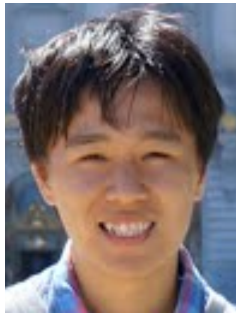
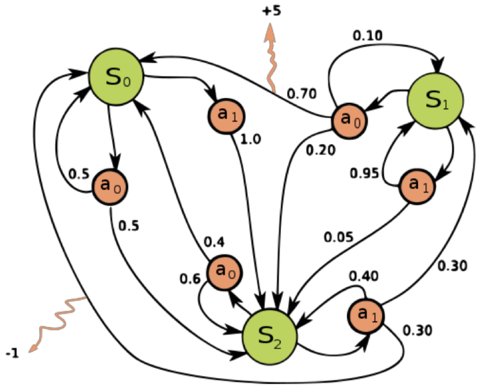
On Oracle-Efficient PAC RL with Rich Observations



Christoph Dann[1], Nan Jiang[2], Akshay Krishnamurthy[3]
Alekh Agarwal[3], John Langford[3], Robert E. Schapire[3]

[1]Carnegie Mellon University
[2]University of Illinois at Urbana-Champaign
[3]Microsoft Research

Exploration in RL

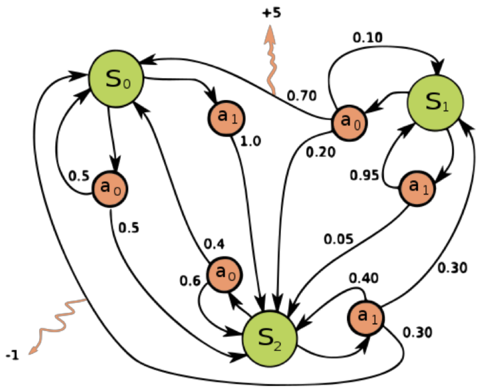


state	value
s_1	v_1
s_2	v_2
s_3	v_3

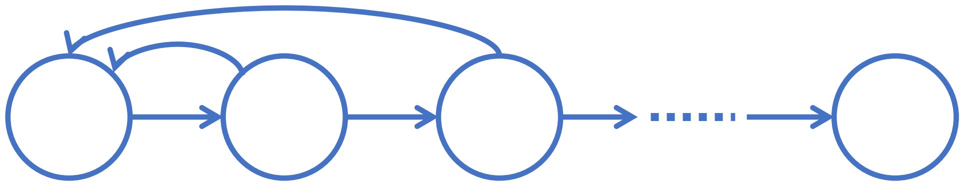
“Tabular”

[Kearns & Singh’98, Brafman & Tennenholtz’02, etc.]

Exploration in RL



state	value
s_1	v_1
s_2	v_2
s_3	v_3

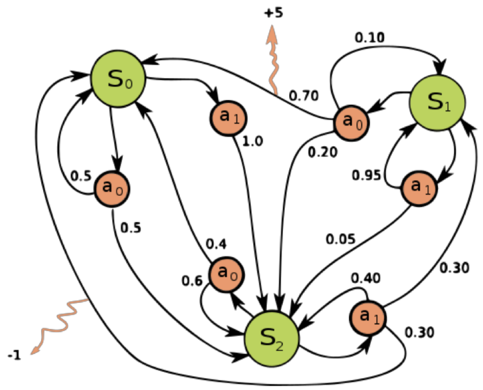


+ ϵ-greedy ⇒ exp. sample complexity!

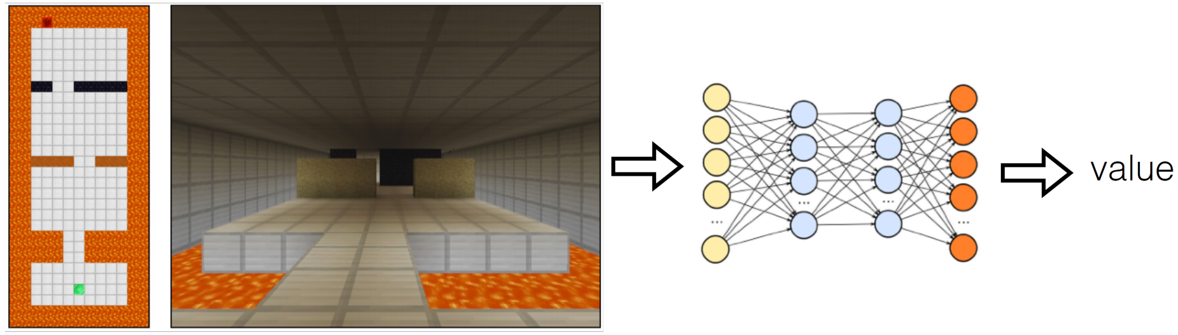
“Tabular”

[Kearns & Singh’98, Brafman & Tennenholtz’02, etc.]

Exploration in RL



state	value
s_1	v_1
s_2	v_2
s_3	v_3



+ ϵ-greedy $\Rightarrow$ exp. sample complexity!

“Tabular”

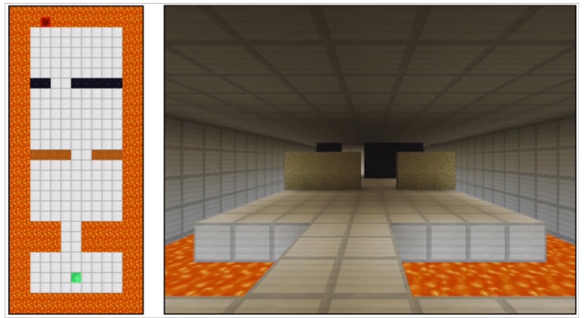
[Kearns & Singh’98, Brafman & Tennenholtz’02, etc.]

Function approximation

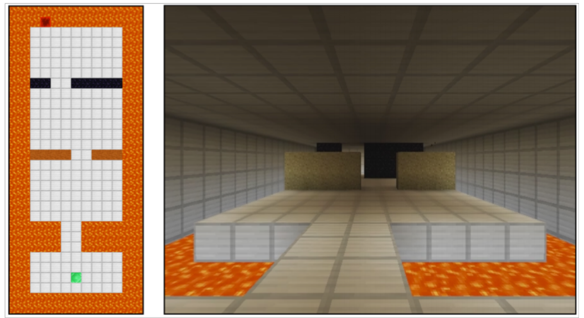
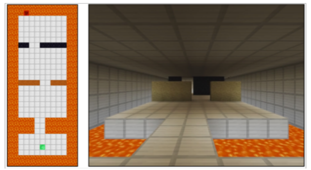
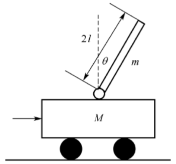
?

Figure from Johnson et al’16

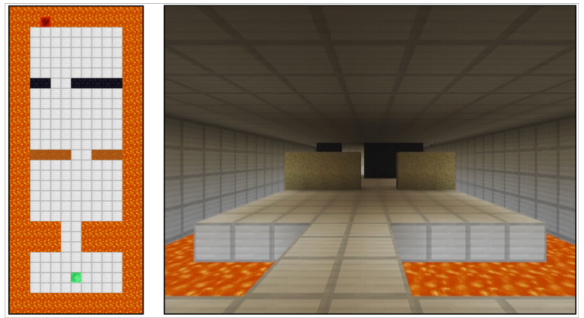
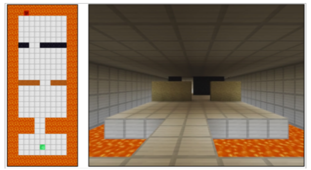
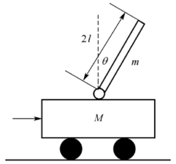
Prior Work & Contribution

Algorithm	LSVEE [KAL'16]		
Setting	Contextual decision processes (CDP) w/ deterministic hidden state dynamics and stochastic rich observations 		
Sample complexity	*poly*(#hidden states) ✓		
Computation	Enumerate functions X		

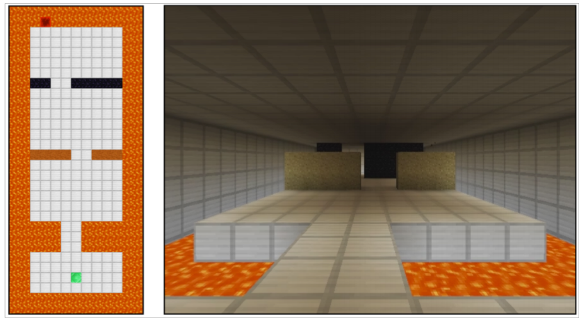
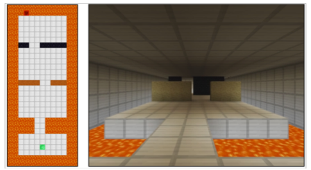
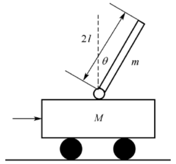
Prior Work & Contribution

Algorithm	LSVEE [KAL'16]		OLIVE [JKALS'17]
Setting	Contextual decision processes (CDP) w/ deterministic hidden state dynamics and stochastic rich observations 		 CDPs with stochastic dynamics  LQRs … anything with low *Bellman rank*
Sample complexity	*poly*(#hidden states) ✓		*poly*(*Bellman rank*) ✓
Computation	Enumerate functions X		Enumerate functions X

Prior Work & Contribution

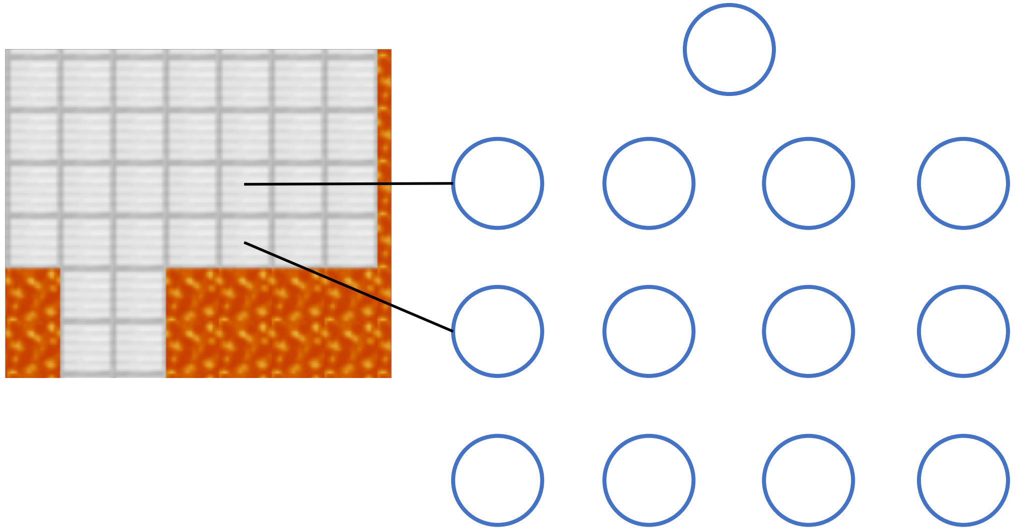
Algorithm	LSVEE [KAL’16]	**VALOR** **[this work]**	OLIVE [JKALS’17]
Setting	Contextual decision processes (CDP) w/ deterministic hidden state dynamics and stochastic rich observations 		 CDPs with stochastic dynamics  LQRs … anything with low *Bellman rank*
Sample complexity	*poly*(#hidden states) ✓		*poly*(*Bellman rank*) ✓
Computation	Enumerate functions X	**Linear program + cost-sensitive classification** ✓	Enumerate functions X

Prior Work & Contribution

Algorithm	LSVEE [KAL'16]	**VALOR** **[this work]**	OLIVE [JKALS'17]
Setting	Contextual decision processes (CDP) w/ deterministic hidden state dynamics and stochastic rich observations 		 CDPs with stochastic dynamics  LQRs … anything with low *Bellman rank*
Sample complexity	*poly*(#hidden states) ✓		*poly*(*Bellman rank*) ✓
Computation	Enumerate functions ✗	**Linear program + cost-sensitive classification** ✓	Enumerate functions ✗ **NP-Hard in tabular case**

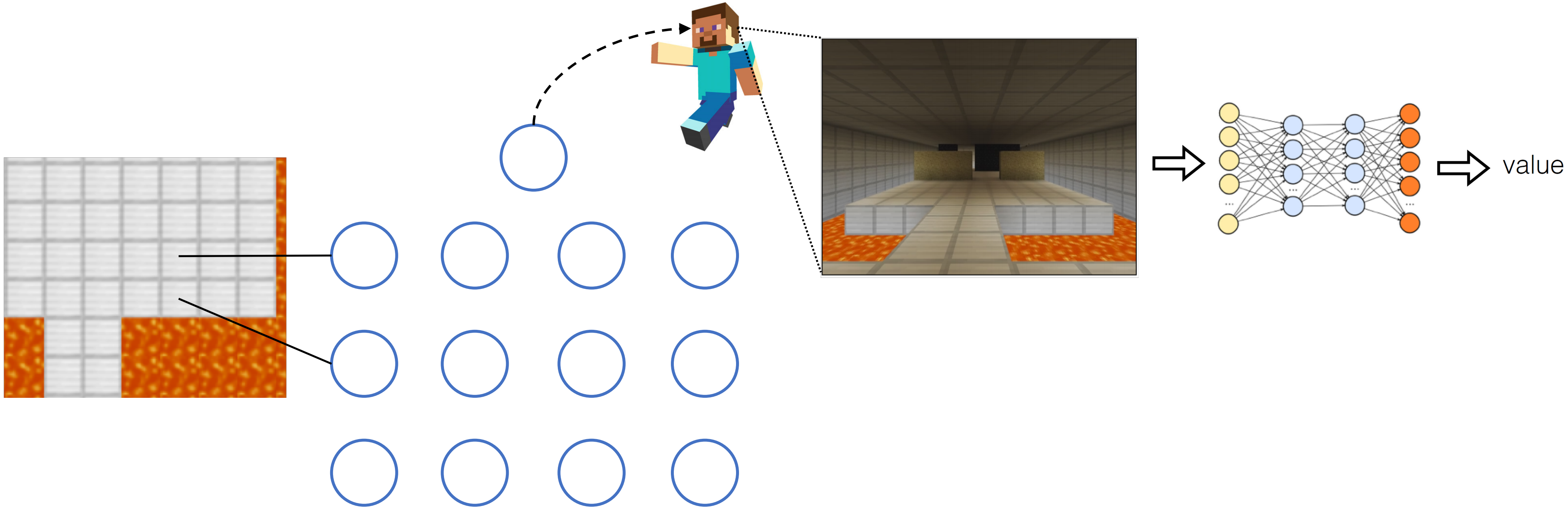
VALOR: efficiently implemented LSVEE

Setting: CDP with deterministic hidden state dynamics



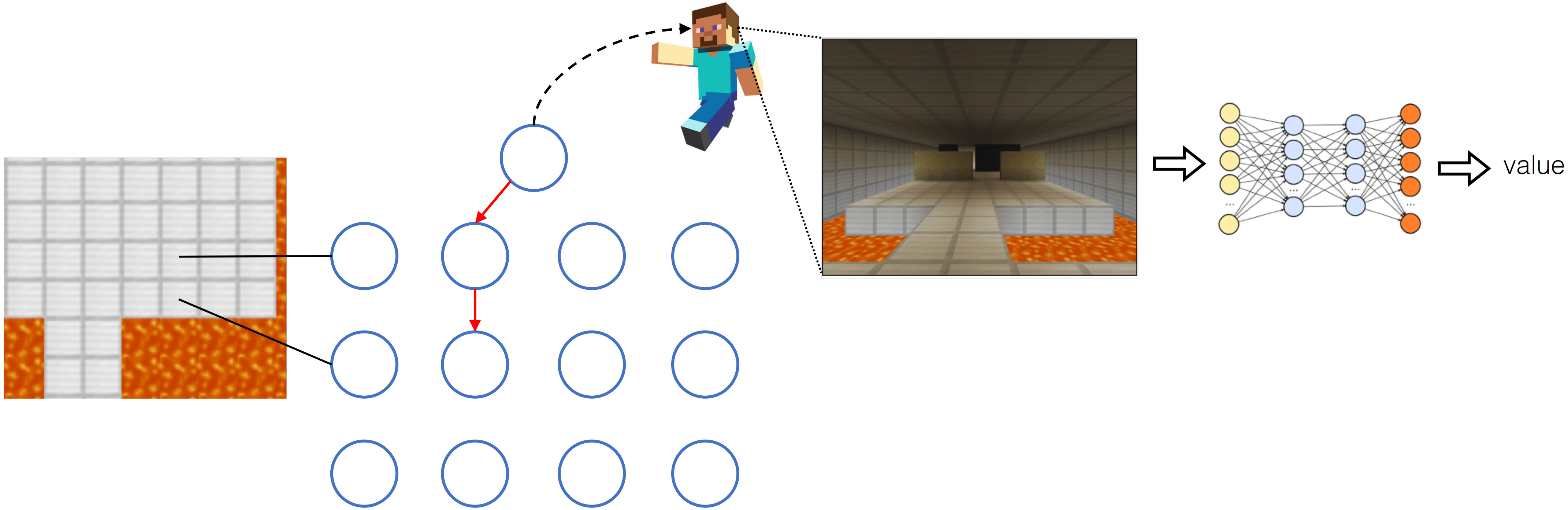
VALOR: efficiently implemented LSVEE

Setting: CDP with deterministic hidden state dynamics



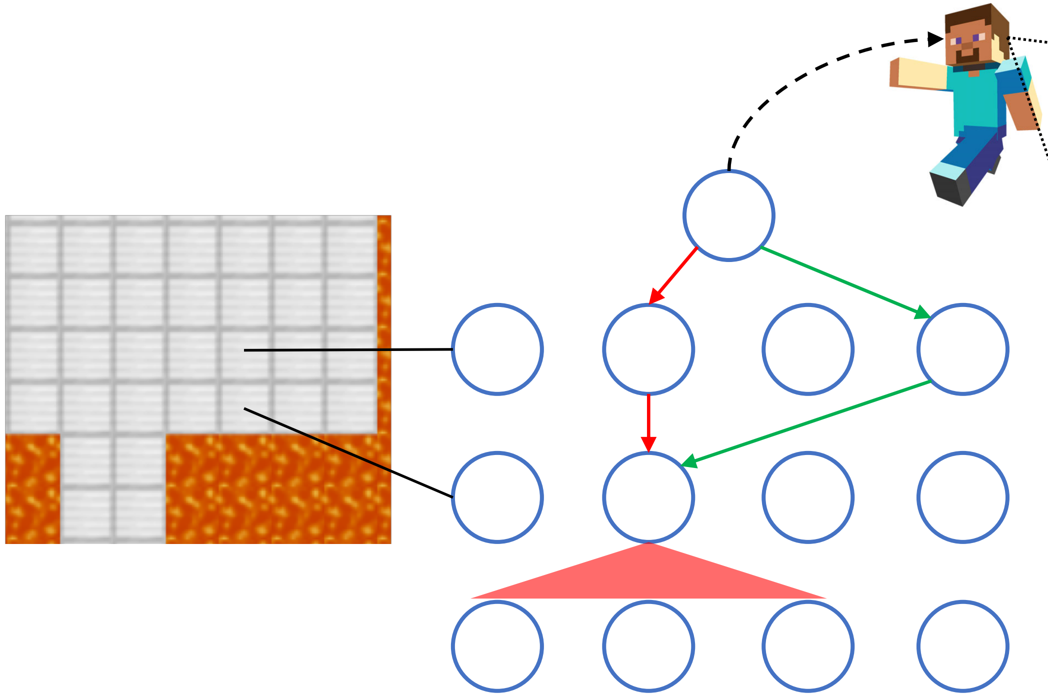
VALOR: efficiently implemented LSVEE

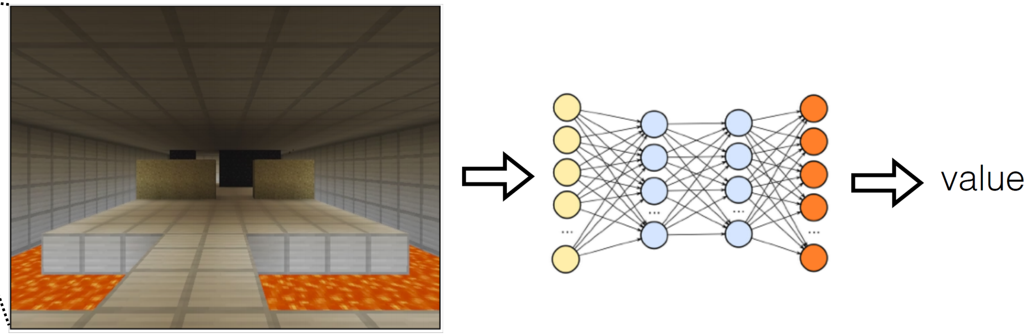
Setting: CDP with deterministic hidden state dynamics



VALOR: efficiently implemented LSVEE

Setting: CDP with deterministic hidden state dynamics

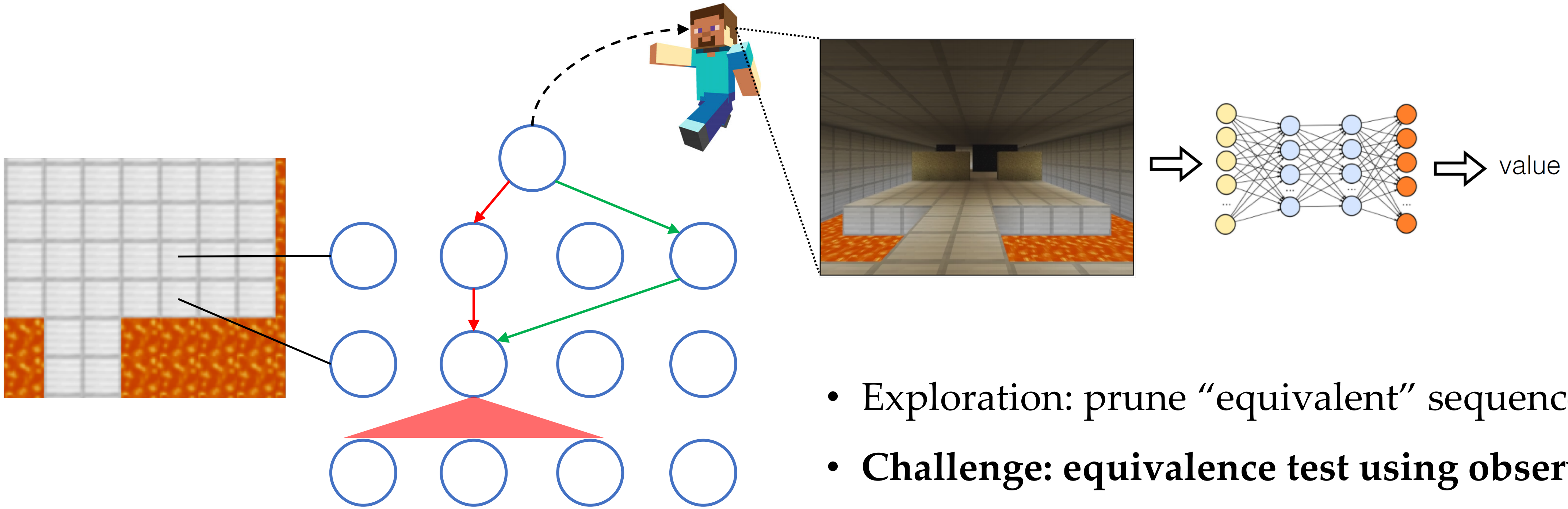




- Exploration: prune “equivalent” sequences

VALOR: efficiently implemented LSVEE

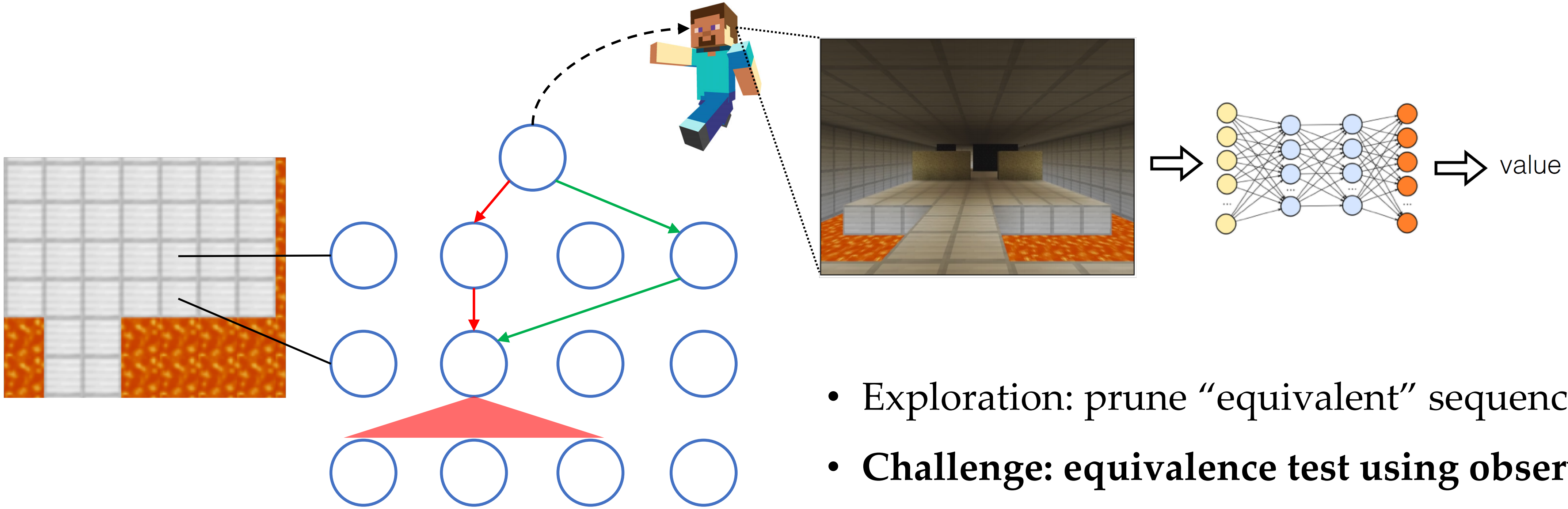
Setting: CDP with deterministic hidden state dynamics



- Exploration: prune "equivalent" sequences
- **Challenge: equivalence test using observations**
  - Model V^* and π^* separately (instead of Q^*)
  - Can be written as a linear program

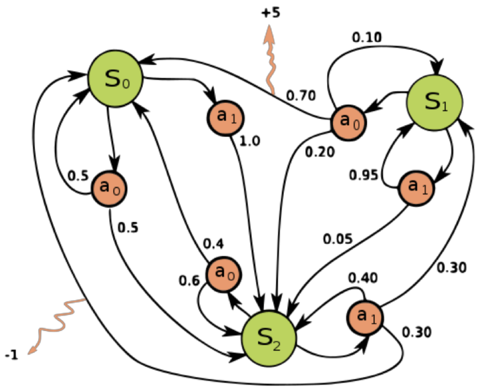
VALOR: efficiently implemented LSVEE

Setting: CDP with deterministic hidden state dynamics



- Exploration: prune “equivalent” sequences
- **Challenge: equivalence test using observations**
  - Model V^* and π^* separately (instead of Q^*)
  - Can be written as a linear program
- **What if we remove determinism?**

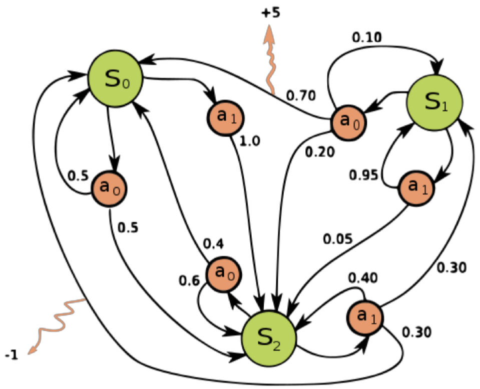
OLIVE is NP-hard in the tabular setting



state	value
s_1	v_1
s_2	v_2
s_3	v_3

OLIVE is NP-hard in the tabular setting

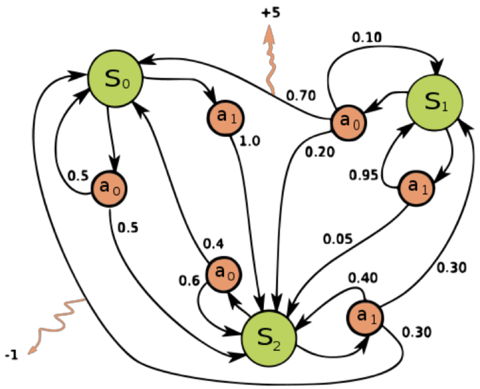
- **But common oracles are efficient in tabular case**
  - e.g., 0-1 loss: majority vote for each *s* separately
  - OLIVE cannot be implemented efficiently with oracles



state	value
s_1	v_1
s_2	v_2
s_3	v_3

OLIVE is NP-hard in the tabular setting

- **But common oracles are efficient in tabular case**
  - e.g., 0-1 loss: majority vote for each *s* separately
  - OLIVE cannot be implemented efficiently with oracles



- **Not the end of story**
  - Lower bound for algorithm, not problem
  - Efficient RL in OLIVE's setting is still an open problem

state	value
s_1	v_1
s_2	v_2
s_3	v_3