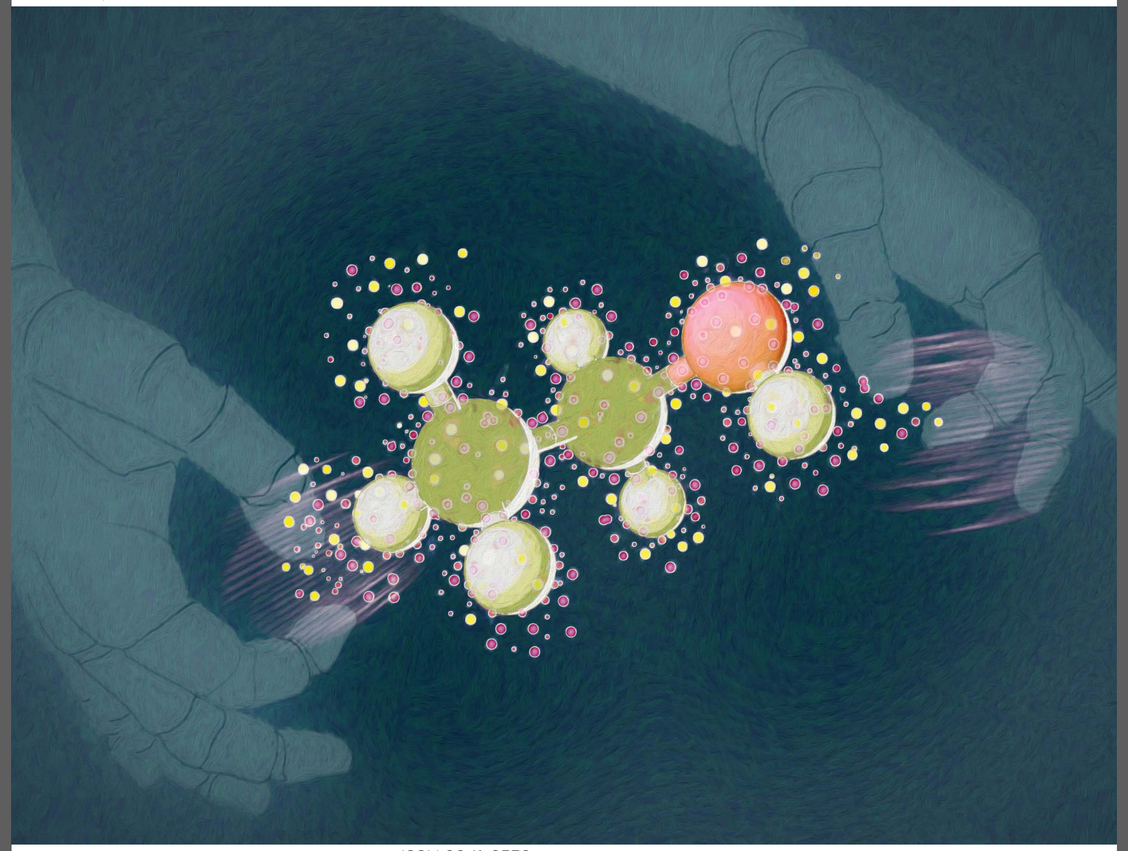


Wang et. al. (2022)
doi: 10.1039/D2SC02739A

Volume 13
Number 41
7 November 2022
Pages 11953–12246

Chemical Science

rsc.li/chemical-science

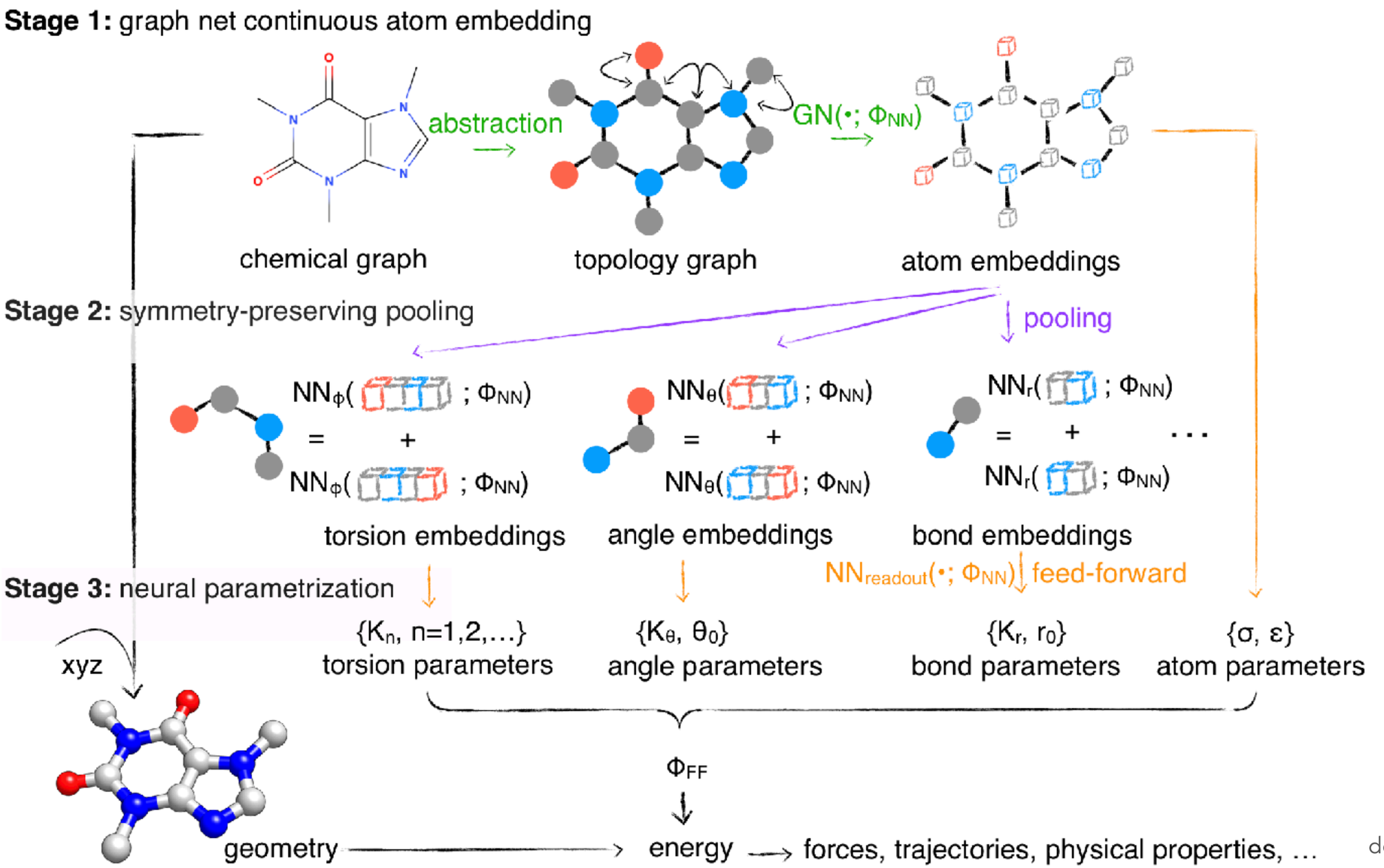


ISSN 2041-6539

ROYAL SOCIETY OF CHEMISTRY

EDGE ARTICLE
Yuanqing Wang, John D. Chodera *et al.*
End-to-end differentiable construction of molecular mechanics force fields

extensible surrogate potential optimized by message-passing



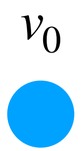
Wang et. al. (2022)
doi: 10.1039/D2SC02739A

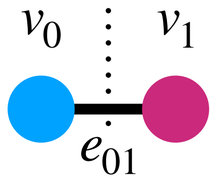
stage 2 and 3: Janossy pooling assigns embeddings and parameters in a symmetry-preserving manner

```
<HarmonicBondForce>
 <Bond type1="ow" type2="hw" length="0.09572" k="462750.4"/>
 <Bond type1="hw" type2="hw" length="0.15136" k="462750.4"/>
 <Bond type1="br" type2="br" length="0.2542" k="103093.76"/>
```

```
<HarmonicAngleForce>
 <Angle type1="hw" type2="ow" type3="hw" angle="1.82421813418" k="836.8"/>
 <Angle type1="hw" type2="hw" type3="ow" angle="2.2294835865" k="0.0"/>
 <Angle type1="br" type2="c1" type3="br" angle="3.14159265359" k="483.33568"/>
```

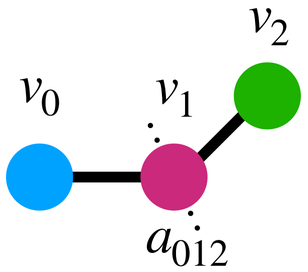
$$\overline{\overline{f}}(|\boldsymbol{h}|, \boldsymbol{h}; \boldsymbol{\theta}^{(f)}) = \frac{1}{|\boldsymbol{h}|!} \sum_{\pi \in \Pi_{|\boldsymbol{h}|}} \vec{f}(|\boldsymbol{h}|, \boldsymbol{h}_\pi; \boldsymbol{\theta}^{(f)}),$$

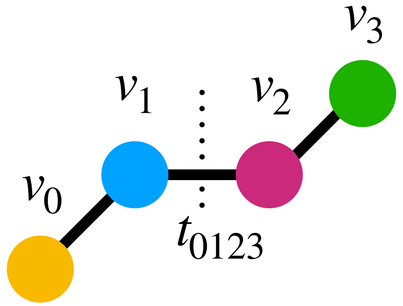




$$\{\sigma_{v_0}, \epsilon_{v_0}\} = NN_v(h_{v_0})$$

$$\{k_{e_{01}}, r_{\mathrm{eq},e_{01}}\} = NN_e(h_{v_0} : h_{v_1}) + NN_e(h_{v_1} : h_{v_0})$$



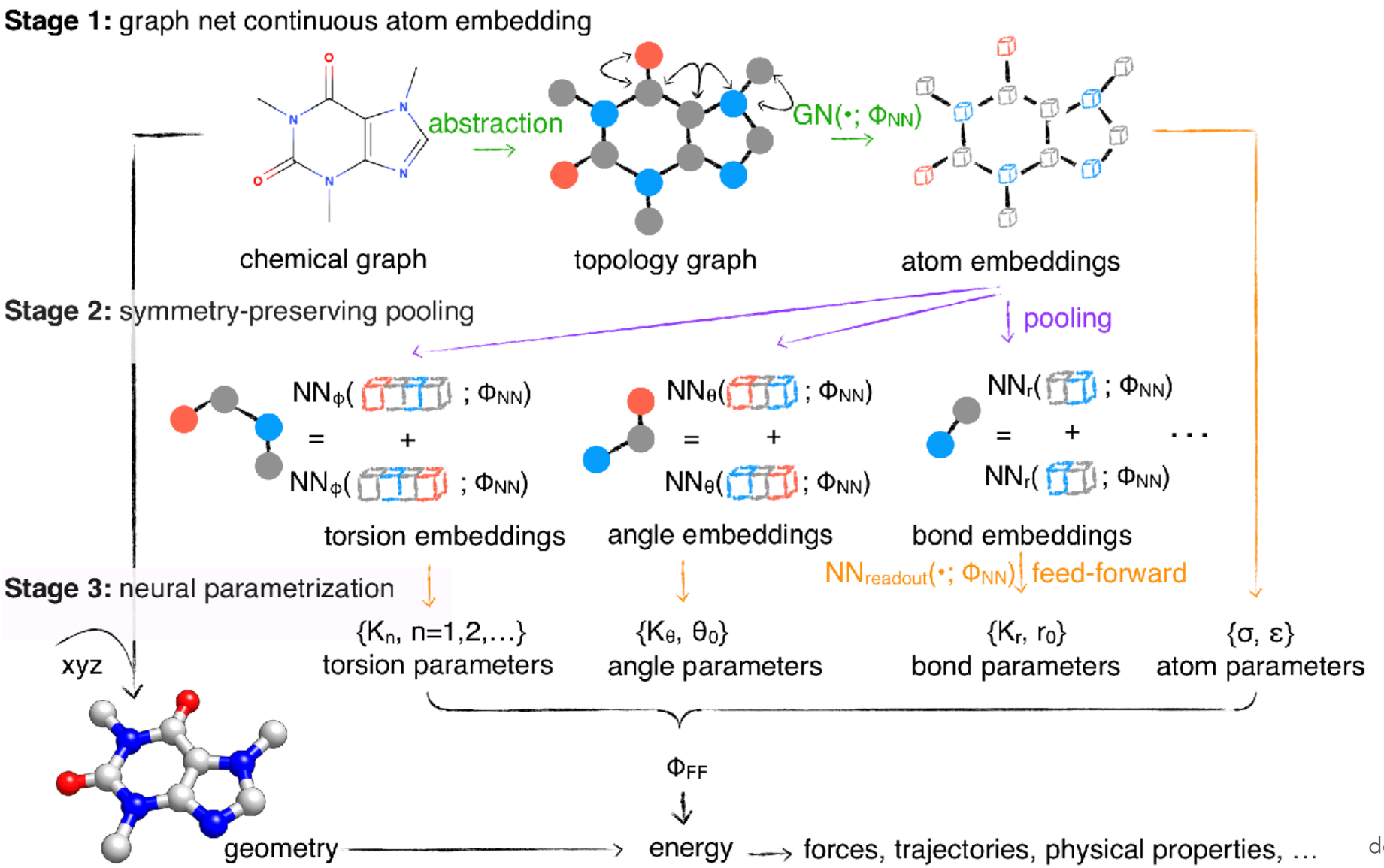


$$\{k_{a_{012}}, \phi_{\mathrm{eq},a_{012}}\} = NN_a([h_{v_0} : h_{v_1} : h_{v_2}]) + NN_a([h_{v_2} : h_{v_1} : h_{v_0}])$$

$$\{k_{i,t_{0123}}, \phi_{\mathrm{eq},i,t_{0123}}\} = NN_t(h_{v_0} : h_{v_1} : h_{v_2} : h_{v_3}) + NN_t(h_{v_3} : h_{v_2} : h_{v_1} : h_{v_0})$$

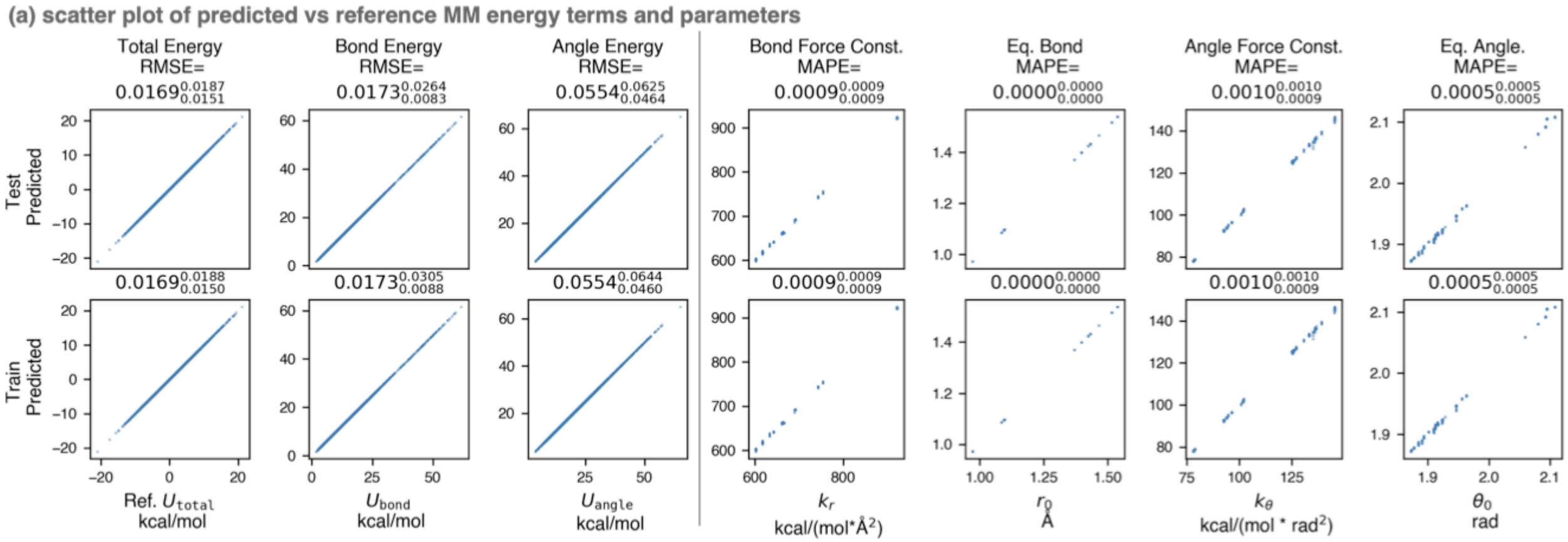
Murphy et al. (2019) arXiv: 1811.01900

extensible surrogate potential optimized by message-passing



Wang et. al. (2022)
doi: 10.1039/D2SC02739A

espaloma recovers MM energies and parameters



espaloma outperforms current force fields in QM accuracy and can be easily trained for heterogeneous systems

	(a) dataset	# mols	# trajs	# snapshots	Espaloma RMSE		Legacy FF RMSE (kcal/mol) (Test molecules)			
					Train	Test	OpenFF 1.2.0	GAFF-1.81	GAFF-2.11	Amber ff14SB
	PhAlkEthOH (simple CHO)	7408	12592	244036	$0.8656^{0.9131}_{0.8225}$	$1.1398^{1.2332}_{1.0715}$	$1.6071^{1.6915}_{1.5197}$	$1.7267^{1.7935}_{1.6543}$	$1.7406^{1.8148}_{1.6679}$	
	OpenFF Gen2 Optimization (druglike)	792	3977	23748	$0.7413^{0.7920}_{0.6914}$	$0.7600^{0.8805}_{0.6644}$	$2.1768^{2.3388}_{2.0380}$	$2.4274^{2.5207}_{2.3300}$	$2.5386^{2.6640}_{2.4370}$	
	VEHICLe (heterocyclic)	24867	24867	234326	$0.4476^{0.4690}_{0.4273}$	$0.4233^{0.4414}_{0.4053}$	$8.0247^{8.2456}_{7.8271}$	$8.0077^{8.2313}_{7.7647}$	$9.4014^{9.6434}_{9.2135}$	
	PepConf (peptides)	736	7560	22154	$1.2714^{1.3616}_{1.1899}$	$1.8727^{1.9749}_{1.7309}$	$3.6143^{3.7288}_{3.4870}$	$4.4446^{4.5738}_{4.3386}$	$4.3356^{4.4641}_{4.1965}$	$3.1502^{3.1859,*}_{3.1117}$
joint	OpenFF Gen2 Optimization	1528	11537	45902	$0.8264^{0.9007}_{0.7682}$	$1.8764^{1.9947}_{1.7827}$	$2.1768^{2.3388}_{2.0380}$	$2.4274^{2.5207}_{2.3300}$	$2.5386^{2.6640}_{2.4370}$	
	PepConf				$1.2038^{1.3056}_{1.1178}$	$1.7307^{1.8439}_{1.6053}$	$3.6143^{3.7288}_{3.4870}$	$4.4446^{4.5738}_{4.3386}$	$4.3356^{4.4641}_{4.1965}$	$3.1502^{3.1859,*}_{3.1117}$

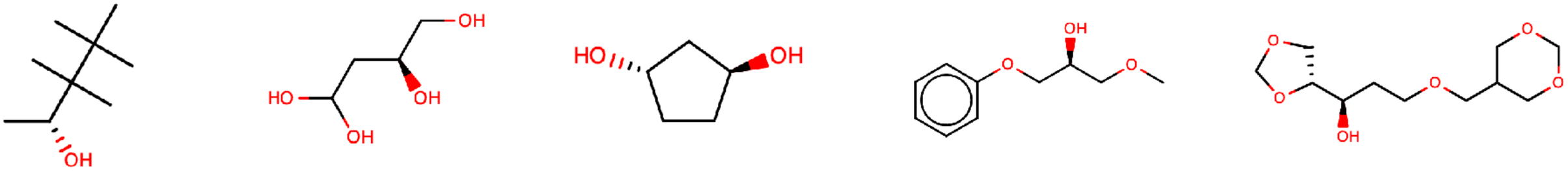
Wang et. al. (2022)
doi: 10.1039/D2SC02739A

espaloma outperforms current force fields in QM accuracy and can be easily trained for heterogeneous systems

(a) dataset	# mols	# trajs	# snapshots	Espaloma RMSE		Legacy FF RMSE (kcal/mol) (Test molecules)			
				Train	Test	OpenFF 1.2.0	GAFF-1.81	GAFF-2.11	Amber ff14SB
PhAlkEthOH (simple CHO)	7408	12592	244036	$0.8656^{0.9131}_{0.8225}$	$1.1398^{1.2332}_{1.0715}$	$1.6071^{1.6915}_{1.5197}$	$1.7267^{1.7935}_{1.6543}$	$1.7406^{1.8148}_{1.6679}$	

PhAlkEthOh: **Ph**enyls, **Alk**anes, **Eth**ers, and alcohols (**OH**)

(a low-complexity chemical space)



PhAlkEthOH doi: https://dx.doi.org/10.1021/acs.jctc.8b00640

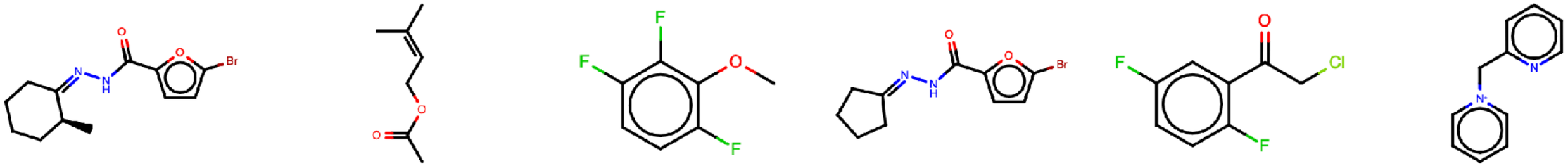
Wang et. al. (2022)
doi: 10.1039/D2SC02739A

how does espaloma compare to discrete chemical perception?

(a) dataset	# mols	# trajs	# snapshots	Espaloma RMSE		Legacy FF RMSE (kcal/mol) (Test molecules)			
				Train	Test	OpenFF 1.2.0	GAFF-1.81	GAFF-2.11	Amber ff14SB
PhAlkEthOH (simple CHO)	7408	12592	244036	$0.8656^{0.9131}_{0.8225}$	$1.1398^{1.2332}_{1.0715}$	$1.6071^{1.6915}_{1.5197}$	$1.7267^{1.7935}_{1.6543}$	$1.7406^{1.8148}_{1.6679}$	
OpenFF Gen2 Optimization (druglike)	792	3977	23748	$0.7413^{0.7920}_{0.6914}$	$0.7600^{0.8805}_{0.6644}$	$2.1768^{2.3388}_{2.0380}$	$2.4274^{2.5207}_{2.3300}$	$2.5386^{2.6640}_{2.4370}$	

OpenFF Gen2 Optimization set: Diverse druglike fragments challenging for force fields

(a moderate-complexity chemical space)



OpenFF Gen2 QM data doi: https://doi.org/10.1021/acs.jctc.1c00571

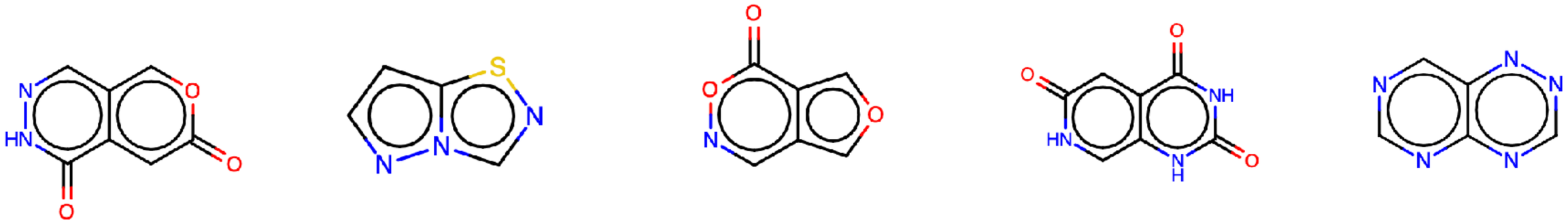
Wang et. al. (2022)
doi: 10.1039/D2SC02739A

how well does espaloma compare to legacy force fields on rare chemistries?

(a) dataset	# mols	# trajs	# snapshots	Espaloma RMSE		Legacy FF RMSE (kcal/mol) (Test molecules)			
				Train	Test	OpenFF 1.2.0	GAFF-1.81	GAFF-2.11	Amber ff14SB
PhAlkEthOH (simple CHO)	7408	12592	244036	$0.8656^{0.9131}_{0.8225}$	$1.1398^{1.2332}_{1.0715}$	$1.6071^{1.6915}_{1.5197}$	$1.7267^{1.7935}_{1.6543}$	$1.7406^{1.8148}_{1.6679}$	
OpenFF Gen2 Optimization (druglike)	792	3977	23748	$0.7413^{0.7920}_{0.6914}$	$0.7600^{0.8805}_{0.6644}$	$2.1768^{2.3388}_{2.0380}$	$2.4274^{2.5207}_{2.3300}$	$2.5386^{2.6640}_{2.4370}$	
VEHICLe (heterocyclic)	24867	24867	234326	$0.4476^{0.4690}_{0.4273}$	$0.4233^{0.4414}_{0.4053}$	$8.0247^{8.2456}_{7.8271}$	$8.0077^{8.2313}_{7.7647}$	$9.4014^{9.6434}_{9.2135}$	

VEHICLe: Virtual exploratory heterocyclic drug scaffold library

(aromatic bicyclic heterocyclic compounds containing C, N, O, S, H)

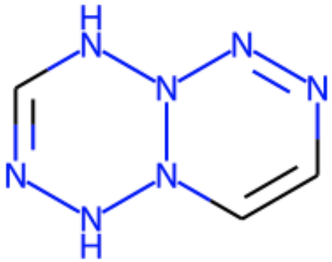


VEHICLe doi: http://doi.org/10.1021/jm801513z

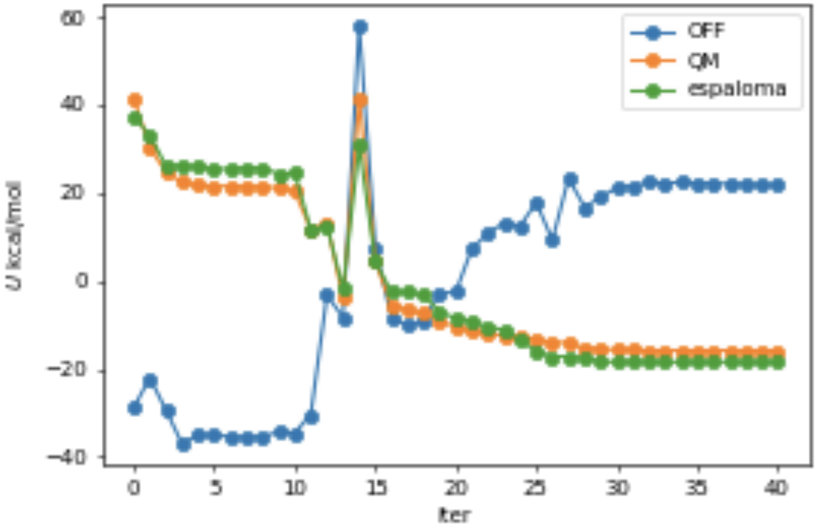
Wang et. al. (2022)
doi: 10.1039/D2SC02739A

how well does espaloma compare to legacy force fields on rare chemistries?

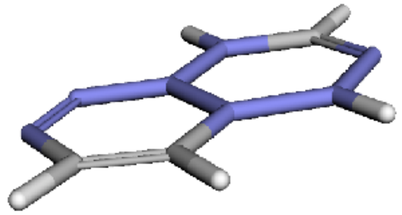
(a) dataset	# mols	# trajs	# snapshots	Espaloma RMSE		Legacy FF RMSE (kcal/mol) (Test molecules)			
				Train	Test	OpenFF 1.2.0	GAFF-1.81	GAFF-2.11	Amber ff14SB
PhAlkEthOH (simple CHO)	7408	12592	244036	$0.8656^{0.9131}_{0.8225}$	$1.1398^{1.2332}_{1.0715}$	$1.6071^{1.6915}_{1.5197}$	$1.7267^{1.7935}_{1.6543}$	$1.7406^{1.8148}_{1.6679}$	
OpenFF Gen2 Optimization (druglike)	792	3977	23748	$0.7413^{0.7920}_{0.6914}$	$0.7600^{0.8805}_{0.6644}$	$2.1768^{2.3388}_{2.0380}$	$2.4274^{2.5207}_{2.3300}$	$2.5386^{2.6640}_{2.4370}$	
VEHICLe (heterocyclic)	24867	24867	234326	$0.4476^{0.4690}_{0.4273}$	$0.4233^{0.4414}_{0.4053}$	$8.0247^{8.2456}_{7.8271}$	$8.0077^{8.2313}_{7.7647}$	$9.4014^{9.6434}_{9.2135}$	



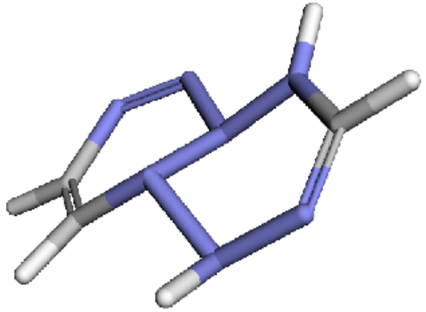
Comparison with QCArchive data



initial



QM minimized



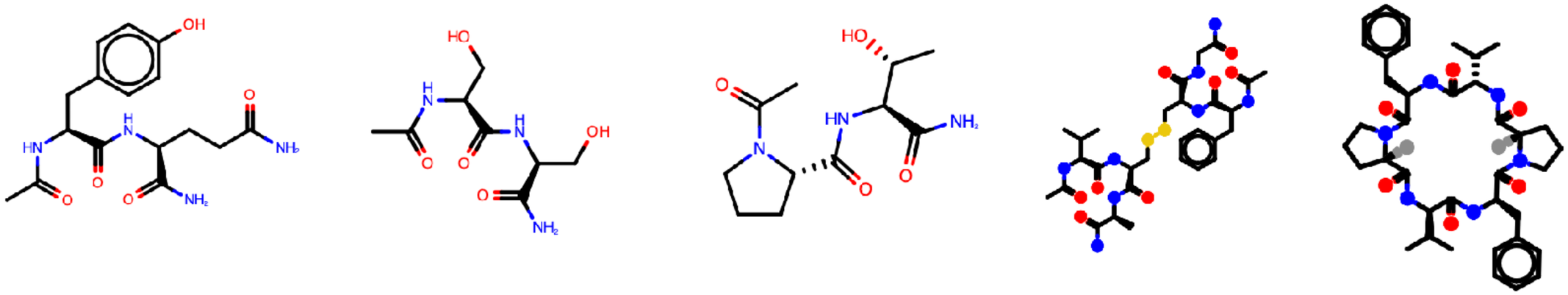
DFT B3LYP-D3(BJ) / DZVP

Wang et. al. (2022)
doi: 10.1039/D2SC02739A

how well does espaloma perform on amino acids?

(a) dataset	# mols	# trajs	# snapshots	Espaloma RMSE		Legacy FF RMSE (kcal/mol) (Test molecules)			
				Train	Test	OpenFF 1.2.0	GAFF-1.81	GAFF-2.11	Amber ff14SB
PhAlkEthOH (simple CHO)	7408	12592	244036	$0.8656^{0.9131}_{0.8225}$	$1.1398^{1.2332}_{1.0715}$	$1.6071^{1.6915}_{1.5197}$	$1.7267^{1.7935}_{1.6543}$	$1.7406^{1.8148}_{1.6679}$	
OpenFF Gen2 Optimization (druglike)	792	3977	23748	$0.7413^{0.7920}_{0.6914}$	$0.7600^{0.8805}_{0.6644}$	$2.1768^{2.3388}_{2.0380}$	$2.4274^{2.5207}_{2.3300}$	$2.5386^{2.6640}_{2.4370}$	
VEHICLe (heterocyclic)	24867	24867	234326	$0.4476^{0.4690}_{0.4273}$	$0.4233^{0.4414}_{0.4053}$	$8.0247^{8.2456}_{7.8271}$	$8.0077^{8.2313}_{7.7647}$	$9.4014^{9.6434}_{9.2135}$	
PepConf (peptides)	736	7560	22154	$1.2714^{1.3616}_{1.1899}$	$1.8727^{1.9749}_{1.7309}$	$3.6143^{3.7288}_{3.4870}$	$4.4446^{4.5738}_{4.3386}$	$4.3356^{4.4641}_{4.1965}$	$3.1502^{3.1859,*}_{3.1117}$

PepConf: Short peptides, including disulfides and cyclic peptides

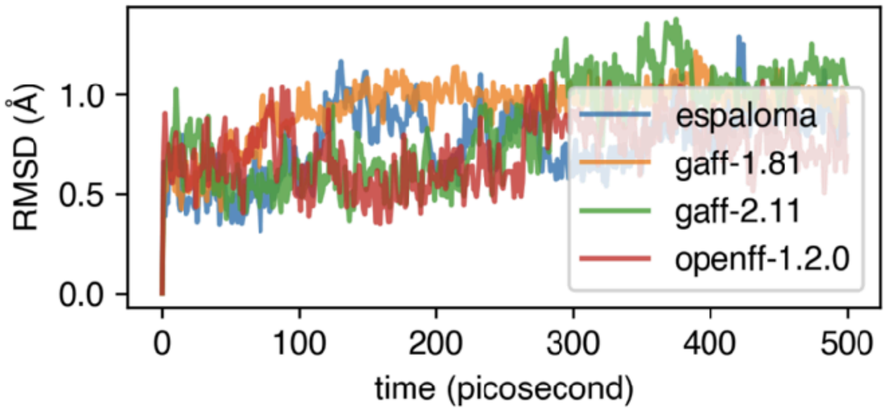


VEHICLe doi: http://doi.org/10.1021/jm801513z

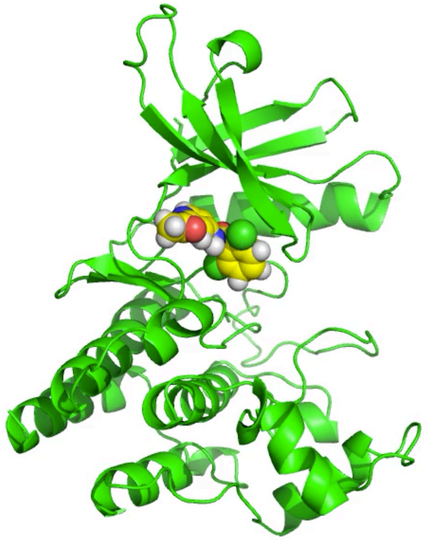
Wang et. al. (2022)
doi: 10.1039/D2SC02739A

espaloma outperforms current force fields in QM accuracy and can be easily trained for heterogeneous systems

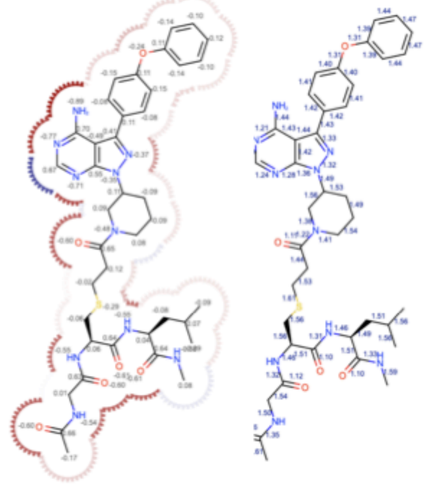
(a) dataset		# mols	# trajs	# snapshots	Espaloma RMSE		Legacy FF RMSE (kcal/mol) (Test molecules)			
					Train	Test	OpenFF 1.2.0	GAFF-1.81	GAFF-2.11	Amber ff14SB
PhAlkEthOH (simple CHO)		7408	12592	244036	$0.8656^{0.9131}_{0.8225}$	$1.1398^{1.2332}_{1.0715}$	$1.6071^{1.6915}_{1.5197}$	$1.7267^{1.7935}_{1.6543}$	$1.7406^{1.8148}_{1.6679}$	
OpenFF Gen2 Optimization (druglike)		792	3977	23748	$0.7413^{0.7920}_{0.6914}$	$0.7600^{0.8805}_{0.6644}$	$2.1768^{2.3388}_{2.0380}$	$2.4274^{2.5207}_{2.3300}$	$2.5386^{2.6640}_{2.4370}$	
VEHICLe (heterocyclic)		24867	24867	234326	$0.4476^{0.4690}_{0.4273}$	$0.4233^{0.4414}_{0.4053}$	$8.0247^{8.2456}_{7.8271}$	$8.0077^{8.2313}_{7.7647}$	$9.4014^{9.6434}_{9.2135}$	
PepConf (peptides)		736	7560	22154	$1.2714^{1.3616}_{1.1899}$	$1.8727^{1.9749}_{1.7309}$	$3.6143^{3.7288}_{3.4870}$	$4.4446^{4.5738}_{4.3386}$	$4.3356^{4.4641}_{4.1965}$	$3.1502^{3.1859,*}_{3.1117}$
joint	OpenFF Gen2 Optimization	1528	11537	45902	$0.8264^{0.9007}_{0.7682}$	$1.8764^{1.9947}_{1.7827}$	$2.1768^{2.3388}_{2.0380}$	$2.4274^{2.5207}_{2.3300}$	$2.5386^{2.6640}_{2.4370}$	
	PepConf				$1.2038^{1.3056}_{1.1178}$	$1.7307^{1.8439}_{1.6053}$	$3.6143^{3.7288}_{3.4870}$	$4.4446^{4.5738}_{4.3386}$	$4.3356^{4.4641}_{4.1965}$	$3.1502^{3.1859,*}_{3.1117}$



Tyk2 benchmark doi: https://doi.org/10.1021/ja512751q



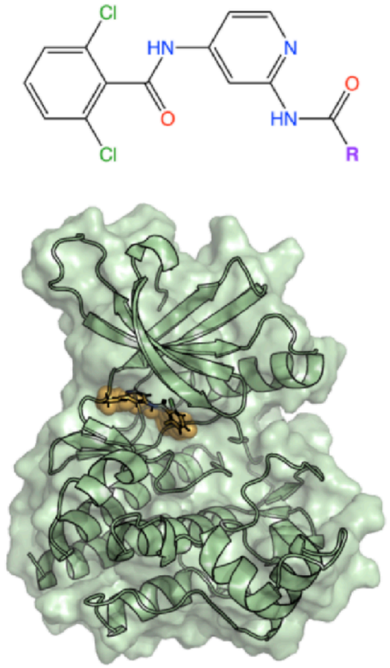
Tyk2 from OpenFF benchmark set
espaloma **joint** model
+ TIP3P water



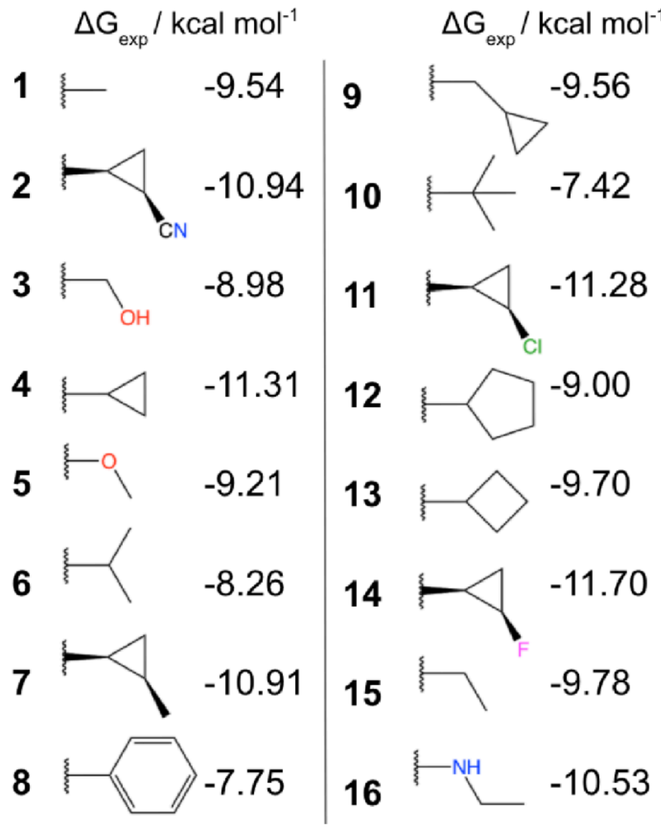
covalent adduct can be parametrized stably

Wang et. al. (2022)
doi: 10.1039/D2SC02739A

Espaloma small molecule parameters perform as well or better than modern biomolecular force fields

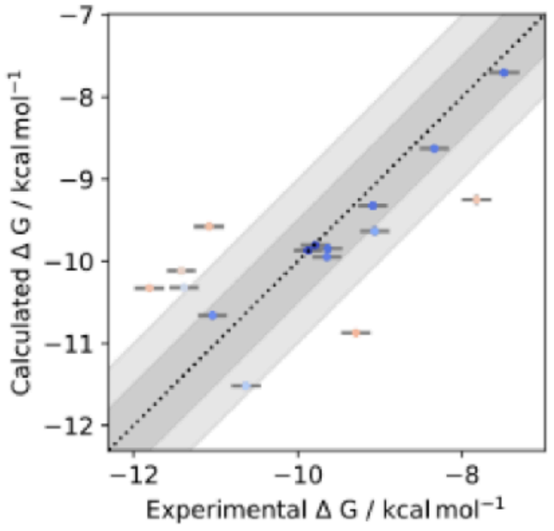


Tyk2
PDBID: 4GIH



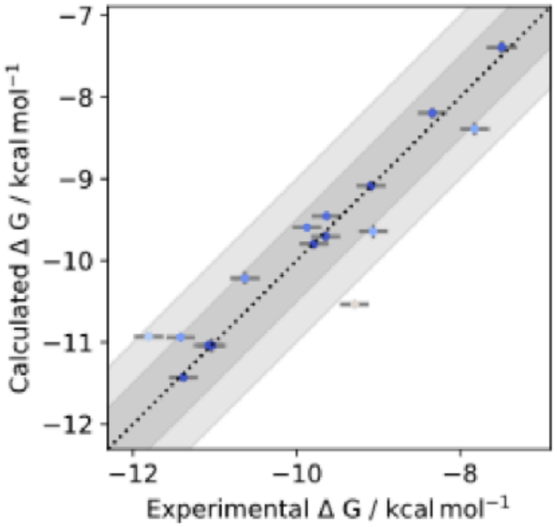
OpenFF 1.2.0 small molecule
Amber ff14SB protein
TIP3P water

```
Absolute binding energies - tyk2
                    tyk2 (N = 16)
RMSE:   0.91 [95%: 0.66, 1.17]
MUE:    0.72 [95%: 0.47, 1.03]
R2:     0.48 [95%: 0.09, 0.78]
rho:    0.69 [95%: 0.28, 0.89]
```



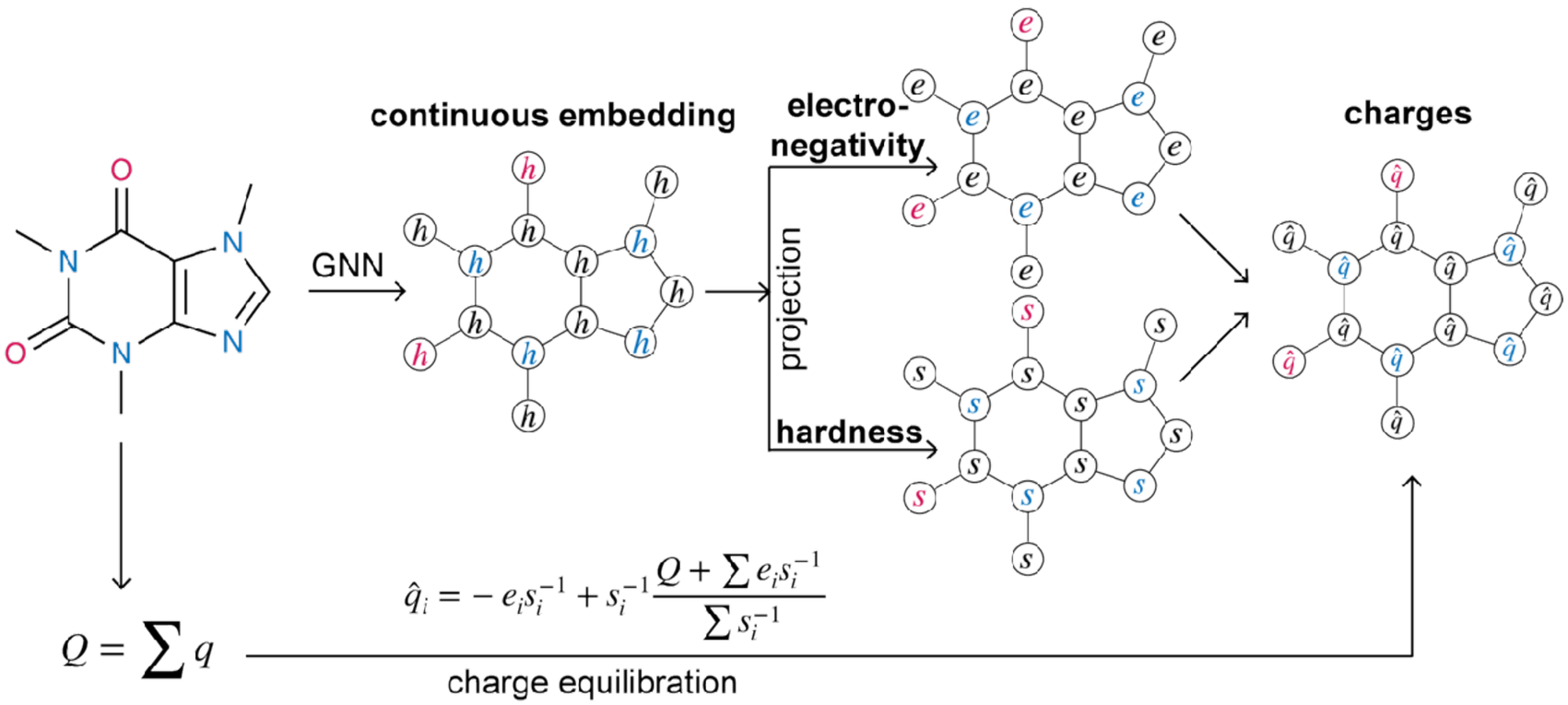
espaloma "joint" 0.2.2 small molecule
Amber ff14SB protein
TIP3P water

```
Absolute binding energies - tyk2
                    tyk2 (N = 16)
RMSE:   0.47 [95%: 0.30, 0.70]
MUE:    0.31 [95%: 0.22, 0.56]
R2:     0.87 [95%: 0.62, 0.96]
rho:    0.93 [95%: 0.80, 0.98]
```



Wang et. al. (2022)
doi: 10.1039/D2SC02739A

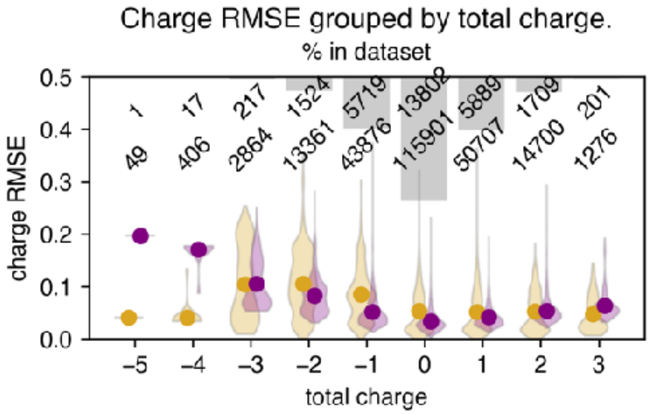
EspalomaCharge: Machine learning-enabled ultra-fast partial charge assignment

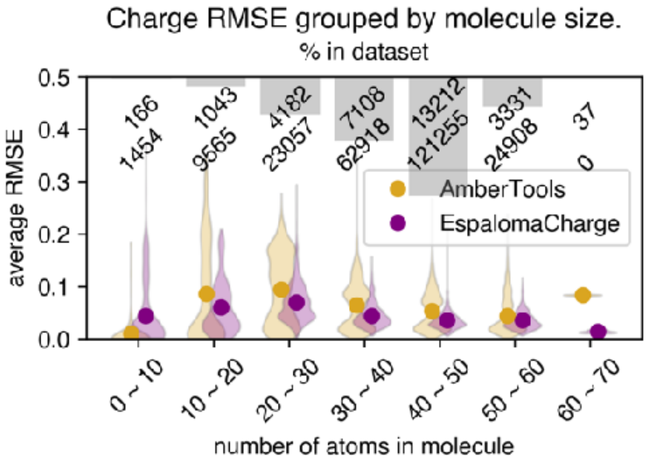


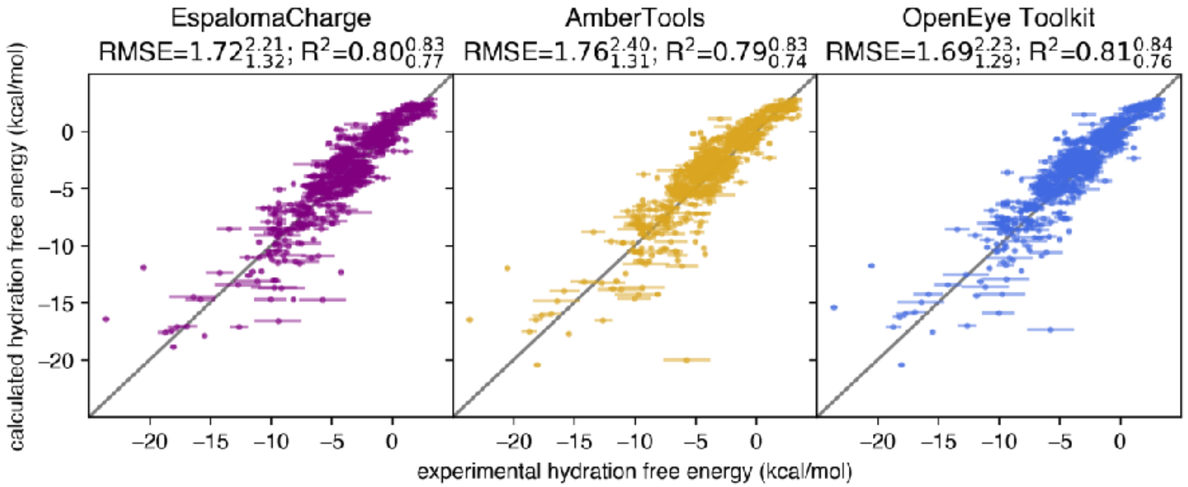
EspalomaCharge introduces minimal discrepancy

dataset	N_{mol}	avg. N_{atoms}	average RMSE (e)		average walltime (s)		
			|EspalomaCharge - OpenEye|	|AmberTools - OpenEye|	EspalomaCharge	AmberTools	OpenEye
SPICE [12] test set	29079	39.36	$\mathbf{0.0435}^{0.0438}_{0.0432}$	$0.0623^{0.0628}_{0.0618}$	**0.05**	93.10	3.79
FDA approved	1019	34.80	$\mathbf{0.0266}^{0.0281}_{0.0255}$	$\mathbf{0.0244}^{0.0263}_{0.0227}$	**0.03**	46.15	1.87
ZINC250K [20]	220250	42.70	$\mathbf{0.0187}^{0.0187}_{0.0187}$	$0.0197^{0.0198}_{0.0197}$	**0.05**	124.89	3.63
FreeSolv [11]	641	18.10	$0.0110^{0.0117}_{0.0104}$	$\mathbf{0.0067}^{0.0077}_{0.0057}$	**0.03**	9.62	0.43
PDB expo [3]	23399	35.94	$\mathbf{0.0186}^{0.0188}_{0.0184}$	$0.0232^{0.0236}_{0.0229}$	**0.04**	88.86	3.63

Table 1. EspalomaCharge accurately and efficiently reproduces AM1-BCC charges for a wide variety of chemical spaces. Here, N_{mol} denotes the number of molecules in the dataset; avg. N_{atoms} denotes the average number of atoms in molecules for the corresponding dataset; average RMSE is the charge RMS deviation between AM1-BCC implementations averaged over all molecules in the dataset, with sub- and superscripts denoting the 95%-confidence interval of the mean (computed by bootstrapping over molecules in the dataset with replacement); average walltime denotes the average wall time for the respective toolkit to assign partial charges for a molecule in the dataset. Boldface statistics denote the best (most accurate or fastest) model or models (in case confidence intervals are indistinguishable) for each statistic.



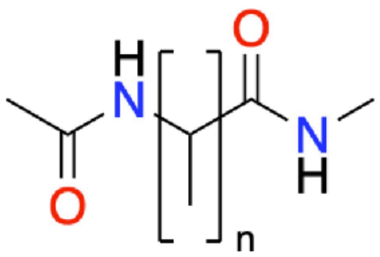


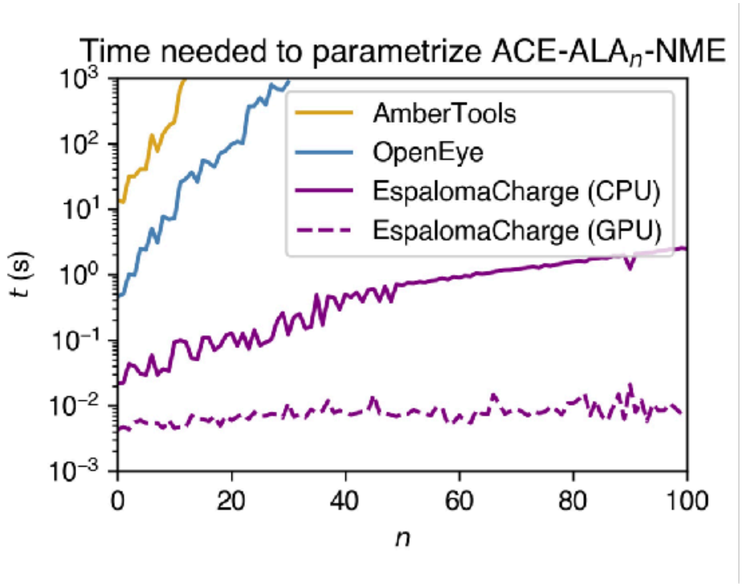


EspalomaCharge is very fast, even for large systems

dataset	N_{mol}	avg. N_{atoms}	average RMSE (e)		average walltime (s)		
			|EspalomaCharge - OpenEye|	|AmberTools - OpenEye|	EspalomaCharge	AmberTools	OpenEye
SPICE [12] test set	29079	39.36	$\mathbf{0.0435}^{0.0438}_{0.0432}$	$0.0623^{0.0628}_{0.0618}$	**0.05**	93.10	3.79
FDA approved	1019	34.80	$\mathbf{0.0266}^{0.0281}_{0.0255}$	$\mathbf{0.0244}^{0.0263}_{0.0227}$	**0.03**	46.15	1.87
ZINC250K [20]	220250	42.70	$\mathbf{0.0187}^{0.0187}_{0.0187}$	$0.0197^{0.0198}_{0.0197}$	**0.05**	124.89	3.63
FreeSolv [11]	641	18.10	$0.0110^{0.0117}_{0.0104}$	$\mathbf{0.0067}^{0.0077}_{0.0057}$	**0.03**	9.62	0.43
PDB expo [3]	23399	35.94	$\mathbf{0.0186}^{0.0188}_{0.0184}$	$0.0232^{0.0236}_{0.0229}$	**0.04**	88.86	3.63

Table 1. EspalomaCharge accurately and efficiently reproduces AM1-BCC charges for a wide variety of chemical spaces. Here, N_{mol} denotes the number of molecules in the dataset; avg. N_{atoms} denotes the average number of atoms in molecules for the corresponding dataset; average RMSE is the charge RMS deviation between AM1-BCC implementations averaged over all molecules in the dataset, with sub- and superscripts denoting the 95%-confidence interval of the mean (computed by bootstrapping over molecules in the dataset with replacement); average walltime denotes the average wall time for the respective toolkit to assign partial charges for a molecule in the dataset. Boldface statistics denote the best (most accurate or fastest) model or models (in case confidence intervals are indistinguishable) for each statistic.





to integrate EspalomaCharge into your pipeline(s) is easy

```
$ pip install espaloma_charge
```

Listing 1. Installing EspalomaCharge via the pip Python package manager.

```
>>> from rdkit import Chem; from espaloma_charge import charge
>>> molecule = Chem.AddHs(Chem.MolFromSmiles("CCO"))
>>> charge(molecule)
array([ 0.95081186, -1.007628  ,  0.93298626, -0.1128267 , -0.1128267 ,
       -0.1128267 , -0.10835528, -0.10835528, -0.32097816], dtype=float32)
```

Listing 2. Example illustrating the EspalomaCharge Python API. Here, EspalomaCharge assigns AM1-BCC ELF10 equivalent partial charges to an RDKit Molecule, returning them in a NumPy array.

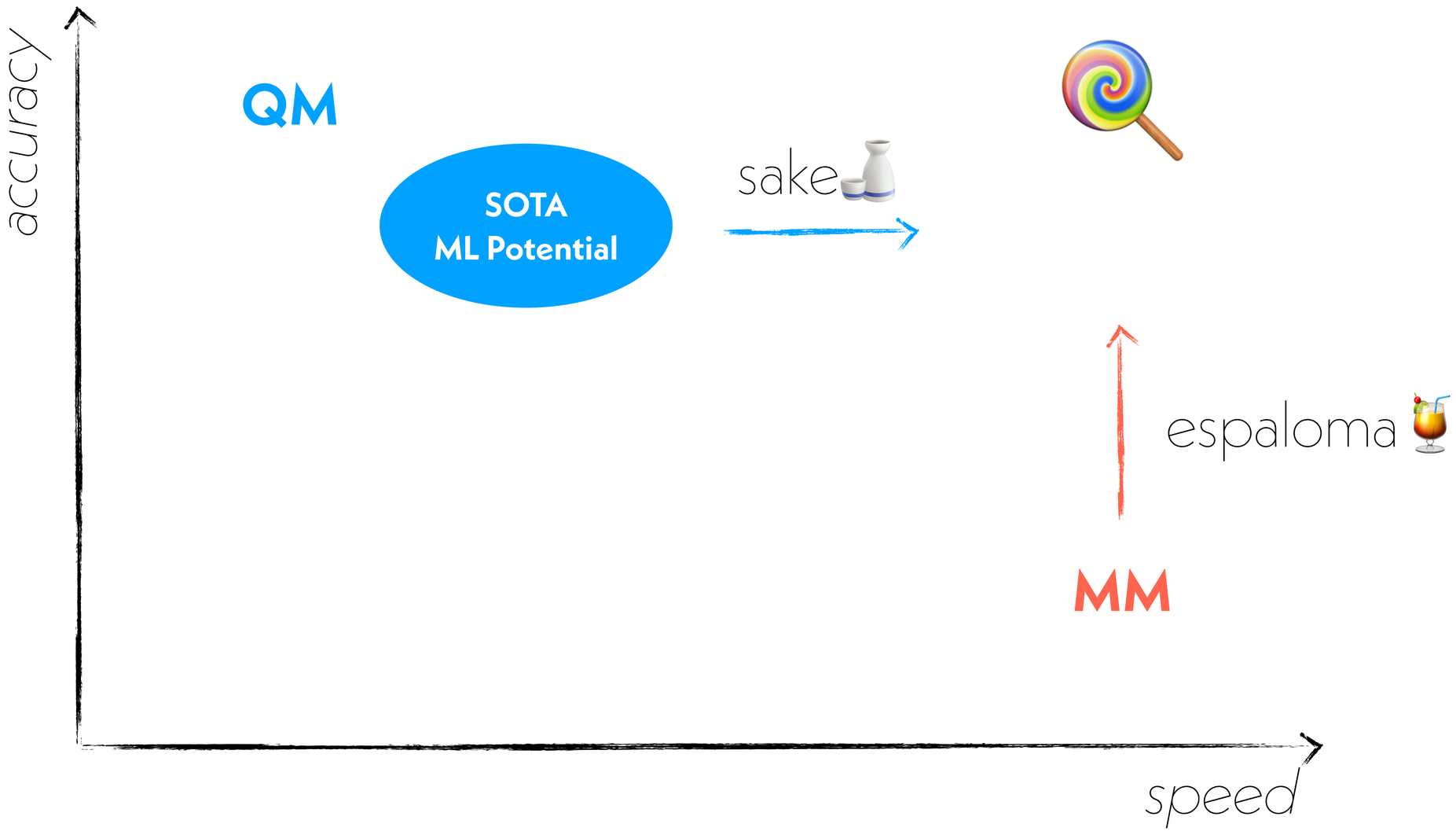
to integrate EspalomaCharge into your pipeline(s) is easy

```
>>> from openff.toolkit.topology import Molecule
>>> from espaloma_charge.openff_wrapper import EspalomaChargeToolkitWrapper
>>> toolkit_registry = EspalomaChargeToolkitWrapper()
>>> molecule = Molecule.from_smiles("CCO")
>>> molecule.assign_partial_charges('espaloma-am1bcc', toolkit_registry=toolkit_registry)
>>> molecule.partial_charges
<Quantity([ 0.95081172 -1.00762811  0.93298611 -0.11282685 -0.11282685 -0.11282685
 -0.10835543 -0.10835543 -0.32097831], 'elementary_charge')>
```

Listing 3. Example illustrating EspalomaCharge integration with the Open Force Field Toolkit. Here EspalomaCharge is used to provide charges via the ToolkitWrapper facility.

```
$ espaloma_charge -i in.mol2 -o in.crg
$ antechamber -fi mol2 -fo mol2 -i in.mol2 -o out.mol2 -c rc -cf in.crg
```

Listing 4. Example illustrating the use of EspalomaCharge as a fast drop-in replacement for `sqm` in an AmberTools `antechamber` workflow. By adding a single line EspalomaCharge can replace the slow `sqm`-based AM1-BCC model to provide fast charges for AmberTools based worklows.


accuracy
QM
SOTA
ML Potential
sake
espaloma
MM
speed

invariant features and equivariant features

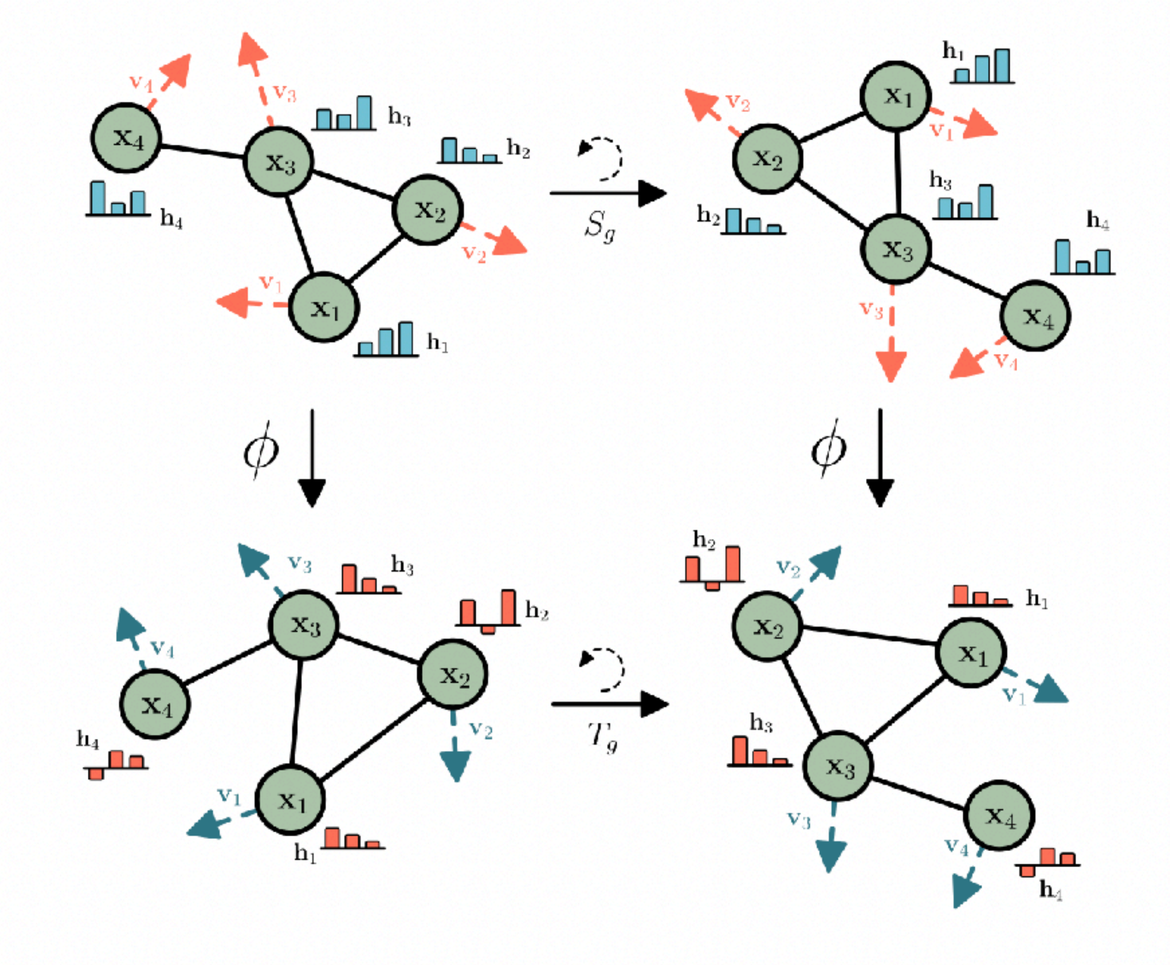


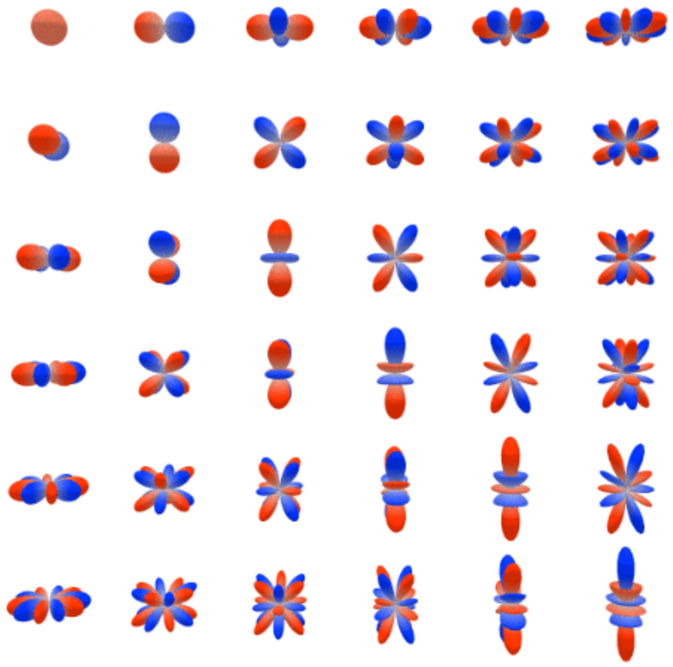
Figure 1. Example of rotation equivariance on a graph with a graph neural network ϕ

invariant features:
- embedding
- energy
- atom properties
- molecular properties

equivariant features:
- position
- velocity
- acceleration

Sarrotas et. al. (2021) https://arxiv.org/abs/2102.09844

you don't need spherical harmonics to describe node environments
spatial attention is universally approximative



e3nn.org

Given a node v with embedding $h_v \in \mathcal{H} = \mathbb{R}^C$ (where C denotes the embedding dimension) and position $\mathbf{x}_v \in \mathcal{X} = \mathbb{R}^n$ (where n denotes the geometry dimension) in a graph $\mathcal{G}$, its neighbors $u \in \mathcal{N}(v)$ with connecting edges $\{e_{uv}\}$, with displacement vector $\vec{\mathbf{e}}_{uv} = \mathbf{x}_v - \mathbf{x}_u$ and embedding $h_{e_{uv}} = \rho^{v \to e}(h_v, h_u)$ with some aggregation function $\rho^{v \to e}$, we define spatial attention $\phi : \mathcal{X} \times \mathcal{H} \to \mathcal{H}$ as

$$\phi^{\mathrm{SA}}(v) = \mu\Big(\bigoplus_{i=1}^{N_\lambda} \Big|\Big| \sum_{u \in \mathcal{N}(v)} \lambda_i(h_{e_{uv}}) f(\vec{\mathbf{e}}_{uv}) \Big|\Big|\Big), \tag{6}$$

where $\lambda_i : \mathcal{H} \to \mathbb{R}^1, i = 1, ..., N_\lambda$ is a set of arbitrary attention weights-generating function that operates on the edge embedding, $f : \mathcal{X} \to \mathcal{X}$ is an *equivariant* function that operate on the edge vector, $\mu : N_\lambda \to \mathcal{H}$ is an arbitrary function that takes the norms of N_λ linear combinations, and $\oplus$ denotes concatenation. We drop the explicit dependence of $\phi^{\mathrm{SA}}(v)$ on both geometric and embedding properties of v and u for simplicity.

Wang and Chodera (2022)
arXiv:2301.08893

invariant task: machine learning potential construction

Table 1: Inference time (ms) and test set energy (E) and force (F) mean absolute error (MAE) (meV and meV/Å) on the MD17 quantum chemical dataset.

Model		SchNet (Schütt et al., 2017)	DimeNet (Klicpera et al., 2020b)	sGDML (Chmiela et al., 2019)	PaiNN (Schütt et al., 2021)	GemNet(T/Q) (Klicpera et al., 2021a)	NequIP (Batzner et al., 2021)	SAKE
Inference time	batch of 32		65			88/376	206	**12**
	batch of 4		31			38/99	197	**4**
Aspirin	E	16.0	8.8	8.2	6.9	-	**5.3**	$6.46^{6.47}_{6.43}$
	F	58.5	21.6	29.5	14.7	9.4	**8.2**	$9.90^{9.92}_{9.88}$
Ethanol	E	3.5	2.8	3.0	2.7	-	**2.2**	$\mathbf{2.26}^{2.27}_{2.25}$
	F	16.9	10.0	14.3	9.7	3.7	3.8	$\mathbf{3.70}^{3.77}_{3.62}$
Malonaldehyde	E	5.6	4.5	4.3	3.9	-	**3.3**	$\mathbf{3.26}^{3.27}_{3.26}$
	F	28.6	16.6	17.8	13.8	6.7	5.8	$\mathbf{5.17}^{5.18}_{5.16}$
Naphtalene	E	6.9	5.3	5.2	5.0	-	**4.9**	$\mathbf{4.94}^{4.95}_{4.93}$
	F	25.2	9.3	4.8	3.3	2.2	**1.6**	$2.44^{2.44}_{2.44}$
Salicylic acid	E	8.7	5.8	5.2	4.9	-	**4.0**	$4.76^{4.77}_{4.75}$
	F	36.9	16.2	12.1	8.5	5.4	**3.9**	$5.14^{5.17}_{5.13}$
Toluene	E	5.2	4.4	4.3	4.1	-	**4.0**	$\mathbf{4.02}^{4.02}_{4.01}$
	F	24.7	9.4	6.1	4.1	2.6	**2.0**	$2.44^{2.44}_{2.43}$
Uracil	E	6.1	5.0	4.8	4.5	-	**4.5**	$\mathbf{4.52}^{4.54}_{4.50}$
	F	24.3	13.1	10.4	6.0	4.2	**3.3**	$4.05^{4.06}_{4.04}$

Table 2: Test set energy (E) and force (F) mean absolute error (MAE) (meV and meV/Å) on known and unknown molecules in ISO17.

		ACE (Kovács et al., 2021)	SchNet (Schütt et al., 2017)	PhysNet (Unke and Meuwly, 2019)	SAKE
known	E	16	16	**4**	$12.17^{12.18}_{12.12}$
	F	43	43	**5**	$12.33^{12.34}_{12.31}$
unknown	E	85	104	127	$\mathbf{53.37}^{53.62}_{53.15}$
	F	85	95	60	$\mathbf{39.46}^{39.59}_{39.35}$

Table 3: QM9 test set performance.

	α Bohr^3	$\Delta\epsilon$ meV	HOMO meV	LUMO meV	μ D	C_v cal/mol K
SchNet (Schütt et al., 2017)	0.235	63	41	34	0.033	0.033
DimeNet++ (Klicpera et al., 2020a)	0.044	33	25	20	0.030	0.023
SE(3)-TF (Fuchs et al., 2020)	0.142	53	35	33	0.051	0.054
EGNN (Satorras et al., 2021)	0.071	48	29	25	0.029	0.031
PaiNN (Schütt et al., 2021)	0.059	36	46	20	**0.012**	**0.024**
TorchMD-Net (Thölke and Fabritiis, 2022)	0.059	36	20	17	**0.011**	**0.023**
SphereNet (Liu et al., 2022)	**0.030**	31	19	23	0.025	**0.022**
SAKE	0.068	**23**	**16**	**13**	**0.014**	0.087

Wang and Chodera (2022)
arXiv:2301.08893

equivariant task: N-body dynamic system forecasting

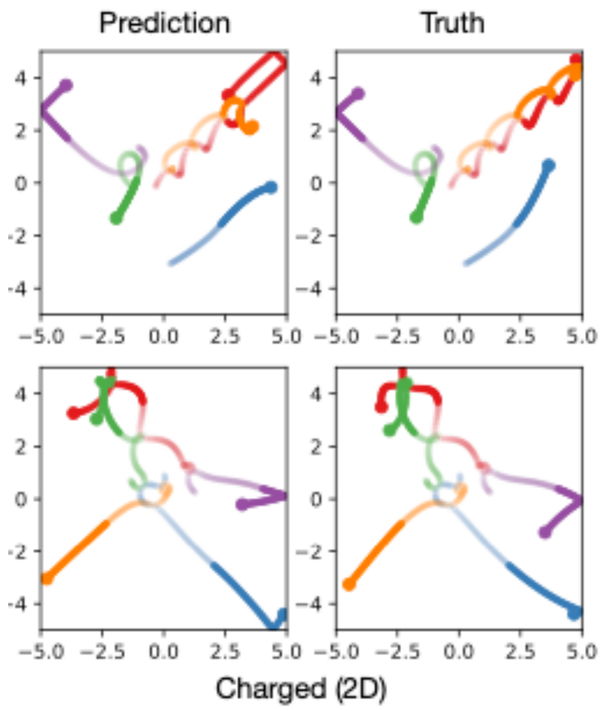


Table 4: Mean Squared Error (MSE) and inference time (ms)for dynamic system forecasting.

Architecture	MSE	Inference time
SE(3)-TF (Fuchs et al., 2020)	0.244	0.1346
TFN (Thomas et al., 2018)	0.155	0.0343
GNN (Kipf and Welling, 2016)	0.0107	0.0032
EGNN (Satorras et al., 2021)	0.0071	0.0062
SAKE	0.0049	0.0079
SEGNN (Brandstetter et al., 2021)	**0.0043**	0.0260

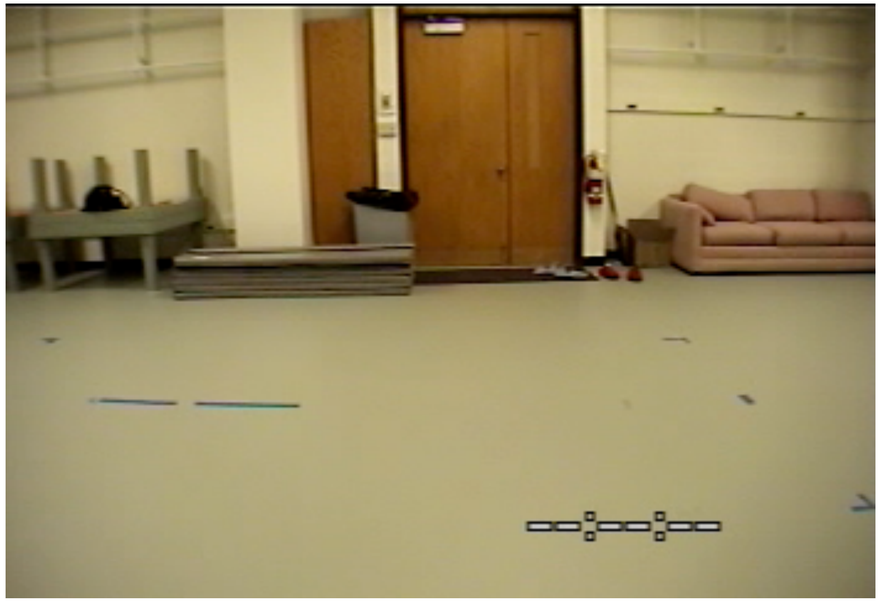
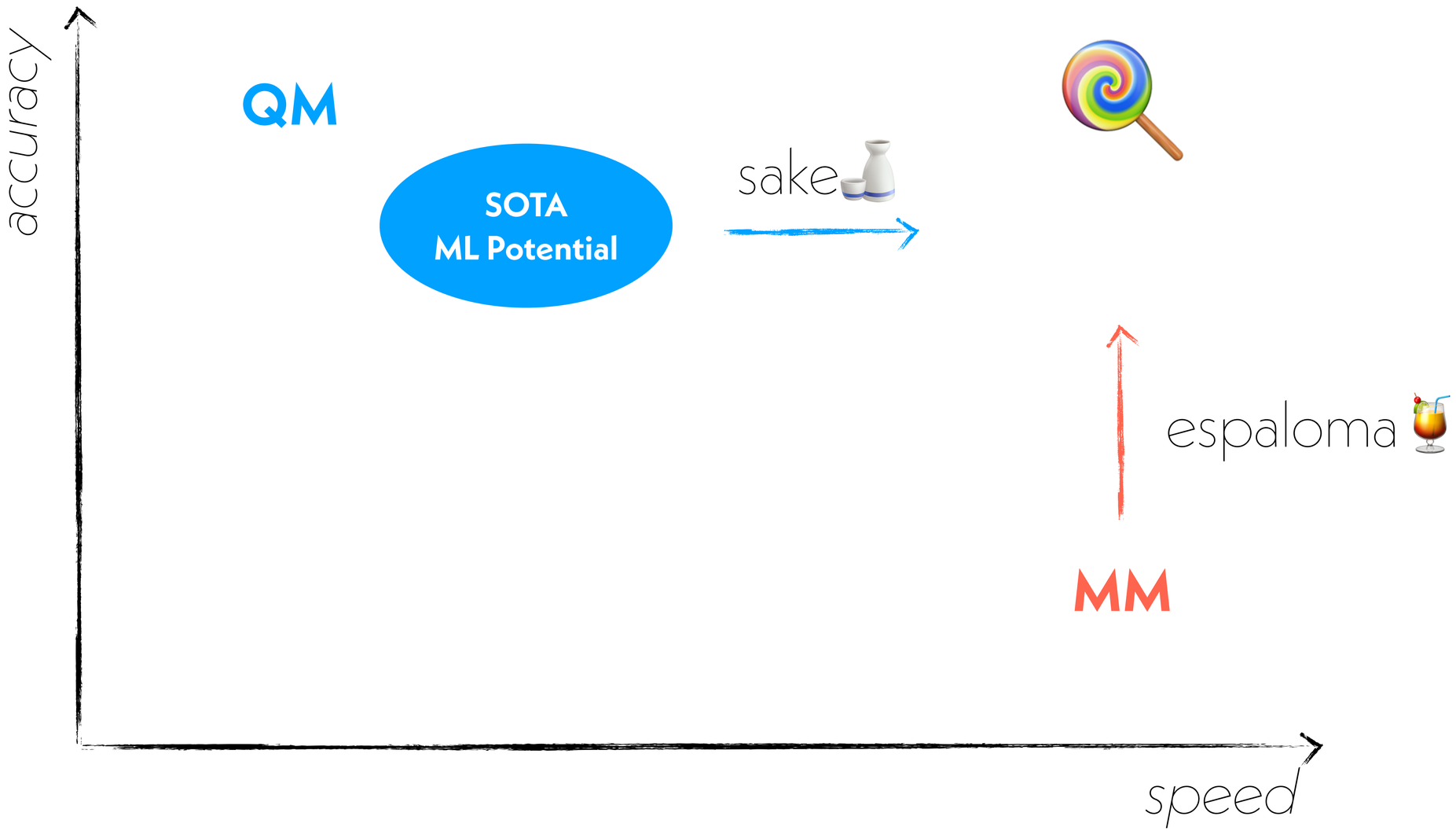


Table 5: Walking motion capture performance.

	GNN	EGNN (Satorras et al., 2021)	GMN (Huang et al., 2022)	SAKE
MAE	67.3±1.1	59.1±2.1	43.9±1.1	**22.7** ±1.6
Epoch time			5.66 s	**1.81** s

Wang and Chodera (2022)
arXiv:2301.08893


accuracy
QM
SOTA
ML Potential
sake
espaloma
MM
speed

espaloma 🍹 **sake** 🍶

only constraining key geometry? Huang et al. (2022)
treat long-range interactions classically?

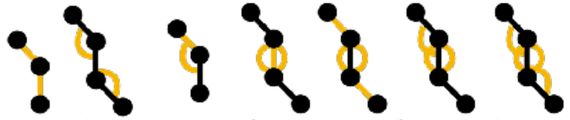
hard-coded topology ⟷ inferred topology

message passing once,
and then local interaction

local NN models? Musaelian et al. (2022) arXiv:2204.05249

message passing
every step during inference

traditional MM
simple, elegant, and interpretable



class-II force fields or
other symmetry fn?

Hwang et al. (1994) doi:10.1021/ja00085a036

blackboxy

tooling to makes this a reality:
google/jax-md
choderalab/espalomax

Can class ii force fields provide NEAR-QM ACCURACY at mm speeds?

$$
\begin{aligned}
E = & \sum_{b} [{}^{2}K_b(b-b_0)^2 + {}^{3}K_b(b-b_0)^3 + {}^{4}K_b(b-b_0)^4] \\
& + \sum_{\theta} [{}^{2}K_\theta(\theta-\theta_0)^2 + {}^{3}K_\theta(\theta-\theta_0)^3 + {}^{4}K_\theta(\theta-\theta_0)^4] \\
& + \sum_{\phi} [{}^{1}K_\phi(1-\cos\phi) + {}^{2}K_\phi(1-\cos 2\phi) + {}^{3}K_\phi(1-\cos 3\phi)] \\
& + \sum_{\chi} K_\chi \chi^2 + \sum_{i>j} \frac{q_i q_j}{r_{ij}} + \sum_{i>j} \epsilon \left[2\left(\frac{r^*}{r_{ij}}\right)^9 - 3\left(\frac{r^*}{r_{ij}}\right)^6 \right] \\
& + \sum_{b} \sum_{b'} K_{bb'}(b-b_0)(b'-b'_0) + \sum_{\theta} \sum_{\theta'} K_{\theta\theta'}(\theta-\theta_0) \times \\
& \quad (\theta'-\theta'_0) \\
& + \sum_{b} \sum_{\theta} K_{b\theta}(b-b_0)(\theta-\theta_0) \\
& + \sum_{\phi} \sum_{b} (b-b_0)[{}^{1}K_{\phi b} \cos\phi + {}^{2}K_{\phi b} \cos 2\phi + {}^{3}K_{\phi b} \cos 3\phi] \\
& + \sum_{\phi} \sum_{b'} (b'-b'_0)[{}^{1}K_{\phi b'} \cos\phi + {}^{2}K_{\phi b'} \cos 2\phi + \\
& \quad {}^{3}K_{\phi b'} \cos 3\phi] \\
& + \sum_{\phi} \sum_{\theta} (\theta-\theta_0)[{}^{1}K_{\phi\theta} \cos\phi + {}^{2}K_{\phi\theta} \cos 2\phi + {}^{3}K_{\phi\theta} \cos 3\phi] \\
& + \sum_{\phi} \sum_{\theta} \sum_{\theta'} K_{\phi\theta\theta'} (\theta-\theta_0)(\theta'-\theta'_0) \cos\phi \qquad (1)
\end{aligned}
$$

bond-bond: angle node

angle-angle: torsion node

bond-angle: angle node

torsion-(center) bond: torsion

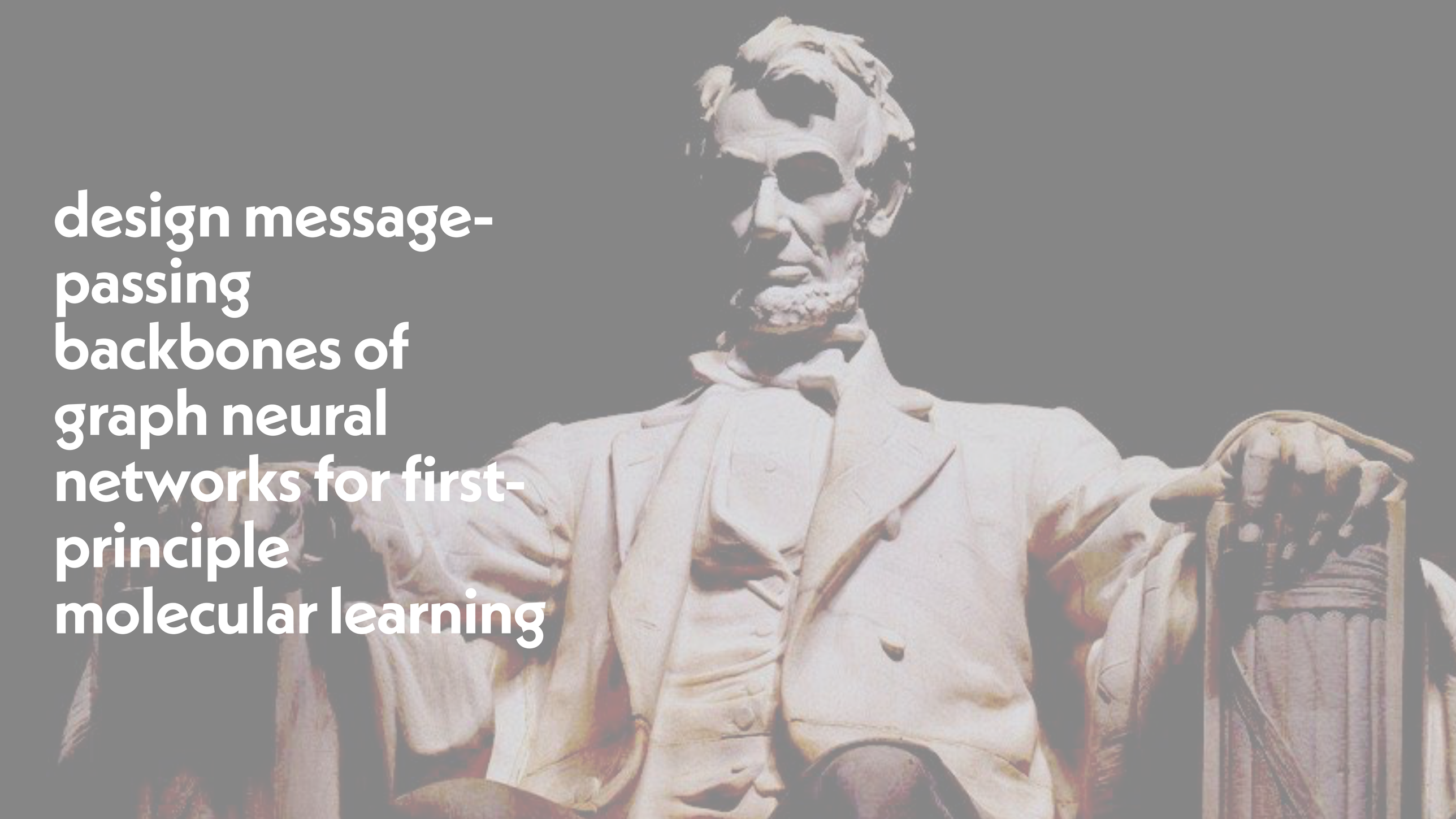
torsion-(side) bond: torsion

torsion-angle: torsion

torsion-angle-angle: torsion

Hwang et al. (1994) http://doi.org/10.1021/ja00085a036

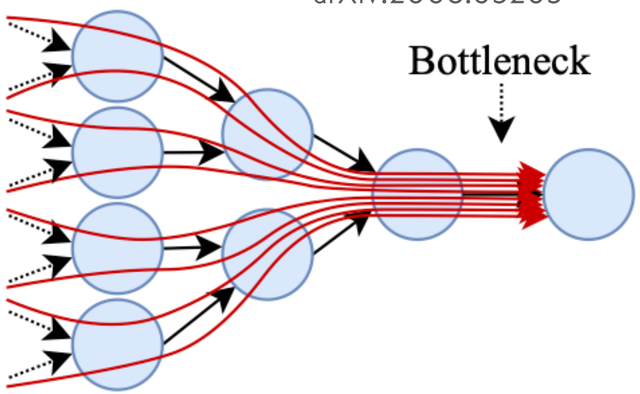
design message-passing backbones of graph neural networks for first-principle molecular learning



original sins of message-passing graph neural networks

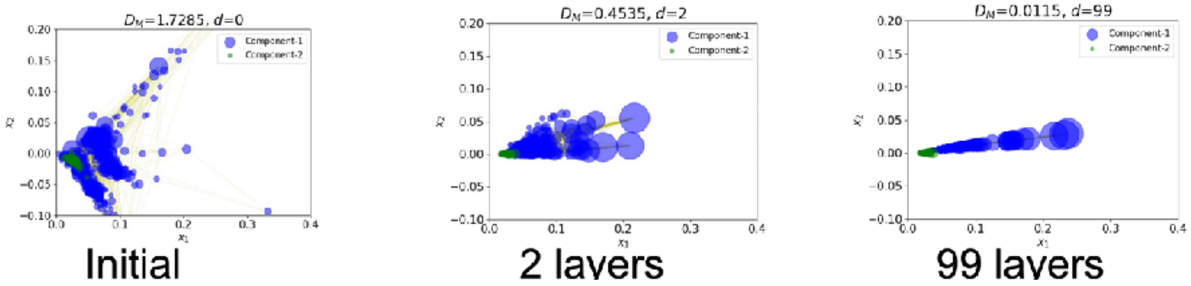
graph convolution is iterative averaging with neighbors

Alon and Yahav(2021)
arXiv:2006.05205



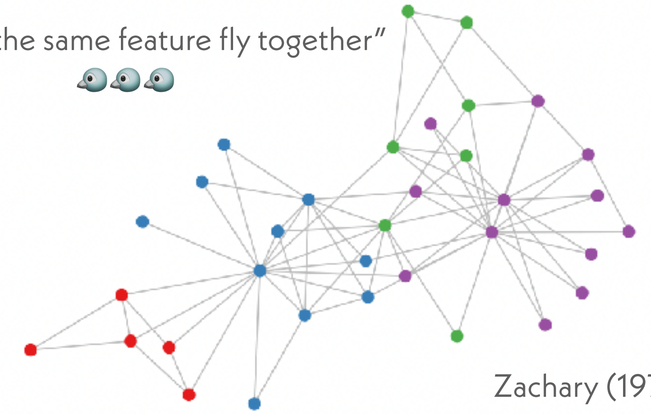
oversmoothing: nodes became more and more similar

Li et. al. (2018)
arXiv:1801.07606



they were built for social networks and encodes the assumption of homophily

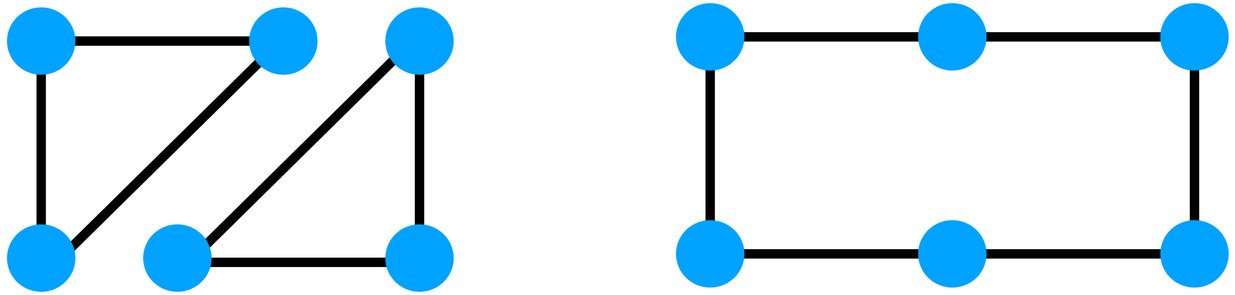
" birds of the same feature fly together"



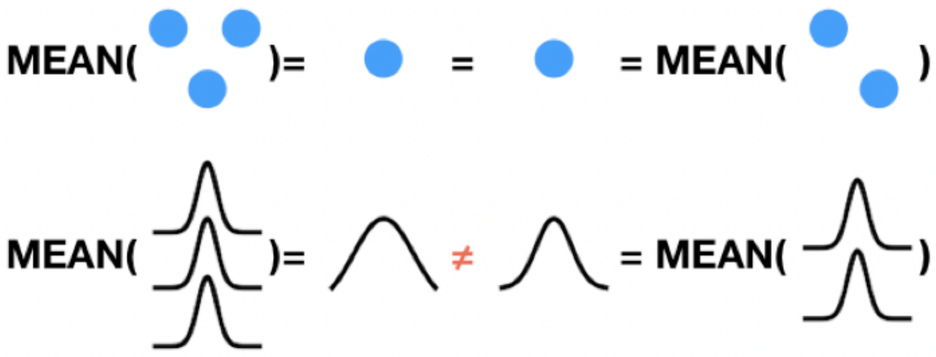
Karate Club Social Network

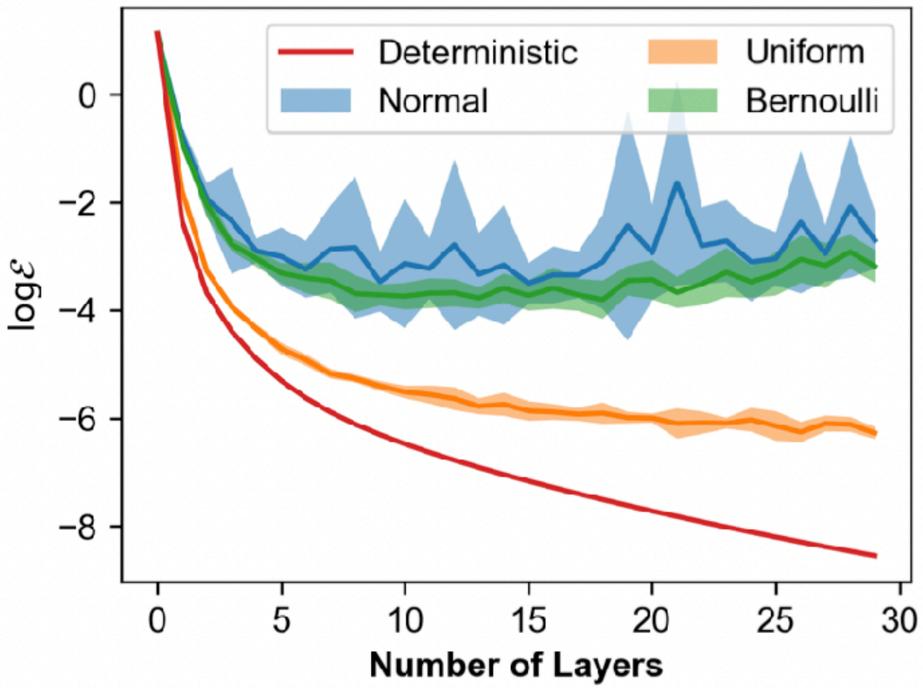
Zachary (1977)
Journal of Anthropological Research
Vol. 33, No. 4 (Winter, 1977), pp. 452-473 (22 pages)

limited expressiveness: no Weisfeiler-Lehman GNN can tell these apart!



passing random variables rather than point masses as messages alleviate limited expressiveness and oversmoothing





Definition 3.1 from Cai and Wang (2020) 1. *Dirichlet energy $\mathcal{E}(f)$ of scalar function f on the graph G is defined as*

$$\mathcal{E}(f) = f^T \widetilde{\Delta} f = \frac{1}{2} \sum A_{ij} \left(\frac{f_i}{\sqrt{1+d_i}} - \frac{f_j}{\sqrt{1+d_j}} \right)^2, \tag{22}$$

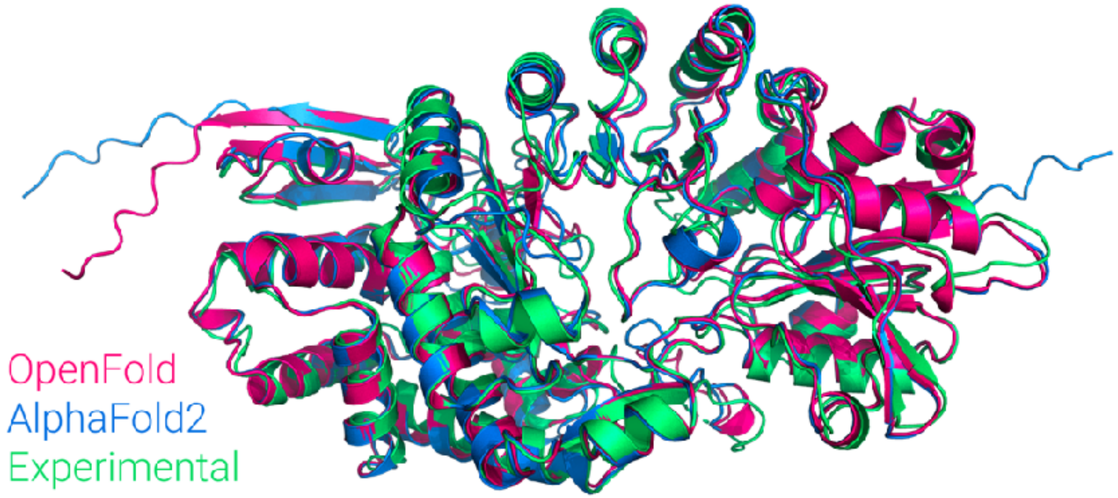
where $\widetilde{\Delta}$ is the normalized Laplacian $\widetilde{\Delta} = \mathbf{I} - \widetilde{D}^{-\frac{1}{2}} \widetilde{A} \widetilde{D}^{-\frac{1}{2}}$ and $d_i = D_{ii}$. For a vector field $\mathbf{X} \in \mathbb{R}^{N \times C}$, Dirichlet energy is defined as

$$\mathcal{E}(\mathbf{X}) = \mathrm{tr}(\mathbf{X}^T \widetilde{\Delta} \mathbf{X}). \tag{23}$$

	ESOL	FreeSolv
$\mathrm{STAG_{VI}}(\mathbb{R})$	0.5956 ± 0.0200	1.1500 ± 0.0359
$\mathrm{STAG_{VI}}(\mathbb{R}^C)$	0.6221 ± 0.0142	1.1561 ± 0.0803
$\mathrm{STAG_{VI}}(\mathbb{R}^{\lvert\mathcal{E}\rvert})$	0.6901 ± 0.0427	1.3349 ± 0.1513
$\mathrm{STAG_{VI}}(\mathbb{R}^{\lvert\mathcal{E}\rvert \times C})$	**0.5928 ± 0.0326**	**0.9958 ± 0.0768**
$\mathrm{STAG_{MLE}}$ (best)	0.5960 ± 0.0375	1.1394 ± 0.0714

Table 4. Performance of STAG with variational inference (VI) on molecule graph datasets

Wang and Karaletsos (2021)
arXiv:2102.12648



Why OpenFold?

Open Source

An open-source project that can be used and improved by academics and companies alike.

Weights Available

Utilize the existing pre-trained weights to get quickly get started fine-tuning your model.

Permissive License

A permissively licensed model that allows commercial and non-commercial use.

Training Pipeline

Provides the tools used to train the model under the same license.

Optimized for Performance

Optimized performance for use on state-of-the-art and widely available GPUs.

PyTorch-Based

A supercomputer scale, distributed training, PyTorch-based training framework

Memorial Sloan Kettering Cancer Center

many thanks! 谢谢

Weill Cornell Medicine

- choderalab/espaloma 🍹
- choderalab/malt 🥃 molecular active learning testbed
- choderalab/gimlet 🍸 graph inference on molecular topology
- choderalab/pinot 🍷 probabilistic inference for novel therapeutics
- yuanqing-wang/sake 🍶 spatial attention kinetic network with equivariance
- yuanqing-wang/galax 🌃 graph learning with JAX

@YuanqingWang Yuanqing-Wang ⌨️ wangyq.net

AMGEN Google anagenex