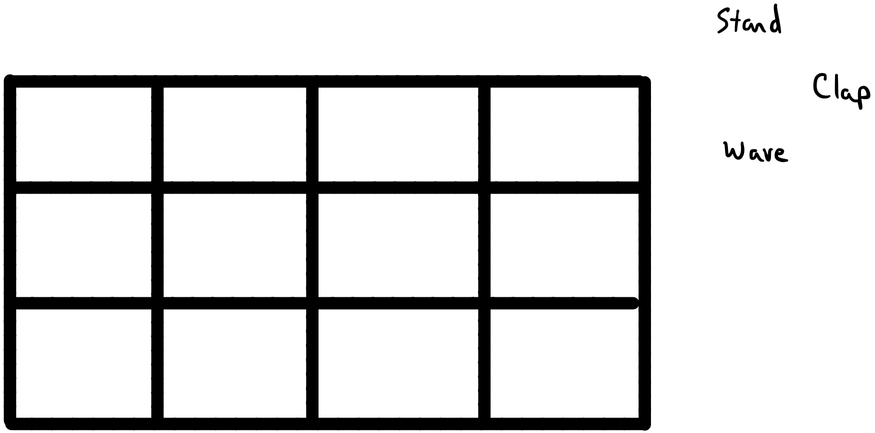
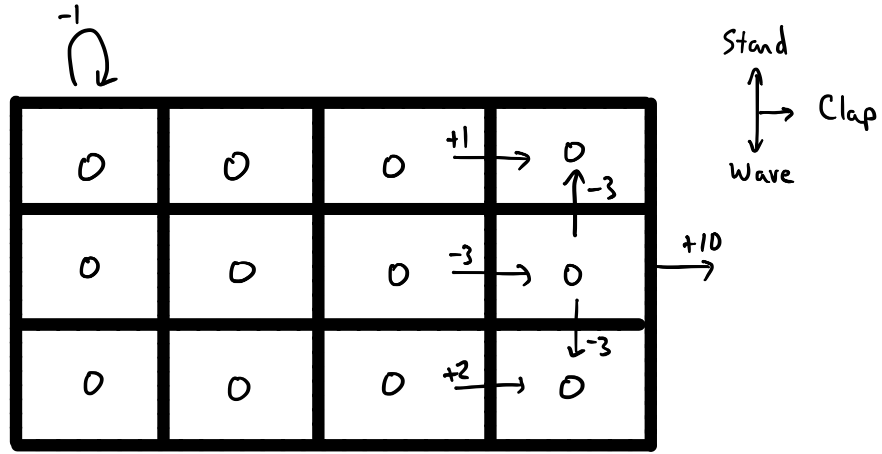
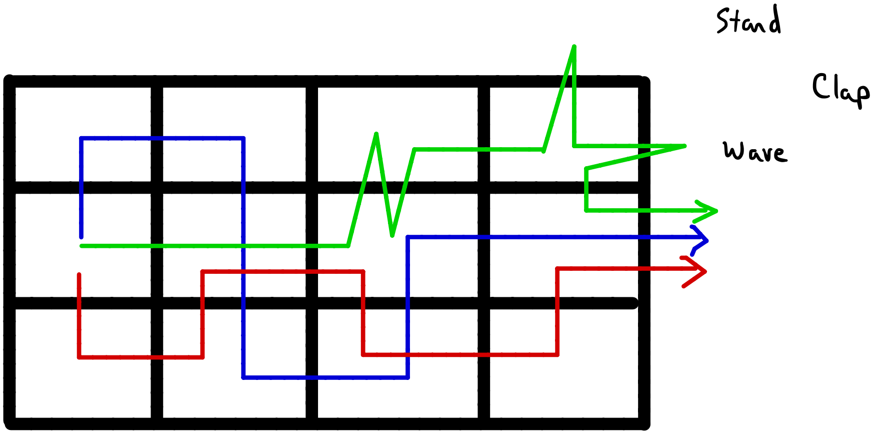
Stand

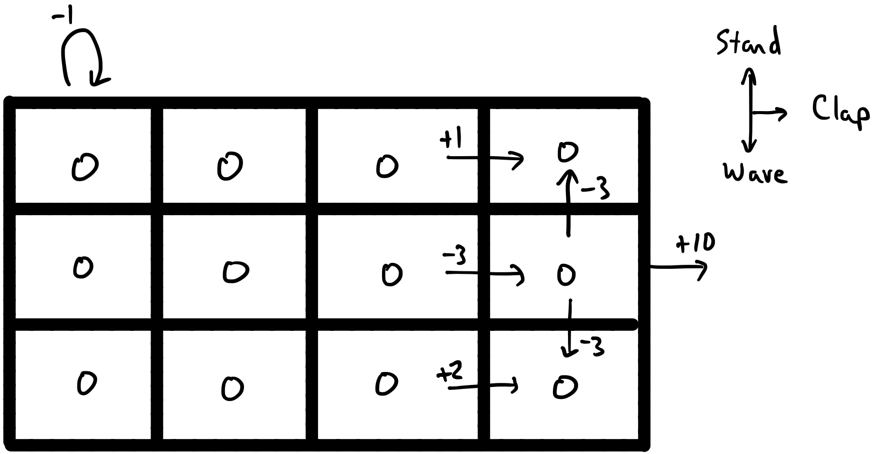
Clap

Wave

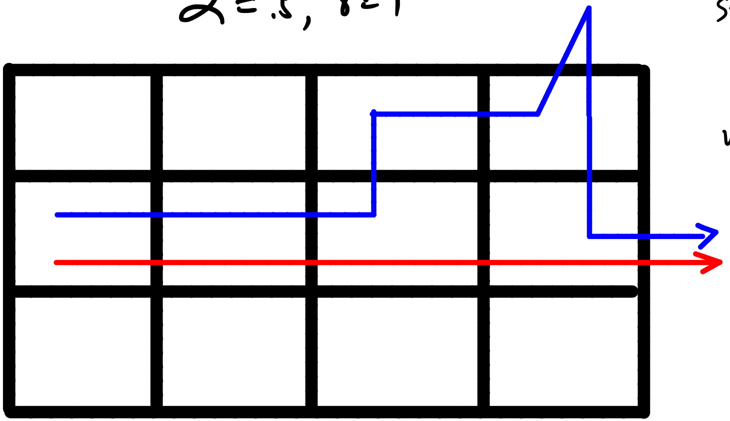

Stand
Clap
Wave


-1
0
0
0
+1
0
-3
0
0
0
-3
0
+10
-3
0
0
0
+2
0
Stand
Clap
Wave


Stand
Clap
Wave


-1
0
0
0
+1
0
-3
0
0
0
-3
0
+10
-3
0
0
0
+2
0
Stand
Clap
Wave

$\alpha = .5, \gamma = 1$



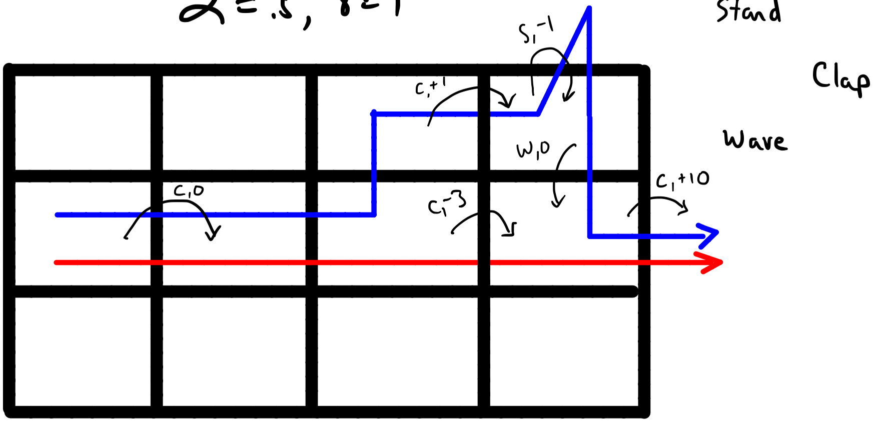
Stand

Clap

Wave

0	0	.5	-.25
0	0	1	7.5
0	0	0	0

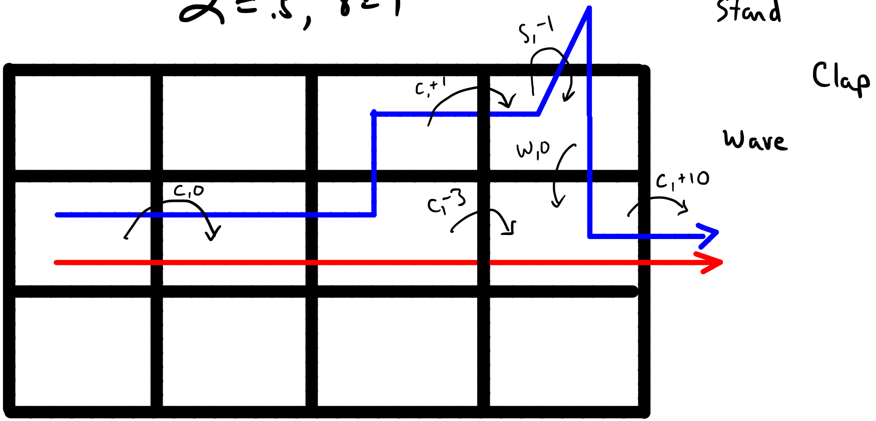
$\alpha = .5, \gamma = 1$



0	0	.5	-.25
0	0	1	7.5
0	0	0	0

			?

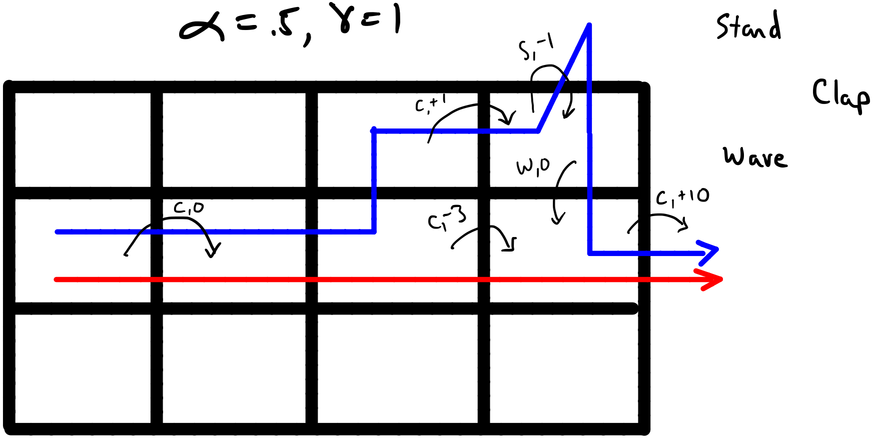
$\alpha = .5,\ \gamma = 1$



0	0	.5	-.25
0	0	1	7.5
0	0	0	0

		?	?
		?	~10

$\alpha = .5, \gamma = 1$



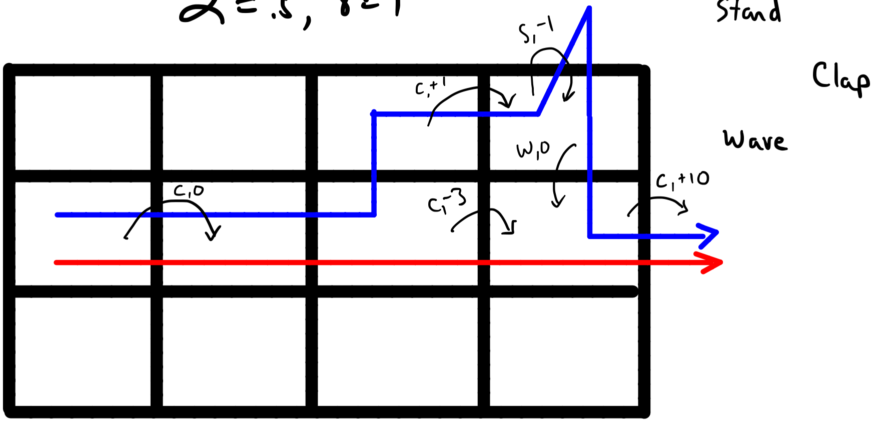
0	0	.5	-.25
0	0	1	7.5
0	0	0	0

			~9.5
			~10

on policy*

* Dyna-Q learns Q values so (↑, 9) and (↓, 10)

$\alpha = .5, \gamma = 1$

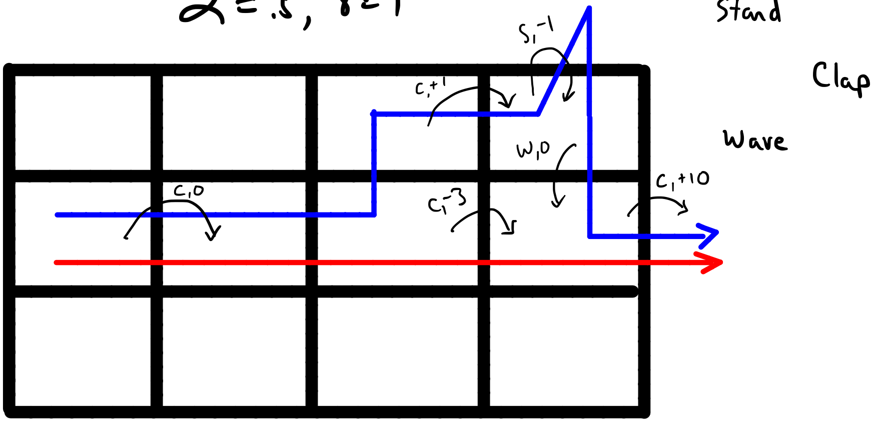


0	0	.5	-.25
0	0	1	7.5
0	0	0	0

on policy

		~10.5	~9.5
~8.75	~8.75	~8.75	~10

$\alpha = .5, \gamma = 1$



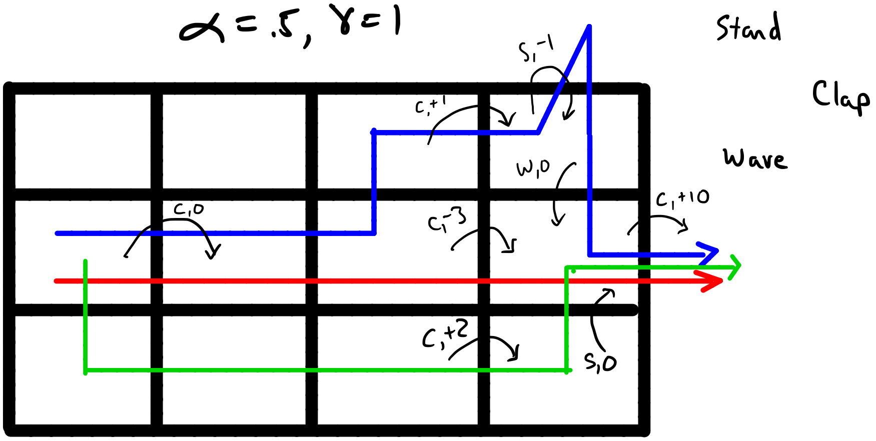
0	0	.5	-.25
0	0	1	7.5
0	0	0	0

on policy

		~10.5	~9.5
~8.75	~8.75	~8.75	~10

- What about unvisited states?
- What if transition function were stochastic?

$\alpha = .5, \gamma = 1$



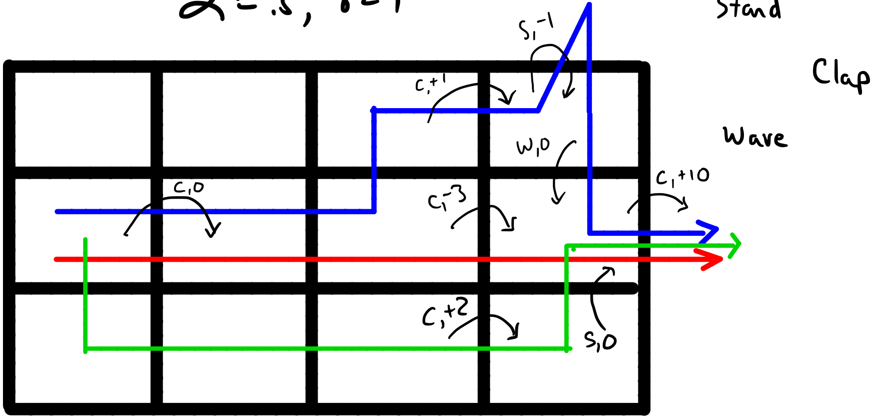
0	0	.5	-.25
0	0	1	7.5
0	0	0	0

0	0	~10.5	~9.5
~8.75	~8.75	~8.75	~10
0	0	0	0

on policy

- What about unvisited states?
- What if transition function were stochastic?
- Does the order of updates matter?

$\alpha = .5, \gamma = 1$



0	0	.5	-.25
0	0	1	7.5
0	0	0	0

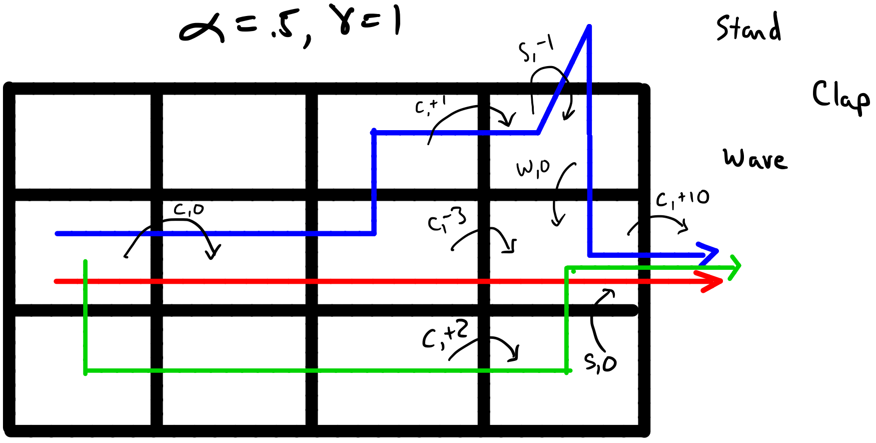
on policy

0	0	~10.5	~9.5
~8.75	~8.75	~8.75	~10
0	0	0	0

Δ0			Δ0
Δ0	Δ0	Δ2	Δ10

- What about unvisited states?
- What if transition function were stochastic?
- Does the order of updates matter?

$\alpha = .5, \gamma = 1$



0	0	.5	-.25
0	0	1	7.5
0	0	0	0

0	0	~10.5	~9.5
~8.75	~8.75	~8.75	~10
0	0	0	0

on policy

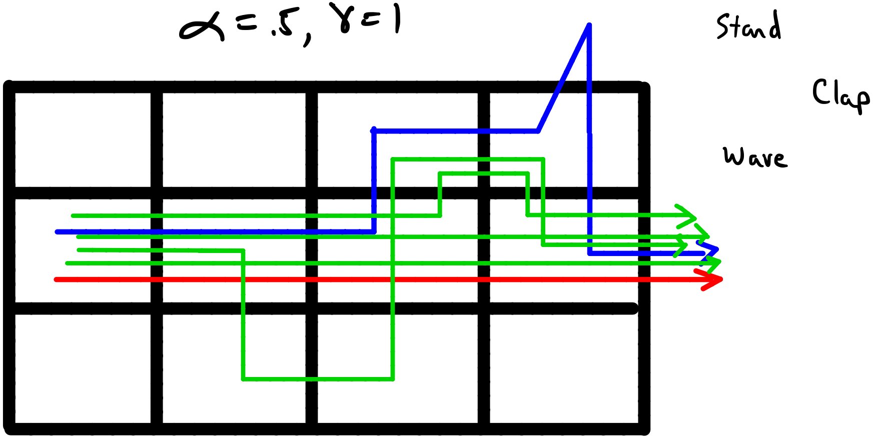
Δ0			Δ0
Δ0	Δ0	Δ2	Δ10

first updates

then here etc.

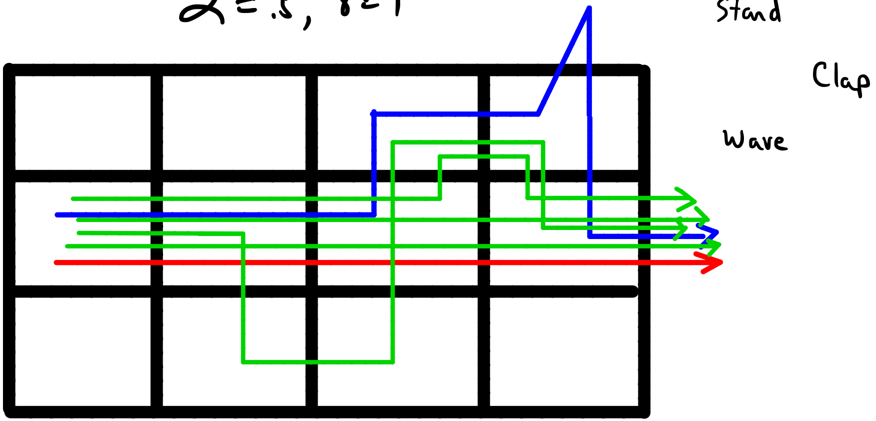
- What about unvisited states?
- What if transition function were stochastic?
- Does the order of updates matter?
  - prioritized sweeping
  - Could actually do update in step (e)

$\alpha = .5, \gamma = 1$



Trajectory sampling: how differs from Dyna here?

$\alpha = .5, \gamma = 1$

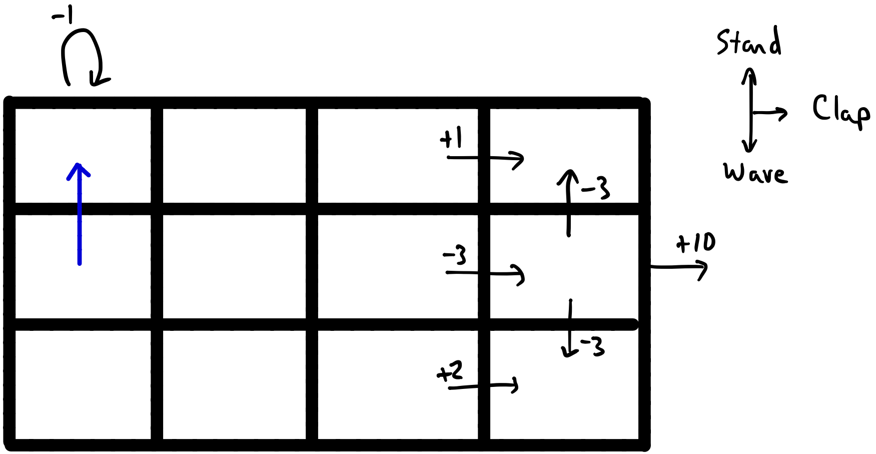


Trajectory sampling: how differs from Dyna here?

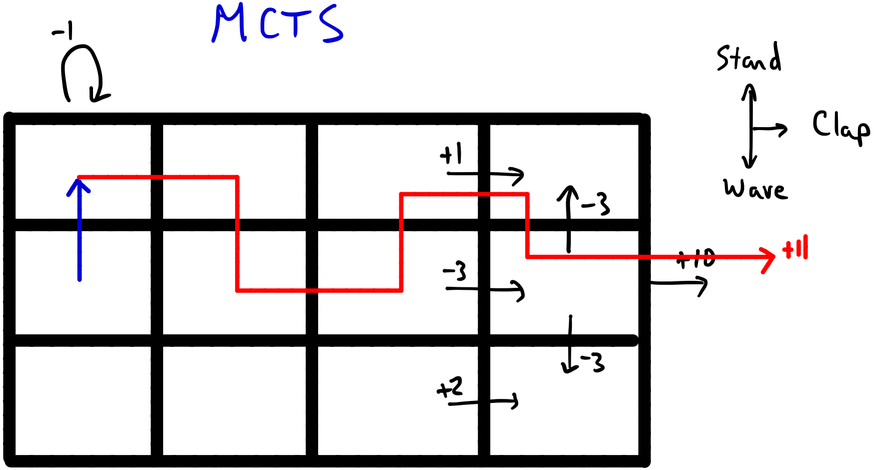
How does Dyna-Q^+ differ from Dyna here?

- relationship to UCB?
- In what way does Dyna-Q^+ violate the authors' principles?
- Could you accomplish something similar w/out changing updates? (see ex. 8.4)

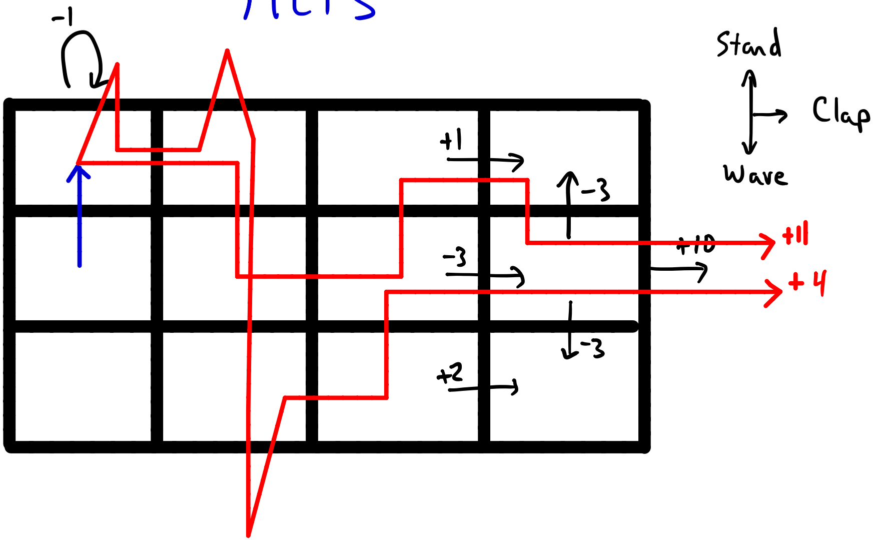
MCTS : Monte Carlo Tree Search - Planning at decision time



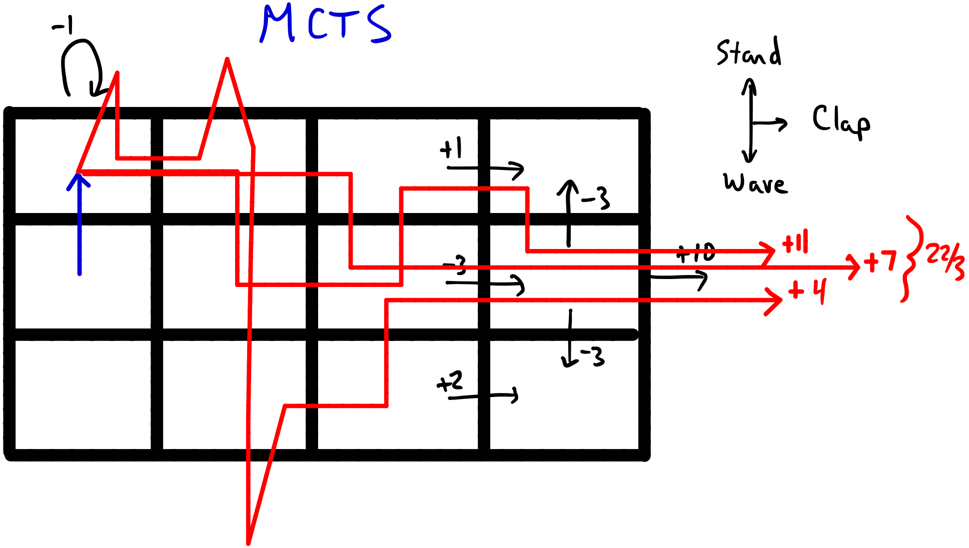
MCTS



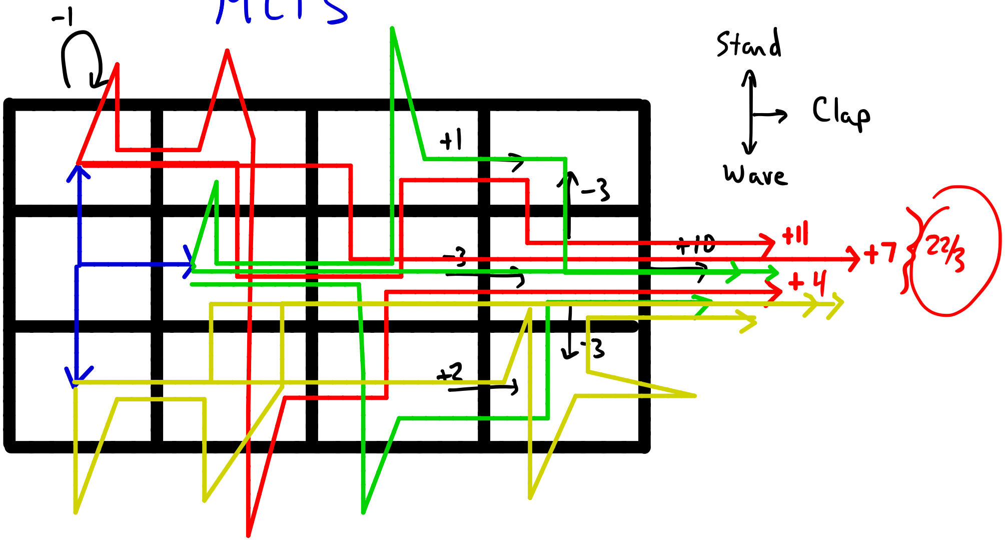
MCTS



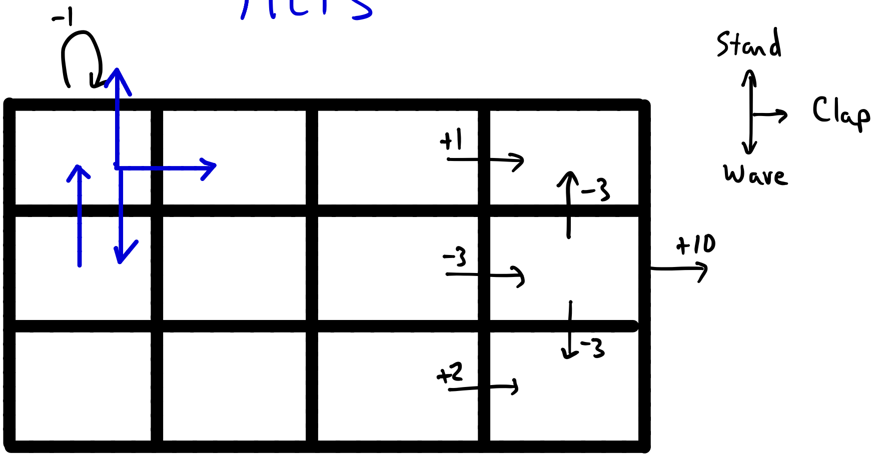
MCTS



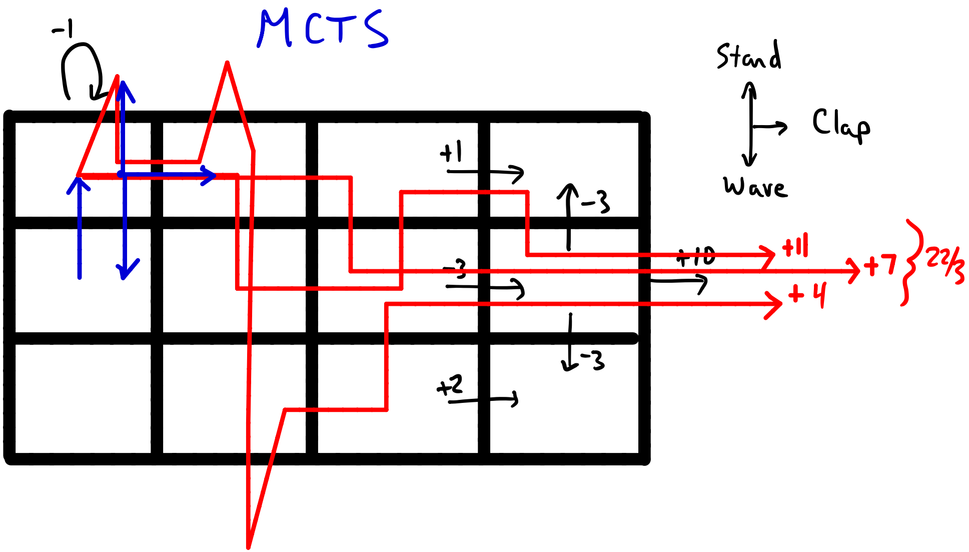
MCTS



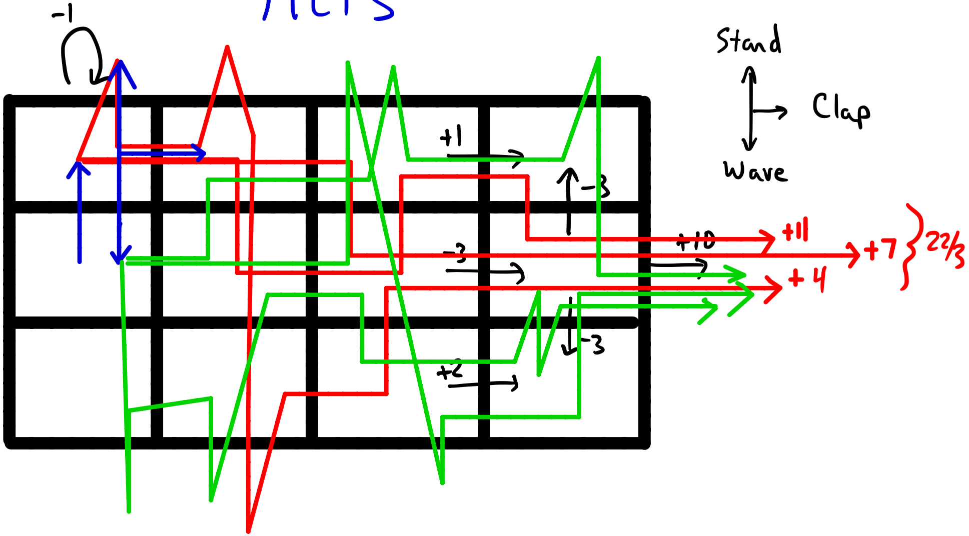
MCTS



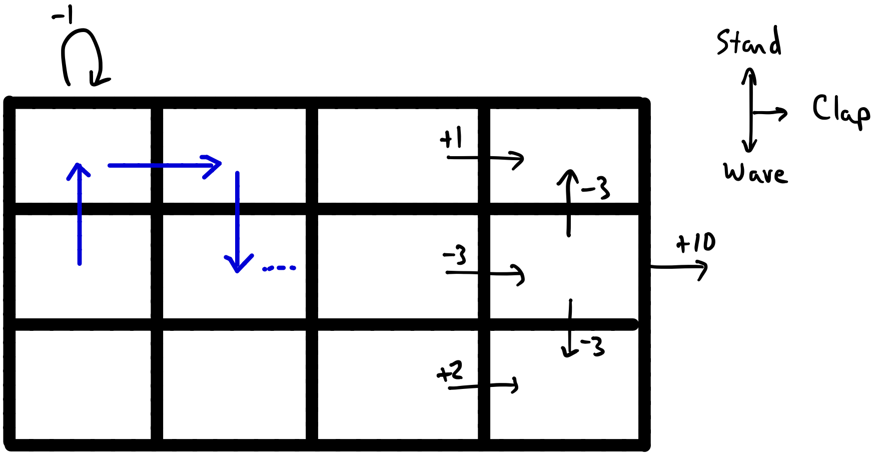
MCTS



MCTS



MCTS : Monte Carlo Tree Search – Planning at decision time



- Interleaving planning and acting : model known
- Focusses search on current state
- Can combine w/ learning a model
- Can combine w/ a learned value function
- Random rollouts especially useful in game playing
- Can use more informed rollouts

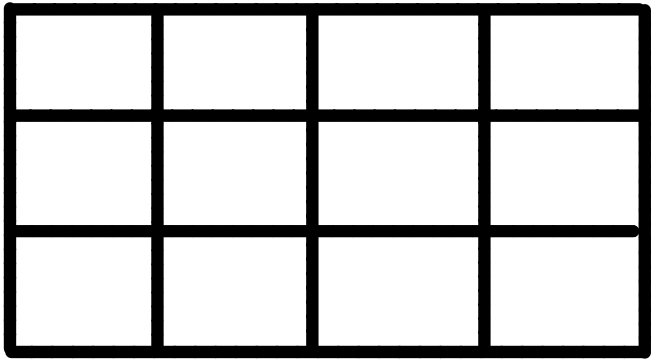
Approximation

Stand

Clap

Wave

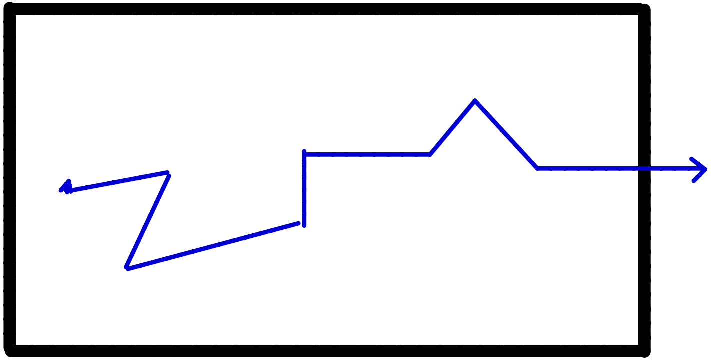
Approximation



Stand

Clap

Wave



Stand

Clap

Wave