MOFFlow: Flow Matching for Structure Prediction of Metal-Organic Frameworks

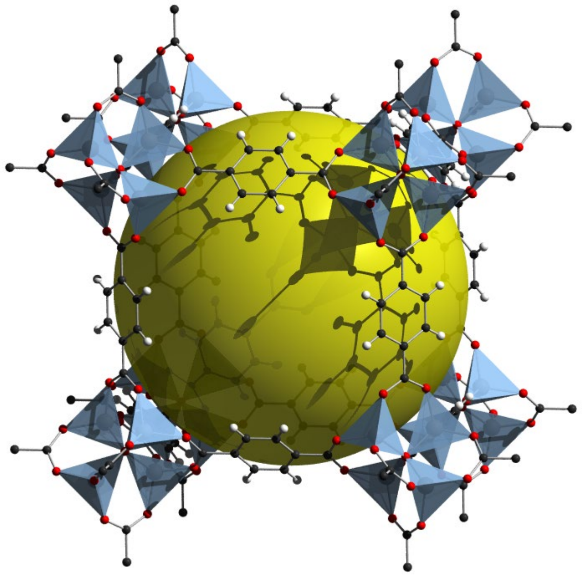
2025.04.01

Nayoung Kim, Seongsu Kim, Minsu Kim, Jinkyoo Park, Sungsoo Ahn

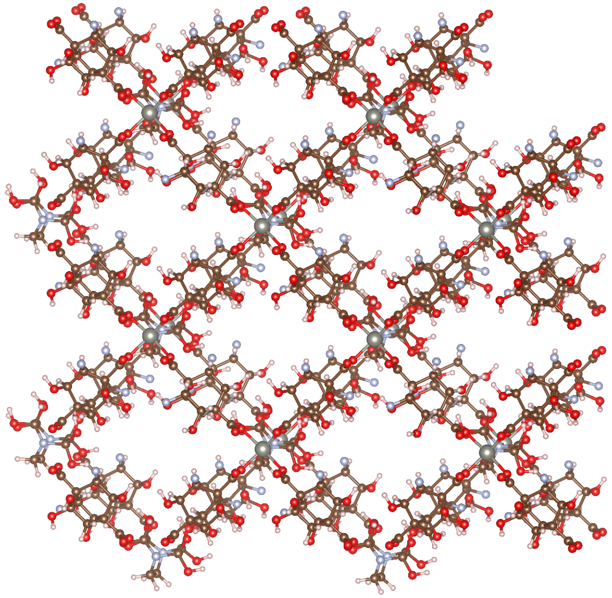
ICLR

What is MOF?

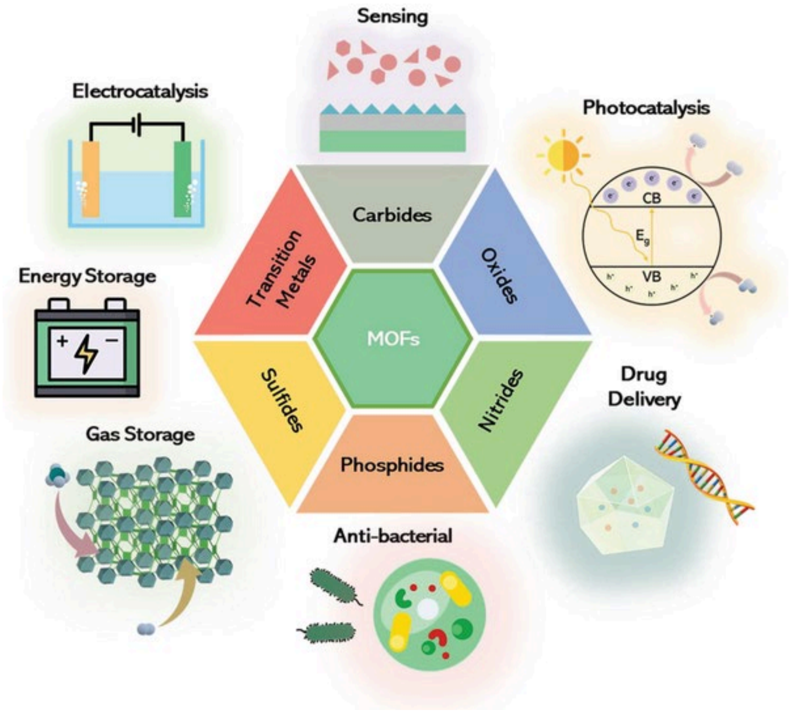
Metal-organic frameworks (MOFs) are **porous**, **crystalline** material with **many applications** – e.g., gas separation, drug delivery, batteries, catalysis, sensing.



An example MOF



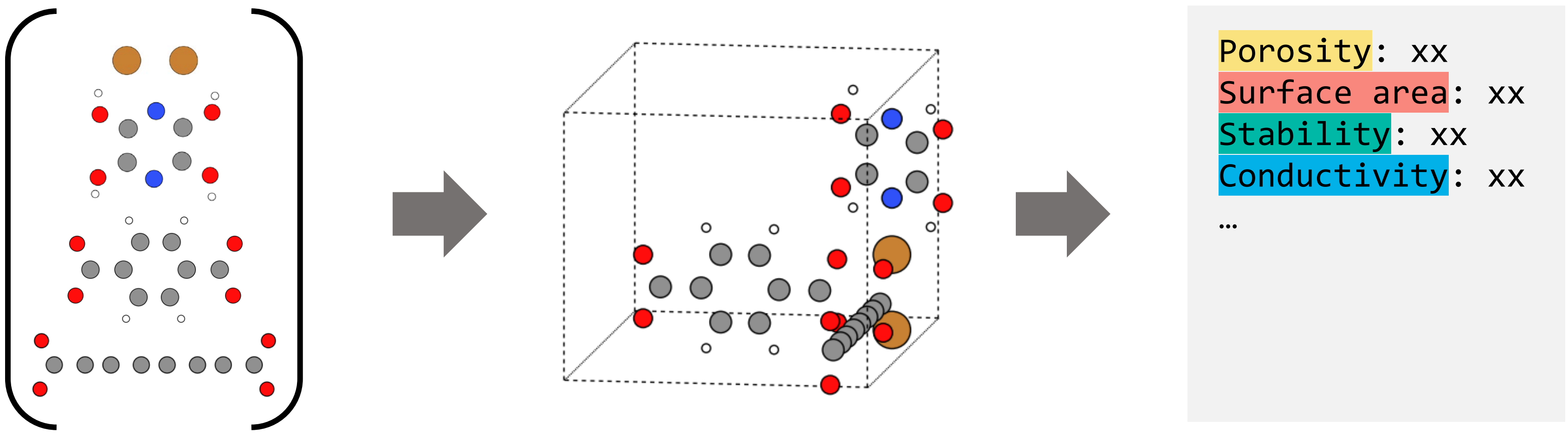
Supercell structure



Various applications of MOF

Structure prediction of MOFs

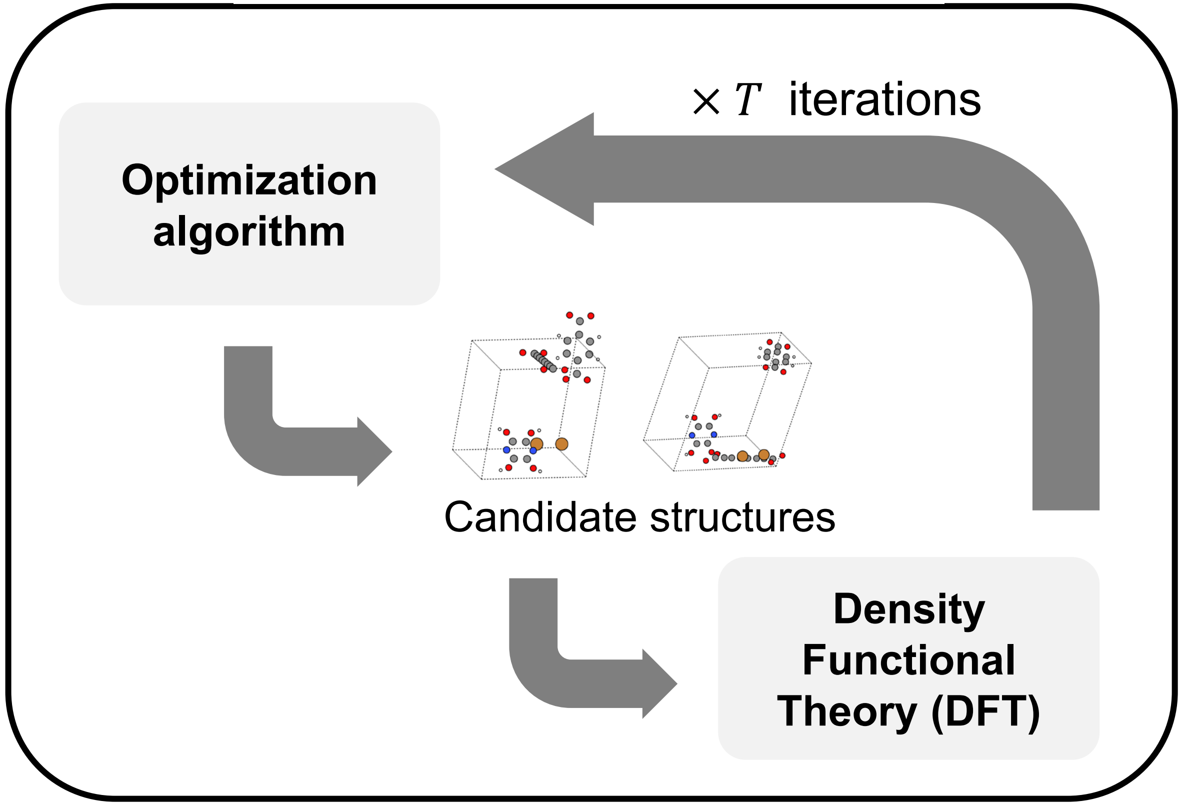
Computational methods for predicting structure of MOFs is useful for directing experimental synt hesis and MOF design.



MOF building blocks ***MOF structure*** ***MOF properties***

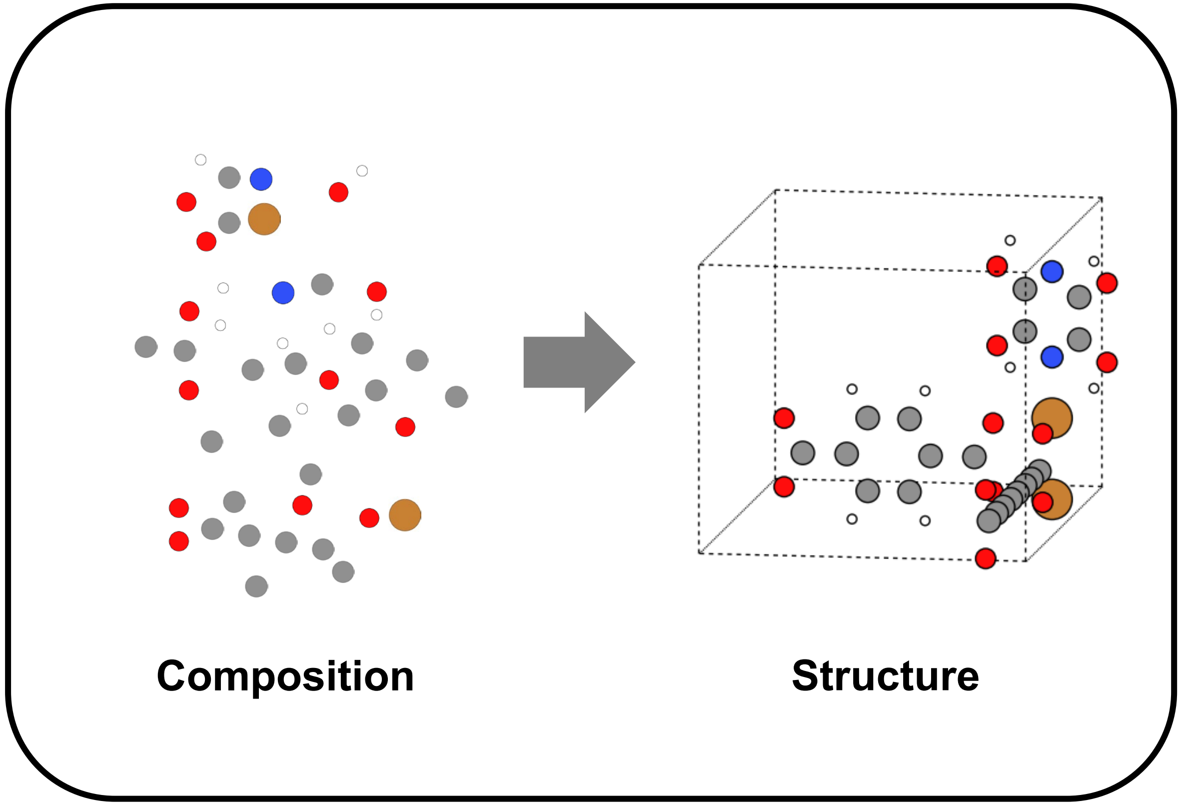
Previous approaches

(1) Traditional approaches



Computationally expensive

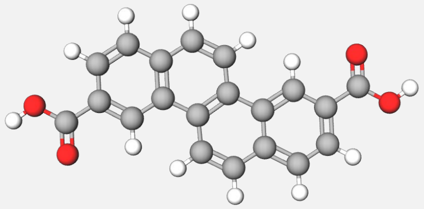
(2) General CSP approaches



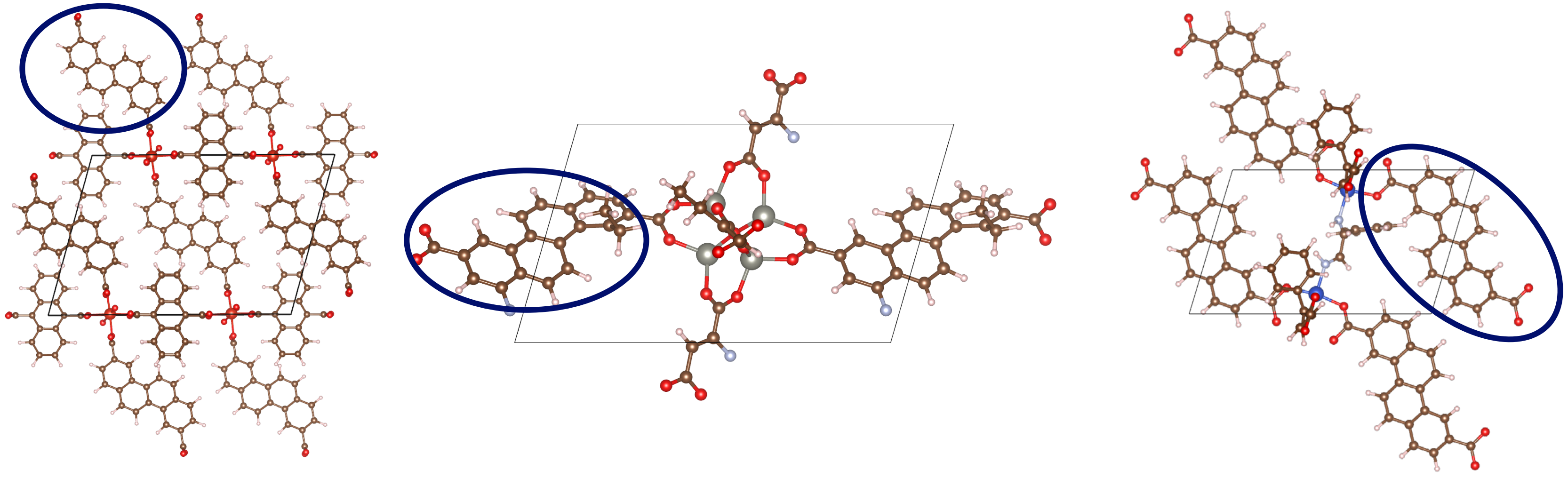
Poor performance and scalability

Motivation

Key observation: Different MOFs share same building blocks with fixed local structures.

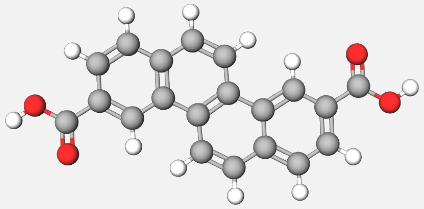


A building block

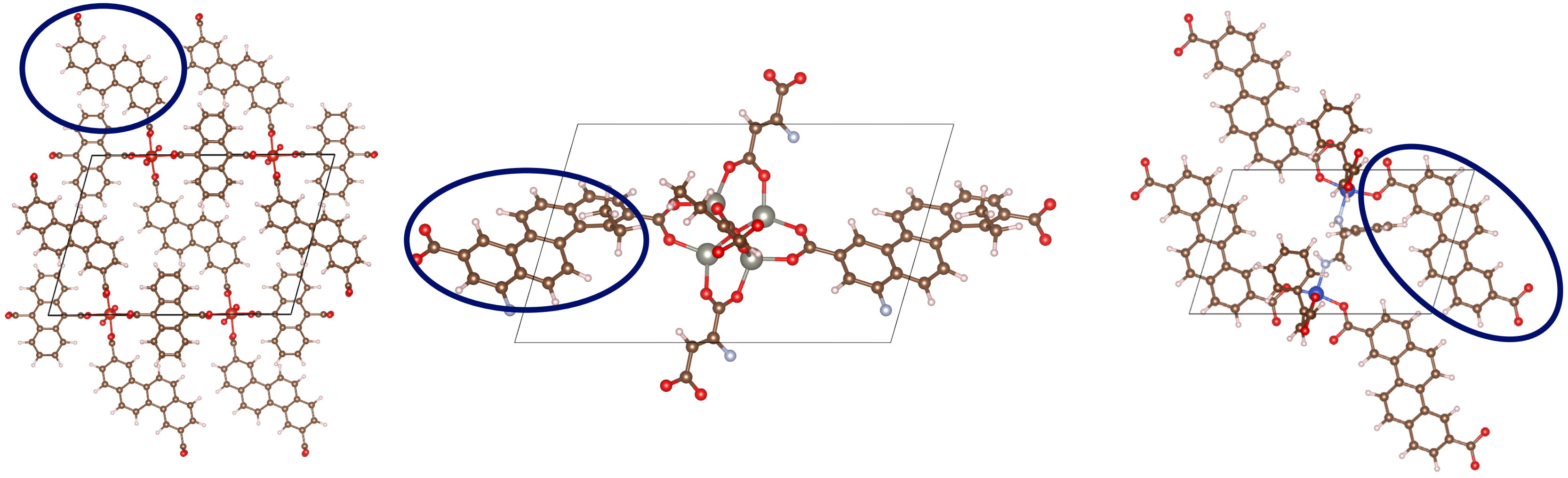


Motivation

Key observation: Different MOFs share same building blocks with fixed local structures.



A building block

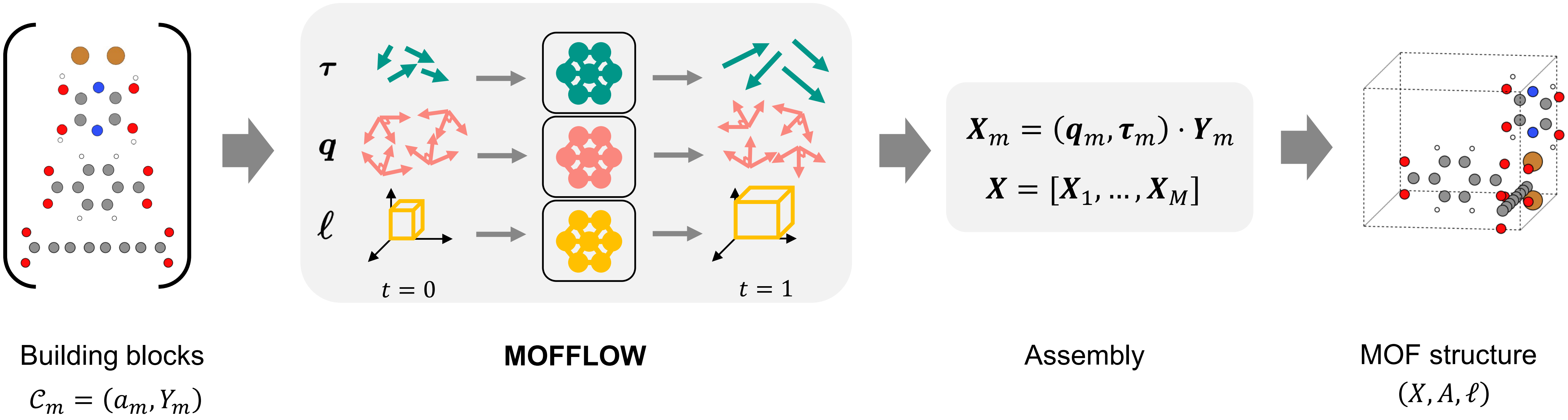


Can we can treat building blocks as *rigid bodies*… and learn to assemble them?

Overview

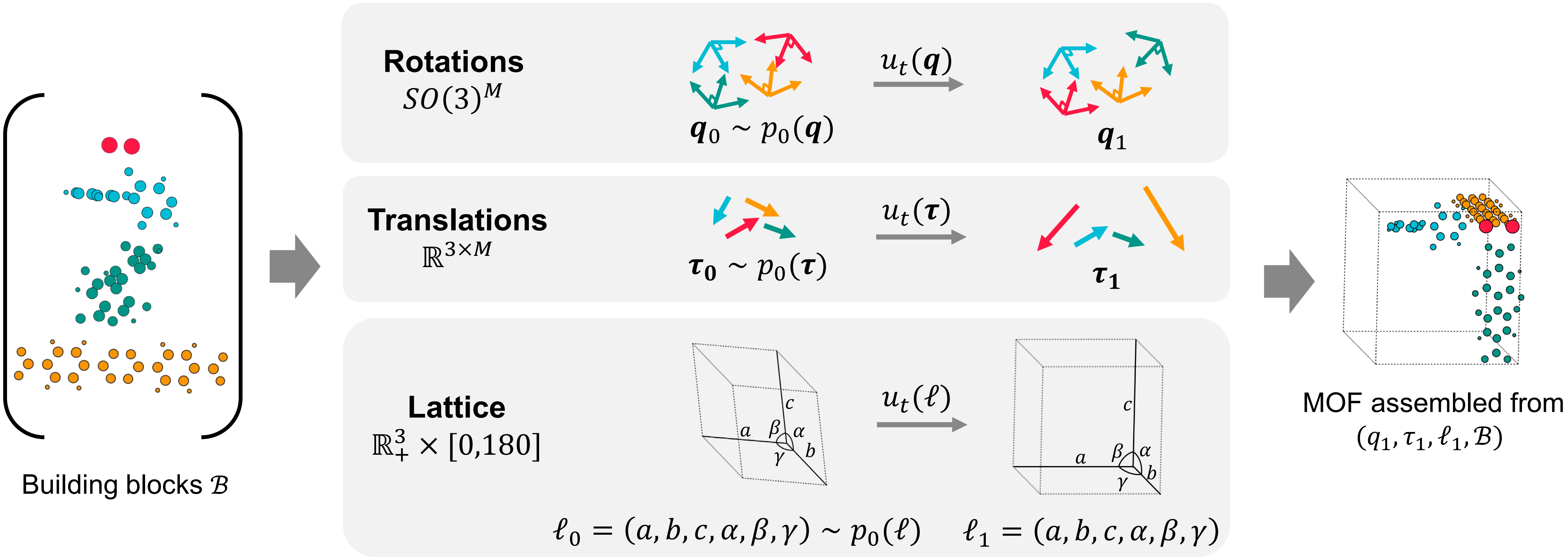
MOFFlow: First deep generative model for structure prediction of MOFs.

Key idea: Learn rotations $\boldsymbol{q}$, translations $\boldsymbol{\tau}$, and lattice ℓ for assembling the building blocks into MOF structure… with Riemannian flow matching!



Training

Goal: Learn a vector field u_t that pushes a prior distribution $p_0(\boldsymbol{q}, \boldsymbol{\tau}, \ell)$ to data distribution $p_1(\boldsymbol{q}, \boldsymbol{\tau}, \ell | \mathcal{B})$

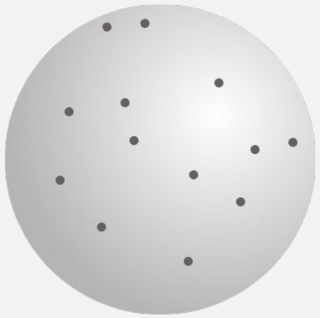


Training

Task 1: Define prior distributions $p_0(\boldsymbol{q}), p_0(\boldsymbol{\tau}), p_0(\ell)$

The virtue of FM framework allows us to choose *any* prior distribution[2,3]!

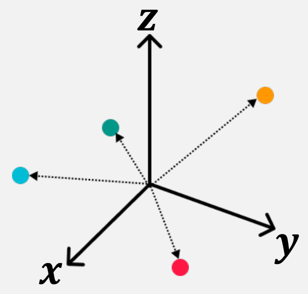
Rotations $SO(3)^M$



$$p_0(q) = \mathcal{U}(SO(3))$$

Uniform distribution on $SO(3)$

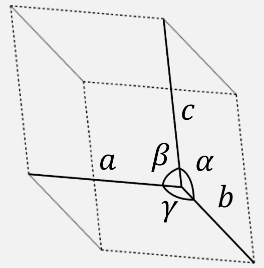
Translations $\mathbb{R}^{3\times M}$



$$p_0(\tau) = \mathcal{N}_c(0, I_3)$$

Centered Gaussian distribution (for translation invariance)

Lattice $\mathbb{R}^3_+ \times [0, \pi]$



$$p_0(a, b, c) = \prod_{\lambda \in (a,b,c)} \mathrm{LogNormal}\left(\lambda \mid \mu_\lambda, \sigma_\lambda\right)$$

$$p_0(\alpha, \beta, \gamma) = \mathcal{U}(60, 120)$$

Log normal and uniform distribution (after Niggli reduction)

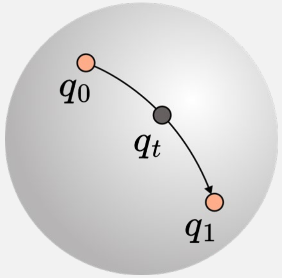
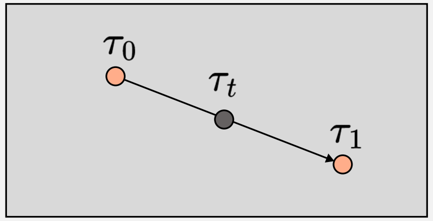
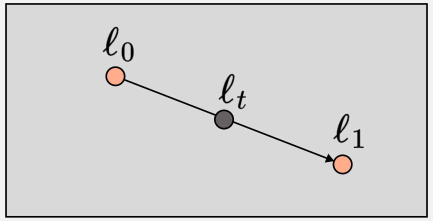
[2] “Multisample Flow Matching” Pooladian et al. (2023). [3] “Minibatch optimal transport” Tong et al. (2024)

Training

Task 2: Define (conditional) vector field to regress towards!

We can define **conditional flows** as a geodesic (shortest path) connecting $x_0 \sim q_0$ and $x_1 \sim q_1$.

We simply take its time-derivative to get **conditional vector field**[4]**.**

	Rotations $SO(3)^M$	**Translations** $\mathbb{R}^{3\times M}$	**Lattice** $\mathbb{R}^3_+ \times [0,\pi]$
Riemannian manifold			
Conditional flow	$q_t = \exp_{q_0}\left(t \log_{q_0} q_1\right)$	$\tau_t = (1-t)\tau_0 + t\tau_1$	$\ell_t = (1-t)\ell_0 + t\ell_1$
Conditional VF	$u_t(q_t \mid q_1) = \dfrac{\log_{q_t} q_1}{1-t}$	$u_t(\tau_t \mid \tau_1) = \dfrac{\tau_1 - \tau_t}{1-t}$	$u_t(\ell_t \mid \ell_1) = \dfrac{\ell_1 - \ell_t}{1-t}$

[4] "Flow Matching on General Geometries" Chen & Lipman (2024)

Training

Parameterization: Instead of directly regressing on conditional vector fields, we parameterize the net work to **predict clean data $\boldsymbol{q_1}, \boldsymbol{\tau_1}, \boldsymbol{\ell_1}$**:

$$u_t(q_t|q_1) = \frac{\log_{q_t}(\boxed{q_1})}{1-t} \qquad u_t(\tau_t \mid \tau_1) = \frac{\boxed{\tau_1} - \tau_t}{1-t} \qquad u_t(\ell_t \mid \ell_1) = \frac{\boxed{\ell_1} - \ell_t}{1-t}$$

$$\underbrace{(\hat{\mathbf{q}}_1, \hat{\tau}_1, \hat{\ell}_1)}_{\text{Prediction of clean data}} = \underbrace{\mathcal{F}(\mathbf{q}_t, \tau_t, \ell_t, \mathcal{B}; \theta)}_{\text{Our neural network!}}$$

Training

Objective: Train neural network $(\hat{q}_1, \hat{\tau}_1, \hat{\ell}_1) = \mathcal{F}_\theta(q_t, \tau_t, \ell_t; \mathcal{B})$ with

$$\mathcal{L}_{CFM}(\theta) = \mathbb{E}_{\mathcal{S}_1,t}\left[\frac{1}{(1-t)^2}\left(\lambda_1 \underbrace{\left\|\log_{q_t}\hat{q}_1 - \log_{q_t} q_1\right\|^2_{SO(3)}}_{\textit{Rotation}} + \lambda_2 \underbrace{\left\|\hat{\tau}_1 - \tau_1\right\|^2_{\mathbb{R}^3}}_{\textit{Translation}} + \lambda_3 \underbrace{\left\|\hat{\ell}_1 - \ell_1\right\|^2_{\mathbb{R}^3}}_{\textit{Lattice}}\right)\right]$$

where $\mathcal{S}_1 = (q_1, \tau_1, \ell_1, \mathcal{B})$ and $t \sim U(0,1)$.

Experiments

Key questions

1. **Accuracy:** How does the structure prediction accuracy of MOFFlow compare to other methods?
2. **Scalability:** How does the performance of MOFFlow vary with increasing number of atoms and building blocks?

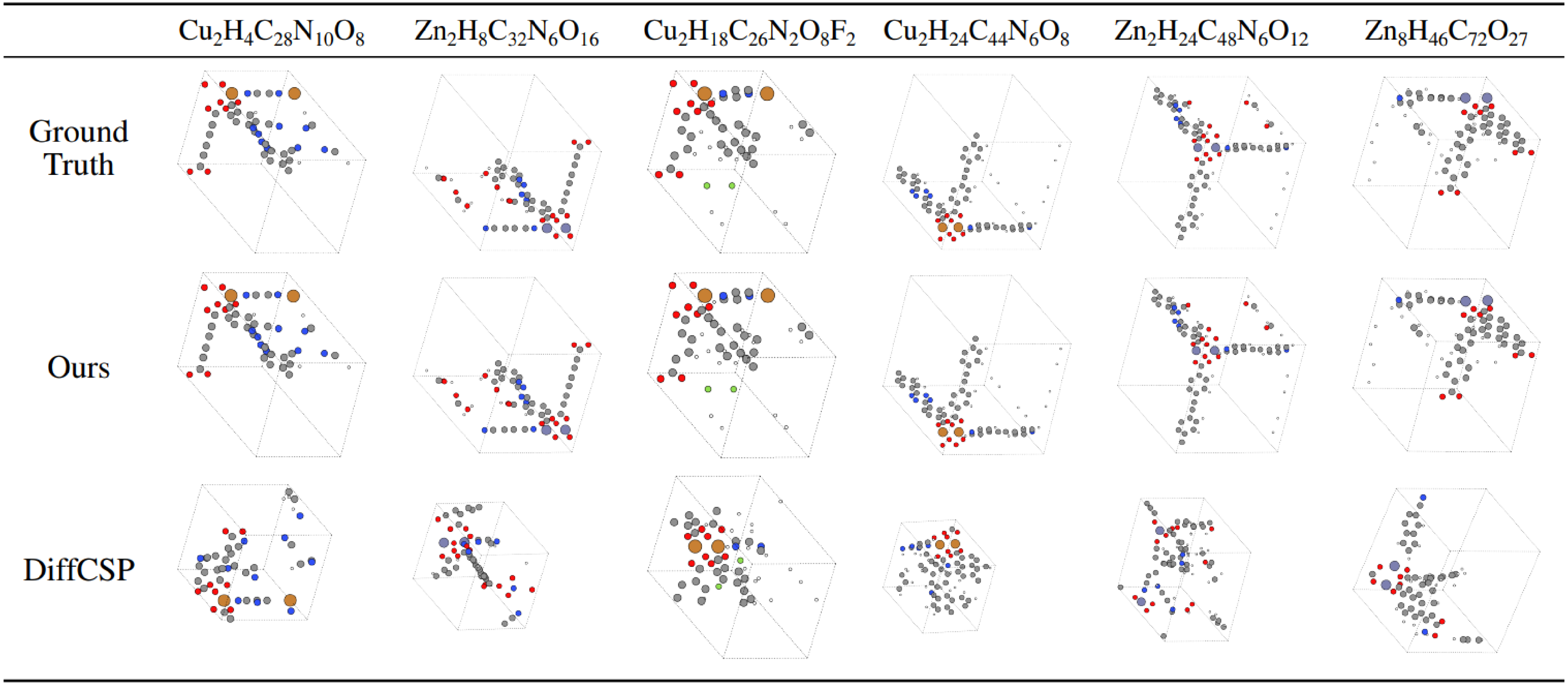
Experiments

✓ **Accuracy:** MOFFlow can accurately predict MOF structures

	# of samples	stol = 0.5		stol = 1.0		Avg. time (s)↓
		MR (%) ↑	RMSE ↓	MR (%) ↑	RMSE ↓	
RS (Yamashita et al., 2021)	20	0.00	-	0.00	-	332
EA (Yamashita et al., 2021)	20	0.00	-	0.00	-	1959
DiffCSP (Jiao et al., 2024a)	1	0.09	0.3961	23.12	0.8294	5.37
	5	0.34	0.3848	38.94	0.7937	26.85
MOFFlow (Ours)	1	31.69	0.2820	87.46	0.5183	**1.94**
	5	**44.75**	**0.2694**	**100.0**	**0.4645**	5.69

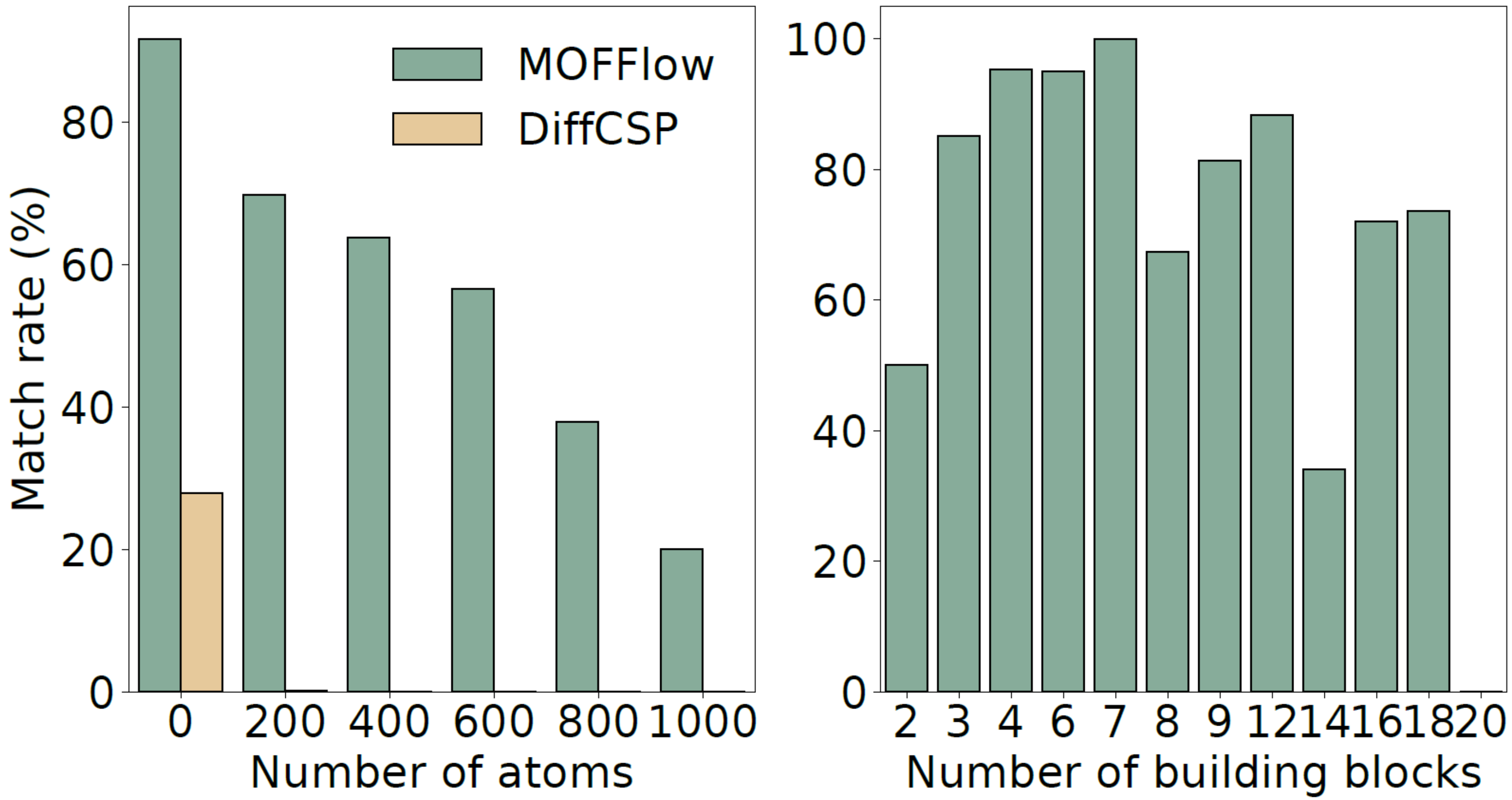
Experiments

✓ **Accuracy:** MOFFlow can accurately predict MOF structures



Experiments

✓ **Scalability:** MOFFlow is scalable to large systems (up to thousands of atoms)



Thank you

Paper

Code