Reinforcement Learning in Neutron Crystallography

Jessica Opsahl-Ong - McLean High
Kate Meuse - Montgomery Blair High
William Ratcliff - Mentor

Goals and Impact

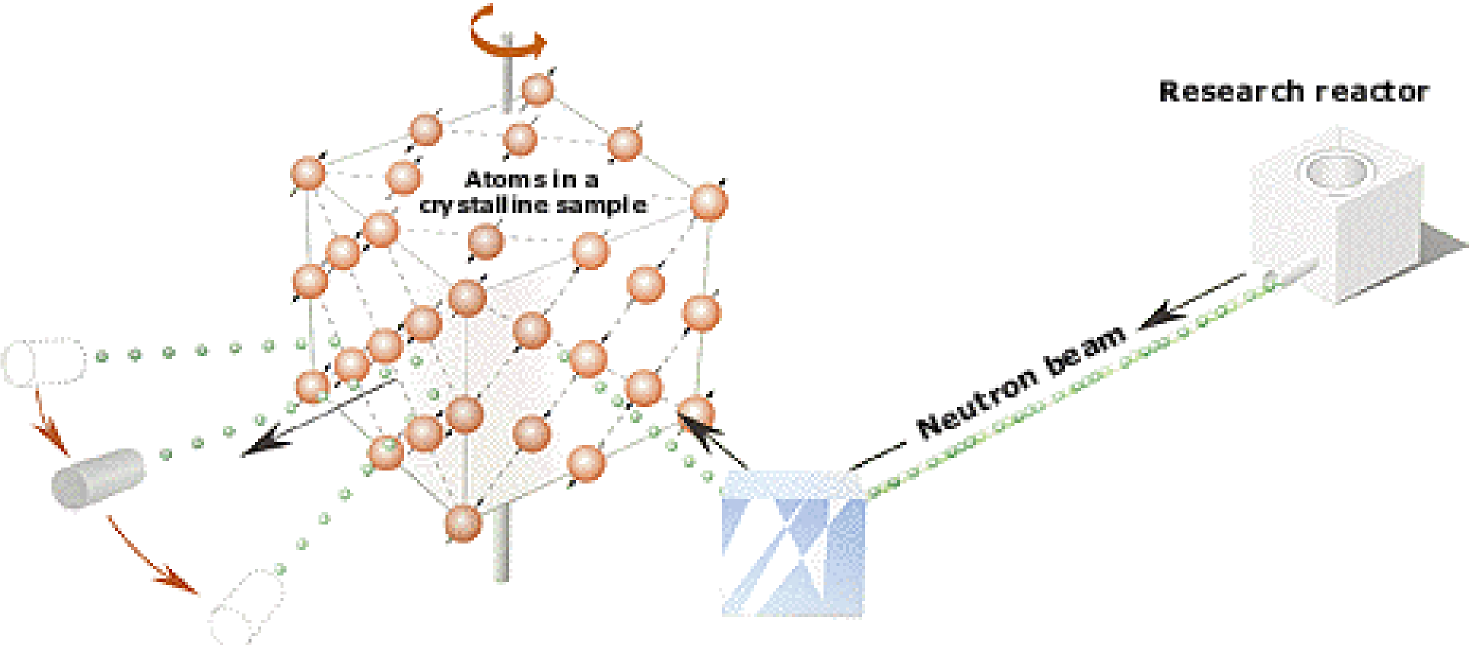
- Beam time is valuable – limited access
- Single crystal spectrometry is slow and often redundant
- Want to take measurements more efficiently
- Software to be implemented on triple axis spectrometer

Crystallography

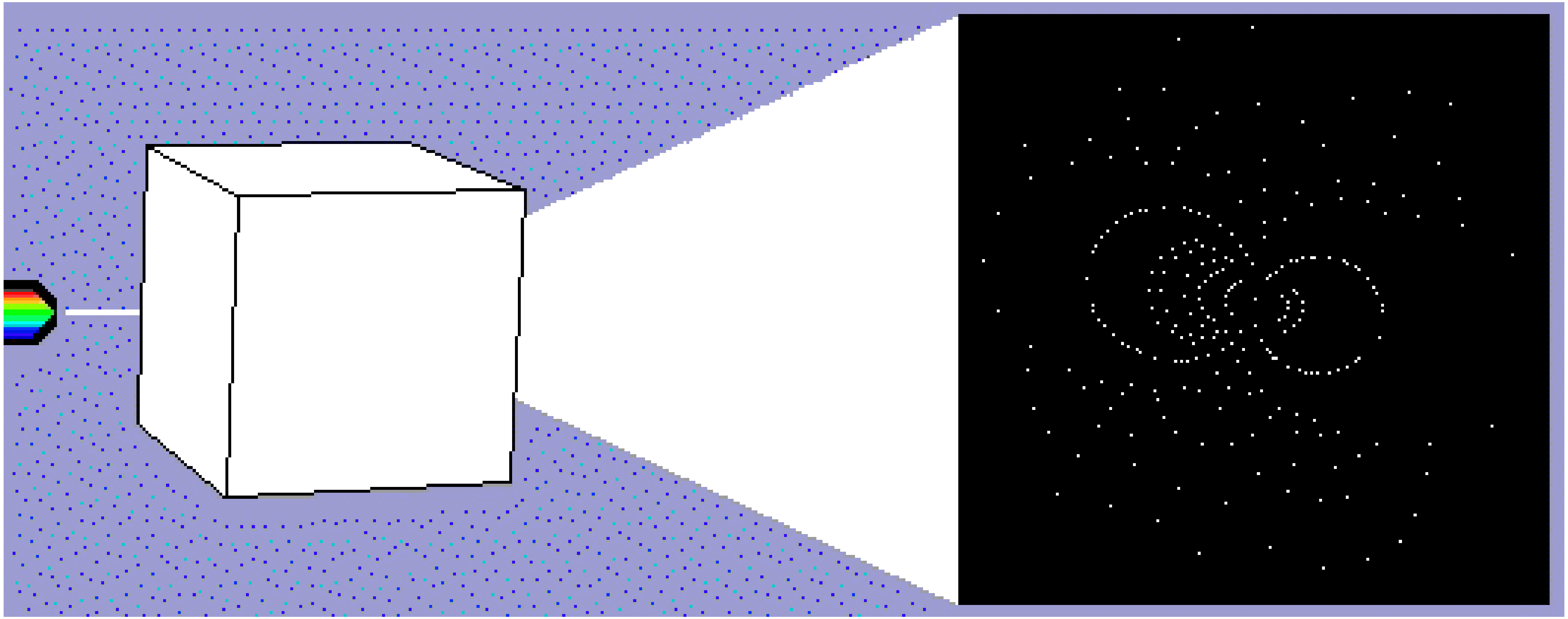
Neutron Crystallography: Using neutron diffraction to study the properties of crystals

Diffraction:

- Neutrons scatter off nuclei
- Neutrons will constructively or destructively interfere
- Left with reciprocal space map
- Measurements give information about the "real space" through a Fourier transform



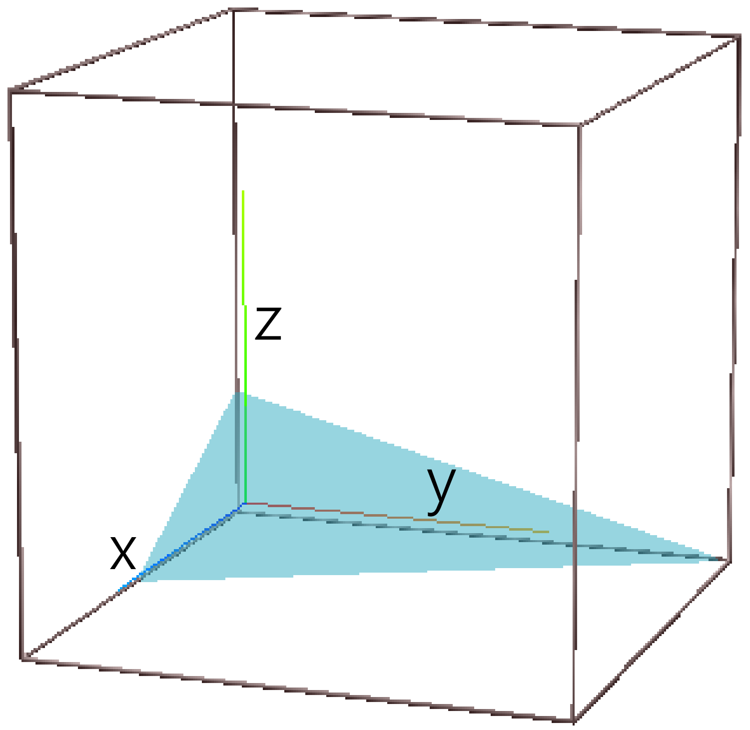
Reciprocal Space



Miller Indices

Taking measurements:

- Miller indices (h, k, l) describe a plane
- Reciprocal of index is location along axis
- Researchers select planes to measure

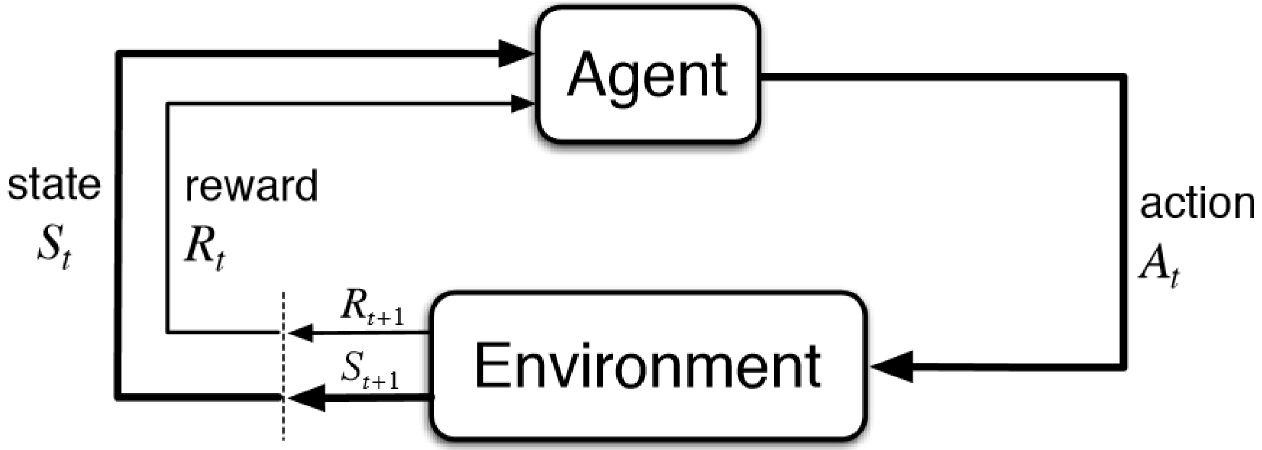


(2 1 4)

Reinforcement Learning

Defined:

- Teaching a computer to make optimal decisions using rewards



How does it work?

- The agent is in an environment
- The environment returns a state
- Agent makes action based on state
- Agent is rewarded after action
- Algorithm learns how to best make actions based on rewards

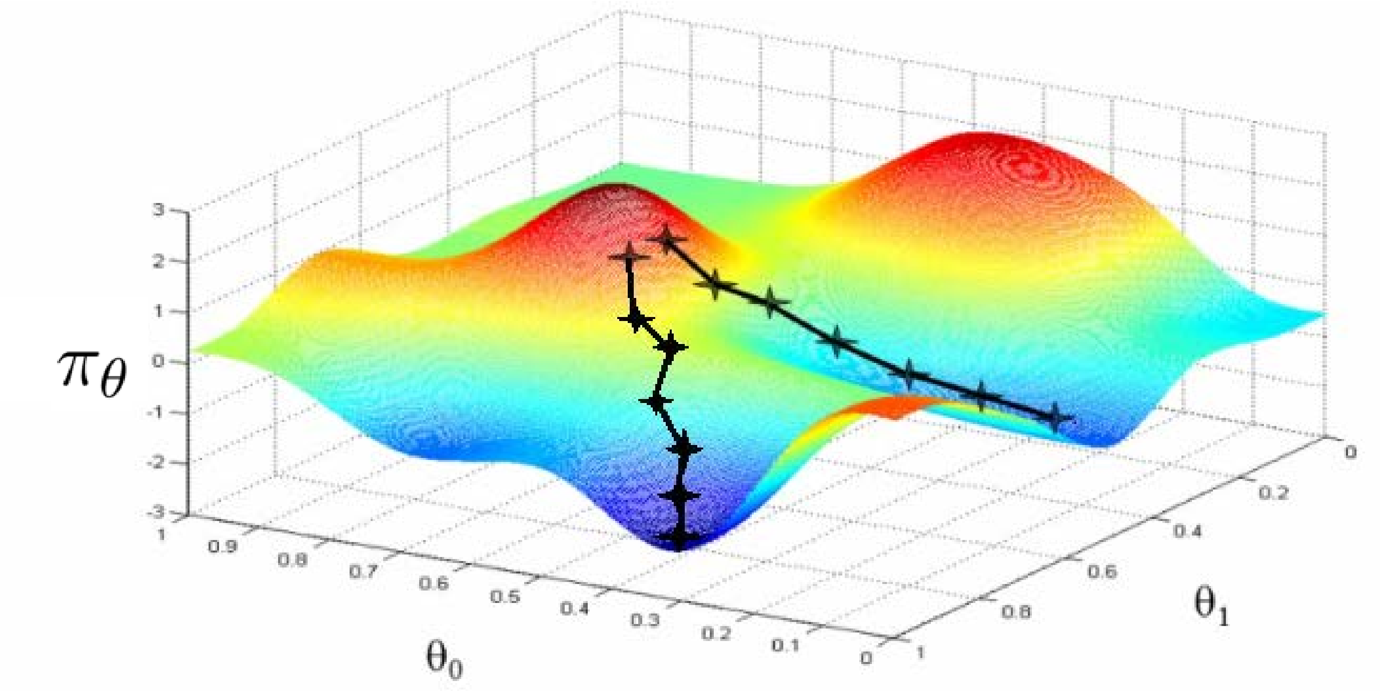
Policy Gradient

What is a policy?

- Algorithm chooses action based on policy function
- Given a state, the policy returns a distribution of how probable each action is

Optimizing policy:

- Done through gradient ascent
- Take derivative of function
- Gradually change parameters until it reaches the maximum



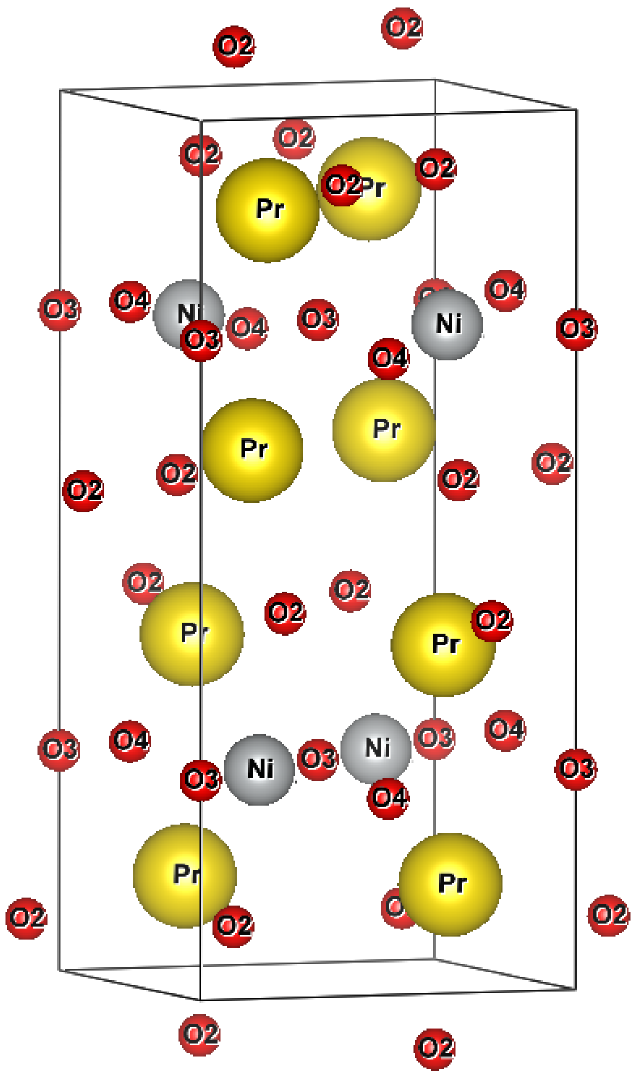
Applying RL to Crystallography

Modeling the problem:

- Testing with a "toy problem"
- Knows everything about the crystal except the z coordinate of one atom

How does this work with RL?

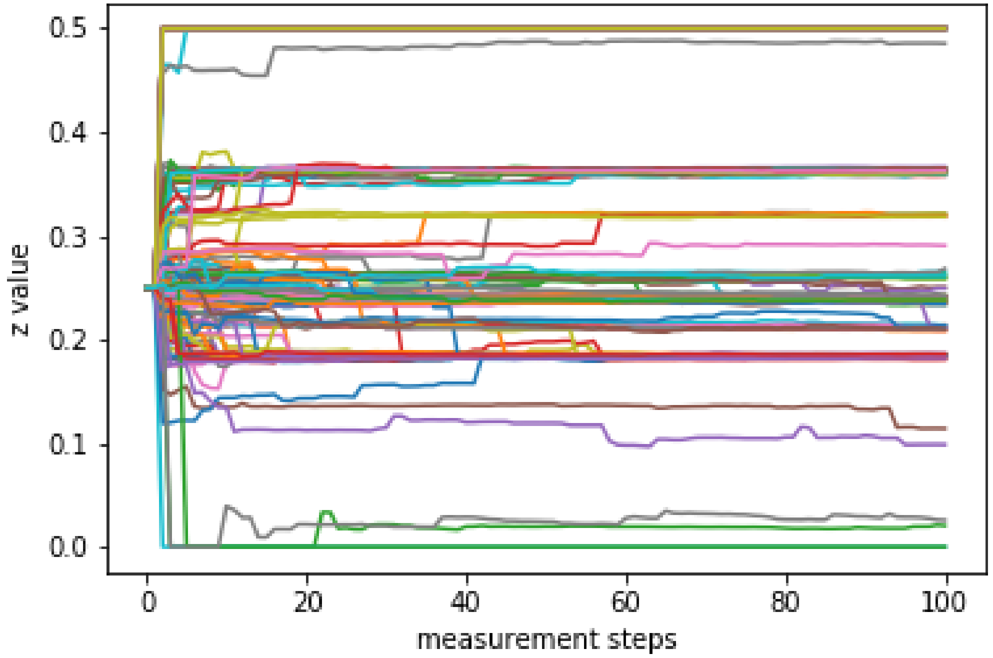
- State: Which Miller indices have been measured
- Action: choosing an hkl to measure
- Rewards: Certainty of z value, chi-squared, fewer steps



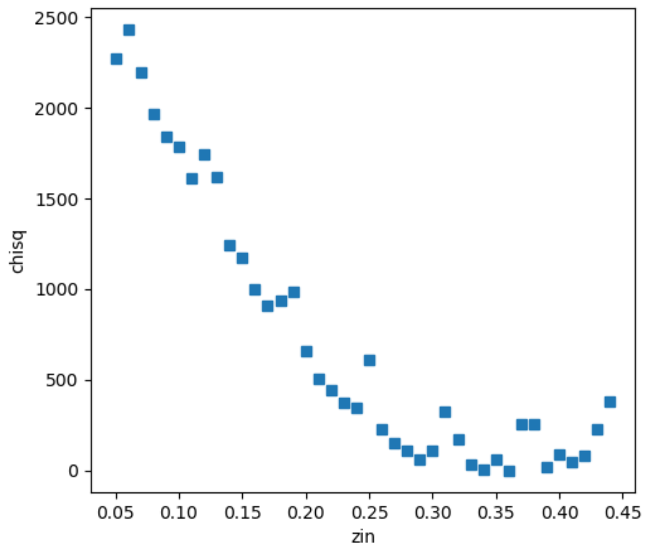
Application Challenges

Three main problems:

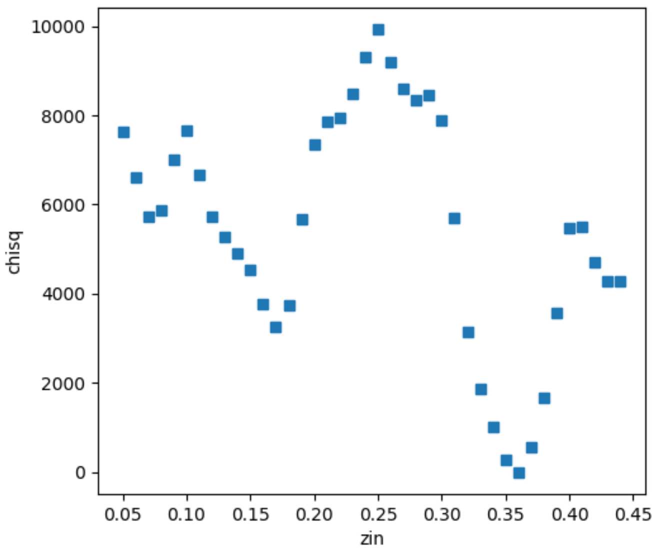
1. Fitter too dependent on starting z value → fits on minimum chi-squared value



Chi-squared by starting z value (5 hkls)



Chi-squared by starting z value (40 hkls)



Application Challenges

Three main problems:

2. Do not want algorithm repeating measurements → applied mask of invalid actions

5	10	3	26	15	9	17

Application Challenges

Three main problems:

2. Do not want algorithm repeating measurements → applied mask of invalid actions

Application Challenges

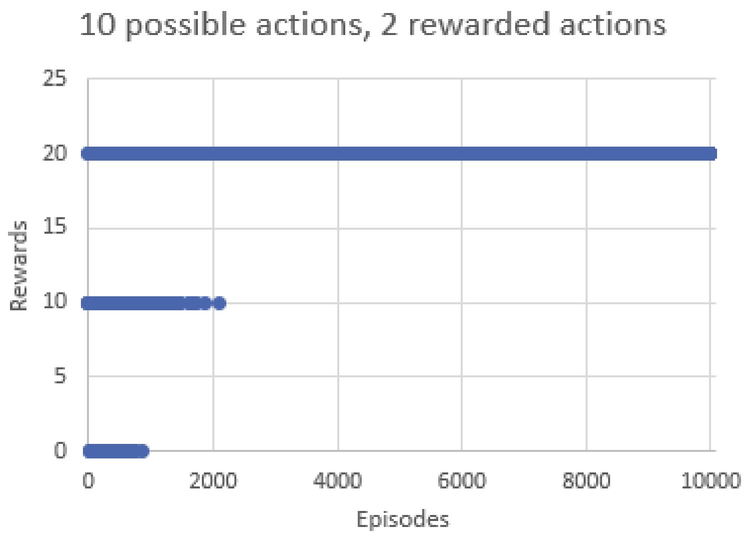
Three main problems:

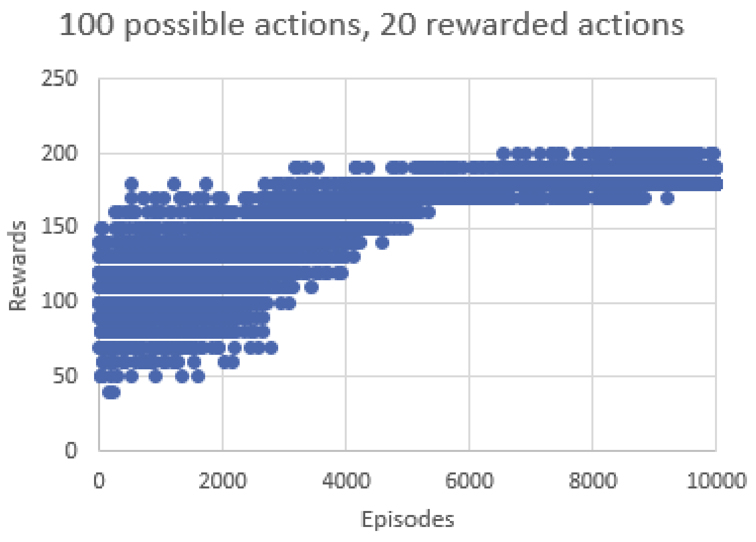
2. Do not want algorithm repeating measurements → applied mask of invalid actions

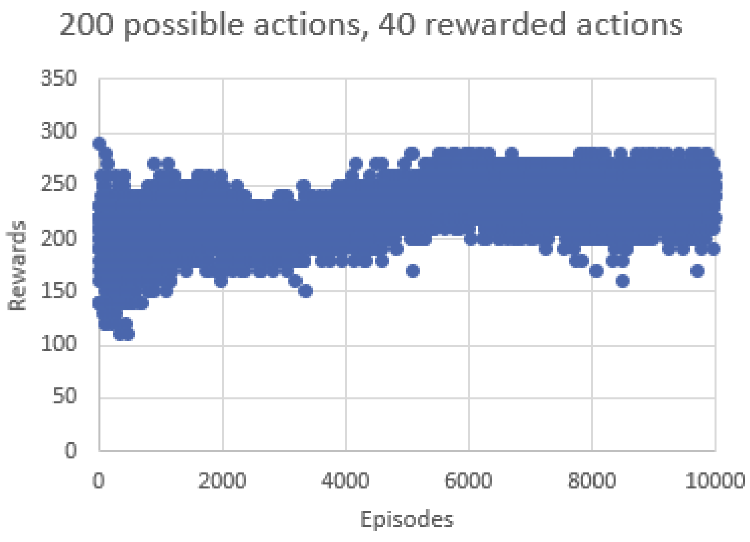
Application Challenges

Three main problems:

3. Large action space → test on dummy environment

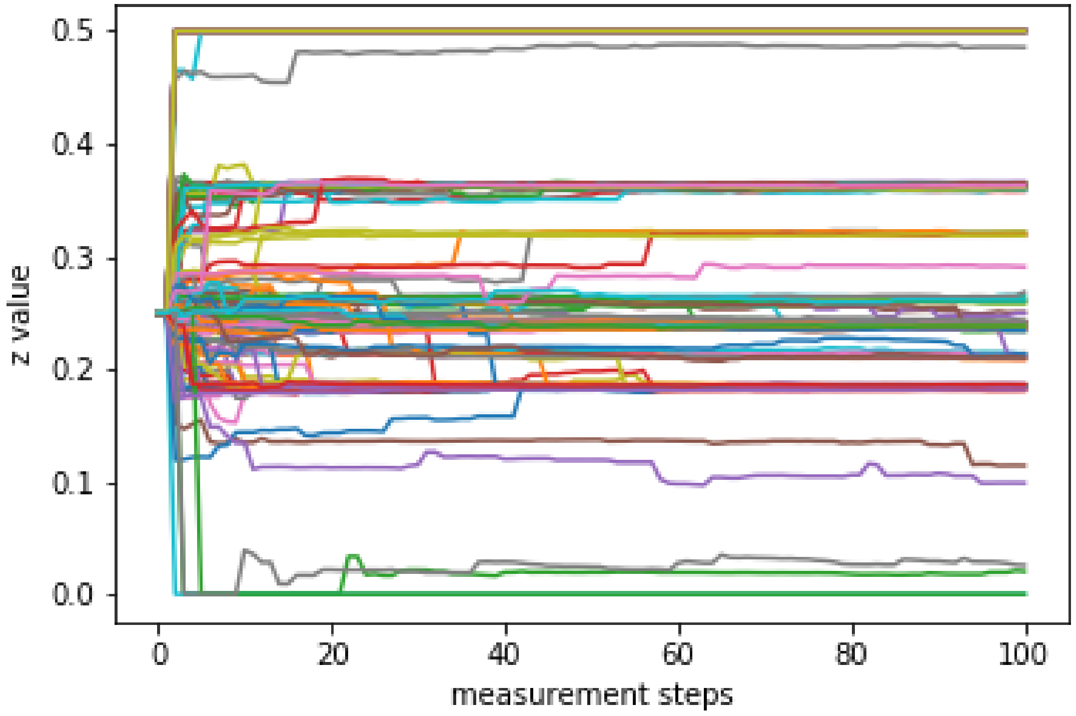




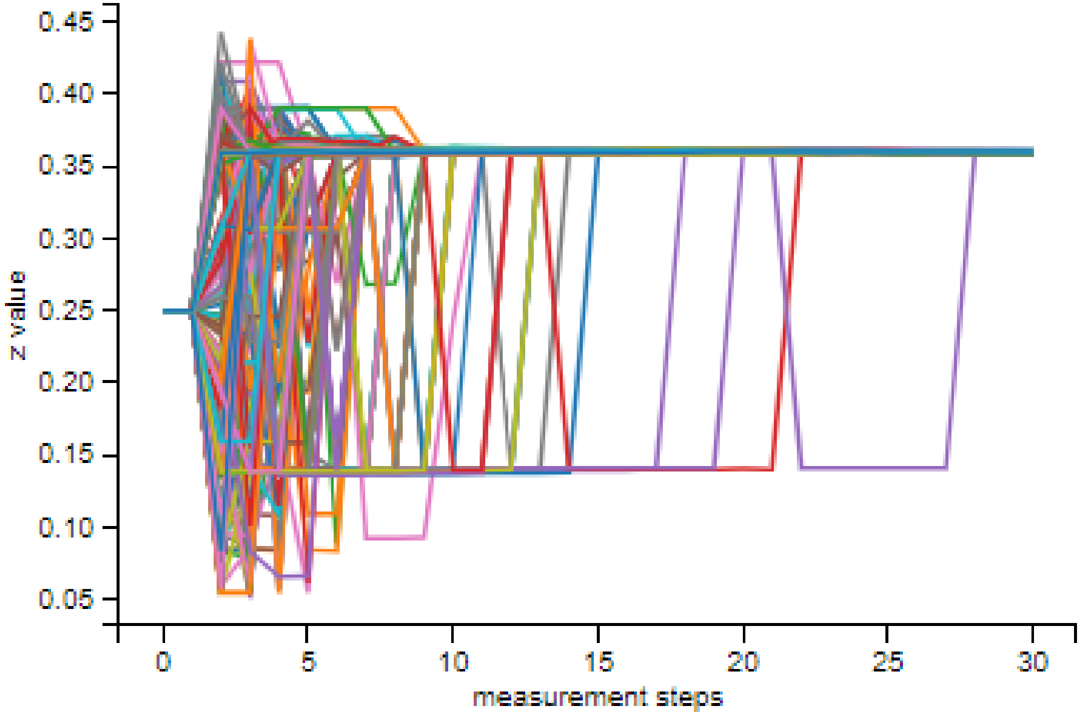


Results

Before

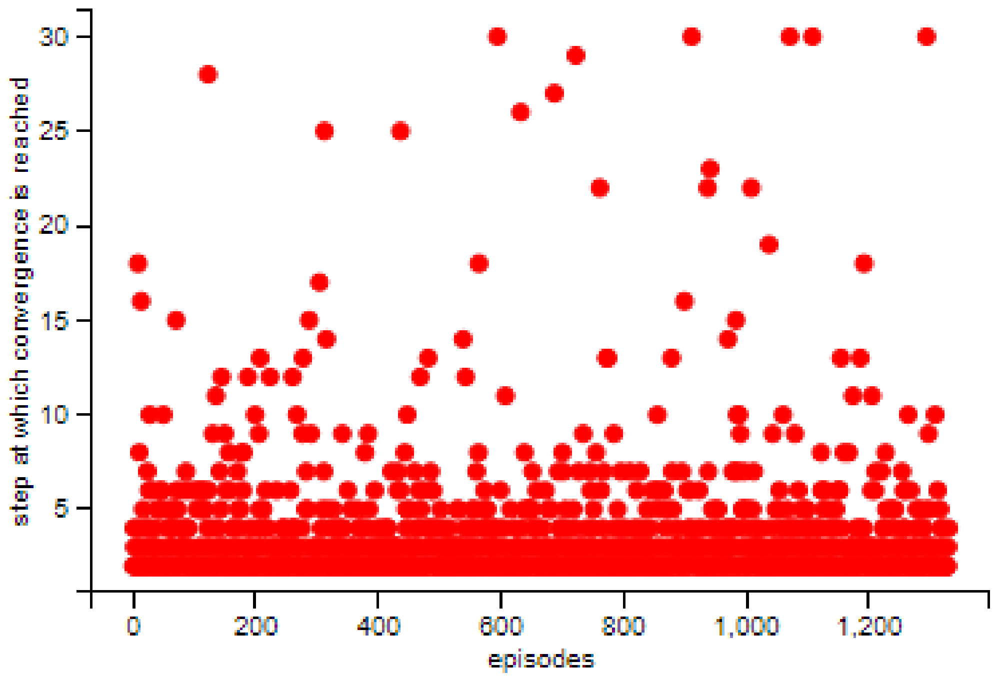


After

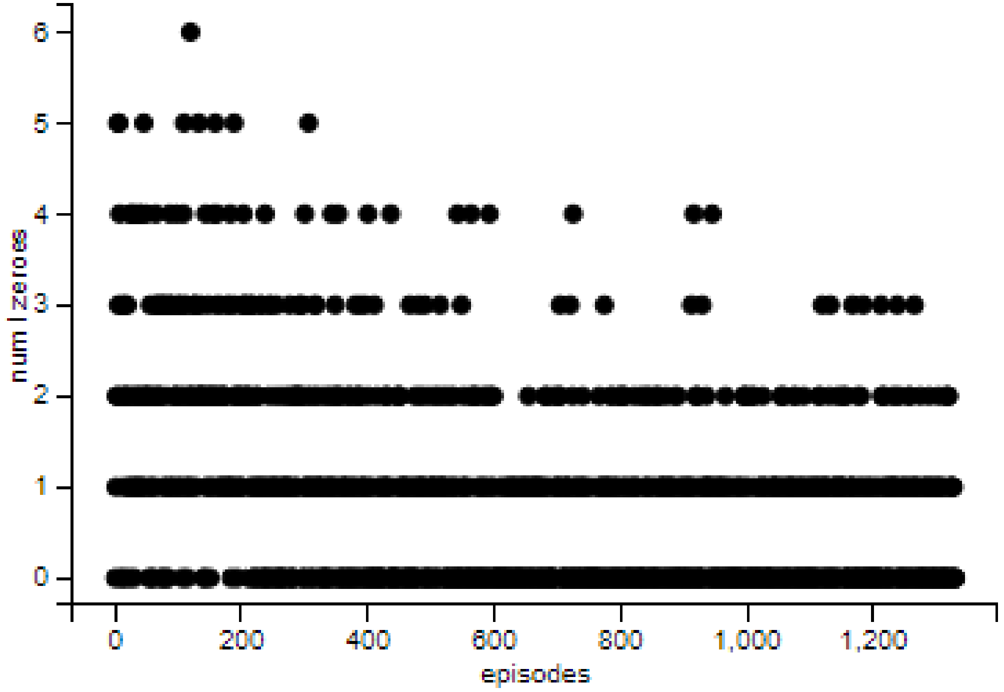


Results

Determining reinforcement learning success:



No decrease in number of steps until convergence



Decrease in number of useless plane measurements

Future Steps

- Run with more episodes so algorithm can learn
- Test on other algorithms
- Move to more complicated test problem
- Find permanent fitter (that handles more parameters)

Acknowledgements

Thanks to the SHIP program directors, as well as SHIP interns and mentors.

Special thanks to William Ratcliff, our mentor; Joseph Rath, SURF intern; and Paul Kienzle.

We are grateful for funding from the Center for High Resolution Neutron Scattering.

Questions?