Trinity College Dublin
Coláiste na Tríonóide, Baile Átha Cliath
The University of Dublin

Learning Shortcuts in the Chemical Space

James O’Reilly

Supervised by Alessandro Lunghi

SS TP Capstone Presentation 2024

Objective?

To leverage **reinforcement learning** methods to screen a large number of molecules at low computational cost for promising high temperature **single-molecule magnet** candidates.

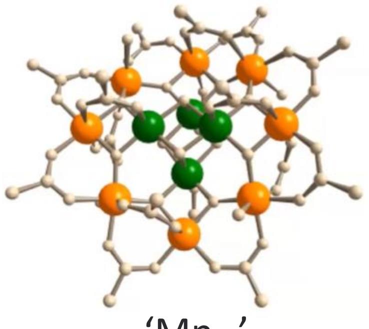
Single-molecule magnets (SMMs)

- Molecules that show a magnetic memory effect, i.e. they retain their magnetisation at low temperatures.

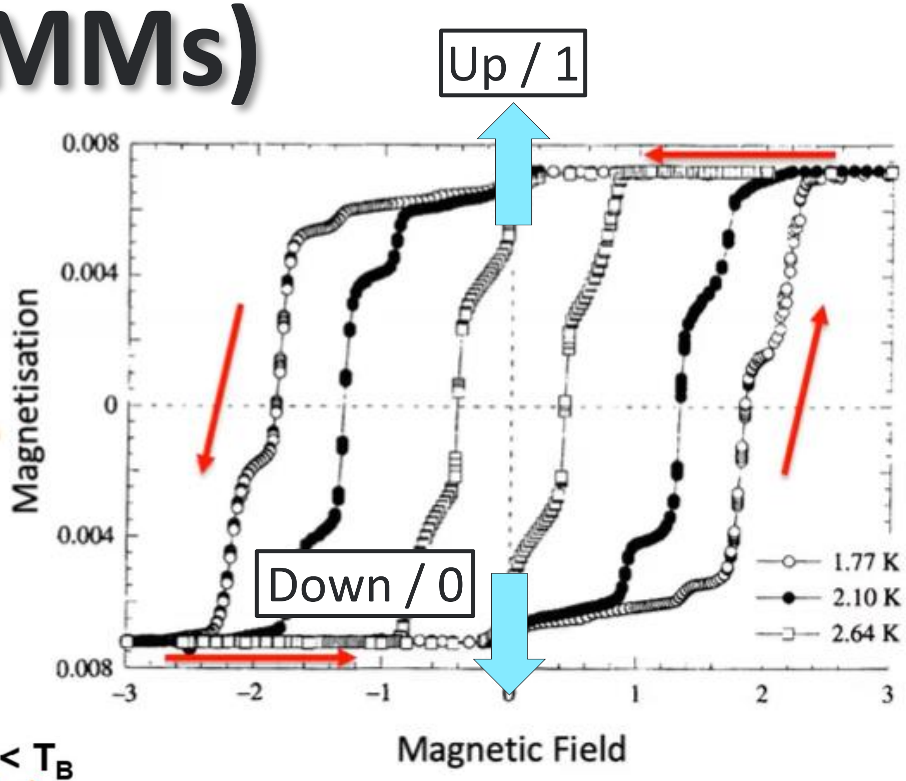
- Why? **Magnetic anisotropy**

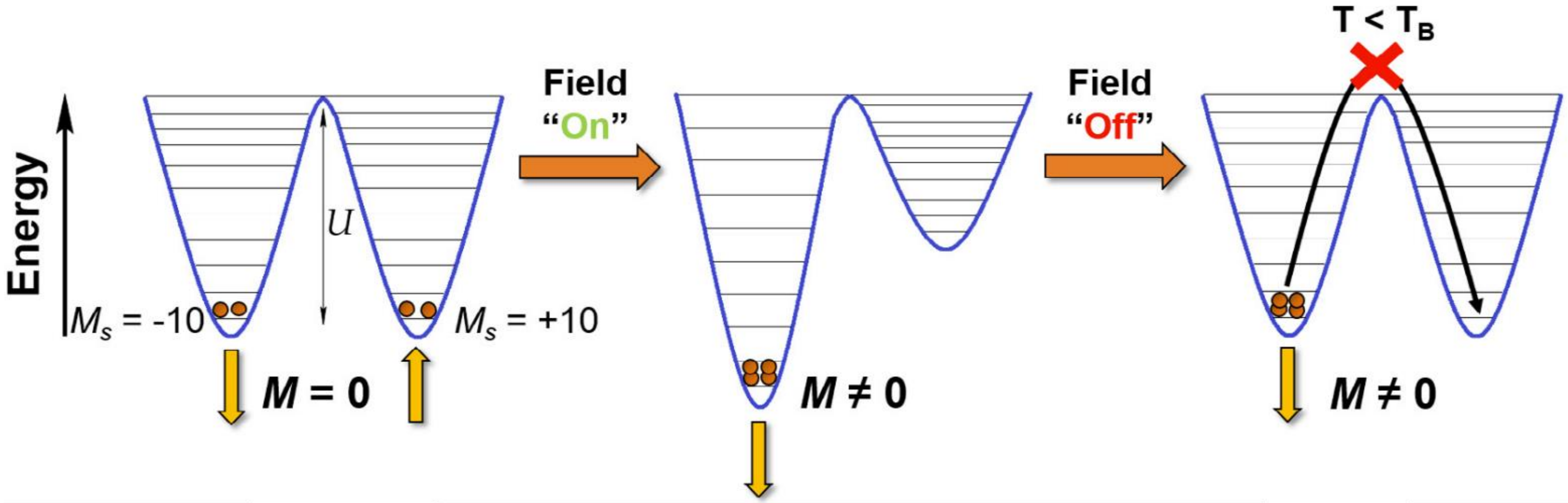
$$E = M_S^2 \cdot D$$

$\Longrightarrow$ Maximal $|M_S|$ in ground state for $D < 0$



‘Mn₁₂’





- Relaxation rate:

$$\tau^{-1} = \tau_0^{-1}\exp[\frac{-U}{k_B T}]$$

$$U \propto |D|$$

We want highly negative D!

Single-molecule magnets (SMMs)

Applications of room temperature single-molecule magnets:

- Data storage:

  - Current HDDs ~ 200GB in^{-2}
  - SMMs ~ 20,000GB in^{-2}

  Data centres 100 times smaller/more efficient

- Quantum computing

- Spintronics

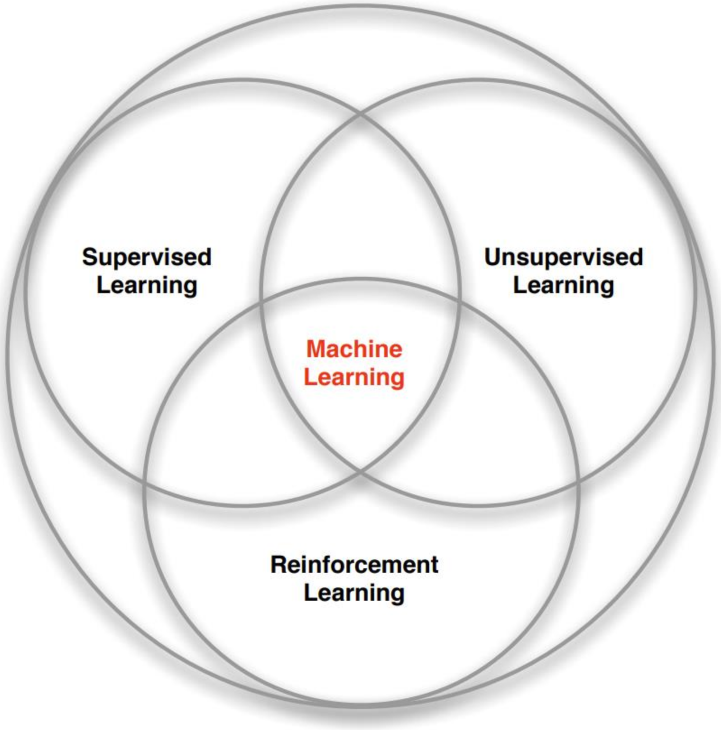
Reinforcement Learning (RL)

- Learn through trial and error by interacting with environment and receiving rewards or penalties based on actions.

- Feedback is delayed.

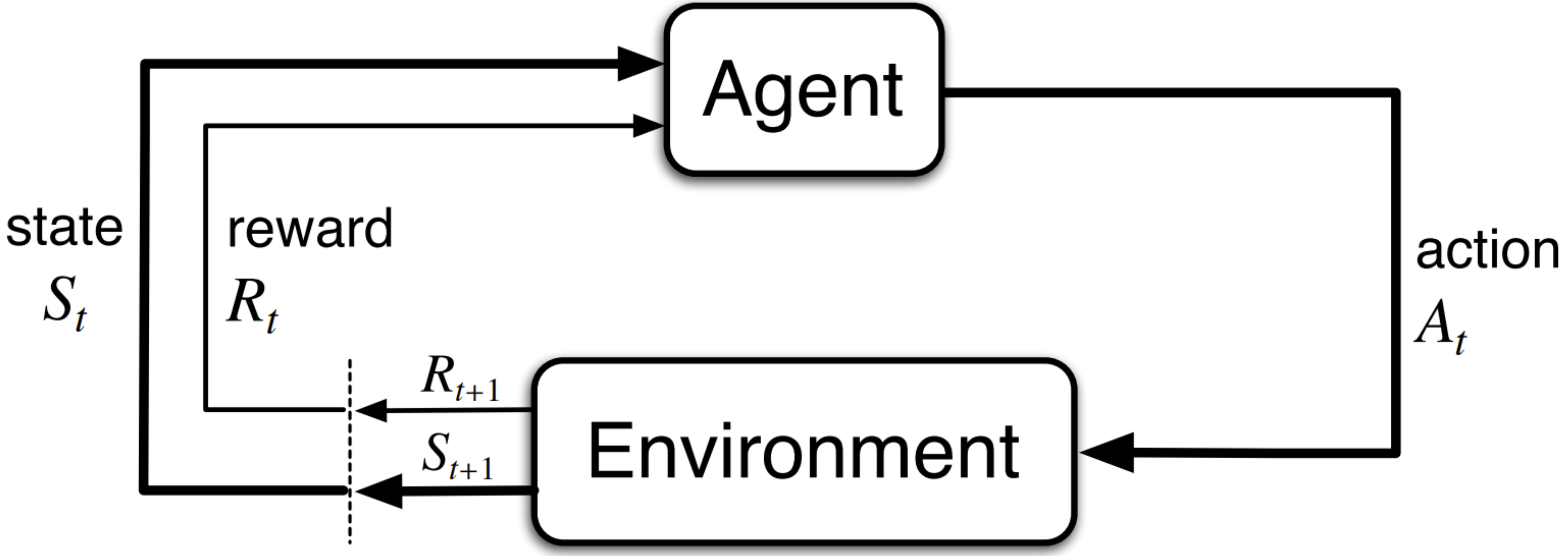
- Importance of balancing exploration vs exploitation.

- Extremely versatile and has had many successes to date:
    - Robotics
    - Autonomous vehicles
    - Finance/trading
    - AlphaGo



Markov Decision Processes (MDPs)

- MDPs provide a mathematical framework for RL problems.



Components of an RL agent

- A policy π maps states to actions: $\pi(a|s) = \mathbb{P}[A_t = a \mid S_t = s]$

- A value function gives a prediction of future reward while following a certain policy.

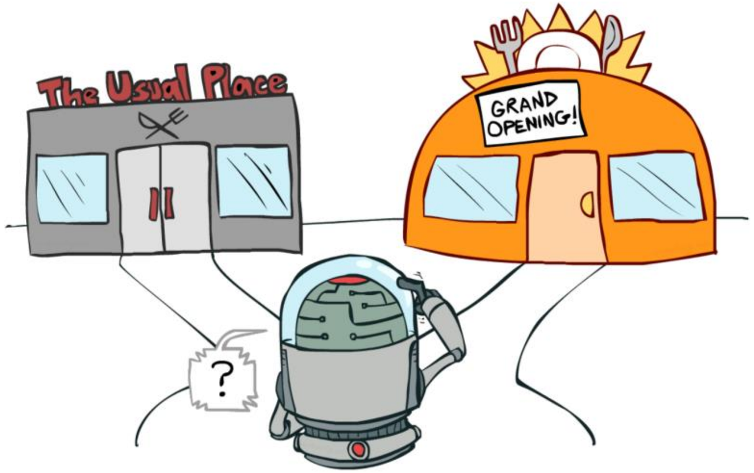
$$q_\pi(s,a) = \mathbb{E}_\pi[R_{t+1} + \gamma R_{t+2} + \gamma^2 R_{t+3} + \cdots \mid S_t = s, A_t = a] \;,\; \gamma \in [0,1]$$

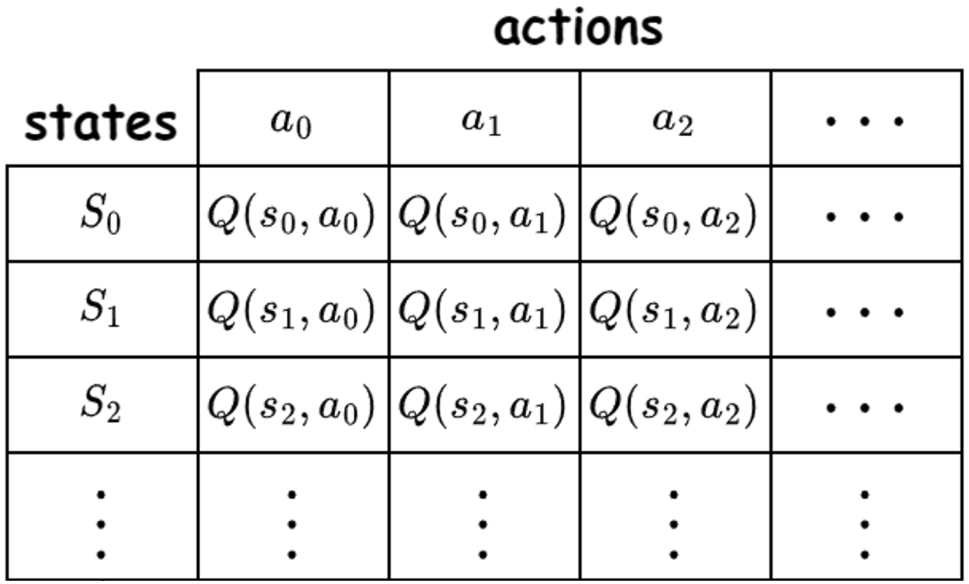
- How do we find the *optimal* policy π_* of our agent?

$$\pi_*(a \mid s) = \begin{cases} 1 \quad if \quad a = \underset{a \in \mathcal{A}}{\operatorname{argmax}}\, q_*(s,a) \\ 0 \quad otherwise \end{cases}$$

Q-Learning

- Tabular Method.
- Epsilon greedy behaviour policy.





actions

states	a_0	a_1	a_2	$\cdots$
s_0	$Q(s_0,a_0)$	$Q(s_0,a_1)$	$Q(s_0,a_2)$	$\cdots$
s_1	$Q(s_1,a_0)$	$Q(s_1,a_1)$	$Q(s_1,a_2)$	$\cdots$
s_2	$Q(s_2,a_0)$	$Q(s_2,a_1)$	$Q(s_2,a_2)$	$\cdots$
$\vdots$	$\vdots$	$\vdots$	$\vdots$	$\vdots$

$$Q(S_t,A_t) \leftarrow Q(S_t,A_t) + \alpha\left(\overbrace{\underbrace{R_{t+1} + \gamma \max_a Q(S_{t+1},a)}_{\text{TD target}} - Q(S_t,A_t)}^{\text{TD error}}\right)$$

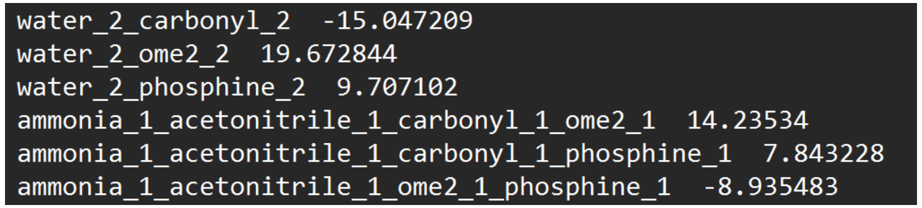
$$Q(s,a) \longrightarrow q_*(s,a)$$

Our Problem

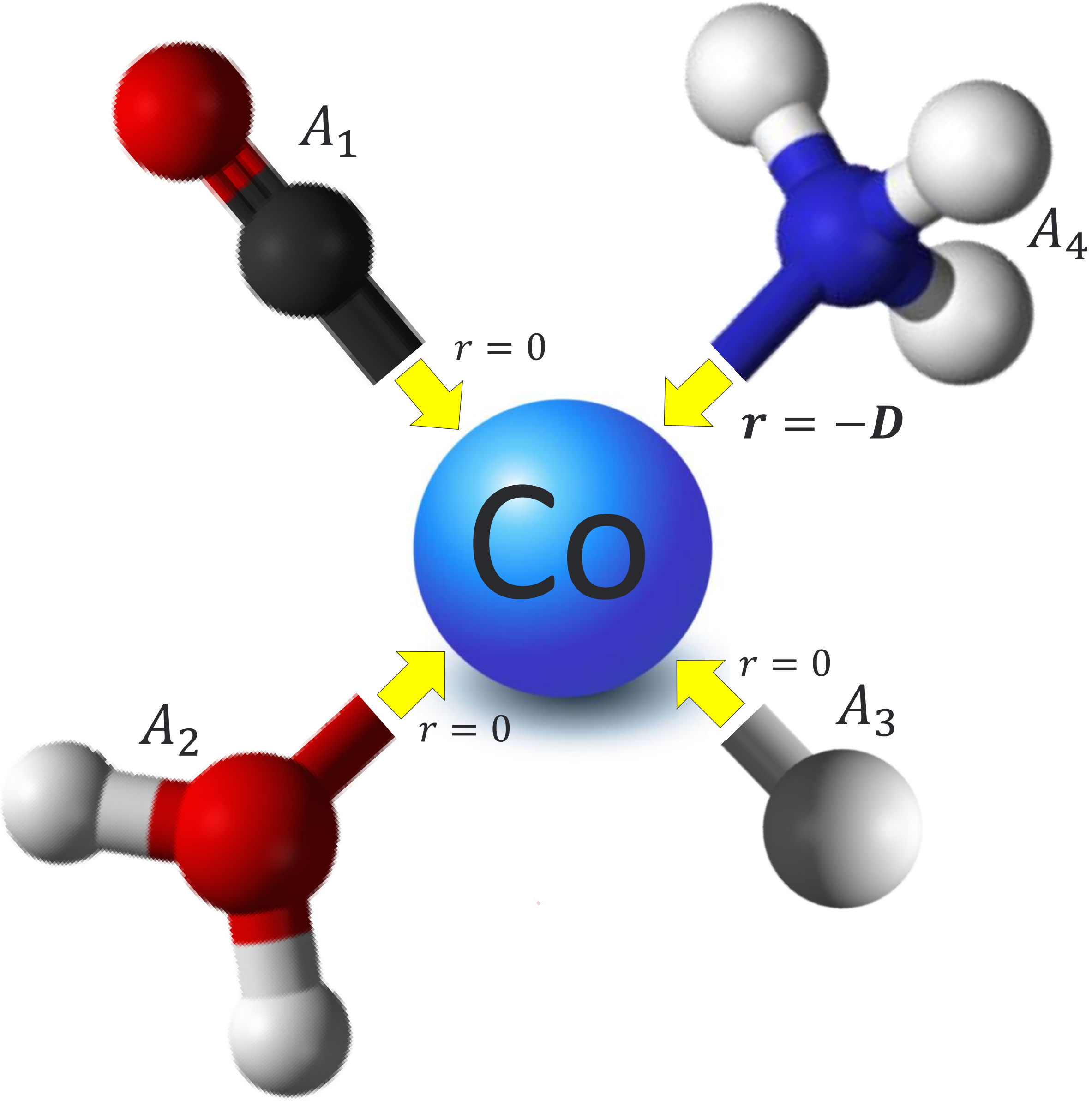
- Construct molecules by adding ligands to a central metal atom.

- Very challenging MDP!
- ➢ Very few no. steps
- ➢ No new info until end of episode

- Data



```
water_2_carbonyl_2  -15.047209
water_2_ome2_2  19.672844
water_2_phosphine_2  9.707102
ammonia_1_acetonitrile_1_carbonyl_1_ome2_1  14.23534
ammonia_1_acetonitrile_1_carbonyl_1_phosphine_1  7.843228
ammonia_1_acetonitrile_1_ome2_1_phosphine_1  -8.935483
```



Synthetic Data

$$n_{states}(N,M) = 1 + \sum_{m=1,\,\dots\,,\,M} \binom{m+N-1}{N-1}$$

$\boldsymbol{N}$ = no. ligands to choose from

$\boldsymbol{M}$ = no. ligands in molecule

M \ N	10	20	30
4	1,001	10,626	46,376
6	8,008	230,230	1,947,792
8	43,758	3,108,105	48,903,492

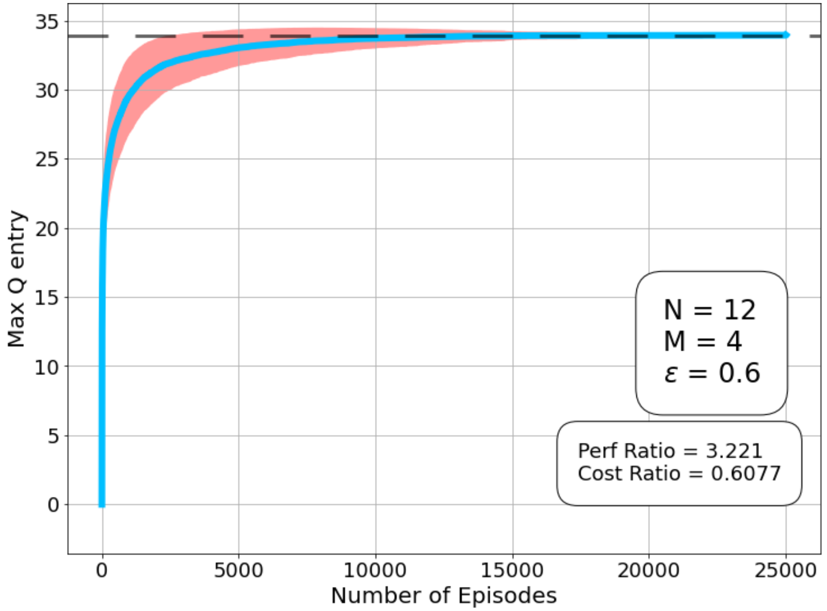
$$L_i \in [-10, 10]$$

$$R_{final} = \sum_{i=1,\dots,M} L_i \quad \Longrightarrow \quad \underbrace{[A, A, \dots, A]}_{M\ times}\ \textbf{best}$$

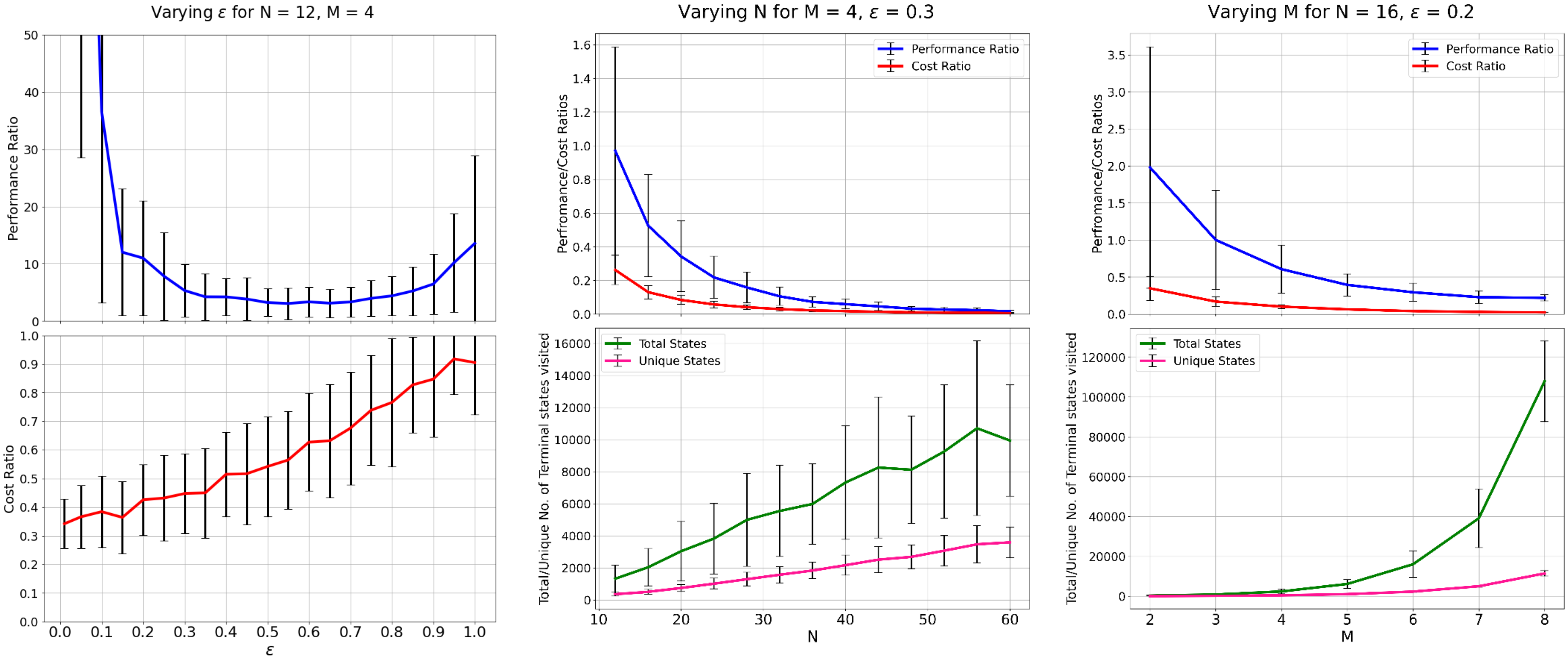
$$\text{Performance Ratio} = \frac{\text{Total no. Terminal States Reached}}{\text{Terminal State Space Size}}$$

$$\text{Cost Ratio} = \frac{\text{Unique no. Terminal States Reached}}{\text{Terminal State Space Size}}$$

Max Q entry vs no. episodes, 1000 runs

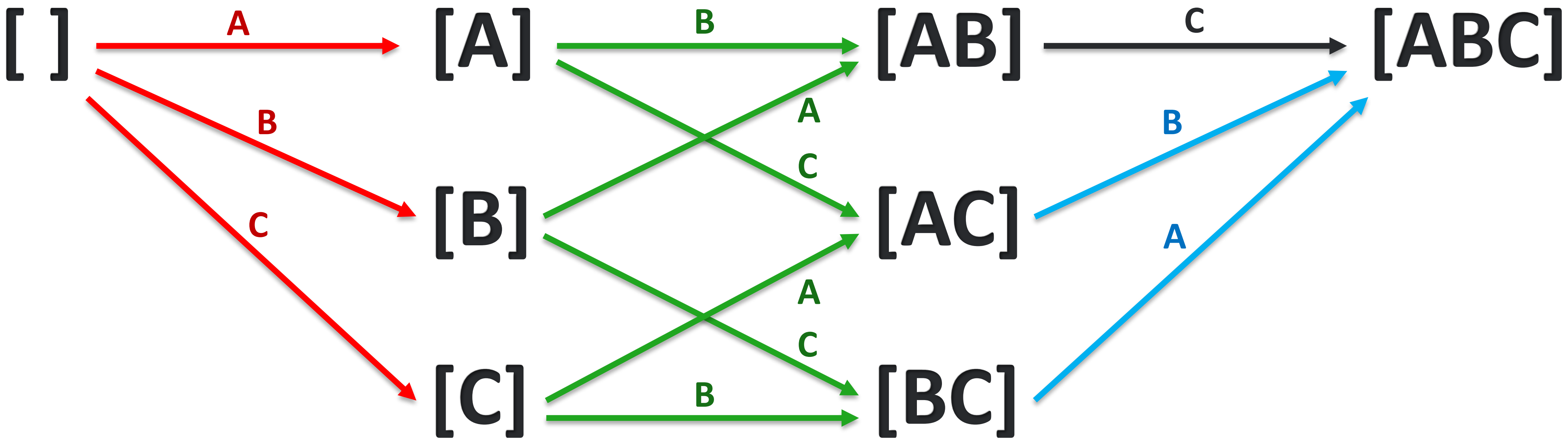


Synthetic Data



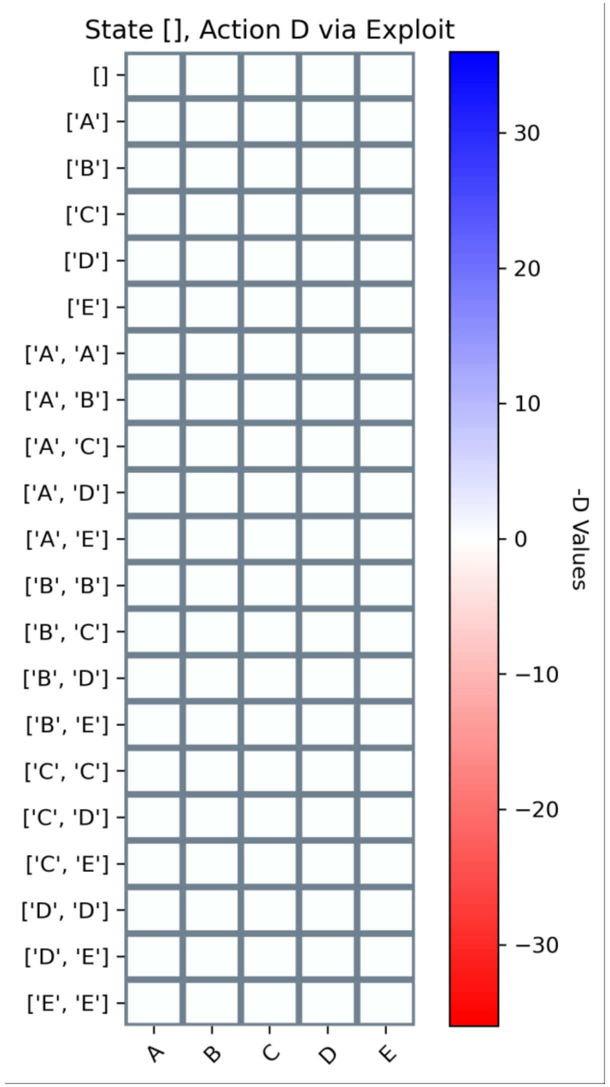
Algorithmic improvements

Order-invariance when adding ligands ⟹ Multiple Q updates per step

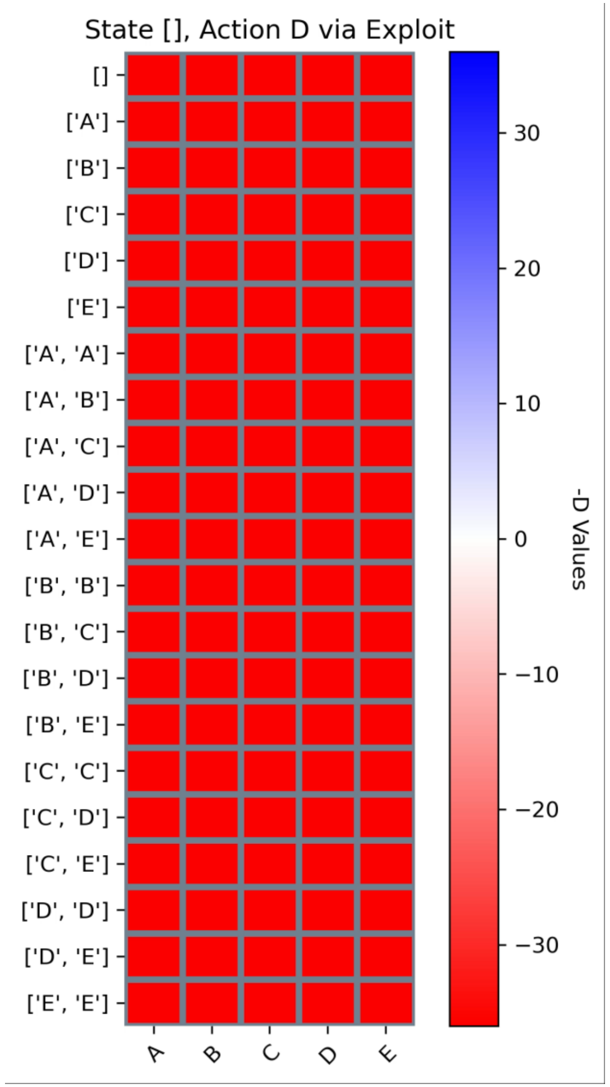


Algorithmic improvements

Vanilla



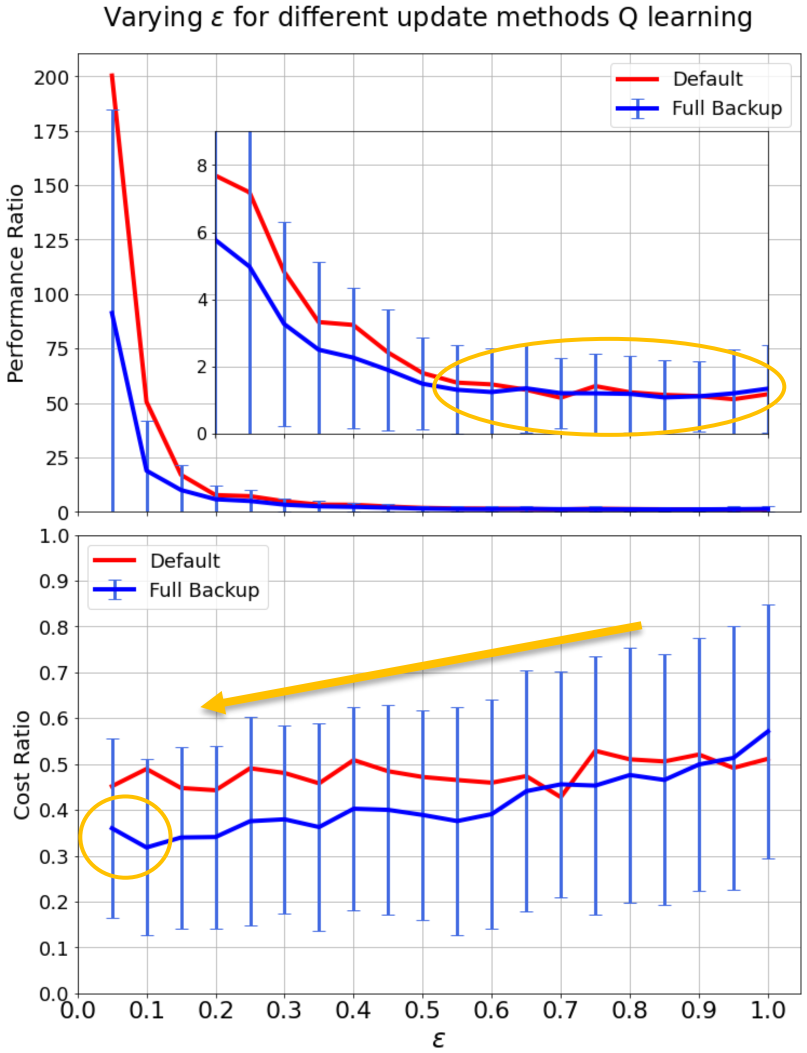
Full backup updates

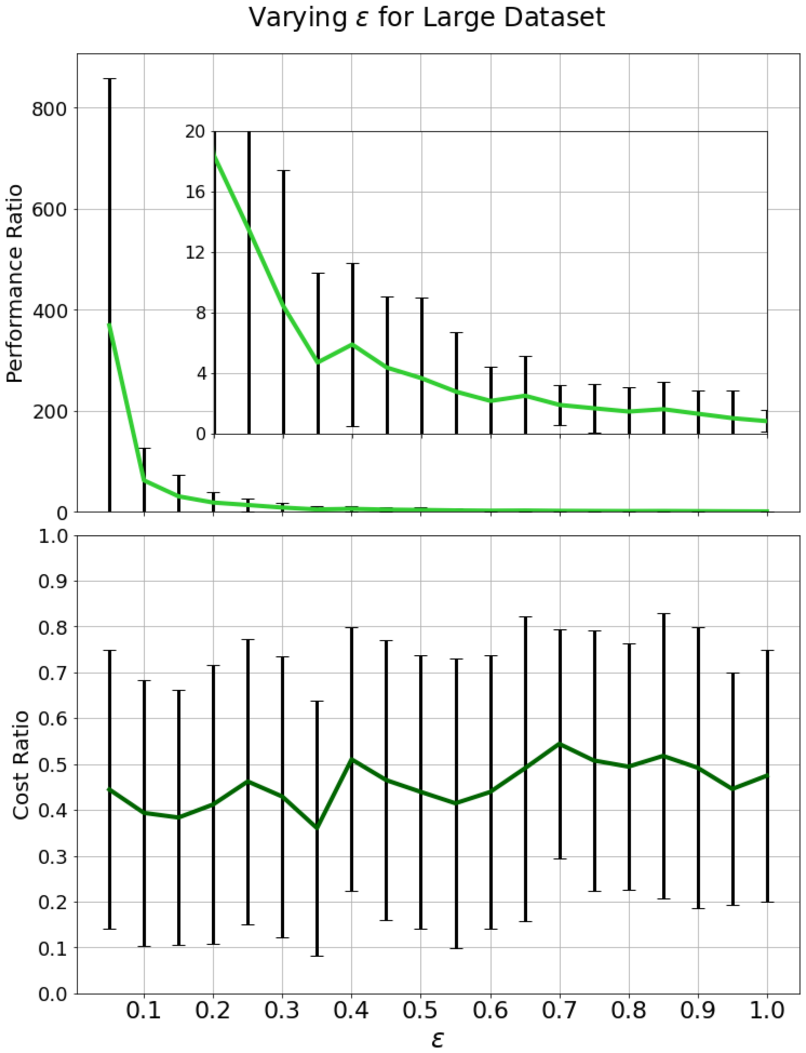


Physical data

Dataset 1:

$$N = 13$$
$$M = 4$$
$$n_{molecules} = 766$$





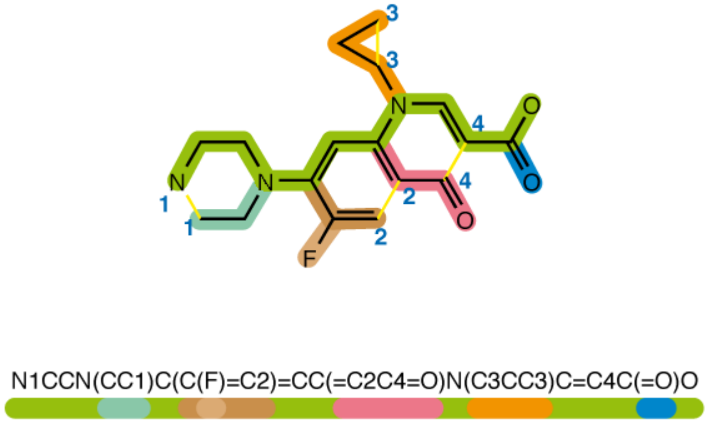
Dataset 2:

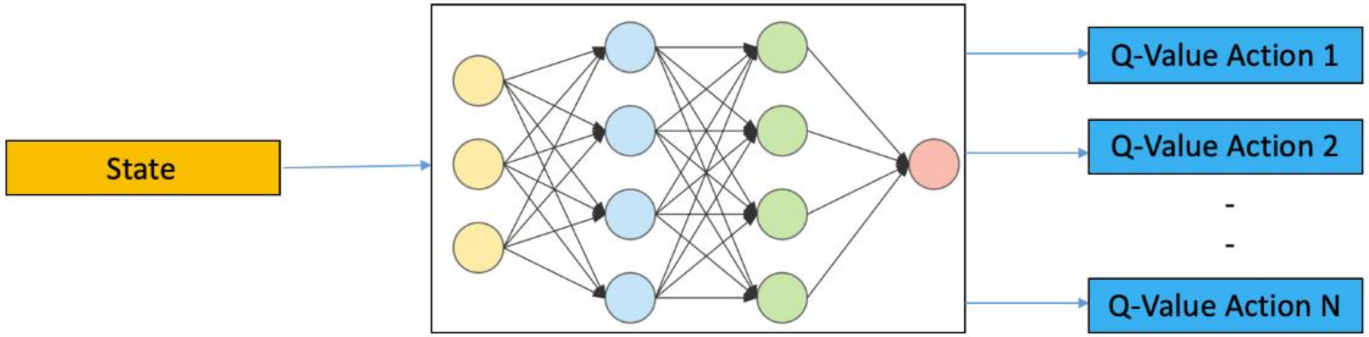
$$N = 206$$
$$M = 2$$
$$n_{molecules} = 21{,}321$$

Next Steps

- Return to synthetic data with more complexity/correlation between ligands in the reward function
  - ➢ Too little of no. steps?
  - ➢ Pattern too complicated?

- Introduce featurisation and use deep Q-Learning:
  - ➢ SMILES
  - ➢ Coordinates
  - ➢ Bispectrum Components





Thank you!

References:

1. R. S. Sutton and A. G. Barto, *Reinforcement Learning: An Introduction*, 2nd ed. (The MIT Press, 2018).

2. D. Silver, Lectures on reinforcement learning, URL: `https://www.davidsilver.uk/teaching/` (2015)

3. A. Zabala-Lekuona, J. M. Seco, and E. Colacio, Coordination Chemistry Reviews 441, 213984 (2021).

4. D. Gatteschi, R. Sessoli, and J. Villain, *Molecular Nanomagnets* (Oxford University Press, 2006).

5. G. Rajaraman, *Computational Modelling of Molecular Nanomagnets* (Springer Cham, 2023)

6. N. Chilton, *Single-molecule magnets: design, measurement and theory,* The University of Manchester (2020), available here.

Trinity College Dublin
Coláiste na Tríonóide, Baile Átha Cliath
The University of Dublin