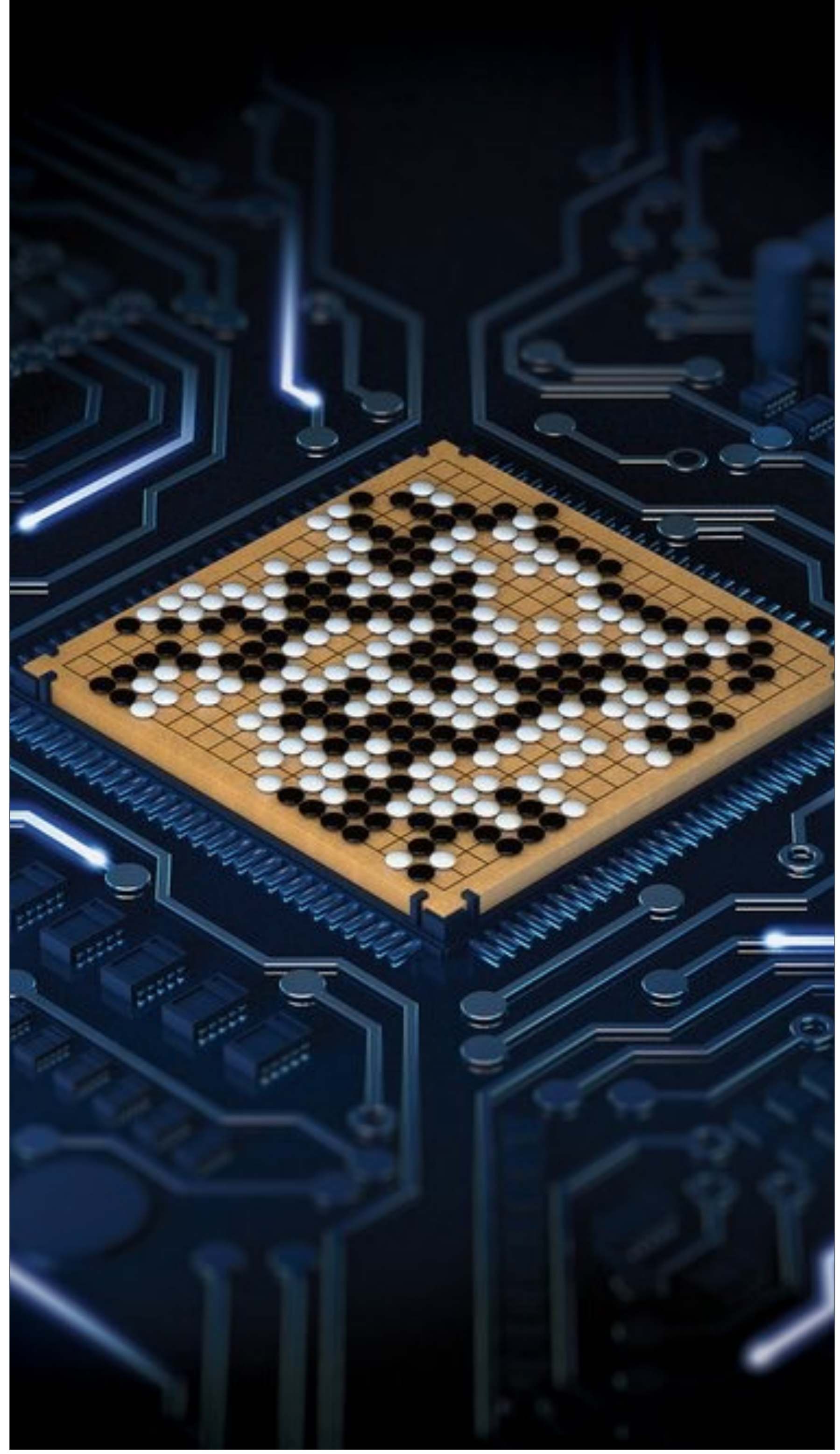


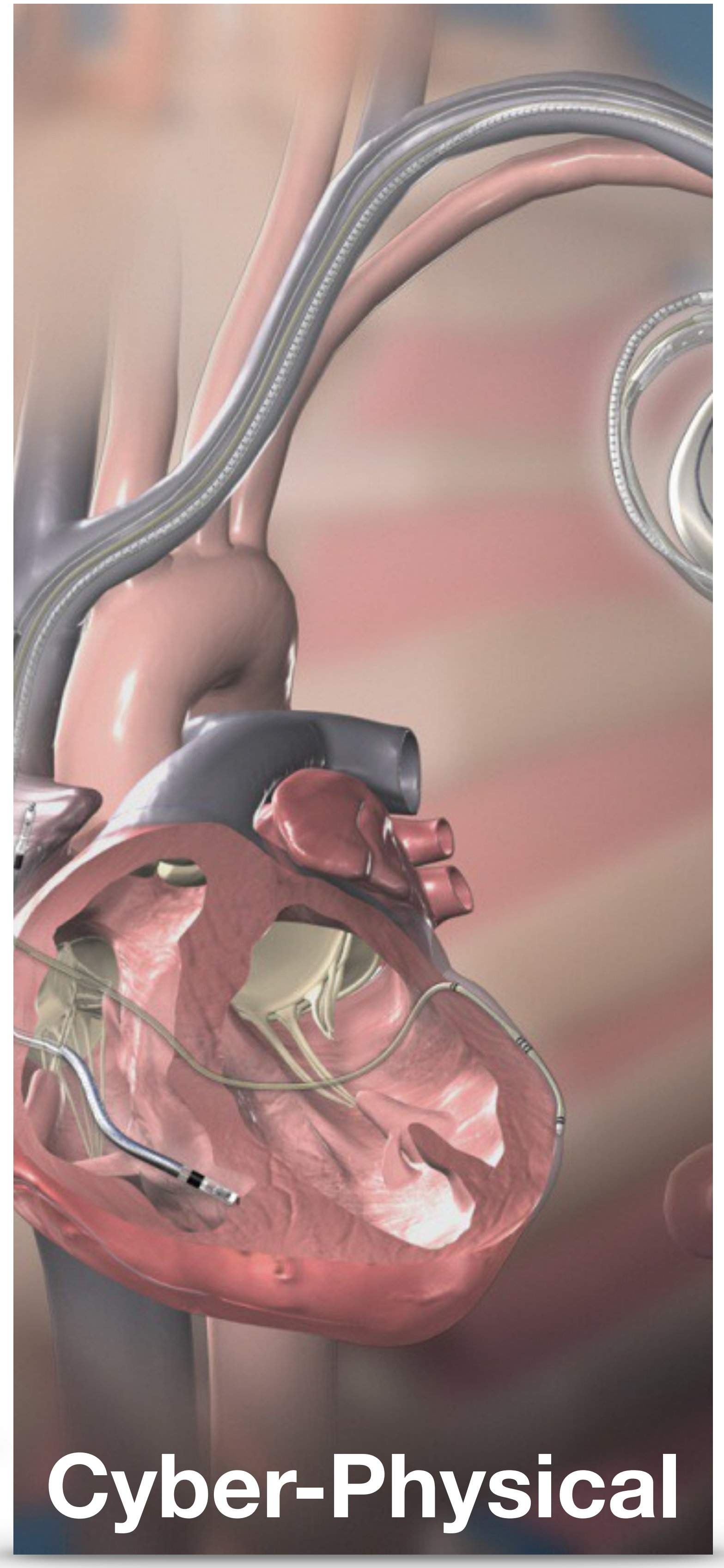
CUPLV

Programmatic Reinforcement Learning for All

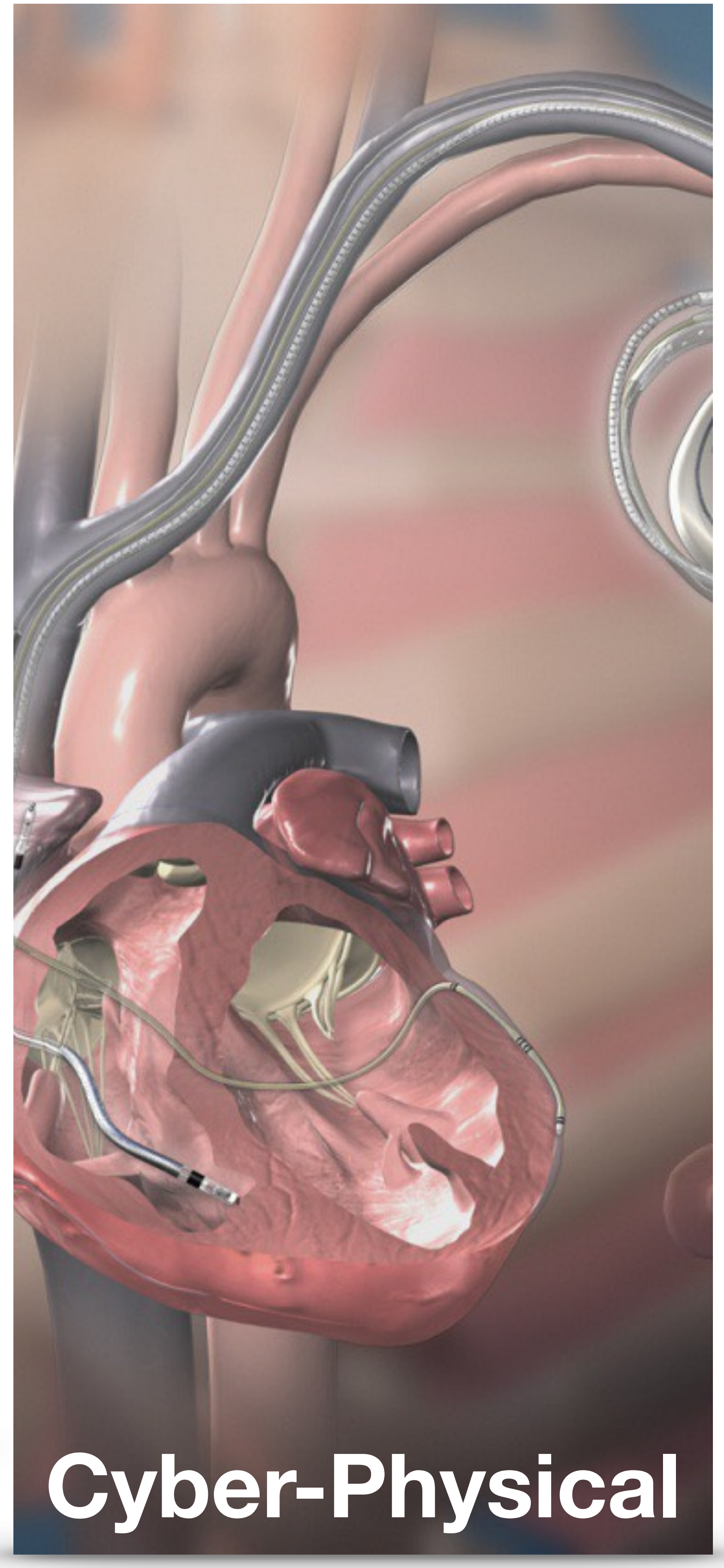
Ashutosh Trivedi
Computer Science and ECEE
University of Colorado Boulder


Hardware


x = True
y = False
z = False
x = False
y = True
z = False
x = False
y = False
z = True
Software

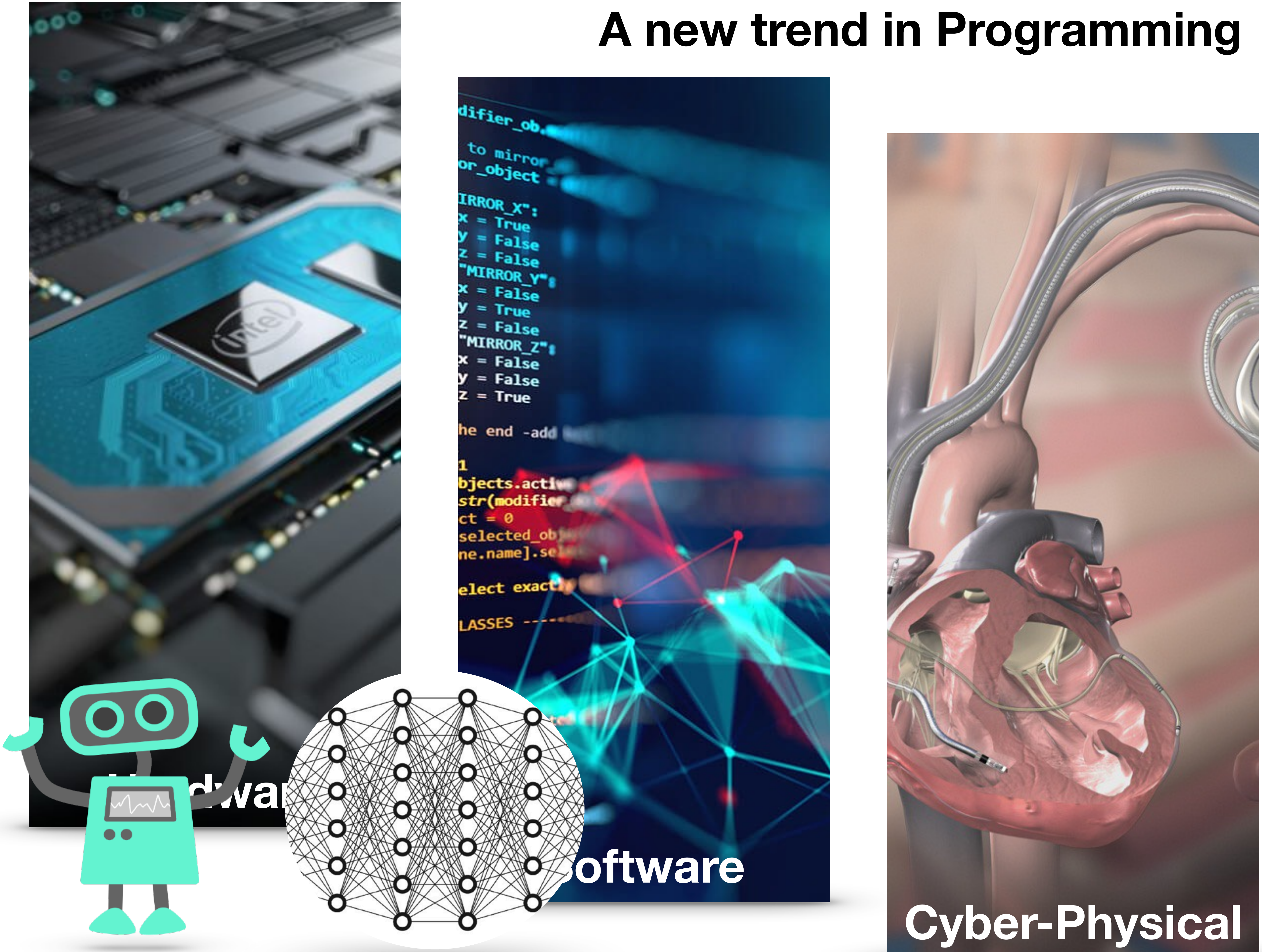

Cyber-Physical

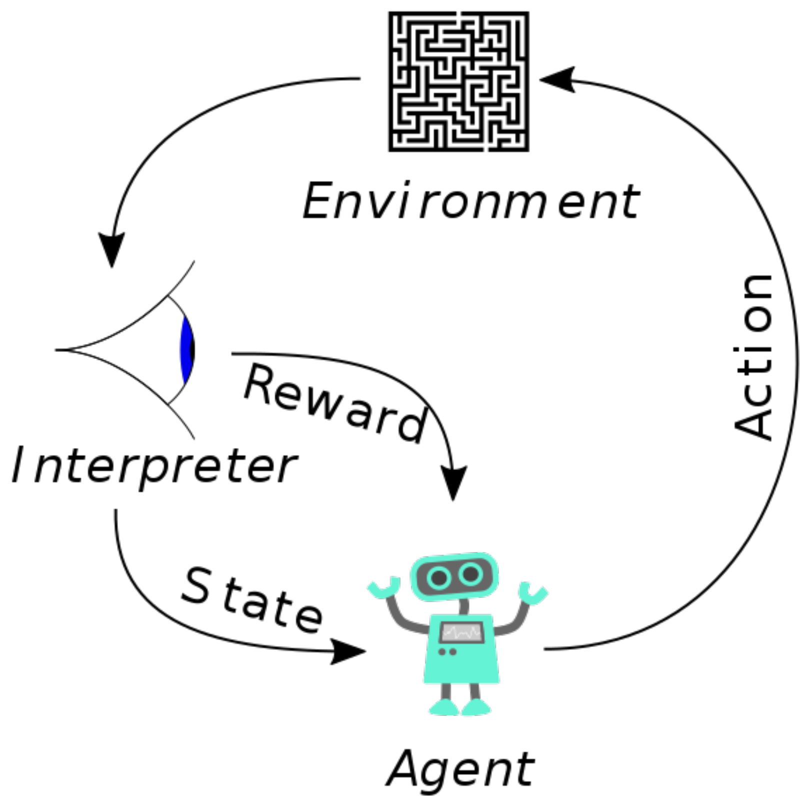
A new trend in Programming



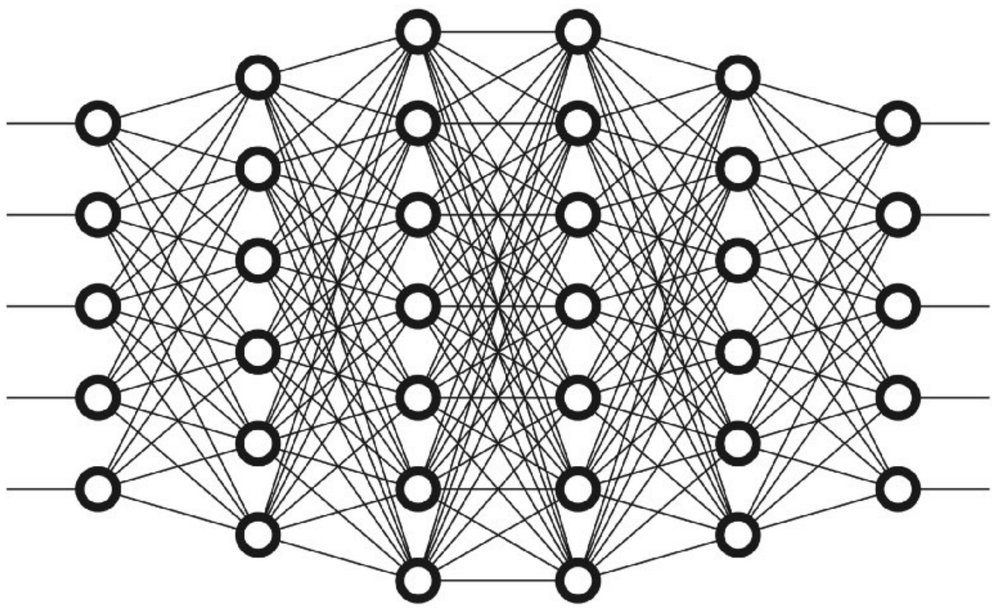
A new trend in Programming

A new trend in Programming

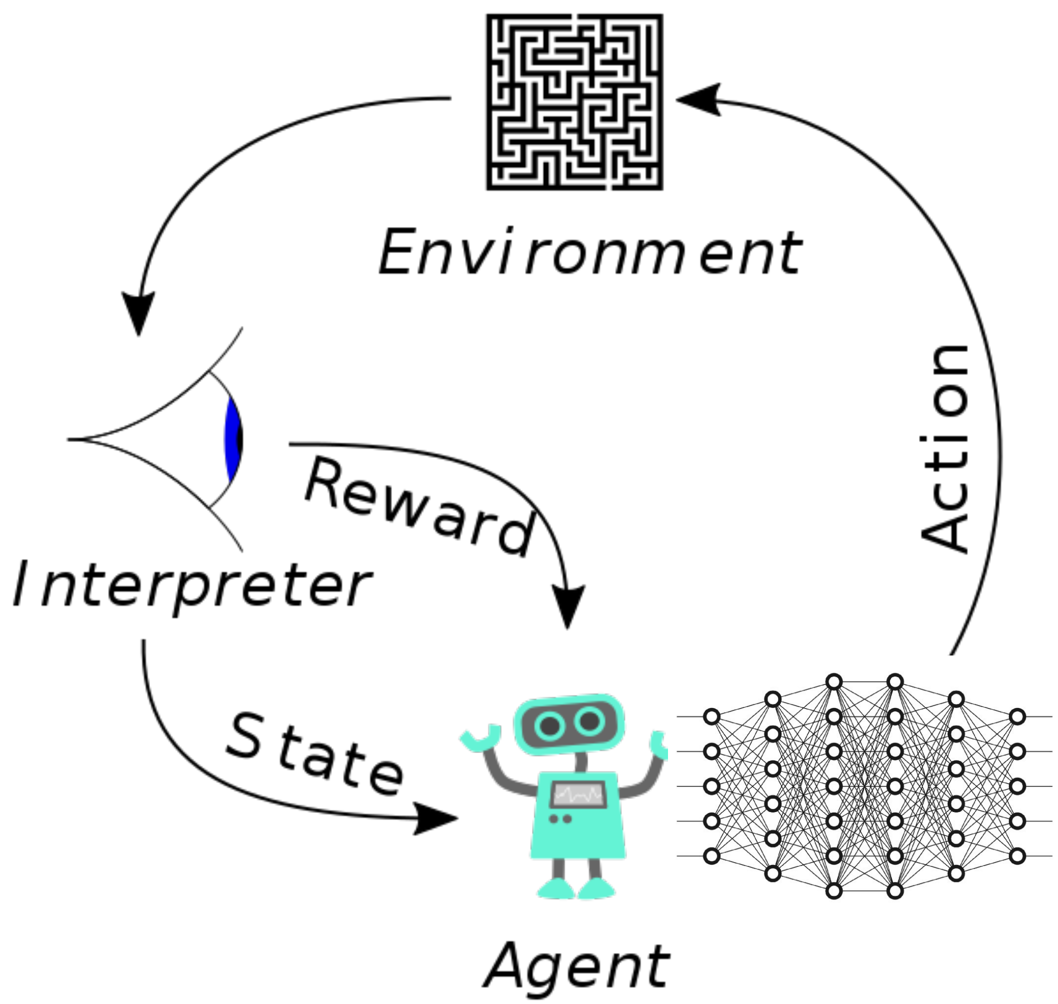




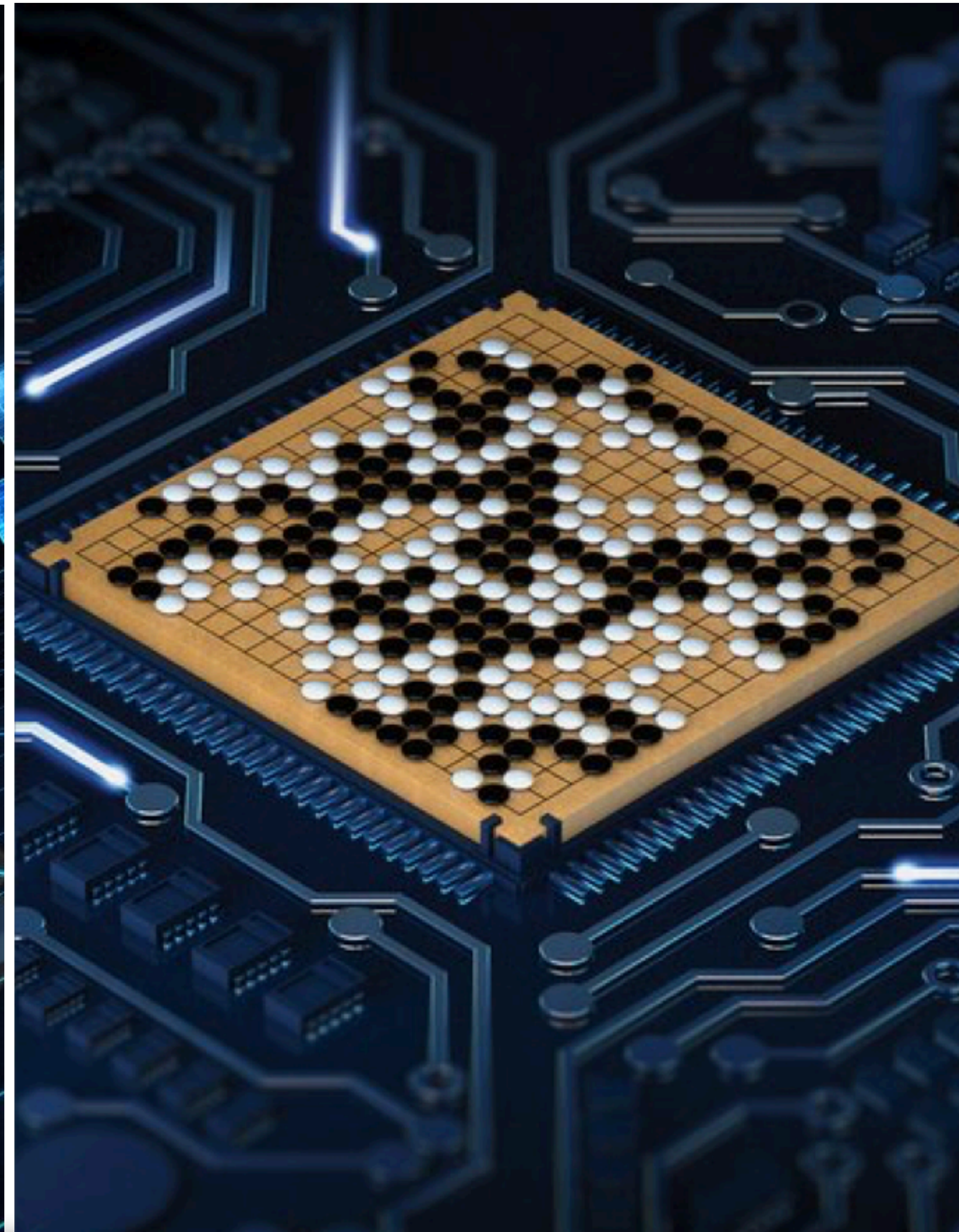
Reinforcement Learning



Deep Neural Networks



Deep Reinforcement Learning



Human-level control through deep reinforcement learning
https://www.nature.com/articles/nature14236

Mastering the game of Go without human knowledge
https://www.nature.com/articles/nature24270

Wayve: Learning to drive in a day

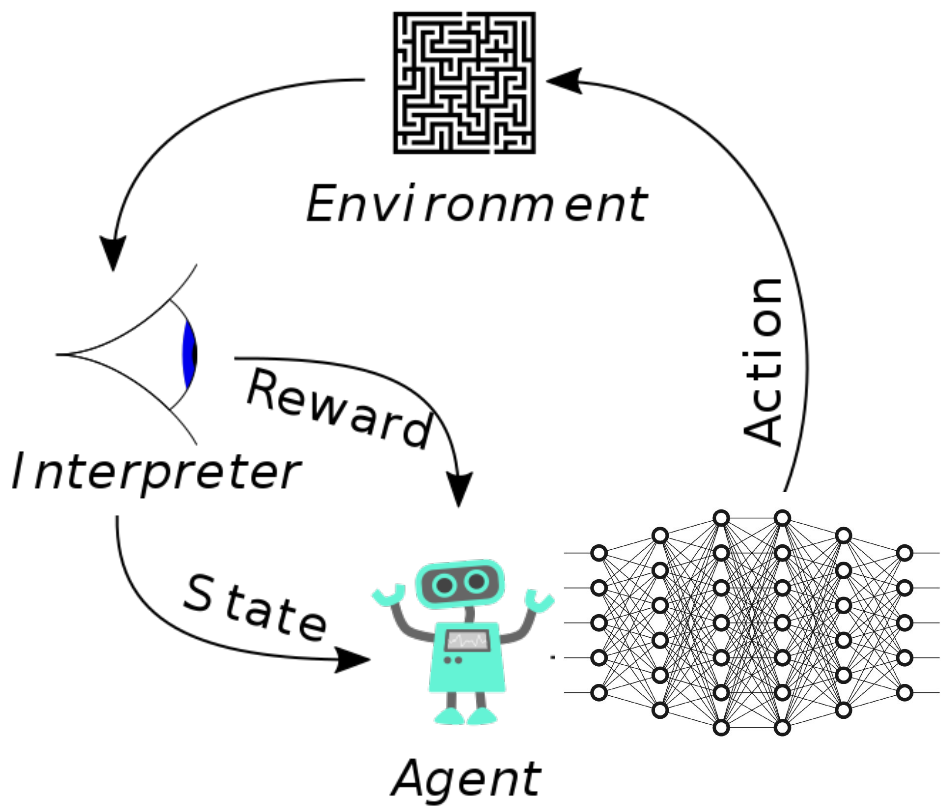
https://www.youtube.com/watch?v=eRwTbRtnT1I

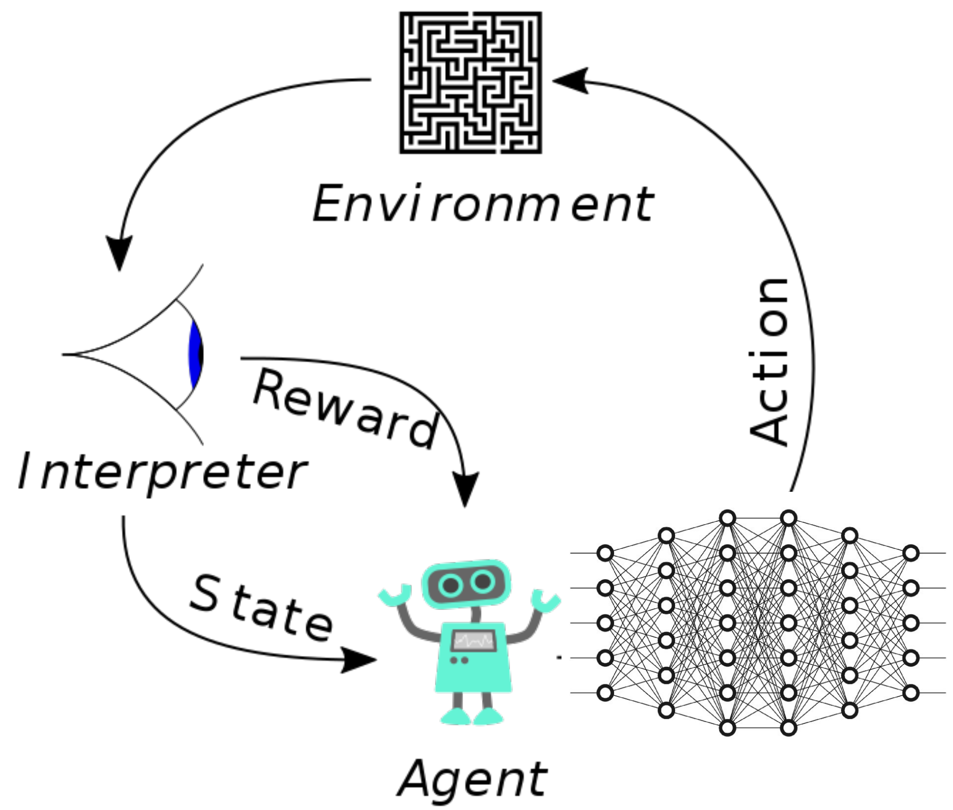
FANUC: Bin-Picking Robots.

https://www.youtube.com/watch?v=ydh_AdWZflA

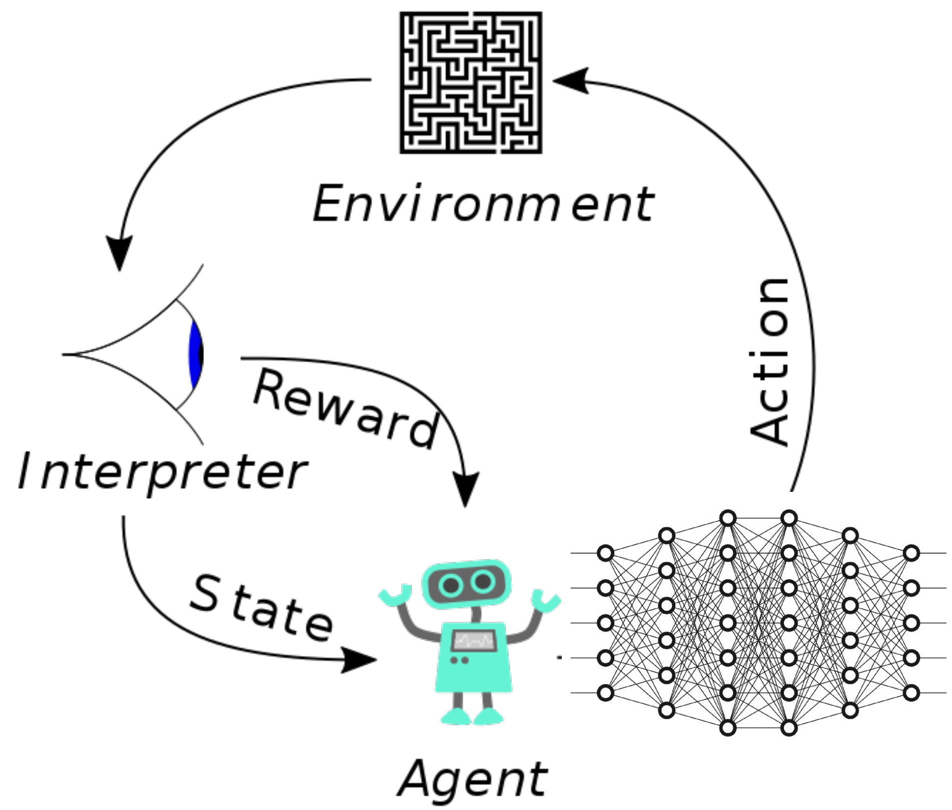
Chip Placement with Deep Reinforcement Learning

https://www.nature.com/articles/nature24270


Environment
Interpreter
Reward
Action
State
Agent

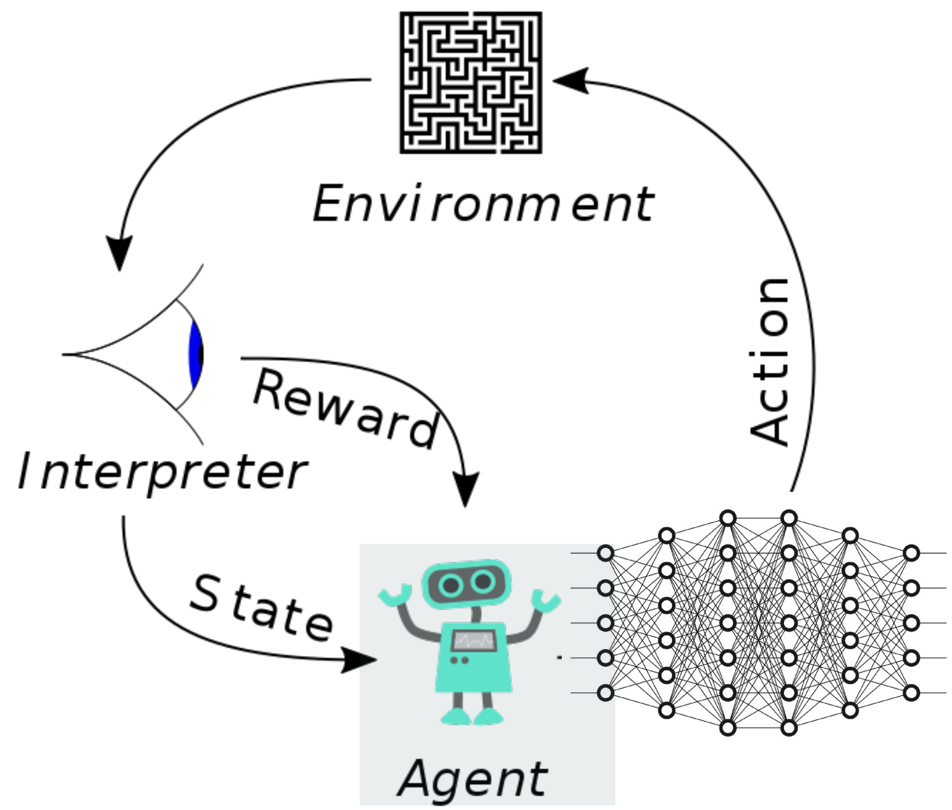


Program Synthesis
(Episodic RL)



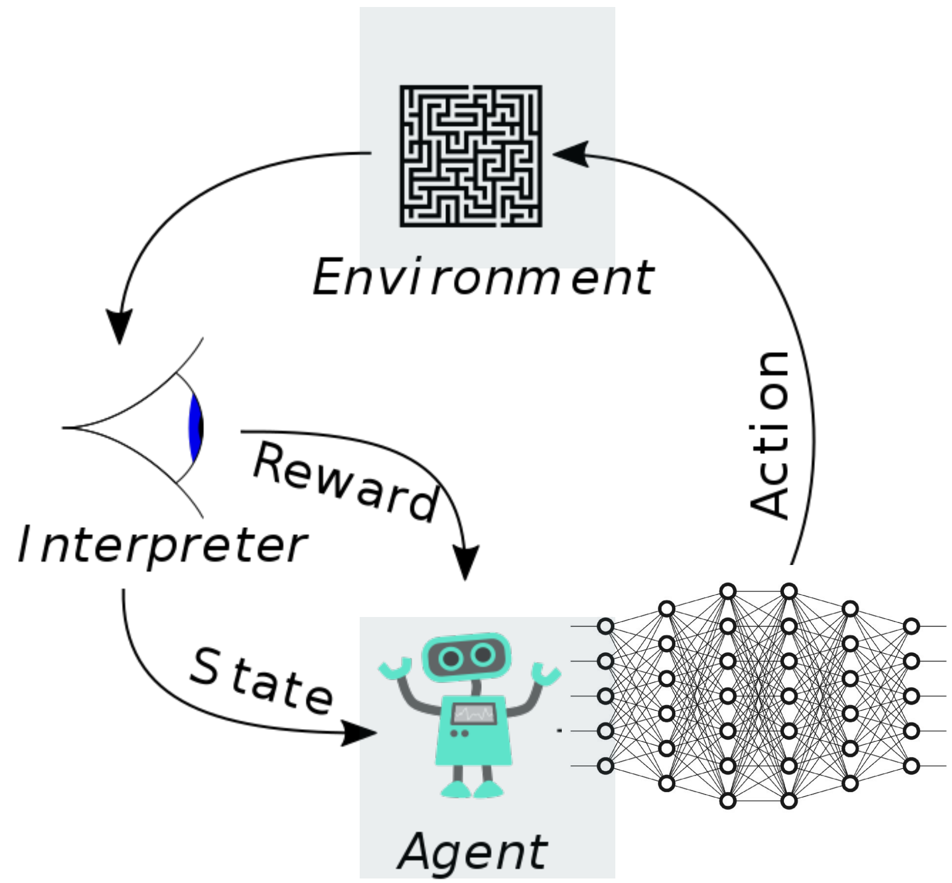
Program Synthesis
(Episodic RL)

Adaptive Programs
(Continual RL)



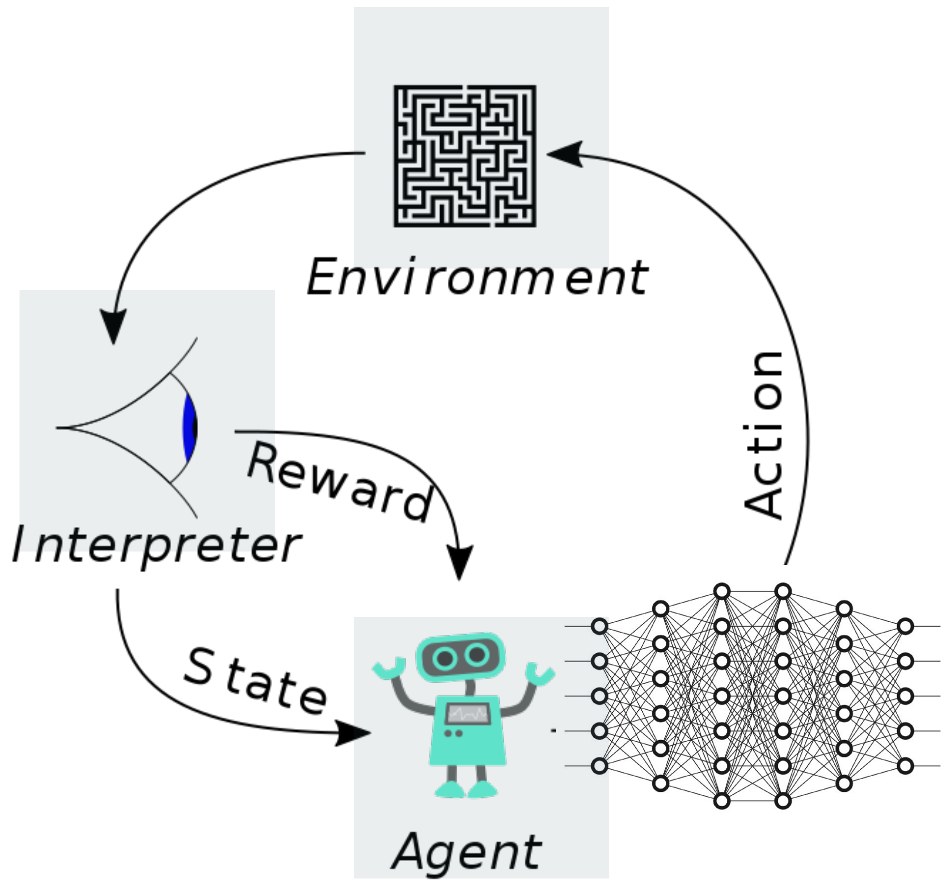
Program Synthesis
(Episodic RL)

Adaptive Programs
(Continual RL)



Program Synthesis
(Episodic RL)

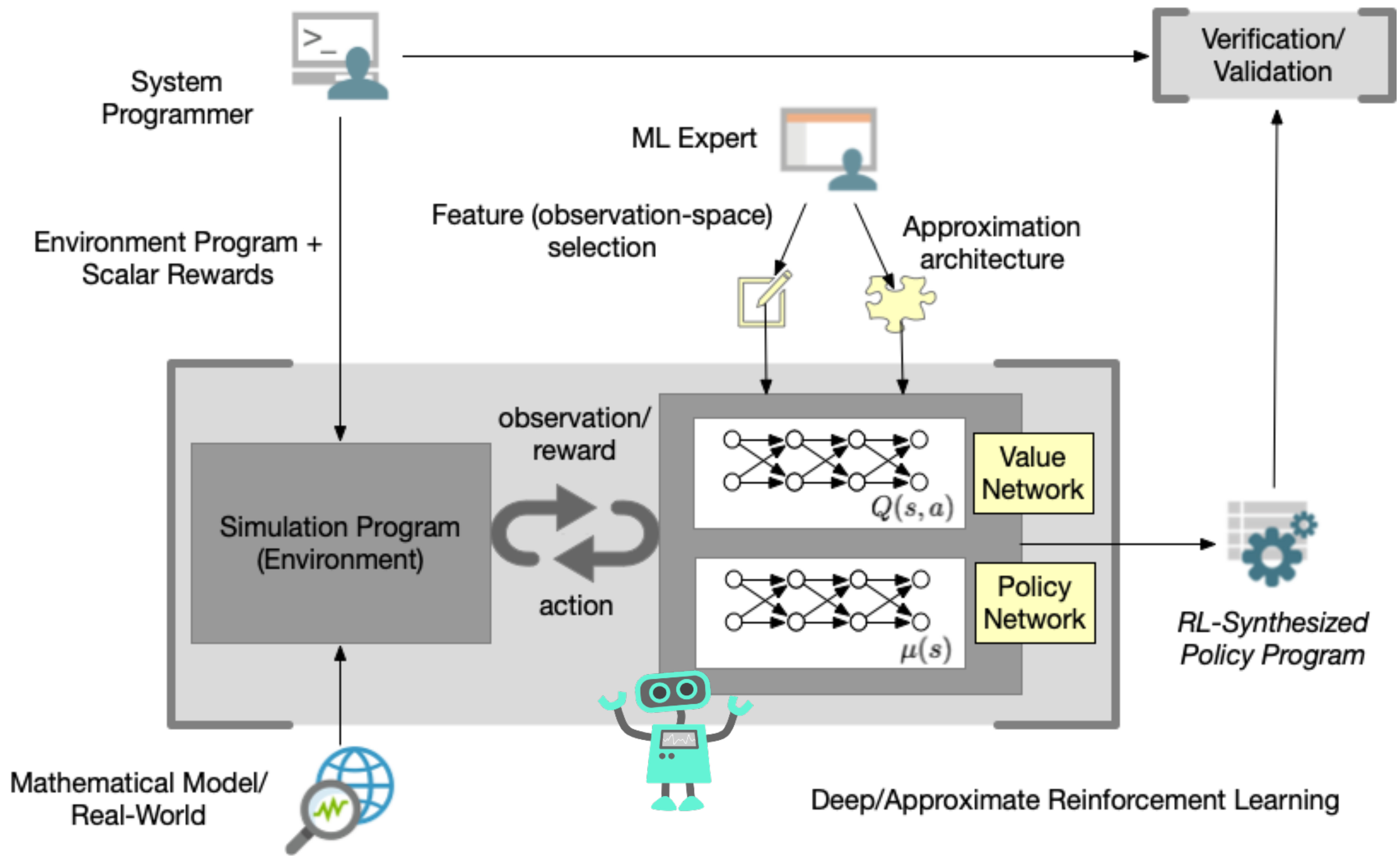
Adaptive Programs
(Continual RL)

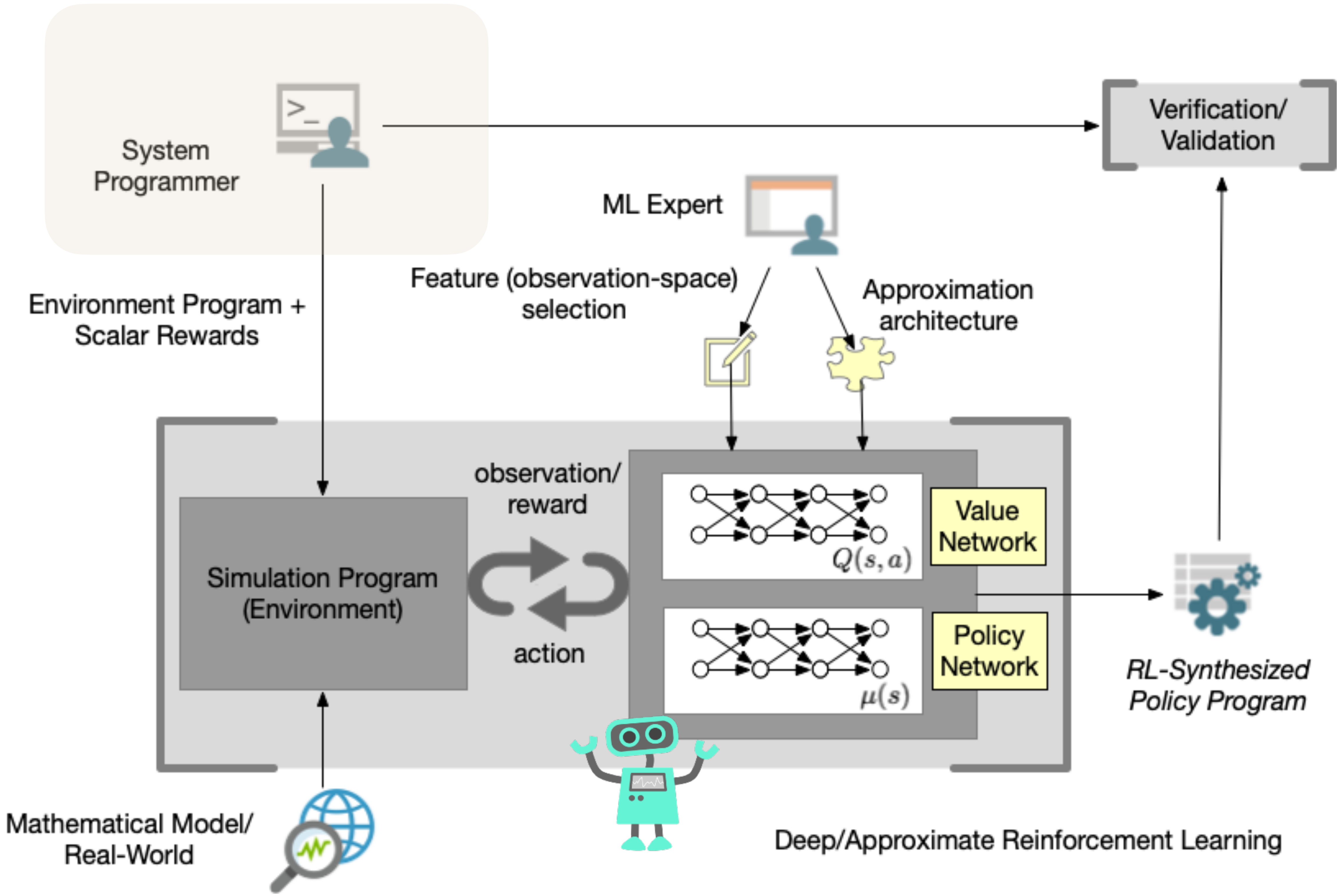


Program Synthesis
(Episodic RL)

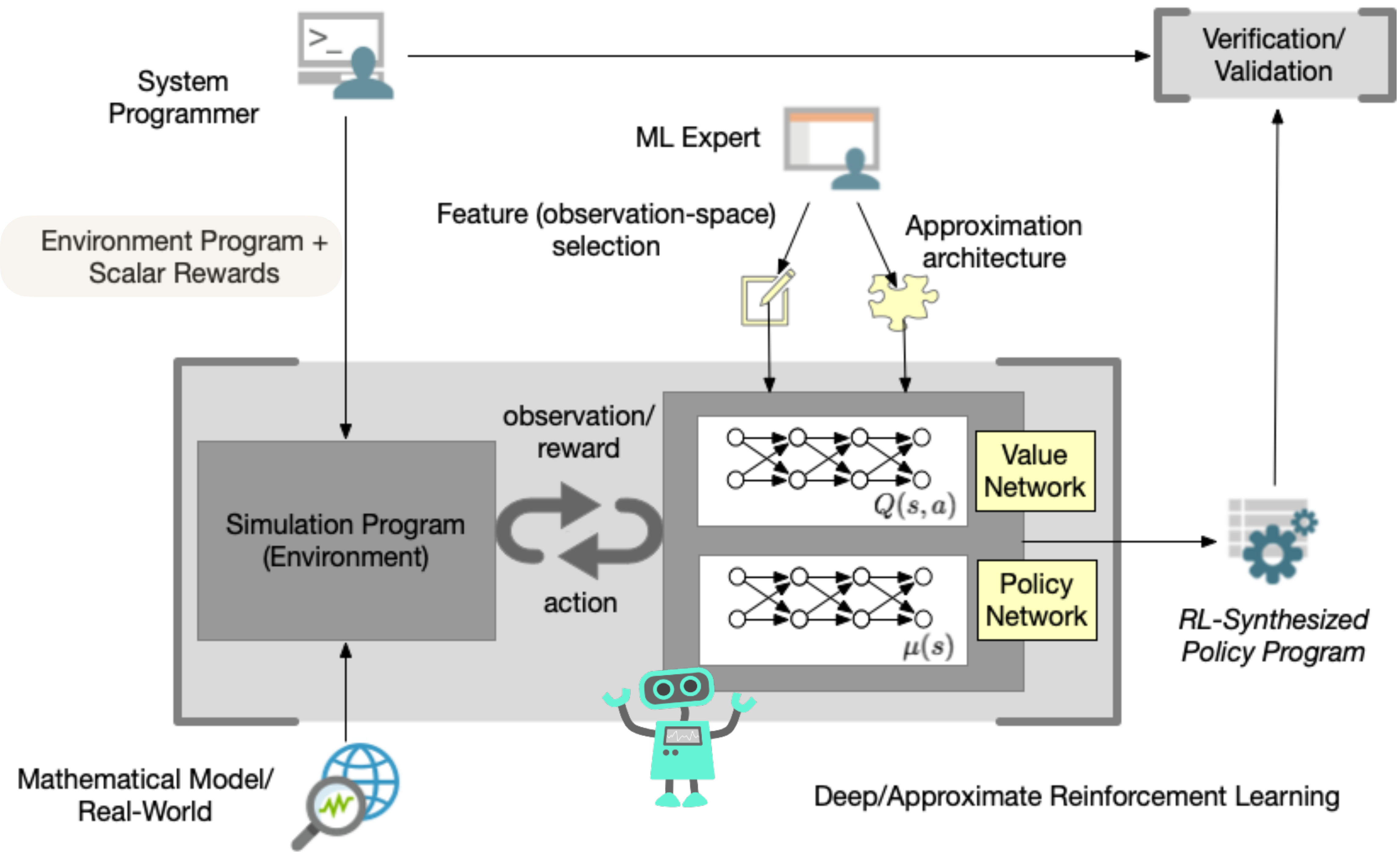
Adaptive Programs
(Continual RL)

Programmatic Reinforcement Learning

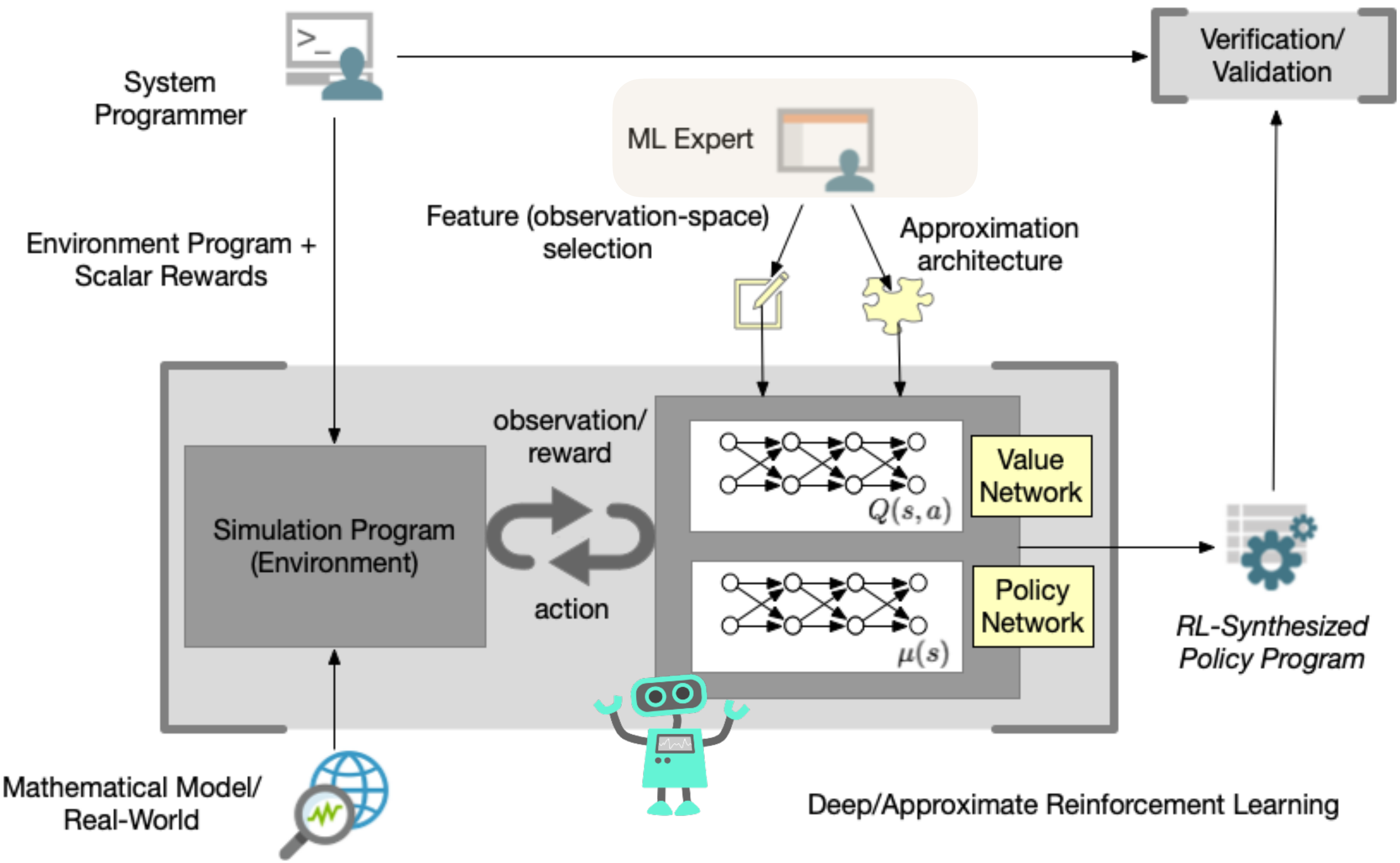

System Programmer
Verification/ Validation
ML Expert
Environment Program + Scalar Rewards
Feature (observation-space) selection
Approximation architecture
observation/ reward
Simulation Program (Environment)
action
Value Network
$Q(s, a)$
Policy Network
$\mu(s)$
RL-Synthesized Policy Program
Mathematical Model/ Real-World
Deep/Approximate Reinforcement Learning



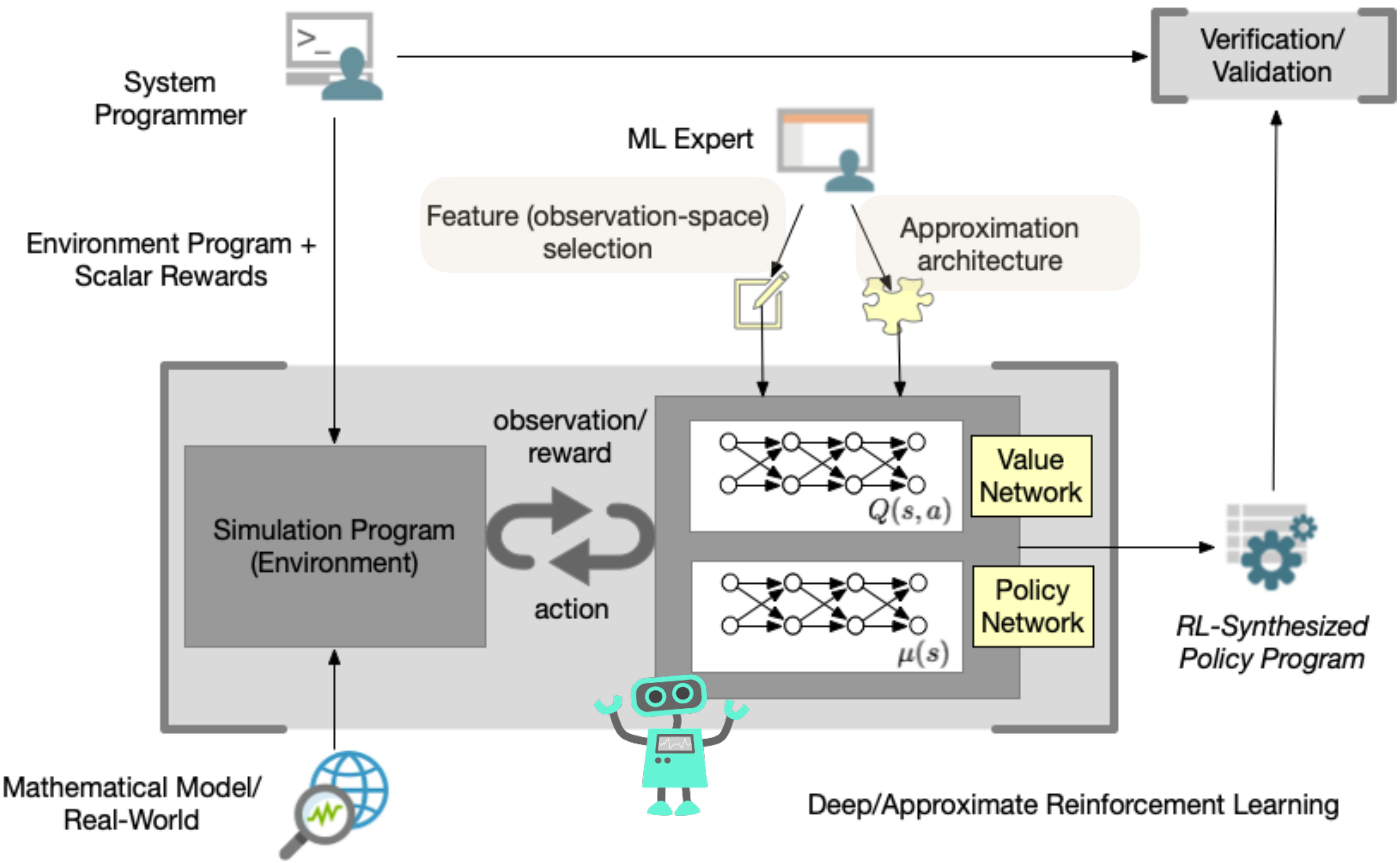
System
Programmer
Verification/
Validation
ML Expert
Environment Program +
Scalar Rewards
Feature (observation-space)
selection
Approximation
architecture
observation/
reward
Simulation Program
(Environment)
action
$Q(s,a)$
Value
Network
$\mu(s)$
Policy
Network
RL-Synthesized
Policy Program
Mathematical Model/
Real-World
Deep/Approximate Reinforcement Learning

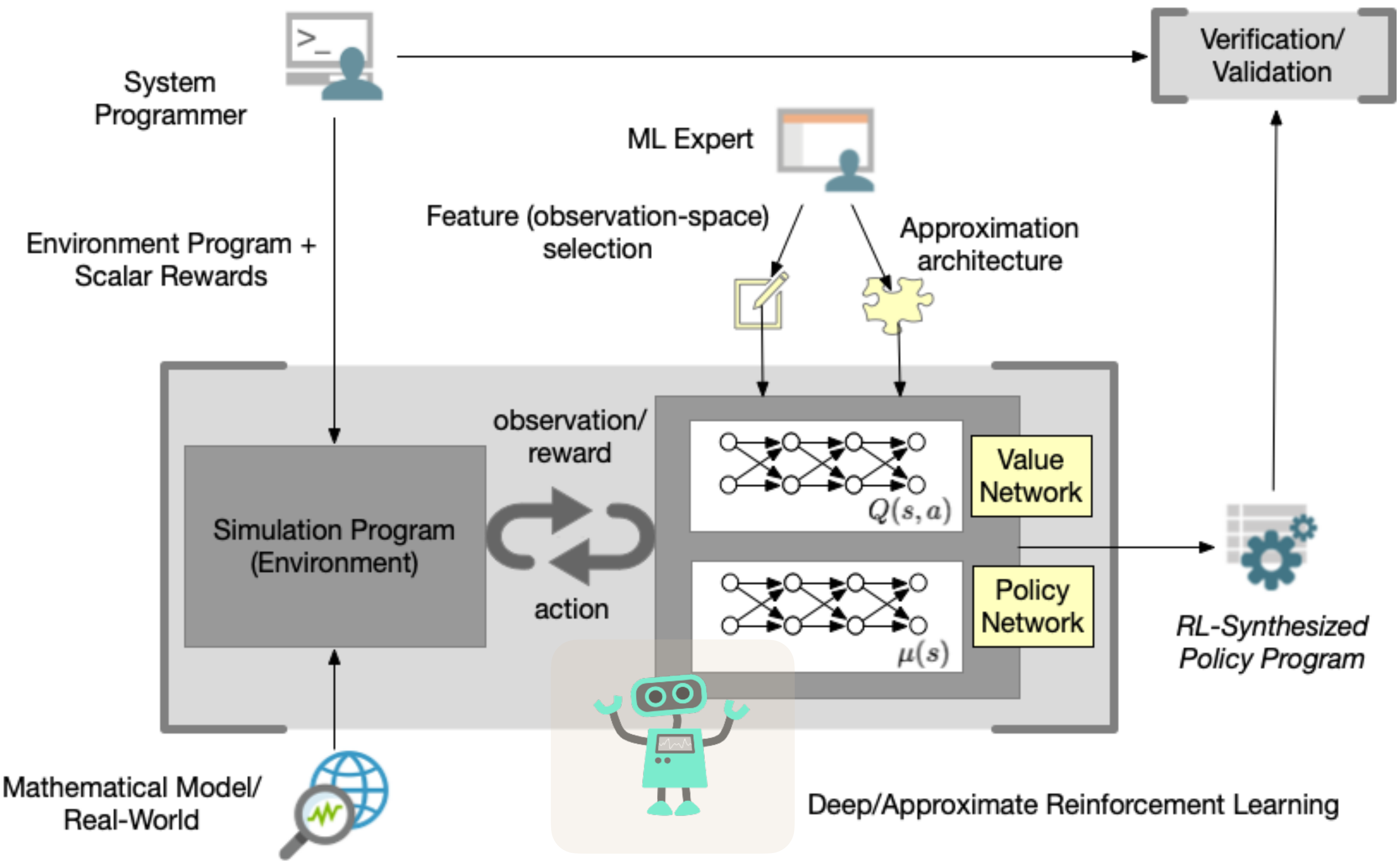


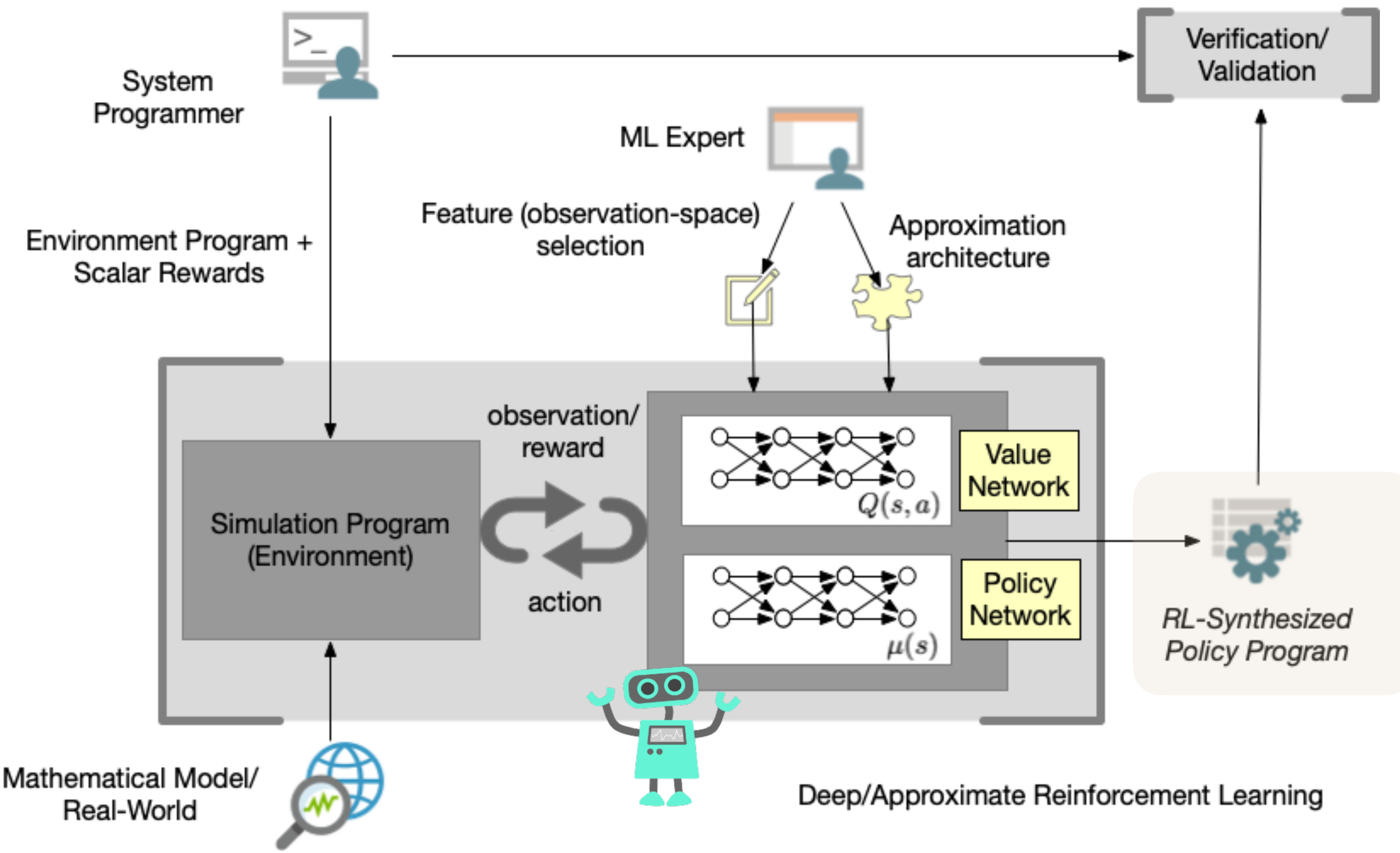
System Programmer
Verification/ Validation
ML Expert
Feature (observation-space) selection
Approximation architecture
Environment Program + Scalar Rewards
observation/ reward
Simulation Program (Environment)
action
Value Network
$Q(s, a)$
Policy Network
$\mu(s)$
RL-Synthesized Policy Program
Mathematical Model/ Real-World
Deep/Approximate Reinforcement Learning



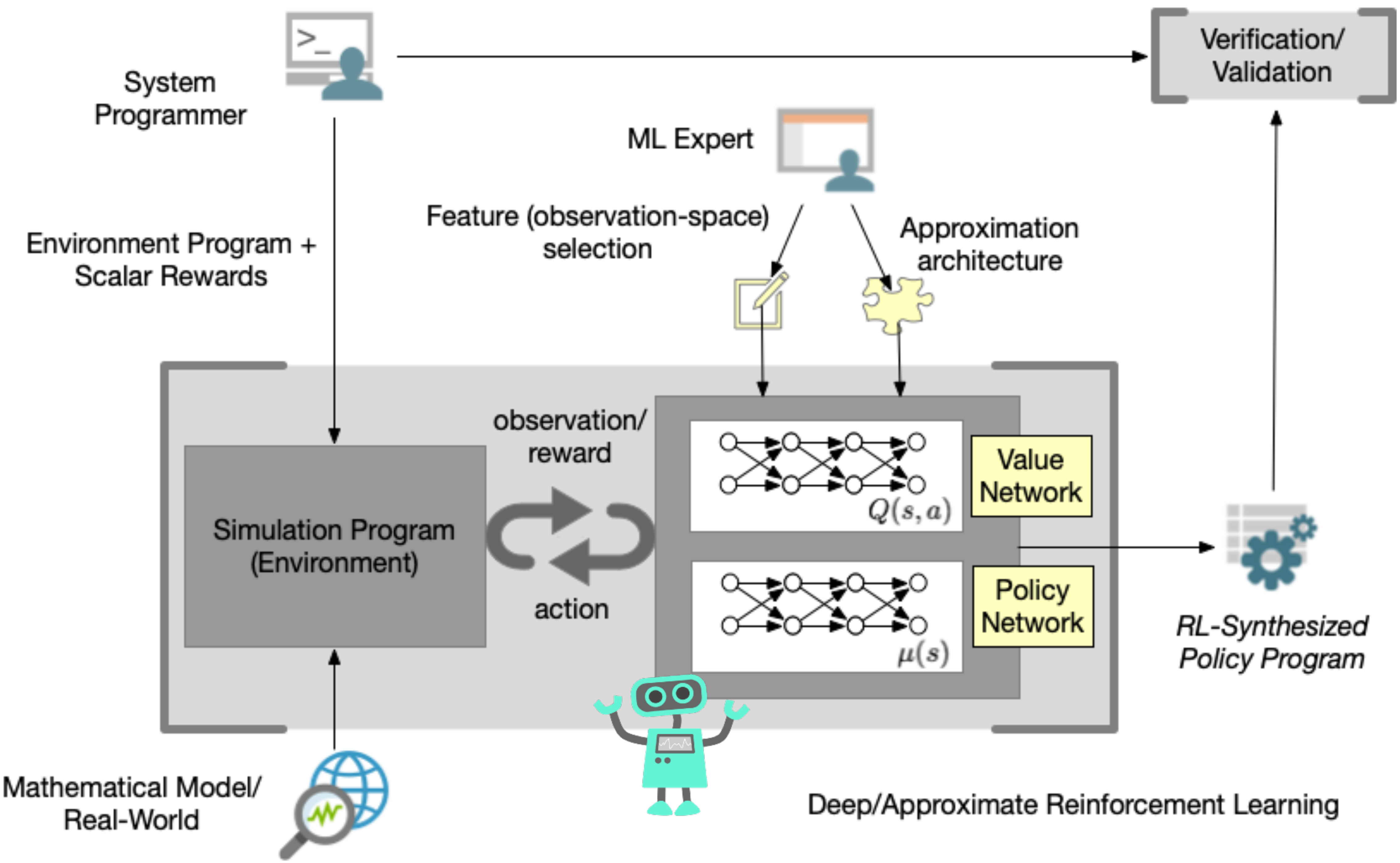
System Programmer
Verification/ Validation
ML Expert
Environment Program + Scalar Rewards
Feature (observation-space) selection
Approximation architecture
observation/ reward
action
Simulation Program (Environment)
Value Network
Policy Network
$Q(s,a)$
$\mu(s)$
RL-Synthesized Policy Program
Mathematical Model/ Real-World
Deep/Approximate Reinforcement Learning


System Programmer
Verification/ Validation
ML Expert
Feature (observation-space) selection
Approximation architecture
Environment Program + Scalar Rewards
observation/ reward
action
Simulation Program (Environment)
$Q(s,a)$
Value Network
$\mu(s)$
Policy Network
RL-Synthesized Policy Program
Mathematical Model/ Real-World
Deep/Approximate Reinforcement Learning


System Programmer
Verification/ Validation
ML Expert
Environment Program + Scalar Rewards
Feature (observation-space) selection
Approximation architecture
observation/ reward
action
Simulation Program (Environment)
Value Network
$Q(s, a)$
Policy Network
$\mu(s)$
RL-Synthesized Policy Program
Mathematical Model/ Real-World
Deep/Approximate Reinforcement Learning

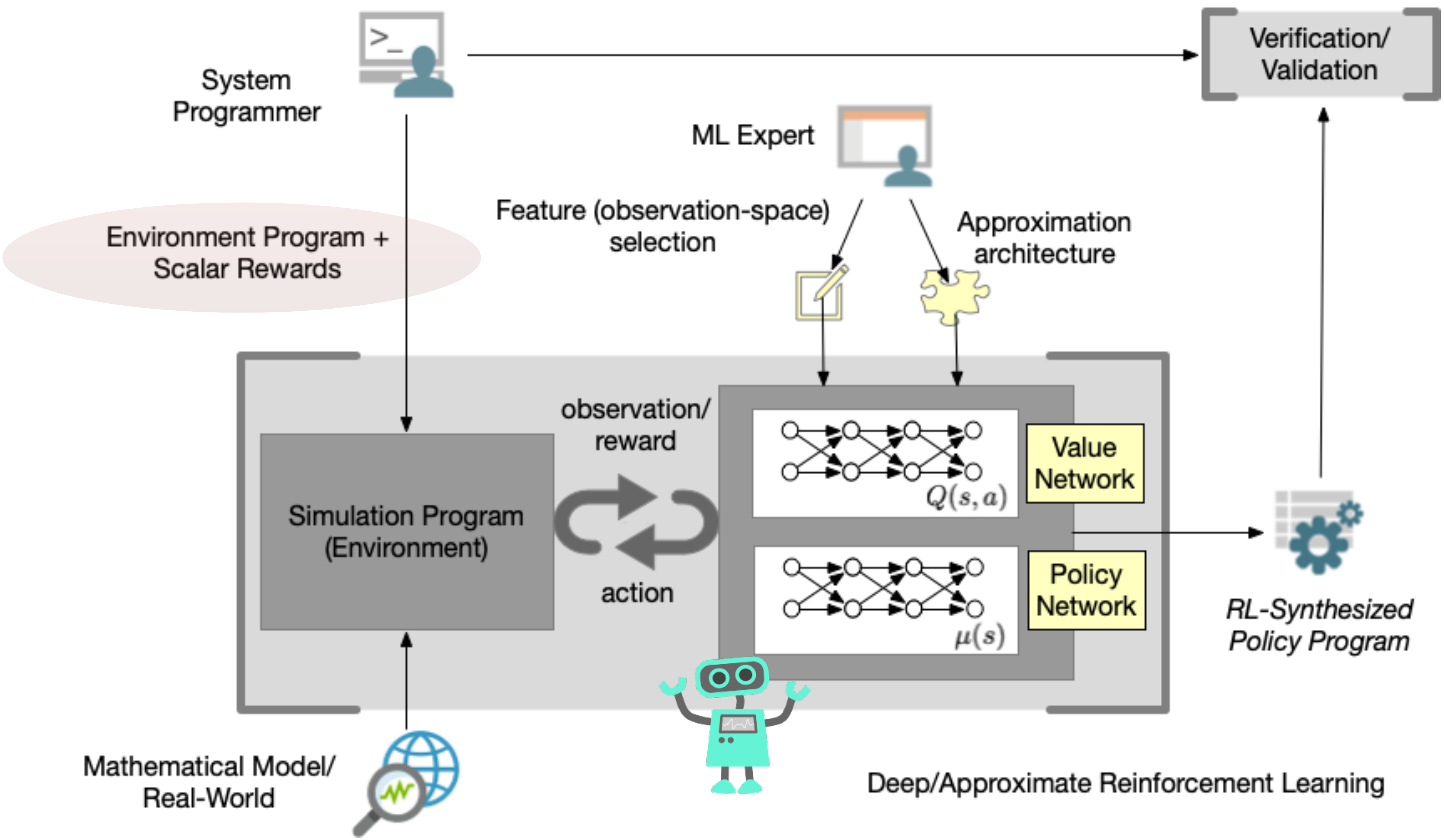

System Programmer
Verification/ Validation
ML Expert
Environment Program + Scalar Rewards
Feature (observation-space) selection
Approximation architecture
observation/ reward
Simulation Program (Environment)
action
$Q(s,a)$
Value Network
$\mu(s)$
Policy Network
RL-Synthesized Policy Program
Mathematical Model/ Real-World
Deep/Approximate Reinforcement Learning

Pain Points



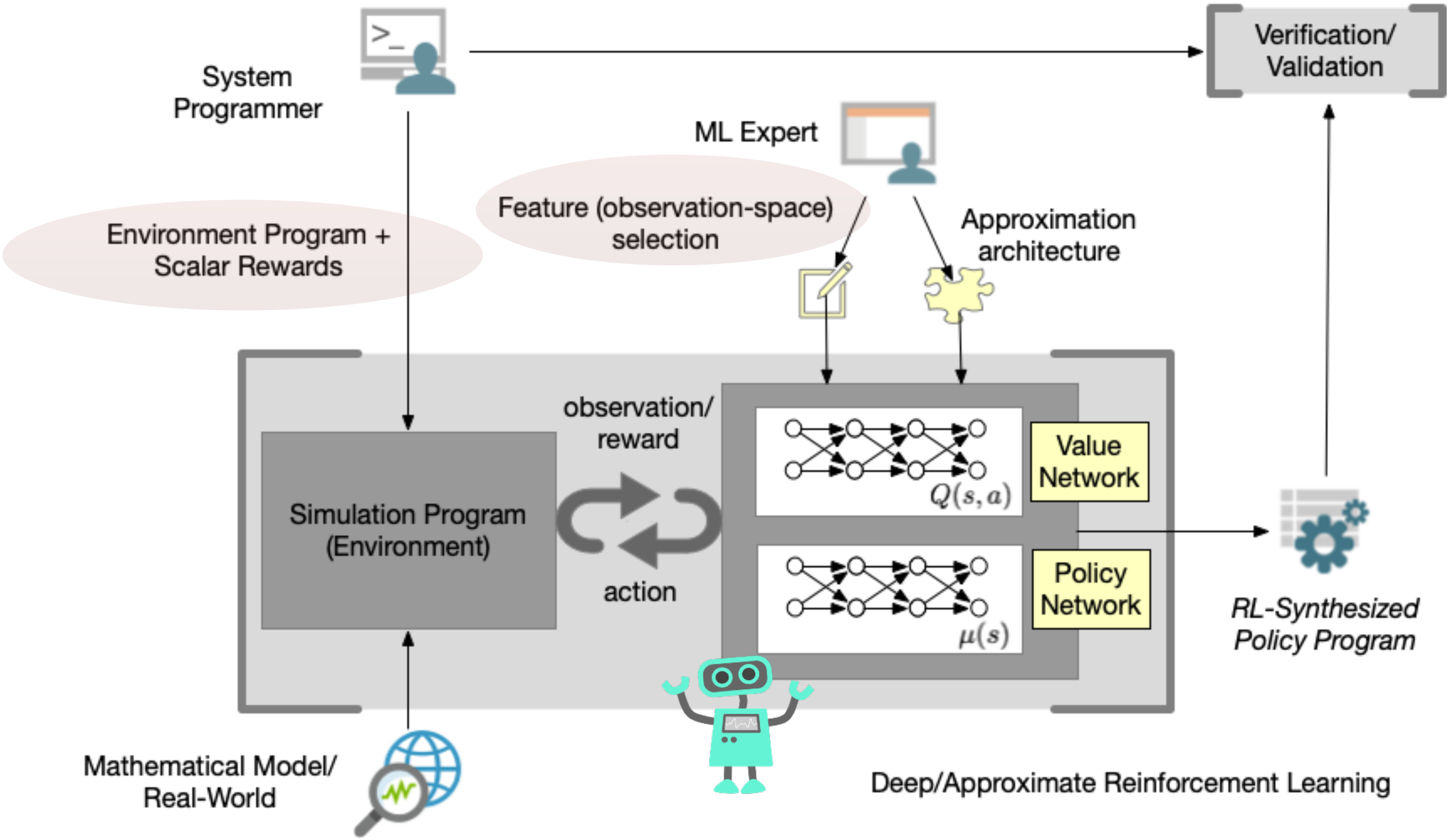
Usability

Pain Points



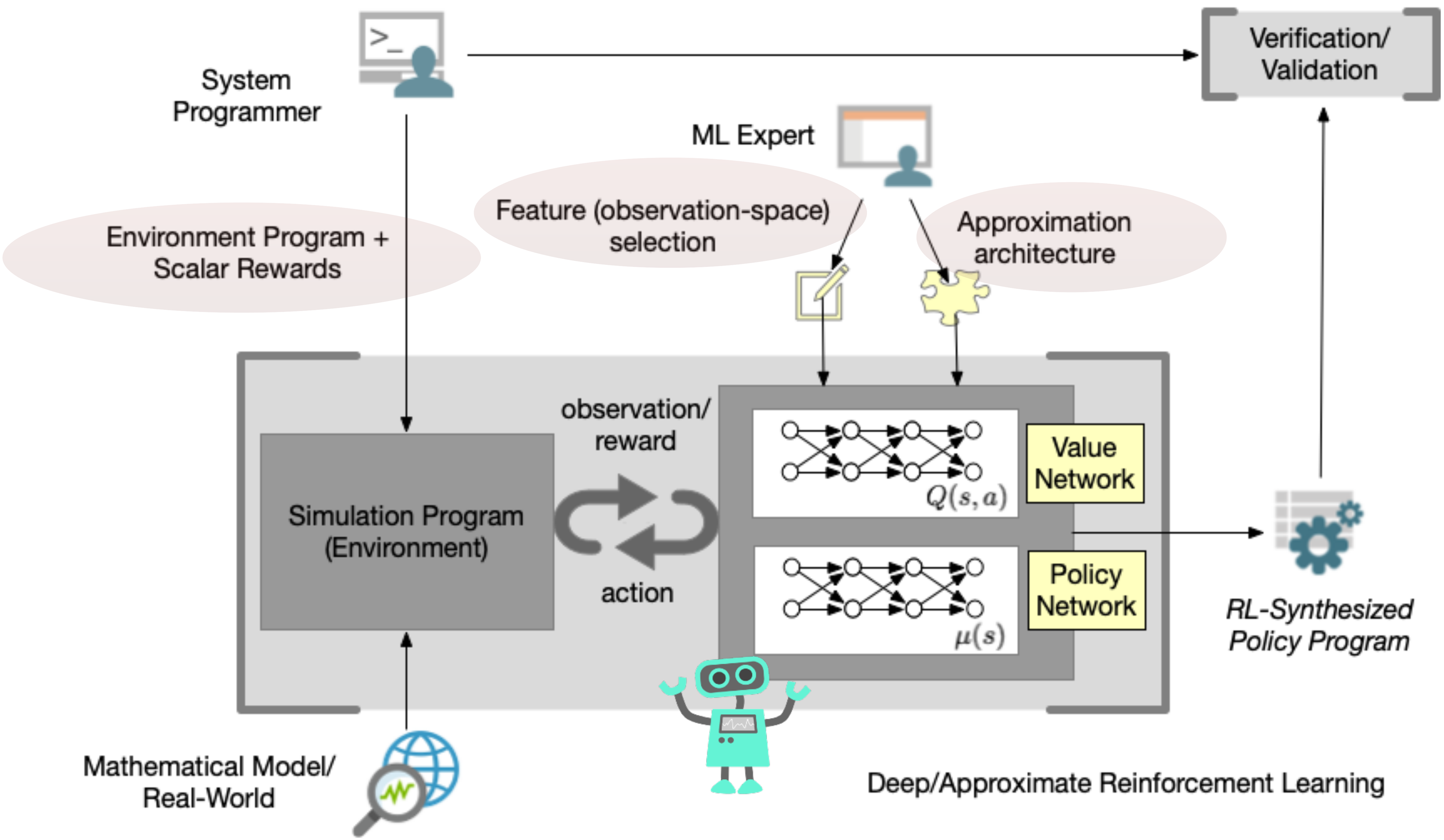
Usability

Pain Points



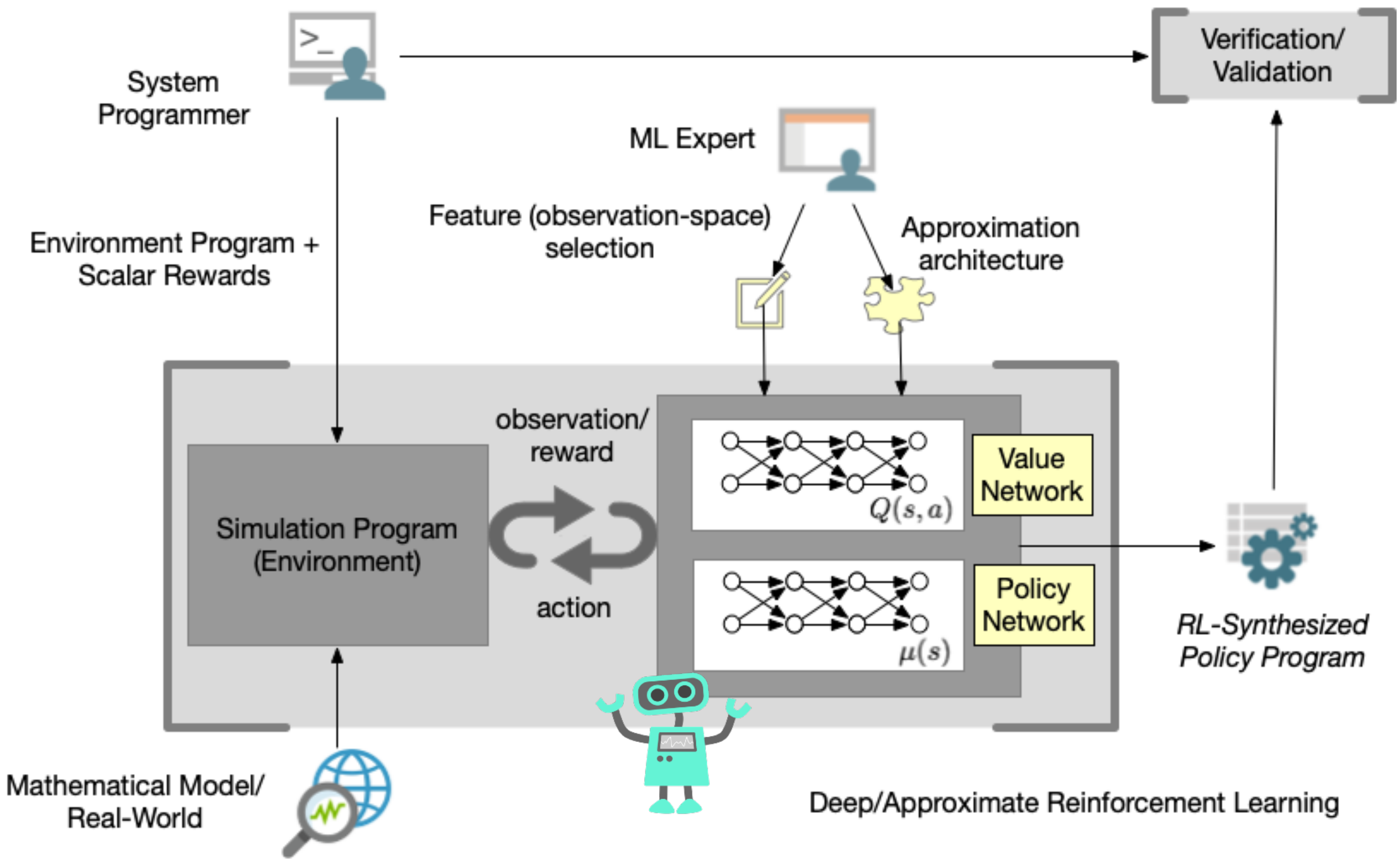
Usability

Pain Points



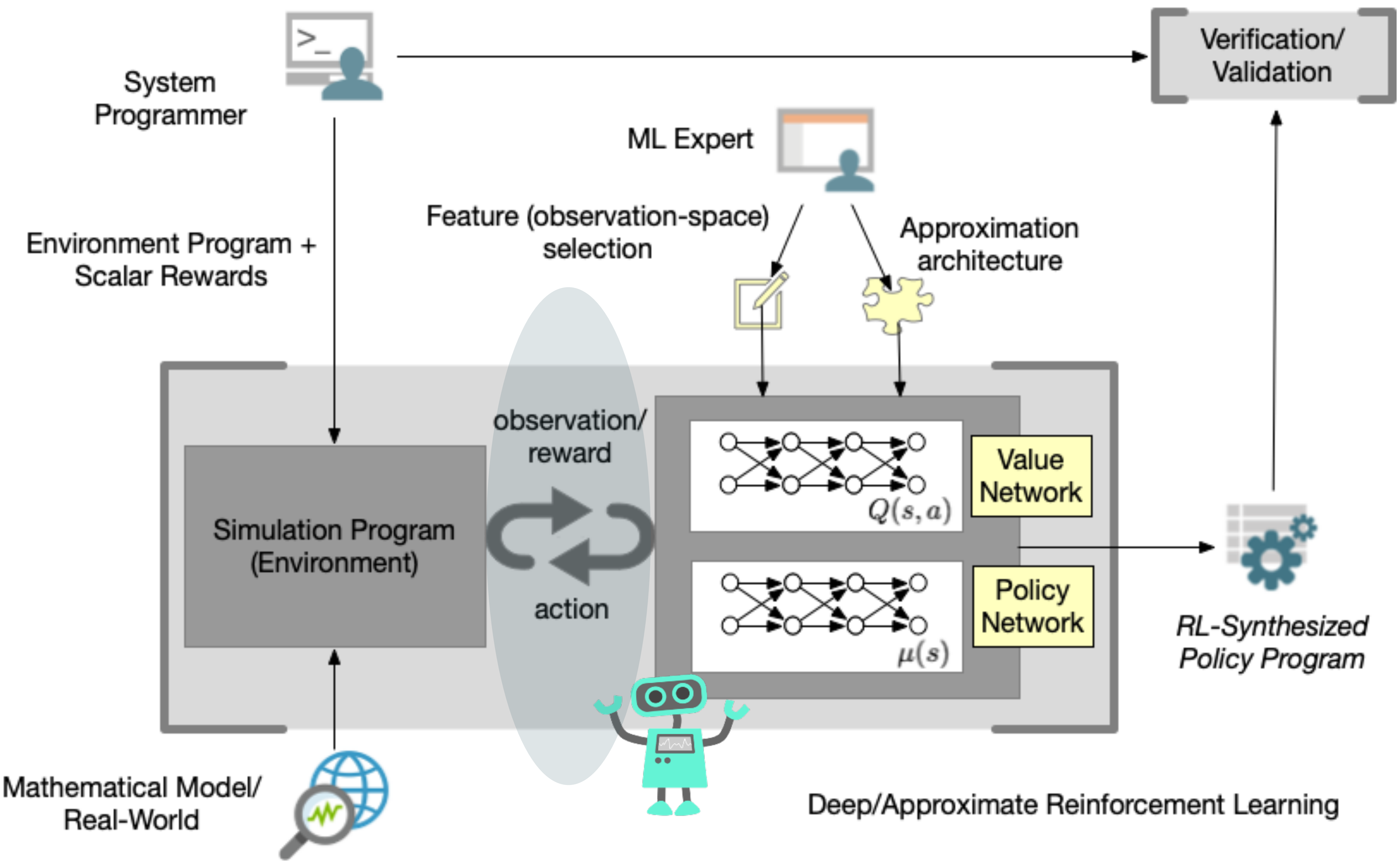
Usability

Pain Points



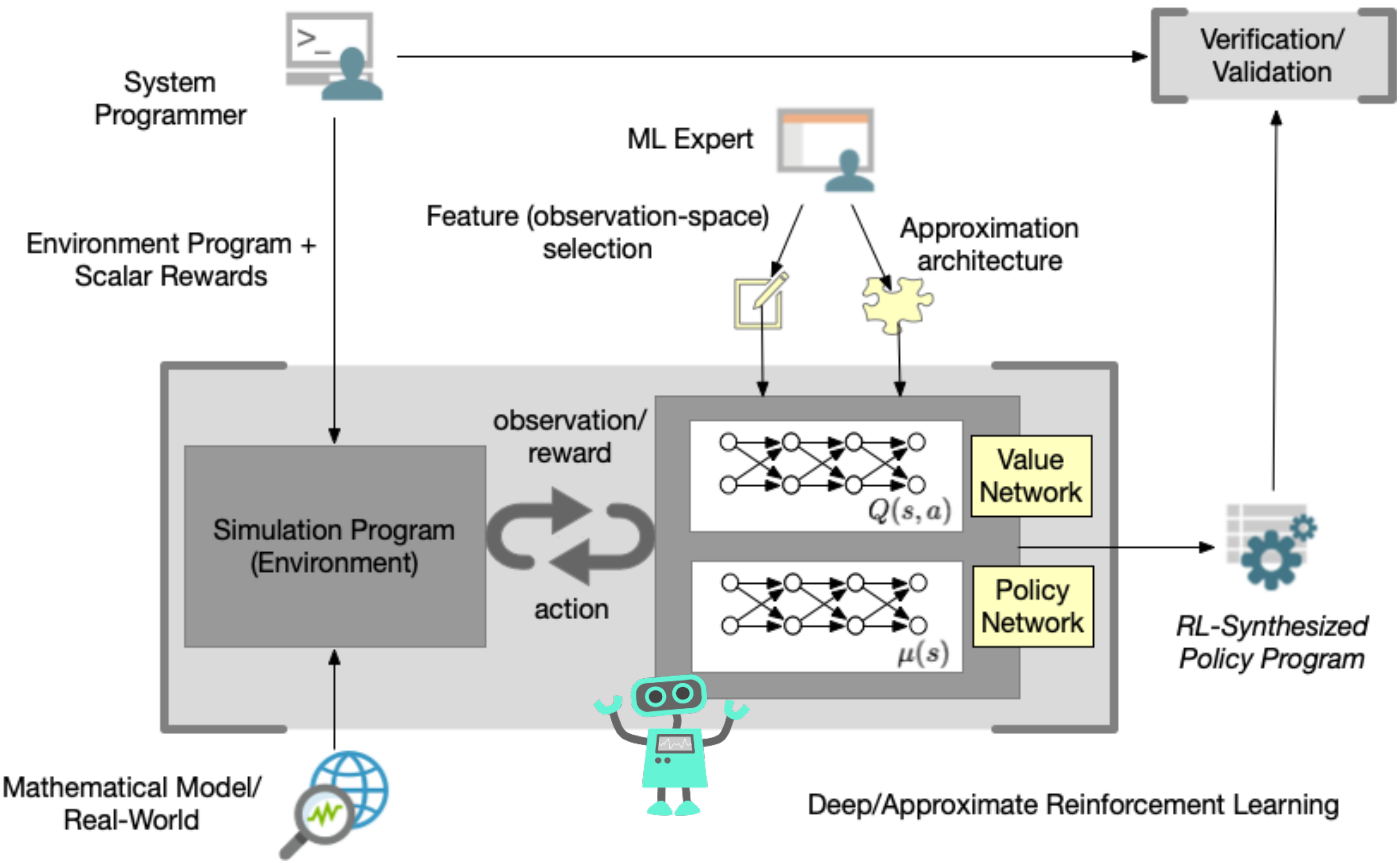
Scalability

Pain Points



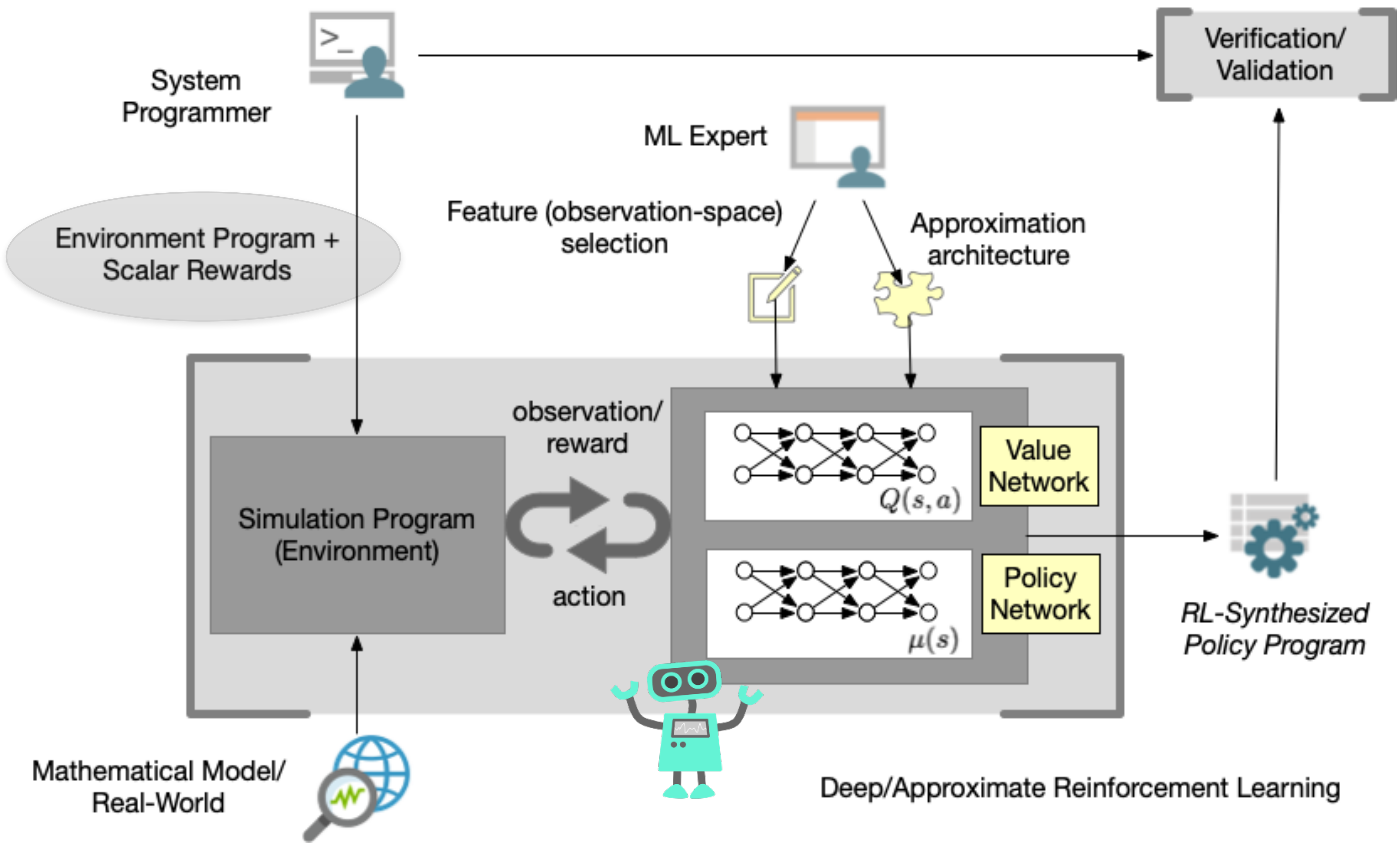
Scalability

Pain Points



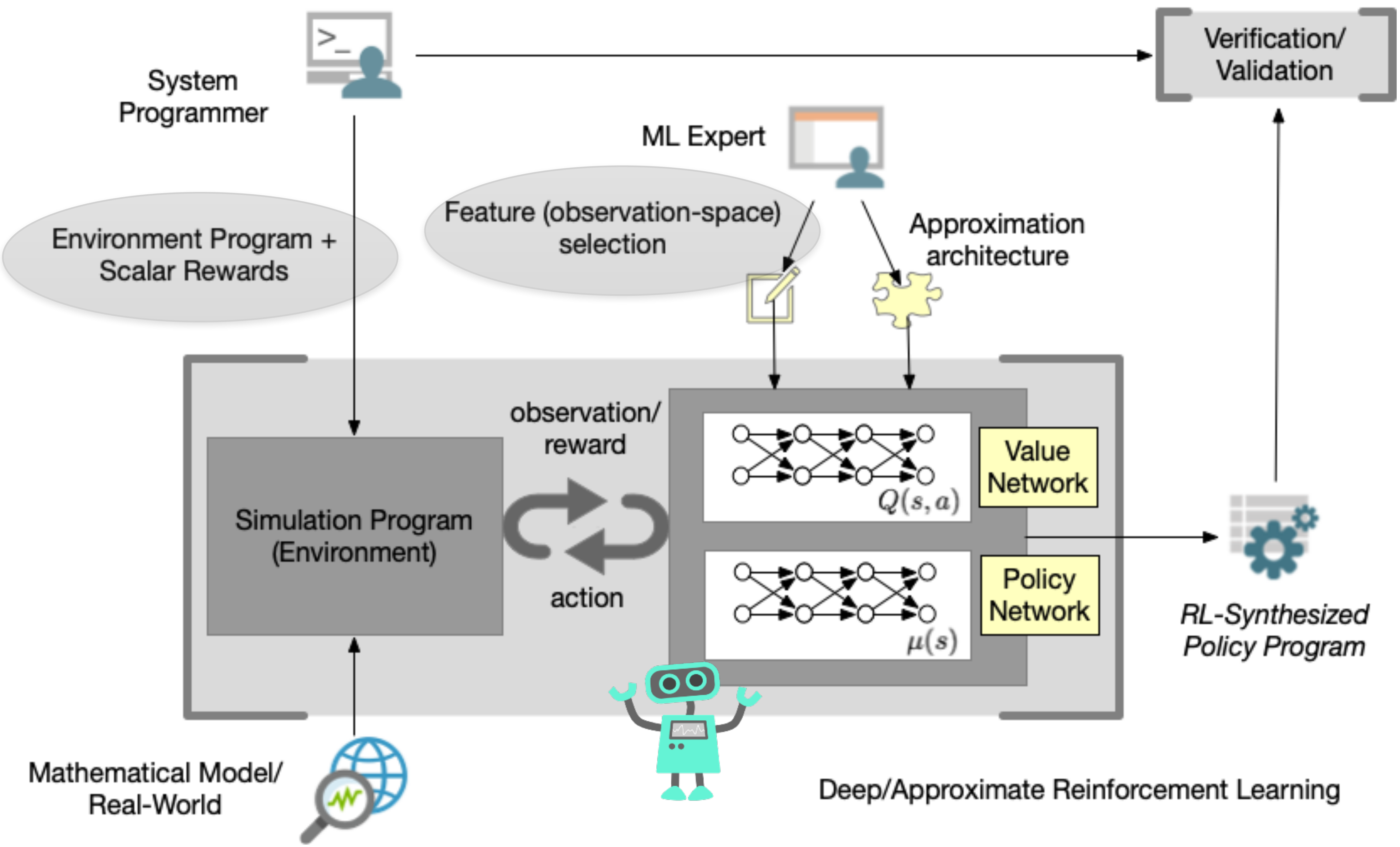
Trust

Pain Points



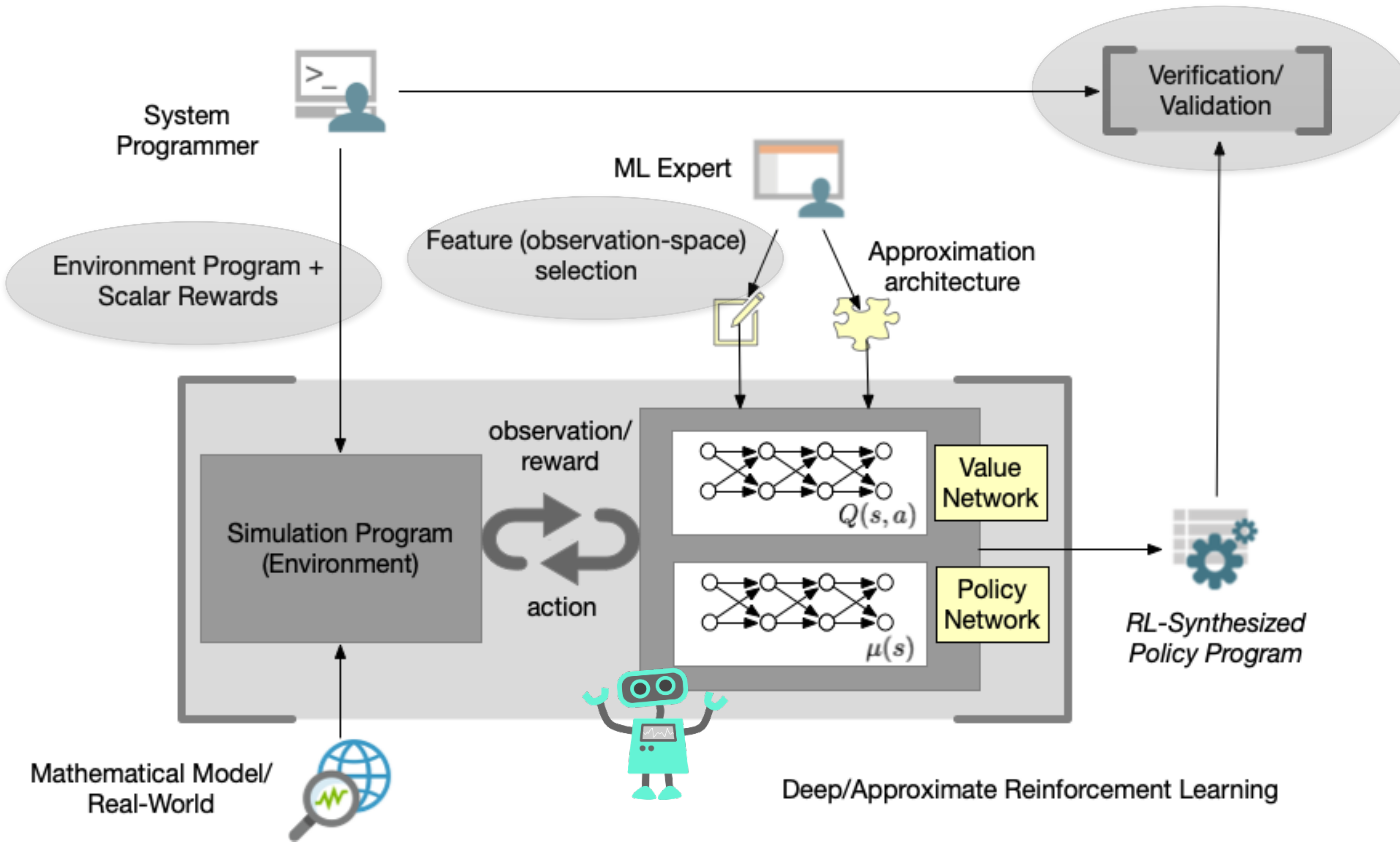
Trust

Pain Points



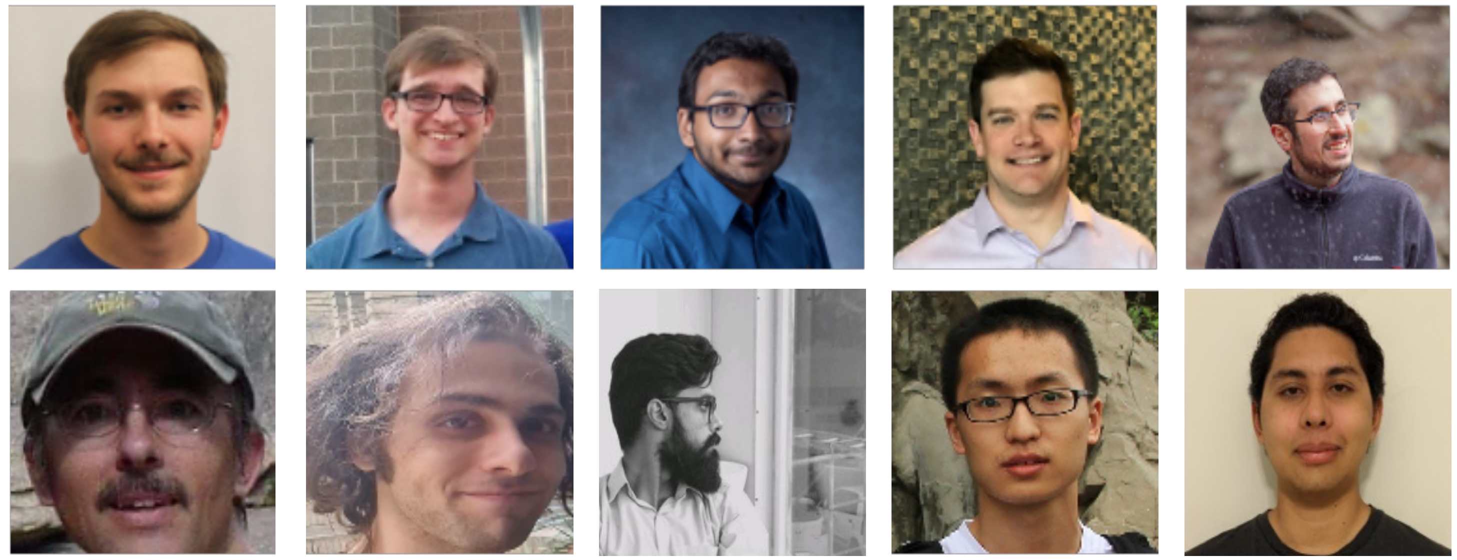
Trust

Pain Points



Trust

*Principled methodologies and powerful tools to improve the **usability**, **scalability**, and **trustworthiness** of programmatic reinforcement learning.*



*Principled methodologies and powerful tools to improve the **usability**, **scalability**, and **trustworthiness** of programmatic reinforcement learning.*

CUPLV
Programming Languages and Verification
University of Colorado Boulder

Improving Usability




I. Programming Language of Rewards

Ernst Moritz Hahn, Mateo Perez, Sven Schewe, Fabio Somenzi, Ashutosh Trivedi, Dominik Wojtczak:
Omega-Regular Objectives in Model-Free Reinforcement Learning. TACAS (1) 2019: 395-412

Ernst Moritz Hahn, Mateo Perez, Sven Schewe, Fabio Somenzi, Ashutosh Trivedi, Dominik Wojtczak:
Model-Free Reinforcement Learning for Stochastic Parity Games. CONCUR 2020: 21:1-21:16

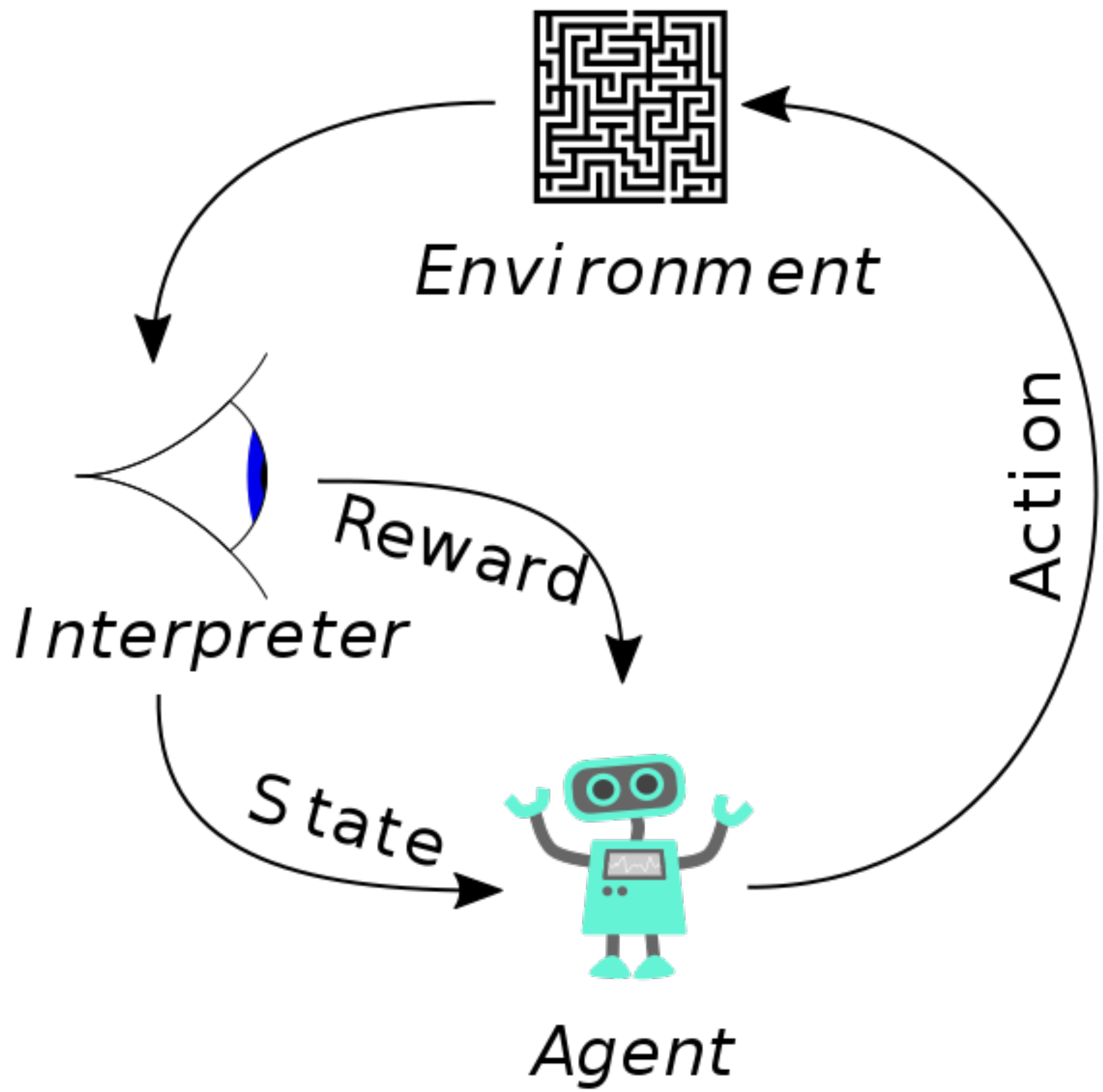
Kalyani Dole, Ashutosh Gupta, John Komp, Shankara Narayanan Krishna, Ashutosh Trivedi:
Event-Triggered and Time-Triggered Duration Calculus for Model-Free Reinforcement Learning. RTSS 2021: 240-252

Ernst Moritz Hahn, Mateo Perez, Sven Schewe, Fabio Somenzi, Ashutosh Trivedi, Dominik Wojtczak:
Model-Free Reinforcement Learning for Lexicographic Omega-Regular Objectives. FM 2021: 142-159

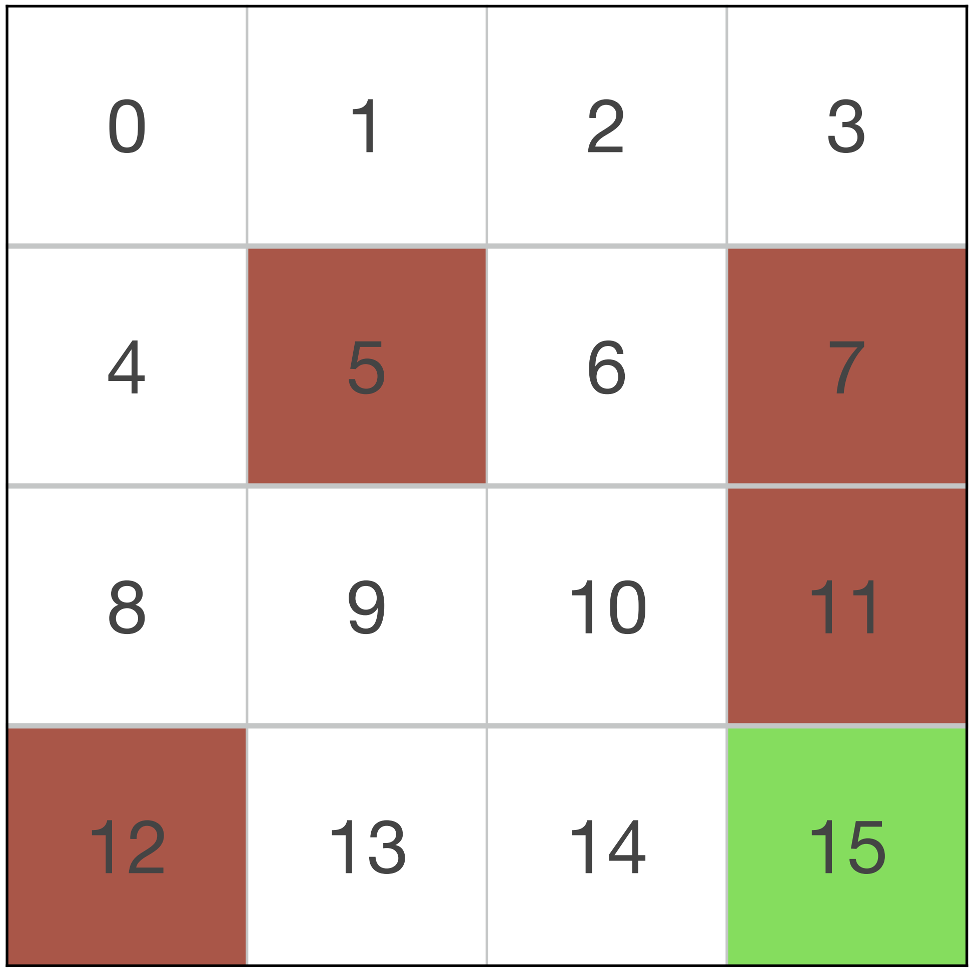
Milad Kazemi, Mateo Perez, Fabio Somenzi, Sadegh Soudjani, Ashutosh Trivedi, Alvaro Velasquez:
Translating Omega-Regular Specifications to Average Objectives for Model-Free Reinforcement Learning. AAMAS 2022: 732-741

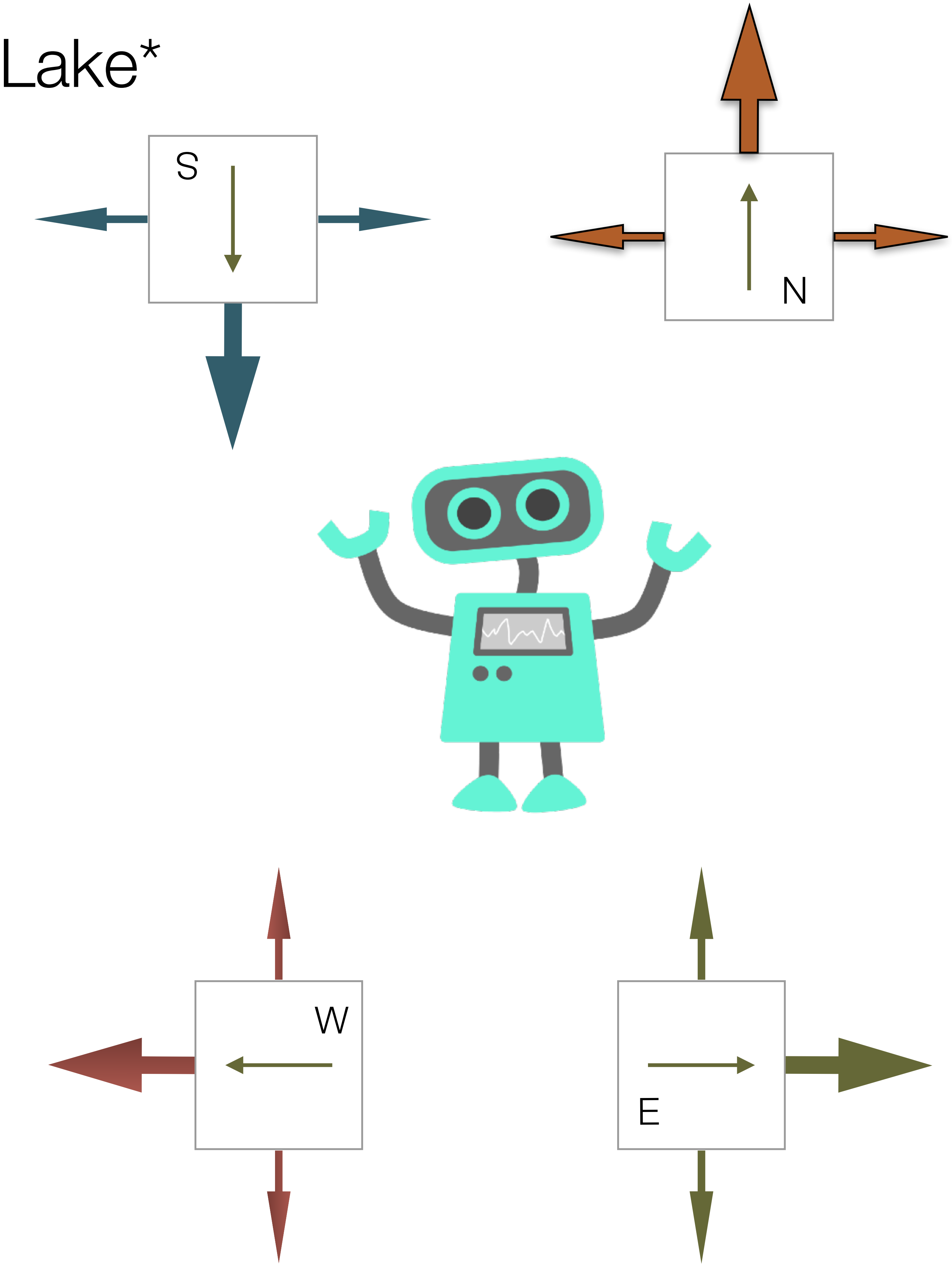
Taylor Dohmen, Noah Topper, George K. Atia, Andre Beckus, Ashutosh Trivedi, Alvaro Velasquez:
Inferring Probabilistic Reward Machines from Non-Markovian Reward Signals for Reinforcement Learning. ICAPS 2022: 574-582

What is Reinforcement Learning?

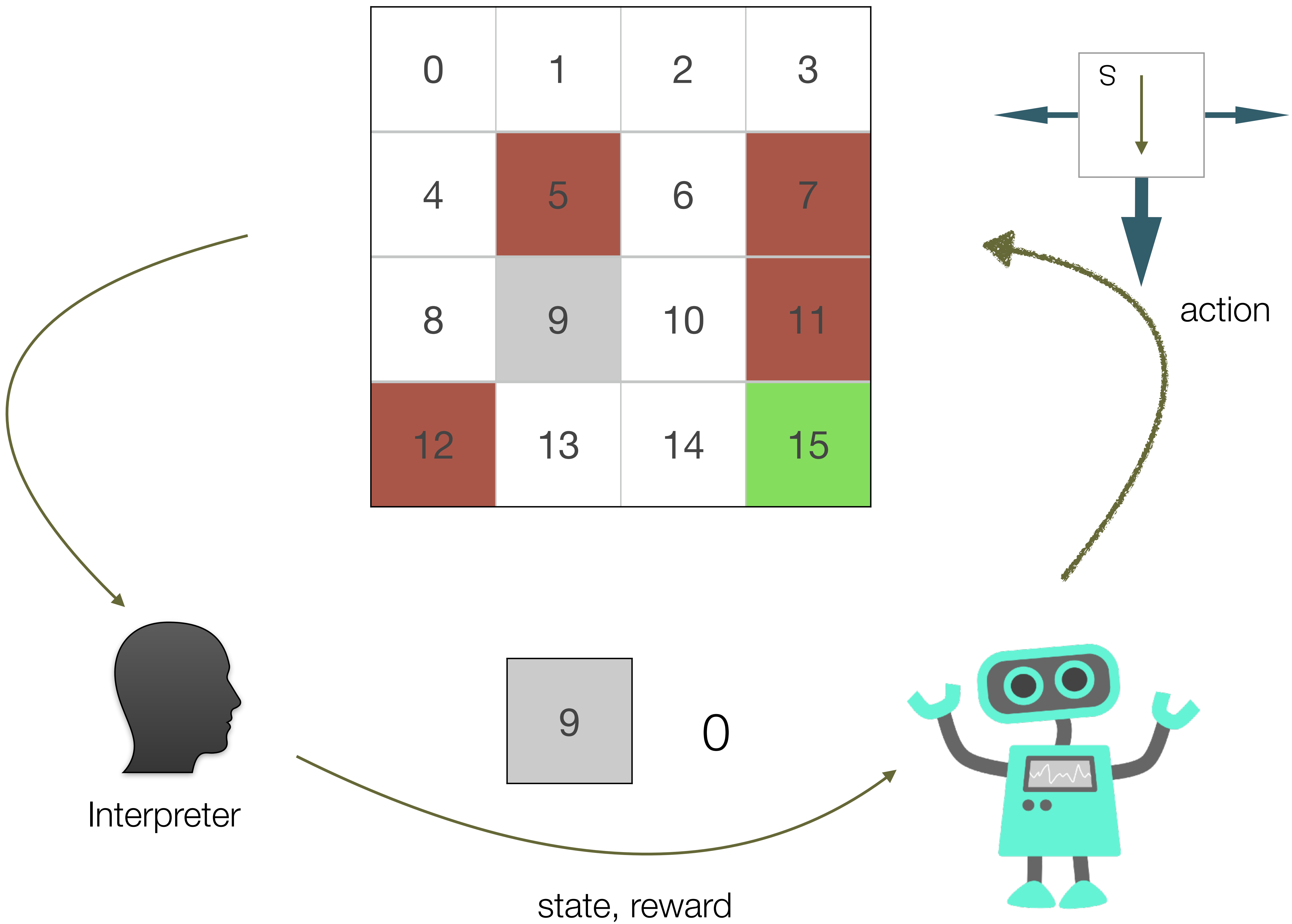

Environment
Interpreter
Reward
State
Agent
Action

Grid-World Example: Frozen Lake*

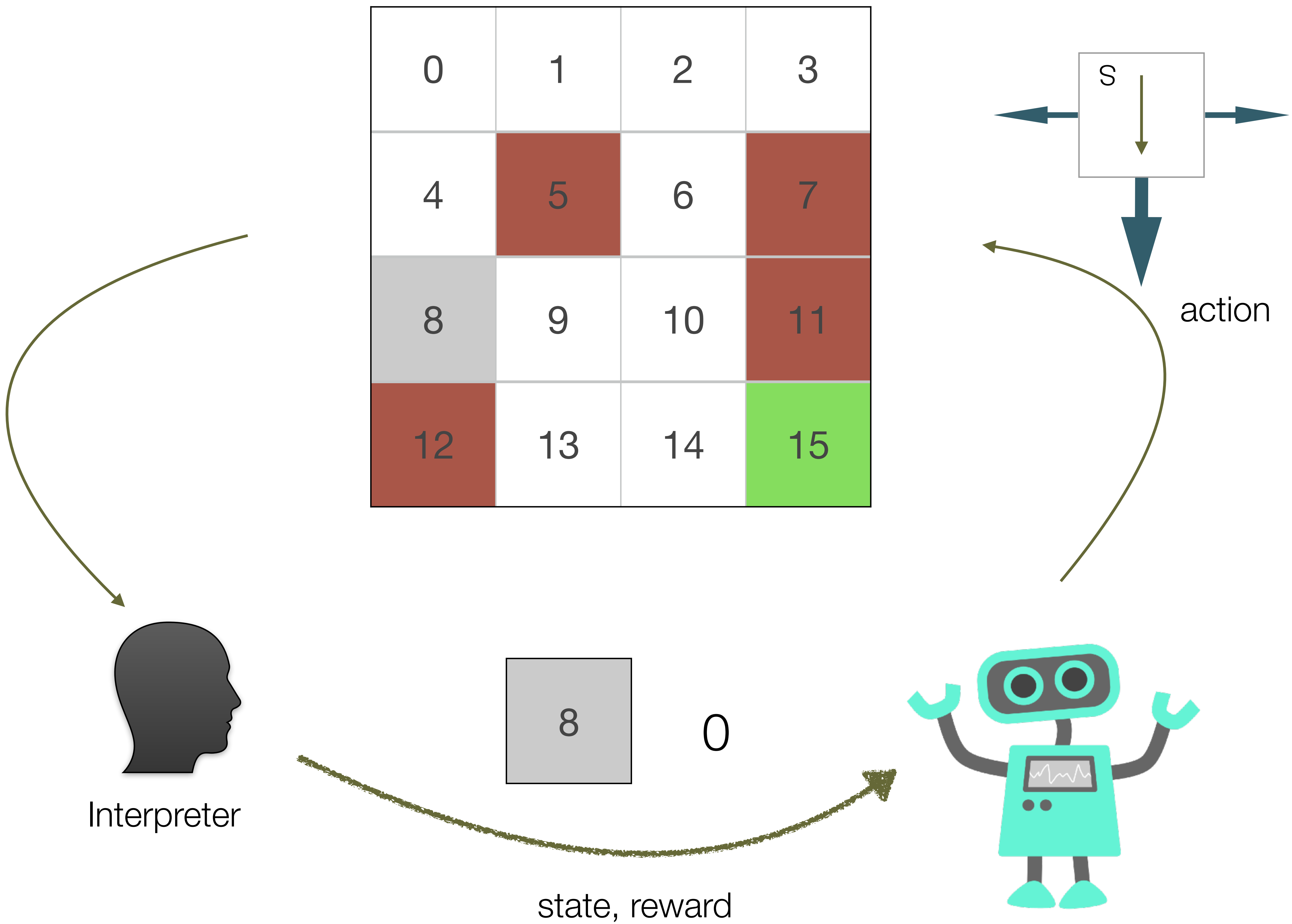




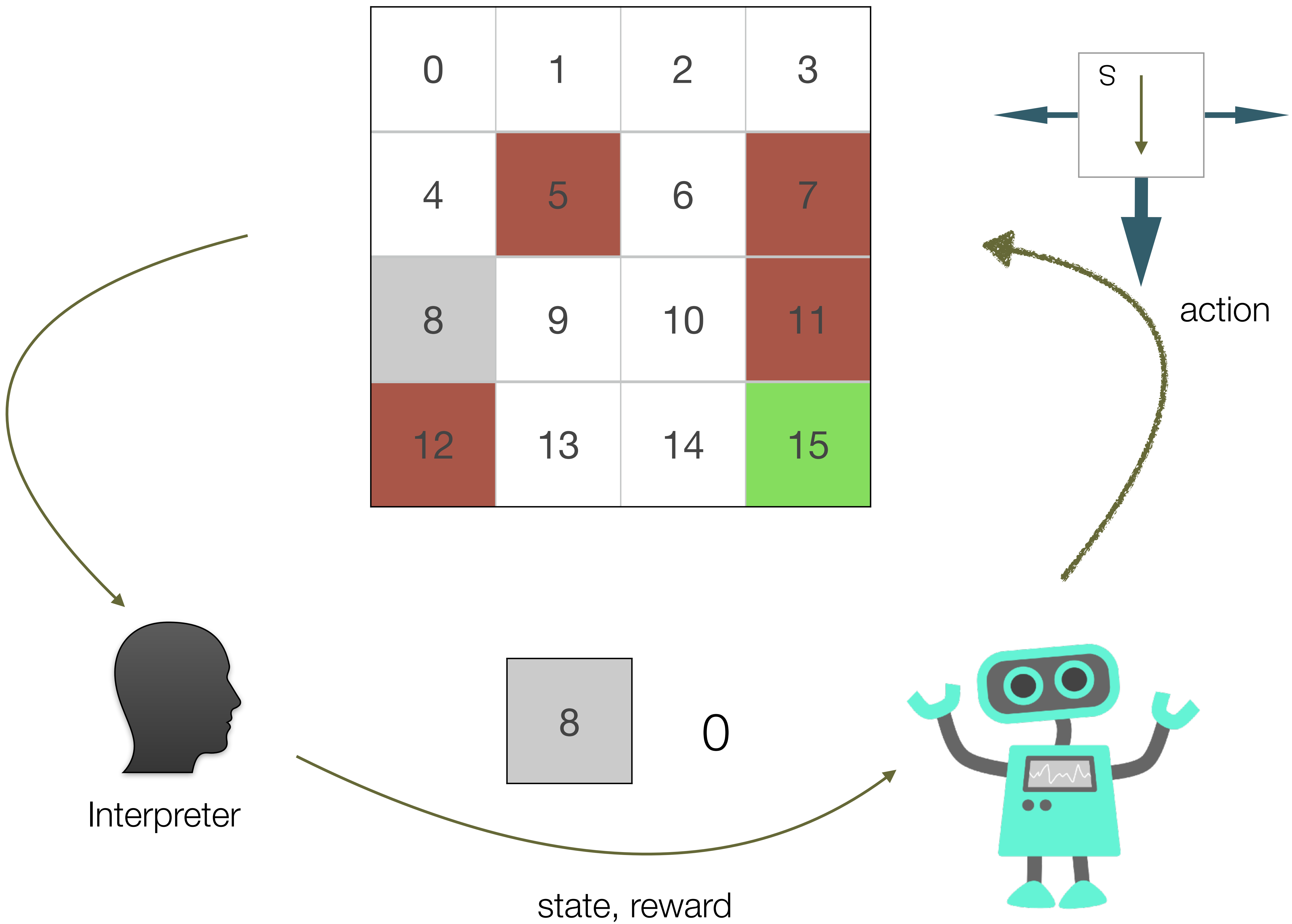
*Frozen Lake (https://gym.openai.com)



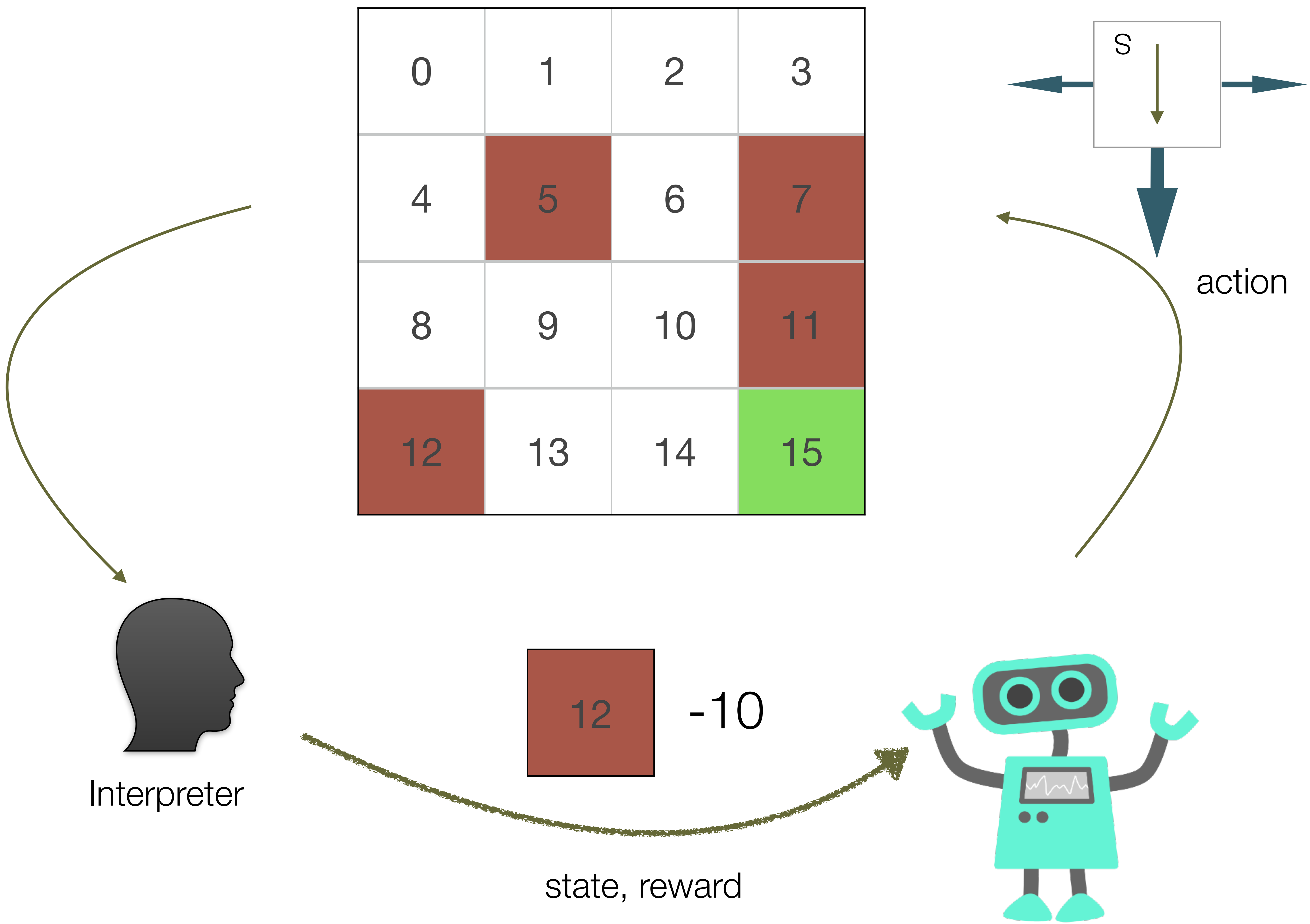
0
1
2
3
4
5
6
7
8
9
10
11
12
13
14
15
S
action
Interpreter
9
0
state, reward

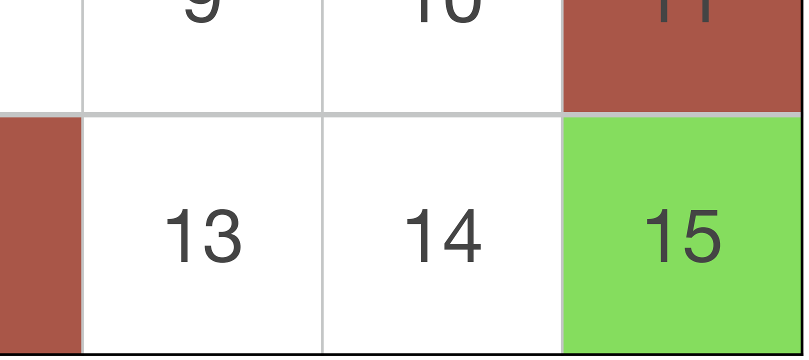


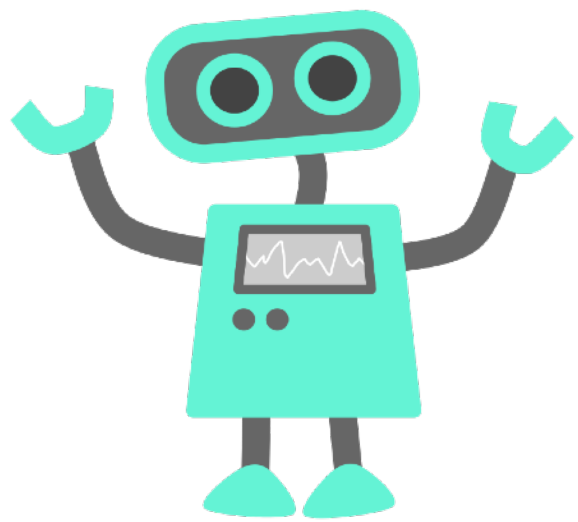
0
1
2
3
4
5
6
7
8
9
10
11
12
13
14
15
S
action
Interpreter
8
0
state, reward



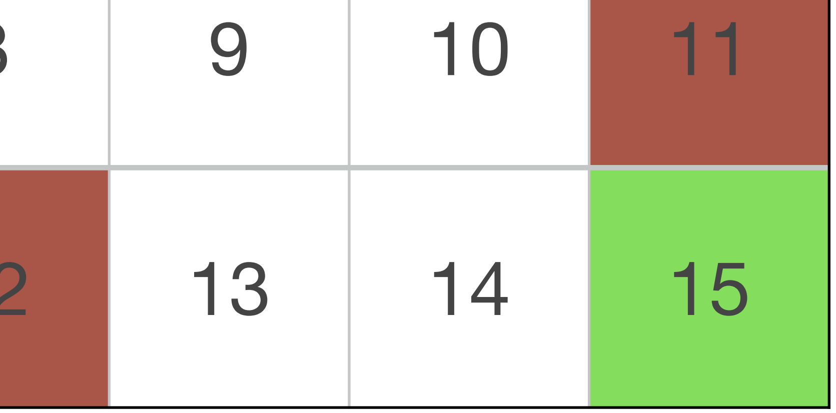
0
1
2
3
4
5
6
7
8
9
10
11
12
13
14
15
S
action
Interpreter
8
0
state, reward

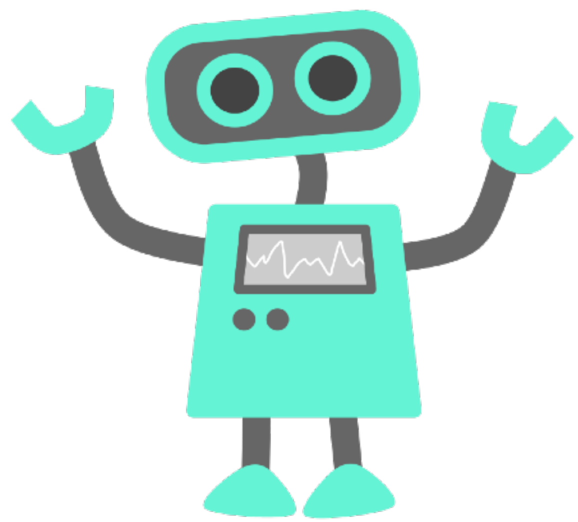

0
1
2
3
4
5
6
7
8
9
10
11
12
13
14
15
S
action
Interpreter
12
-10
state, reward

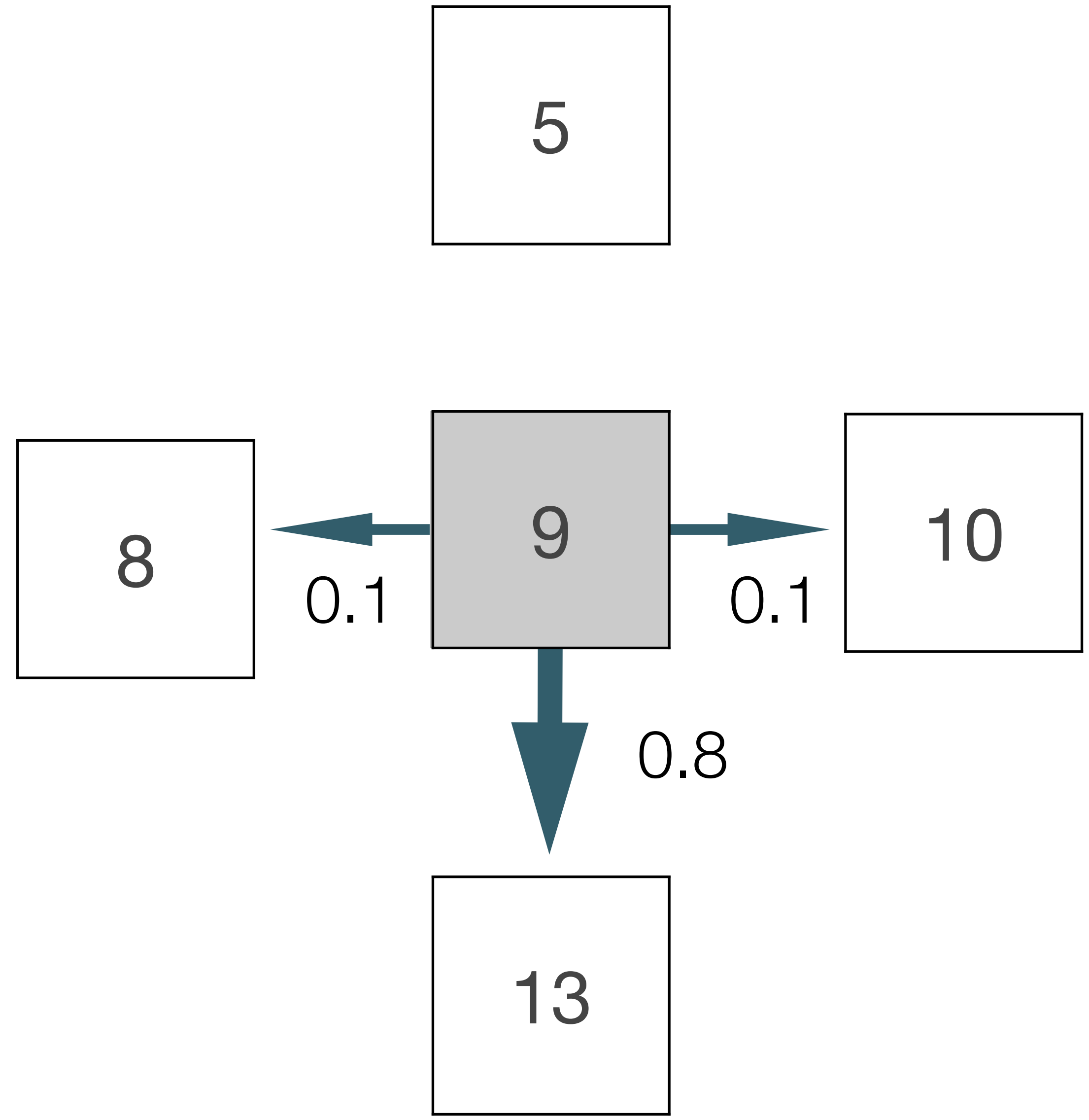




Two flavors of reinforcement learning

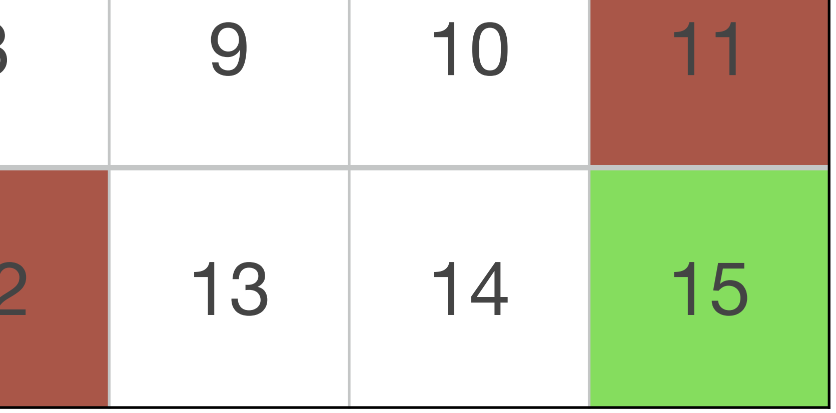




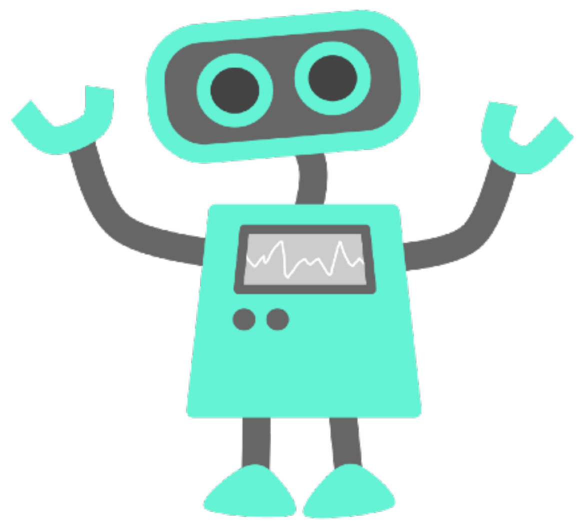


Markov Decision Process

Model-Based Reinforcement Learning

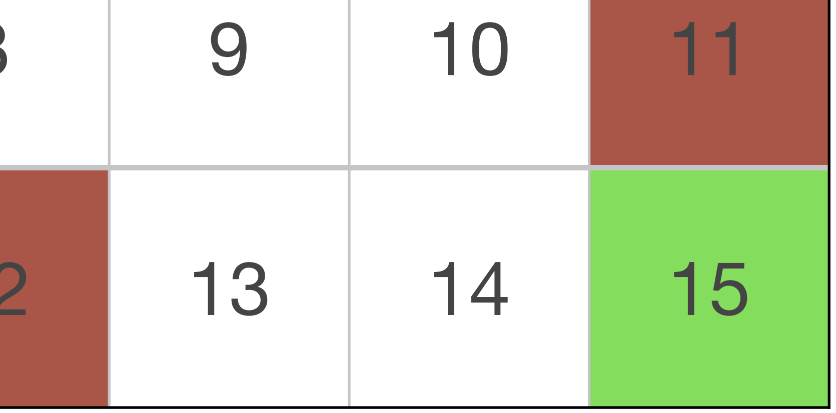


Model-Free Reinforcement Learning

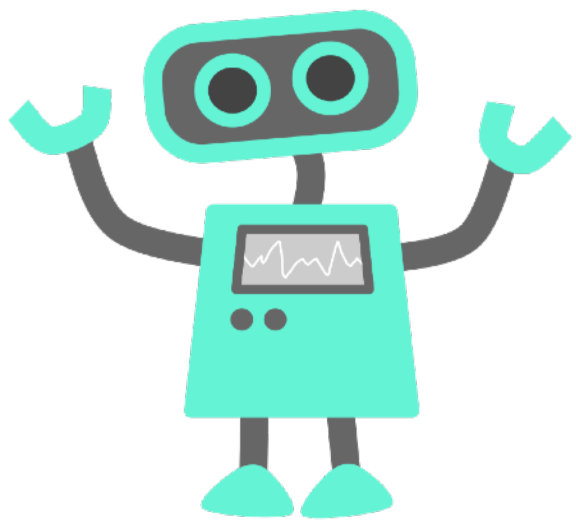


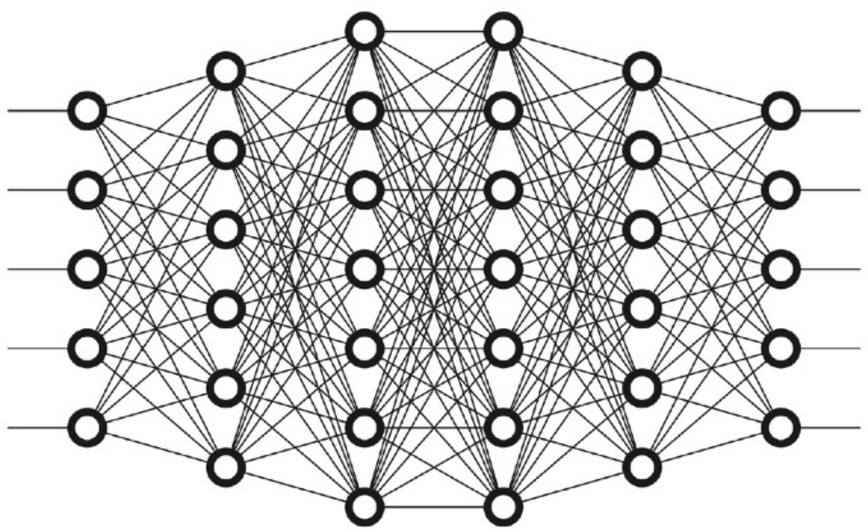
Q	N	W	E	S
1	0.04	0.030	0.092	0.026
2	0.059	-1	0.136	-1
3	0.221	0.495	0.525	-1
4	...	...	...	...
...	...	...	...	...

$$Q_{t+1}(s,a) = \begin{cases} (1-\alpha_t)Q_t(s,a) + \alpha_t(r_{t+1} + \lambda \cdot \max\limits_{a' \in A(s)} Q_t(s_{t+1}, a')), & \text{if } (s,a) = (s_t, a_t) \\ Q_t(s,a), & \text{otherwise,} \end{cases}$$



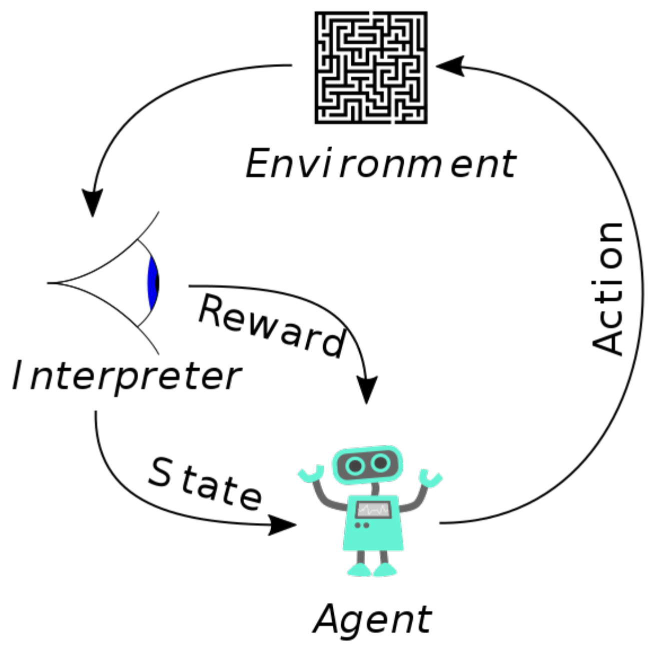
Model-Free Reinforcement Learning





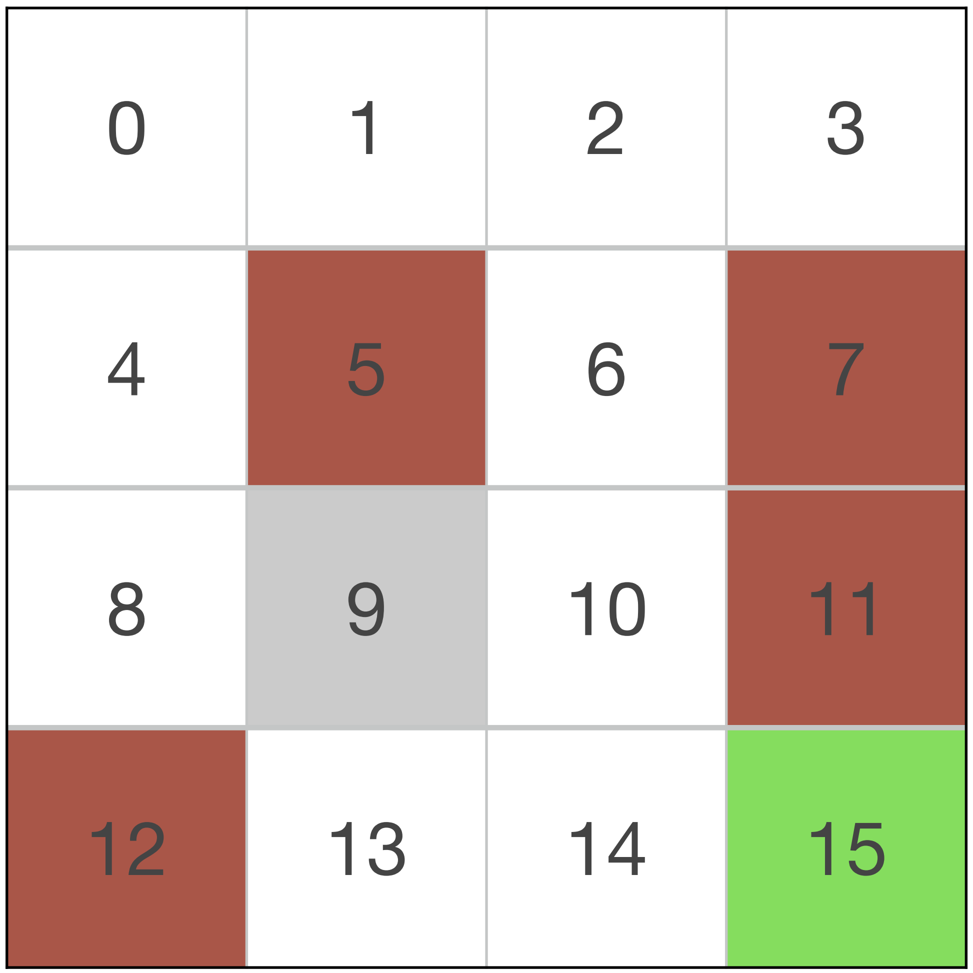
Q	N	W	E	S
1	0.04	0.030	0.092	0.026
2	0.059	-1	0.136	-1
3	0.221	0.495	0.525	-1
4	…	…	…	…
…	…	…	…	…

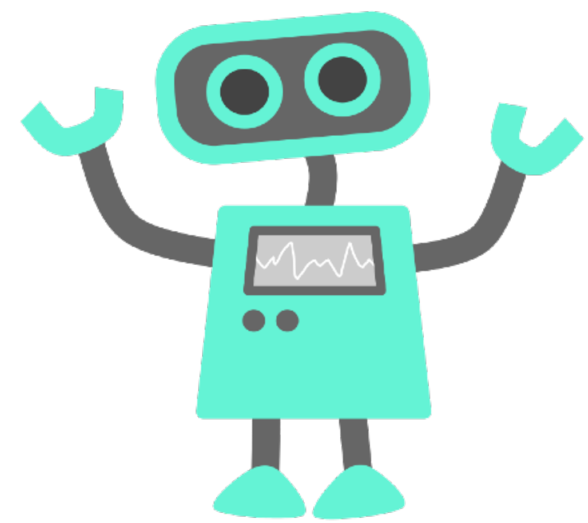
$$Q_{t+1}(s,a) = \begin{cases} (1-\alpha_t)Q_t(s,a) + \alpha_t(r_{t+1} + \lambda \cdot \max_{a' \in A(s)} Q_t(s_{t+1}, a')), & \text{if } (s,a) = (s_t, a_t) \\ Q_t(s,a), & \text{otherwise,} \end{cases}$$



Programming with scalar rewards

Some rewards are easy to express:

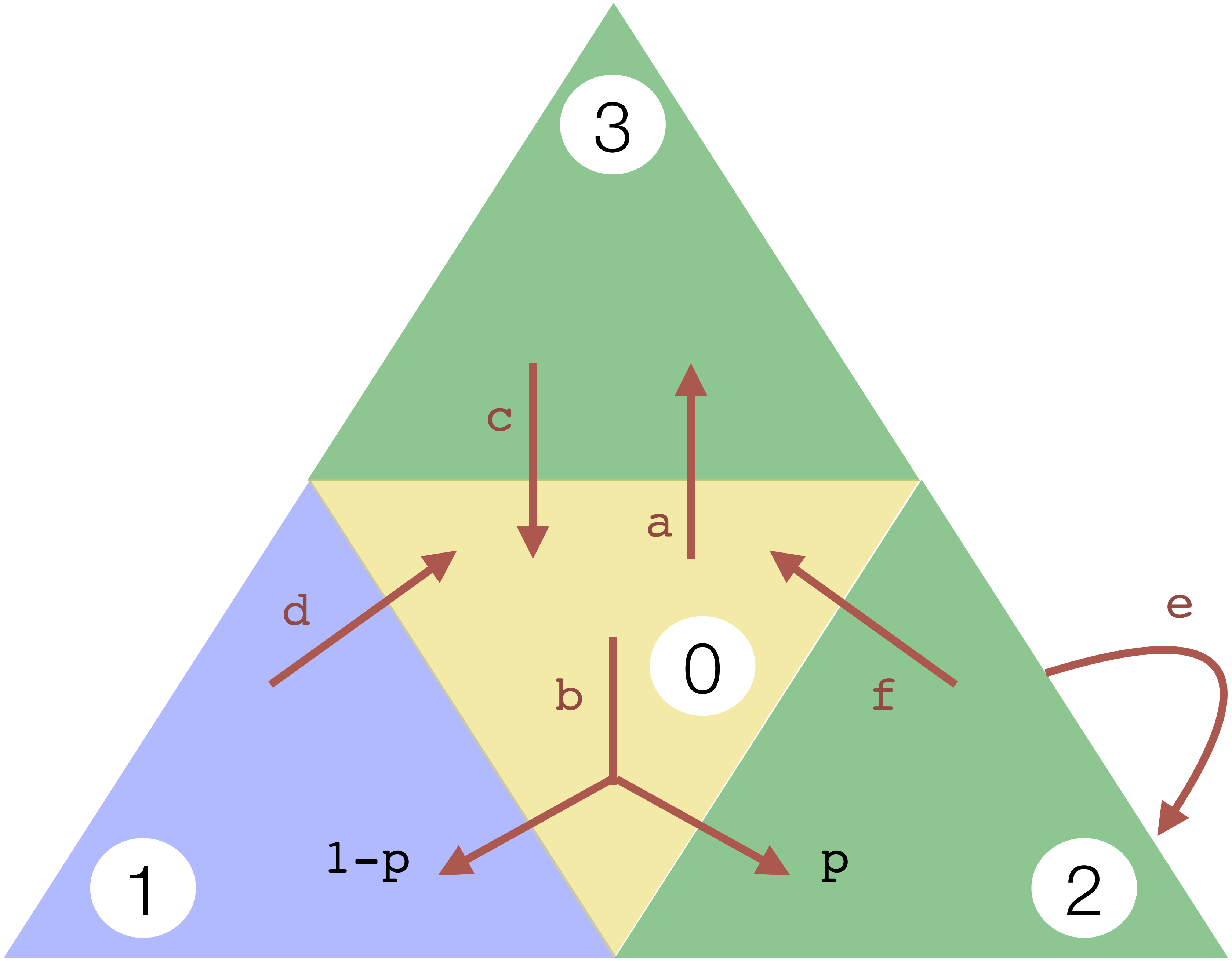


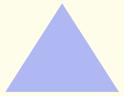
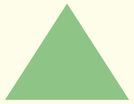


Specification: Reach while Avoiding

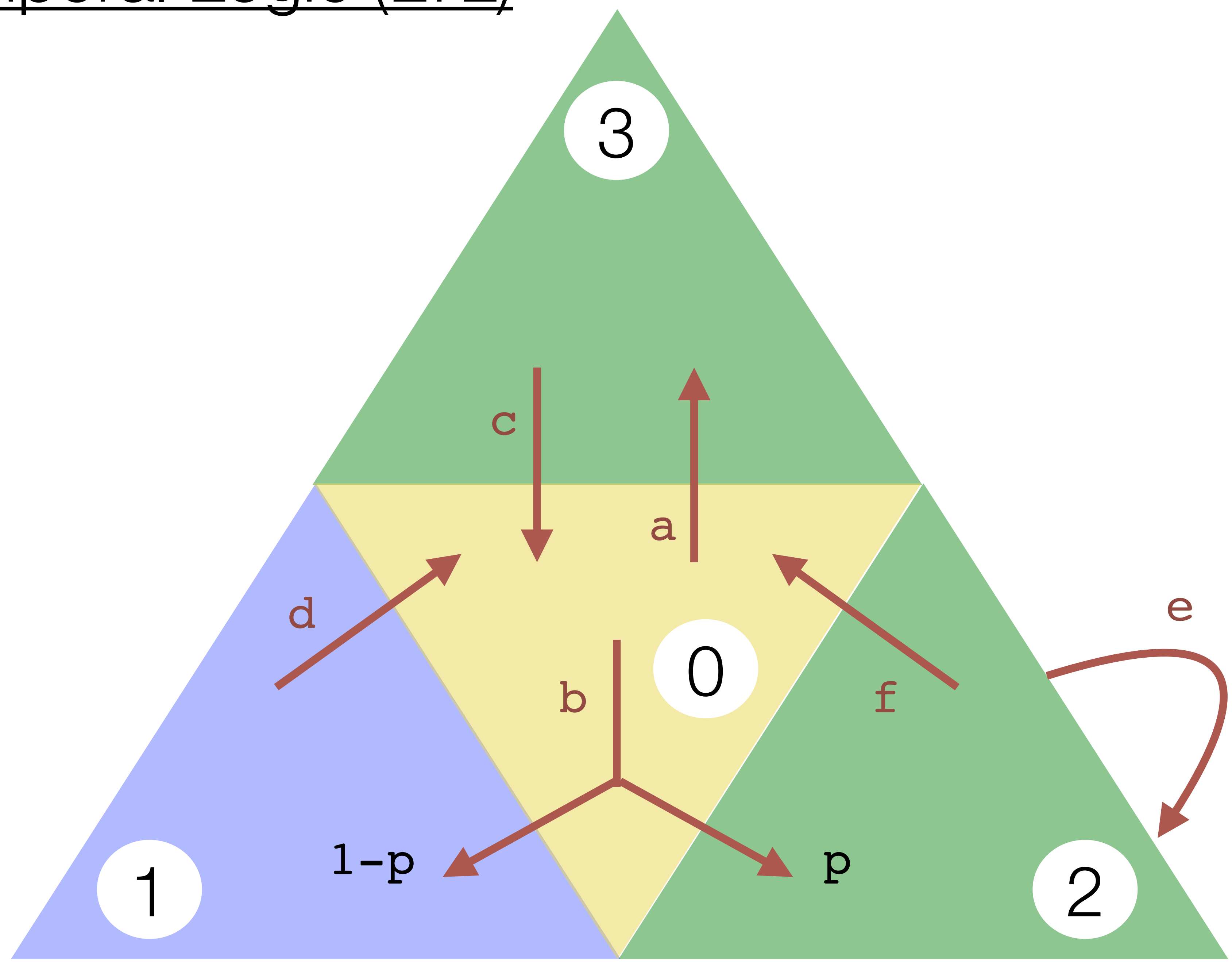
Reward: +1 at -1 at

Some, more involved:



always avoid  and infinitely often visit 

Linear-Temporal Logic (LTL)

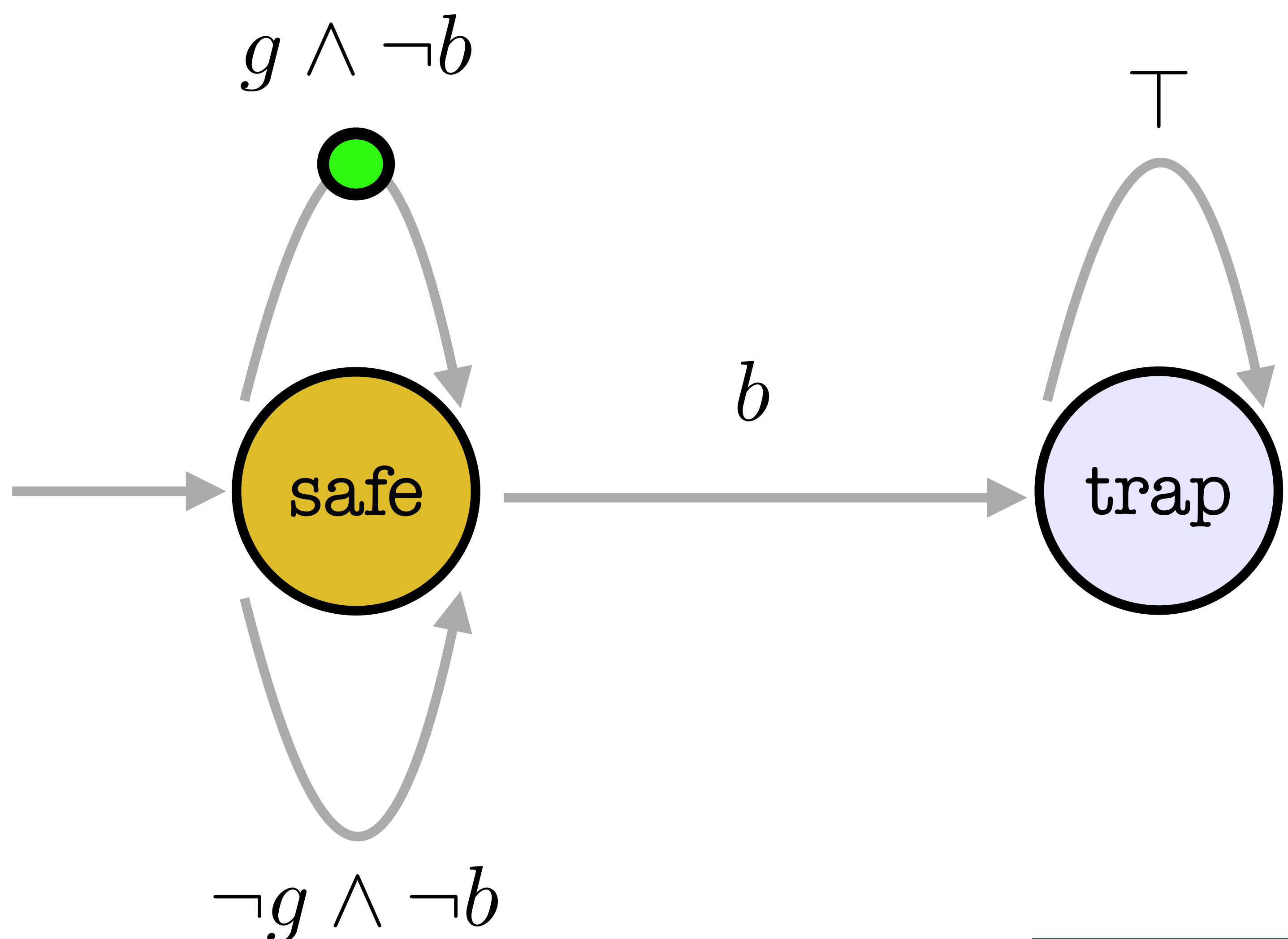


lobally
lobally
inally

$$\varphi = (\mathsf{G}\,\neg\mathsf{b}) \wedge (\mathsf{G}\,\mathsf{F}\,\mathsf{g})$$

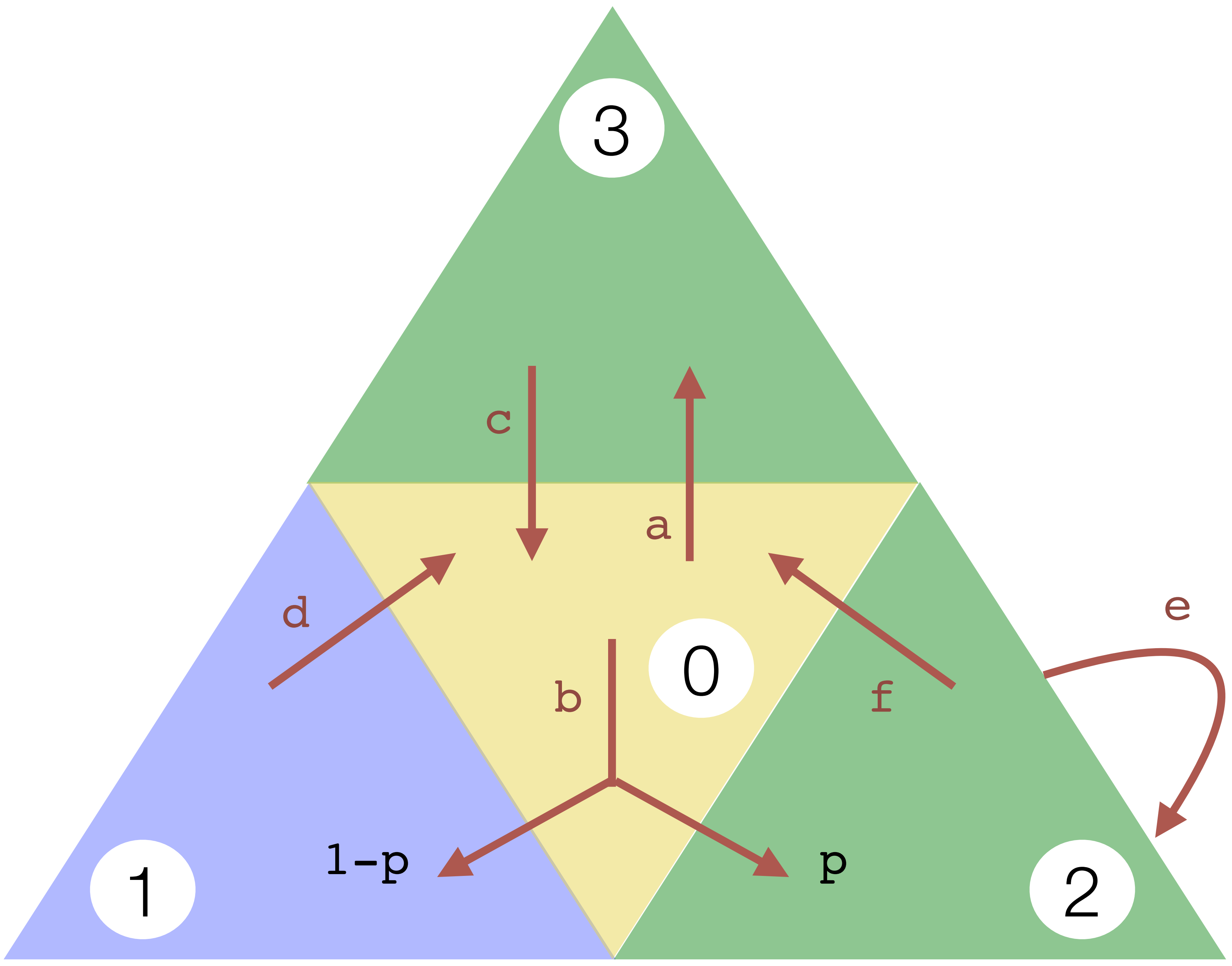
Compiling LTL to Monitor Automata

$$\varphi = (\mathsf{G}\,\neg\mathsf{b}) \wedge (\mathsf{G}\,\mathsf{F}\,\mathsf{g})$$

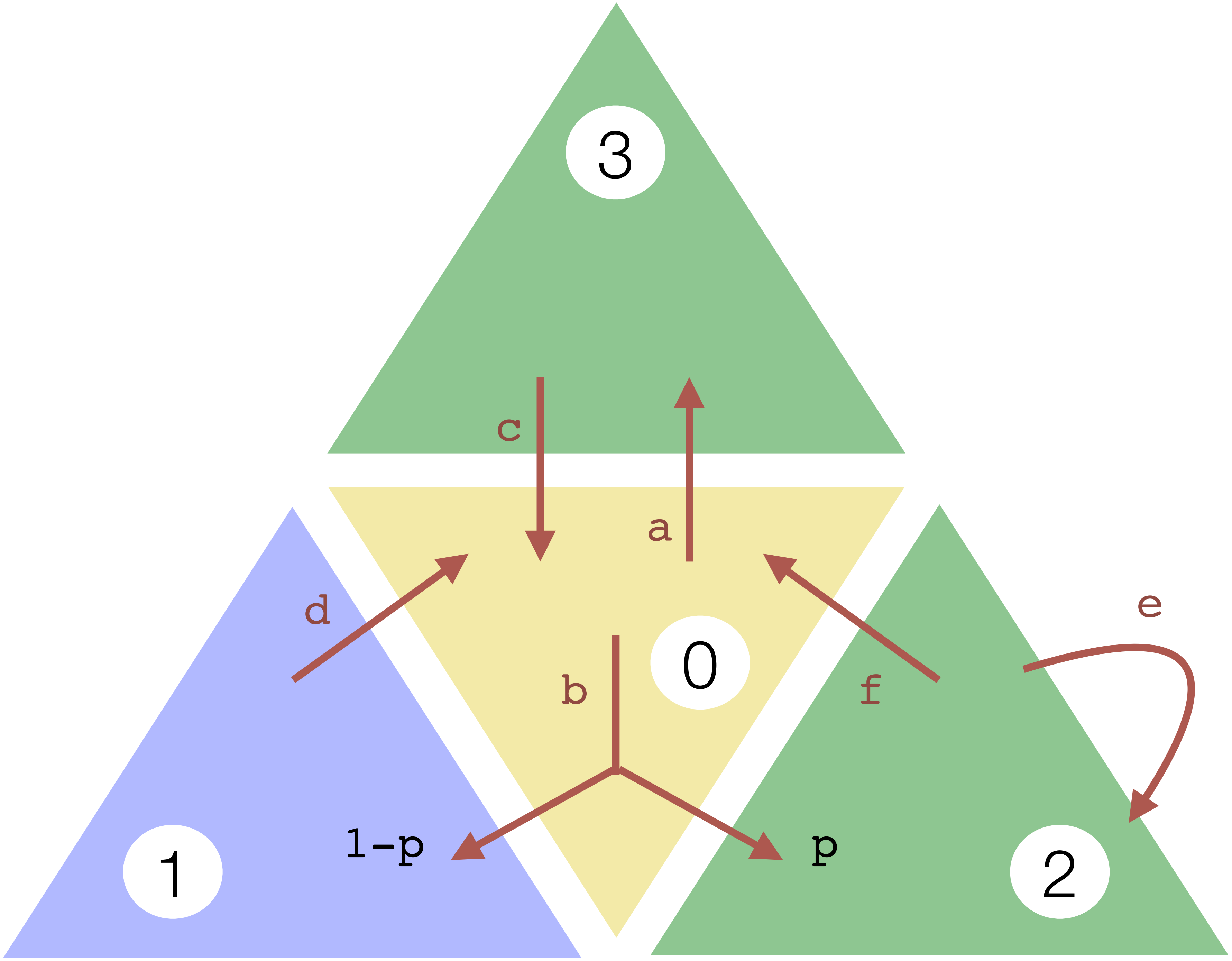


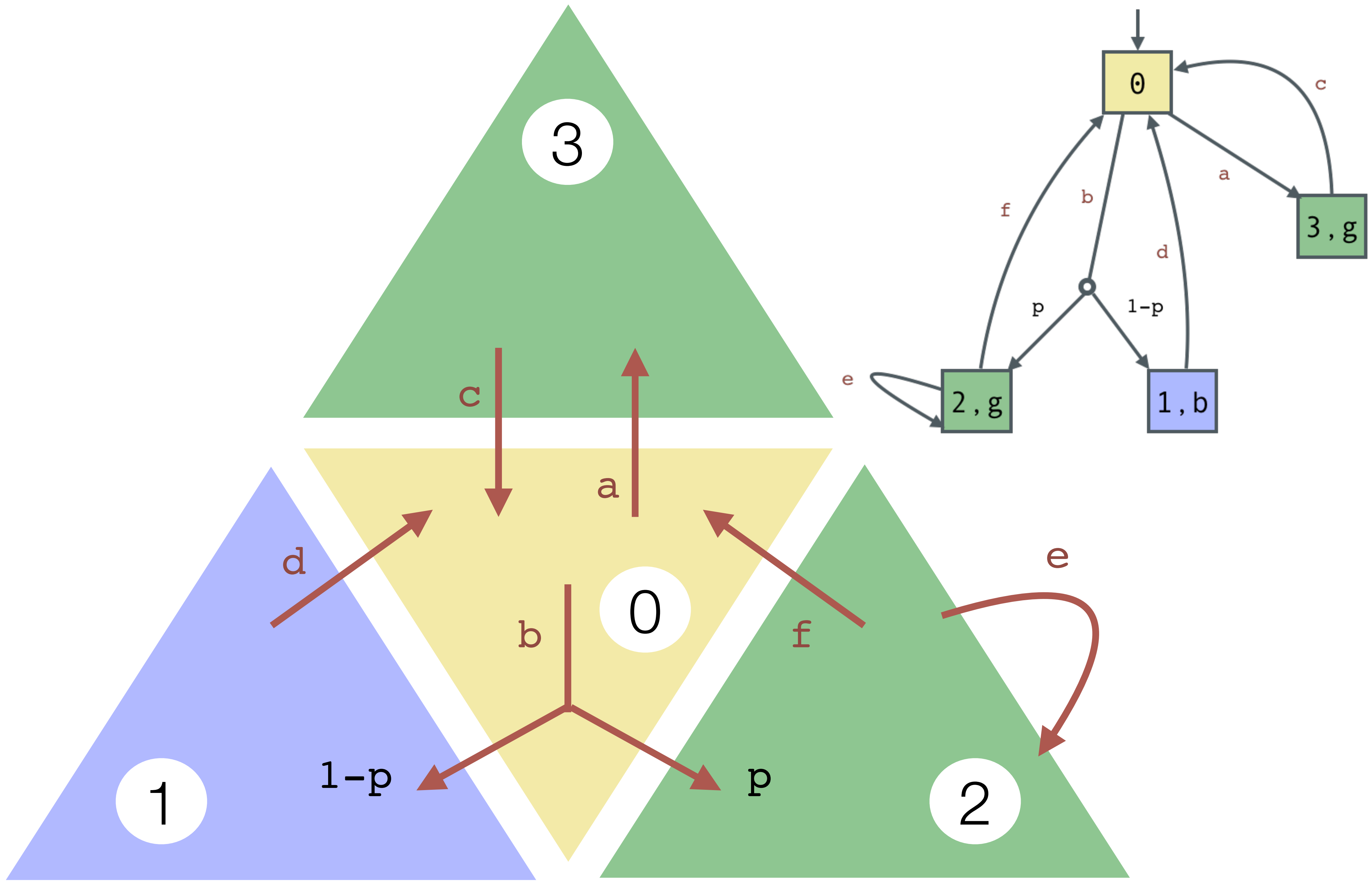
Büchi Automata

How to use Linear Temporal Logic in RL?

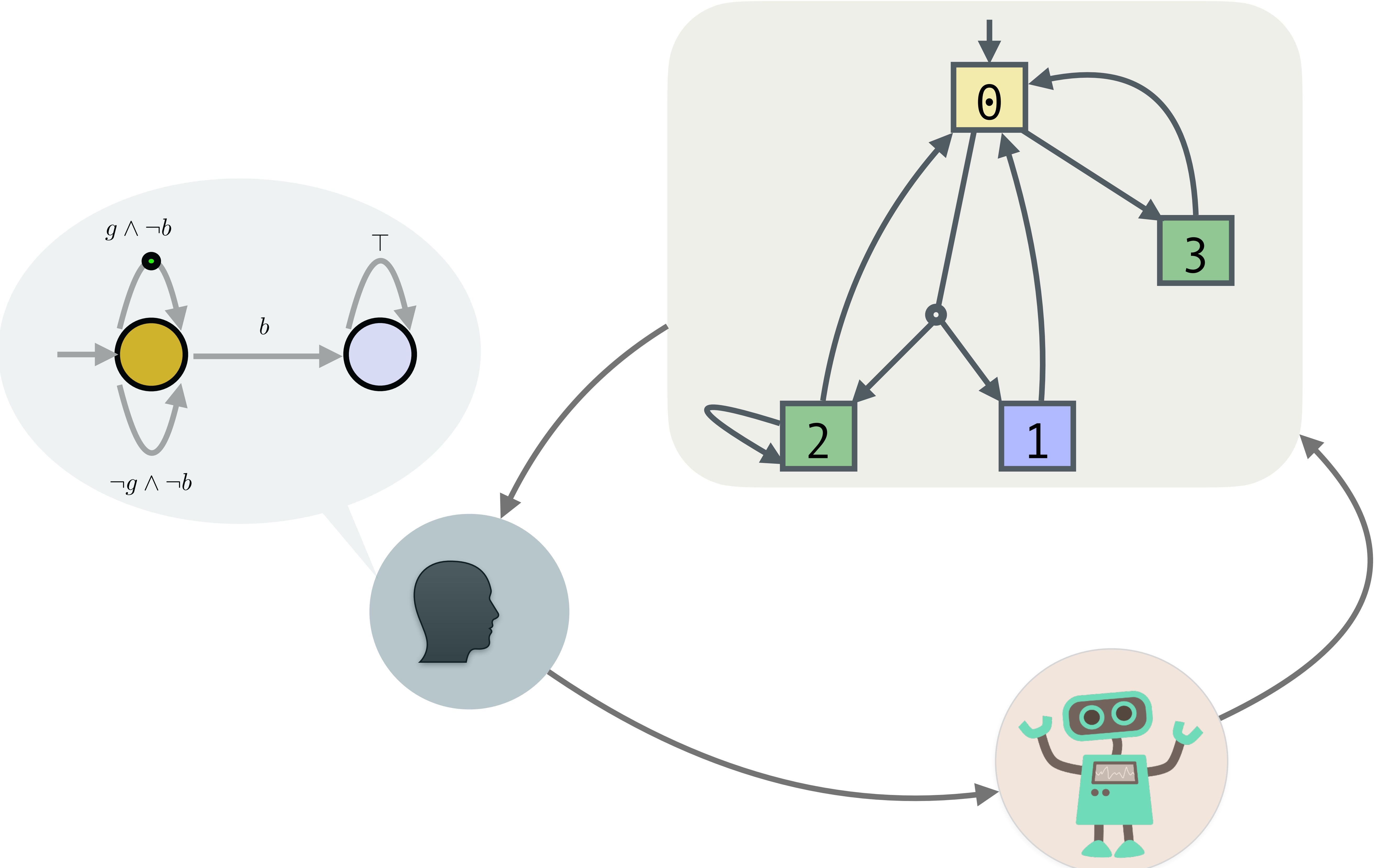


3
c
a
d
0
e
b
f
1-p
p
1
2


3
c
a
d
0
b
f
e
1-p
p
1
2



3
c
a
0
d
b
f
e
1
1-p
p
2
0
c
f
b
a
3,g
d
p
1-p
e
2,g
1,b



Formal Reinforcement Learning

Model-Based Formal Reinforcement Learning

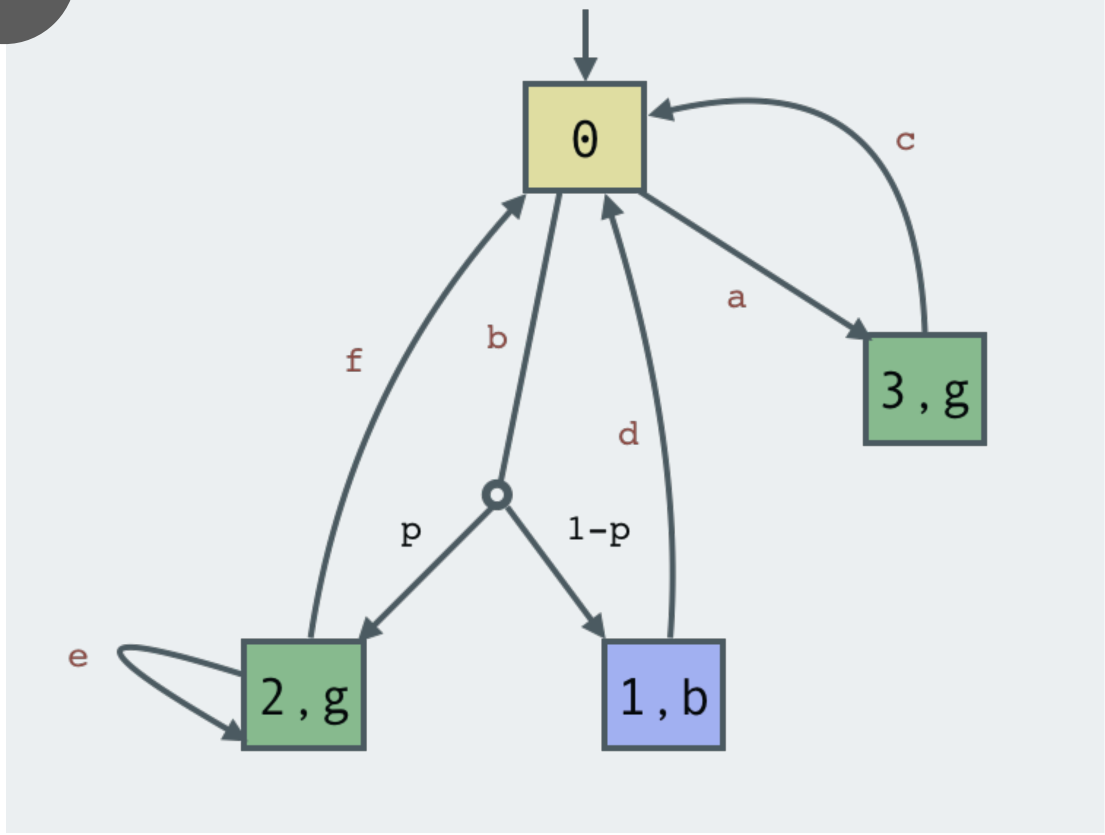
Moshe Y. Vardi:
Automatic Verification of Probabilistic Concurrent Finite-State Programs. FOCS 1985: 327-338

Jie Fu, Ufuk Topcu:
Probably Approximately Correct MDP Learning and Control With Temporal Logic Constraints. Robotics: Science and Systems 2014

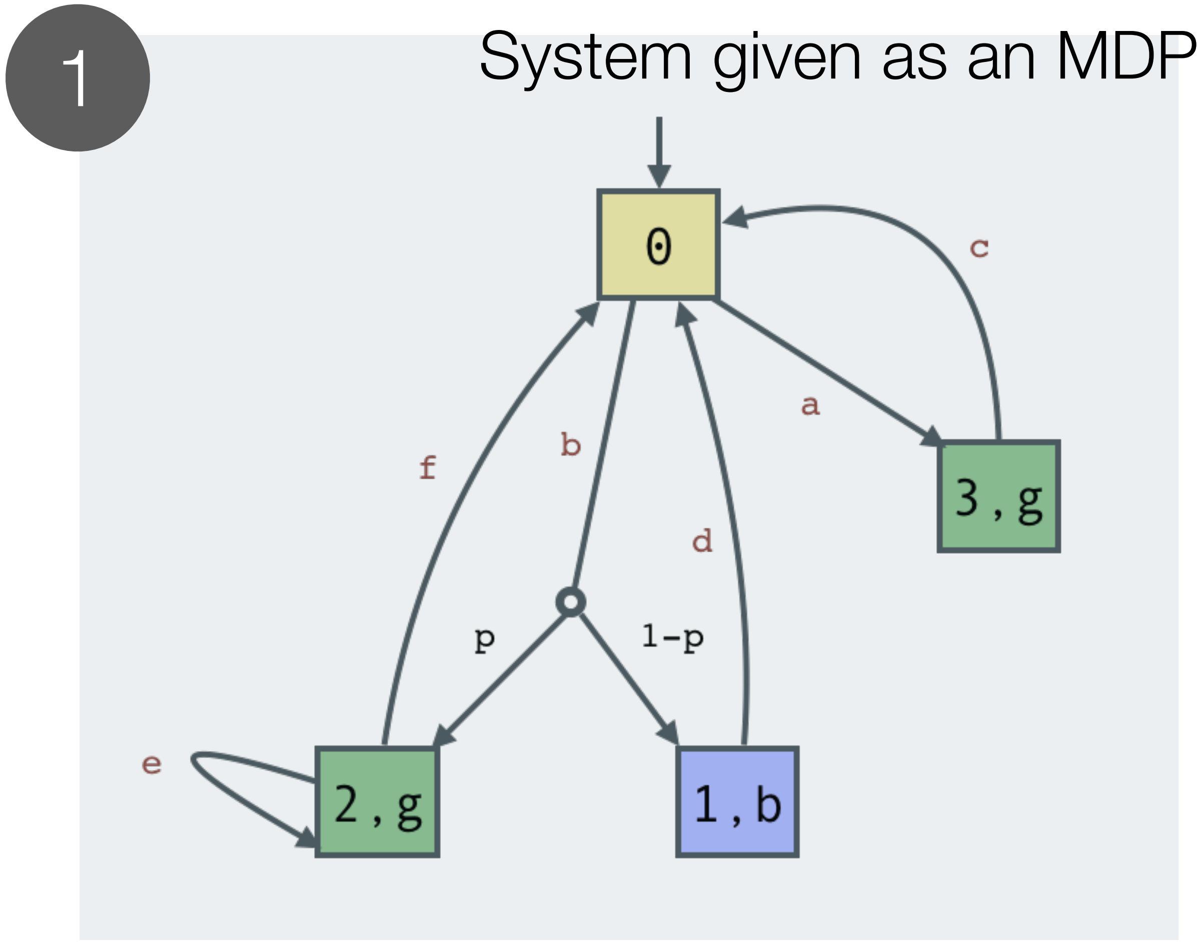
Probabilistic Model Checking

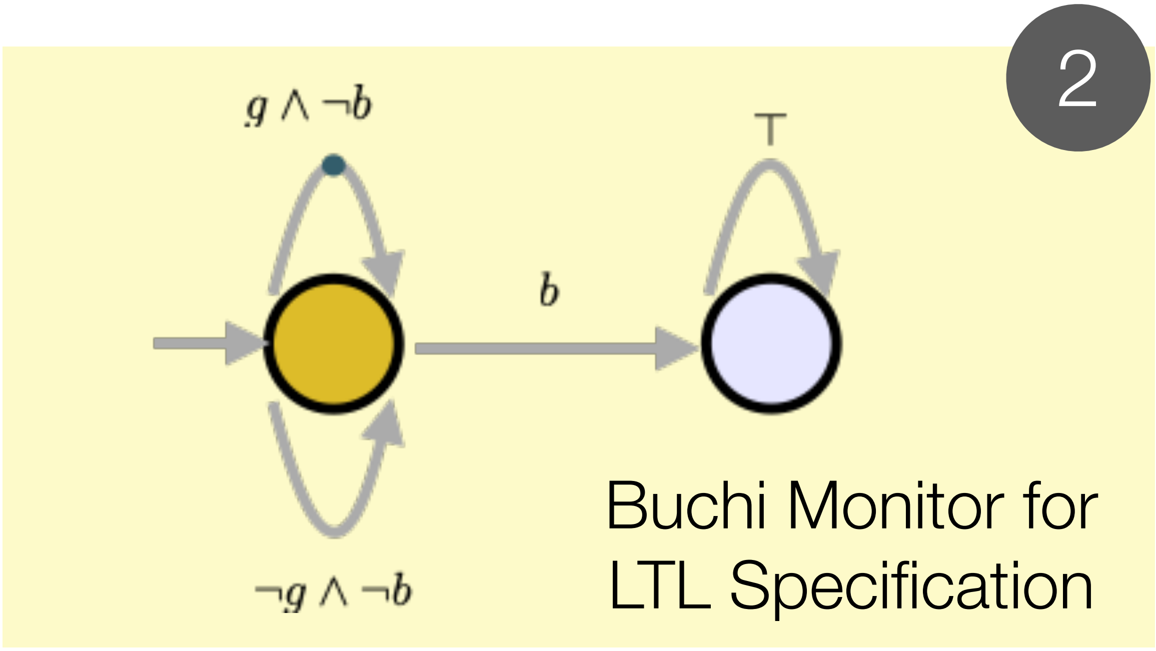
1

System given as an MDP

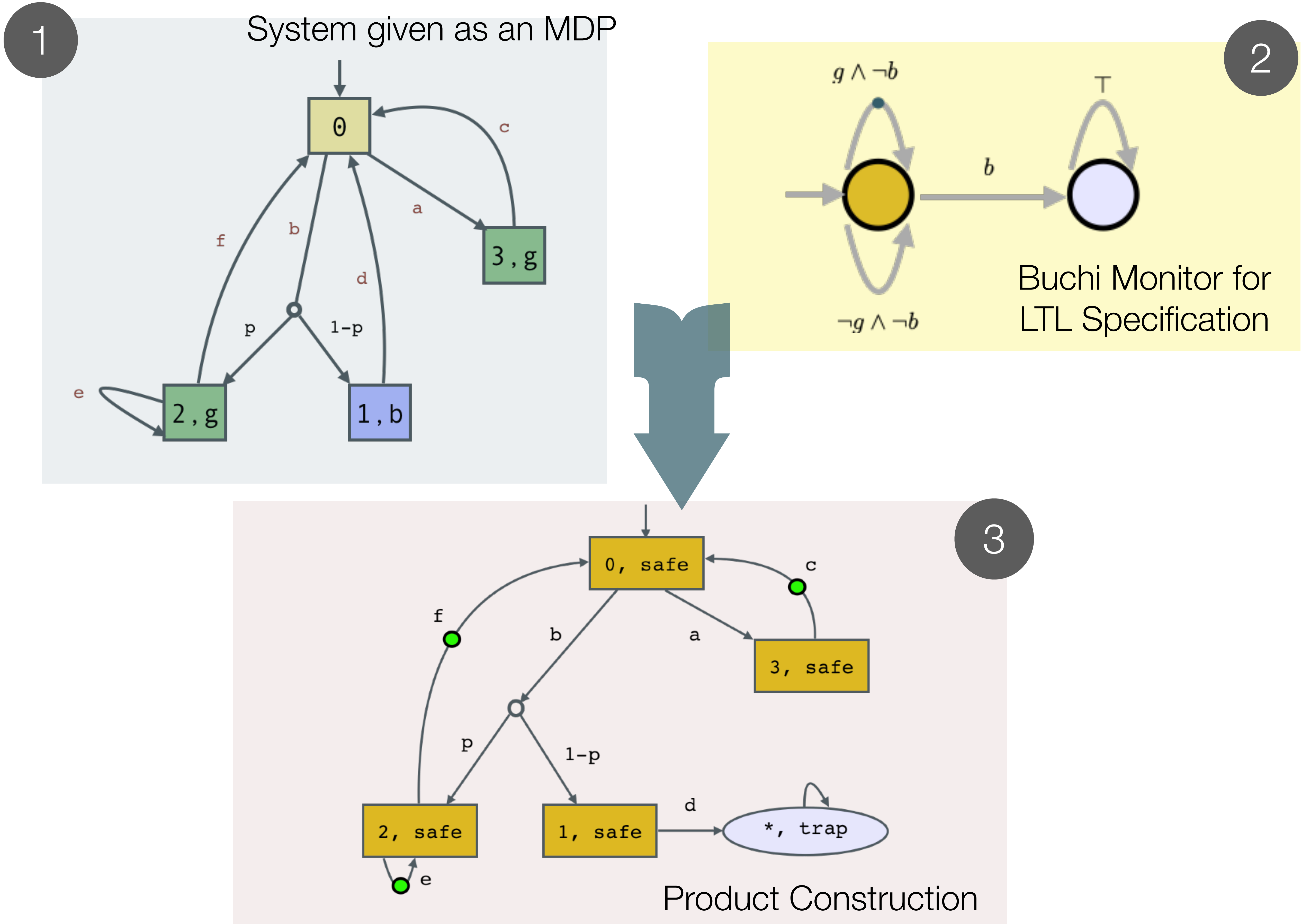


Probabilistic Model Checking





<u>Probabilistic Model Checking</u>



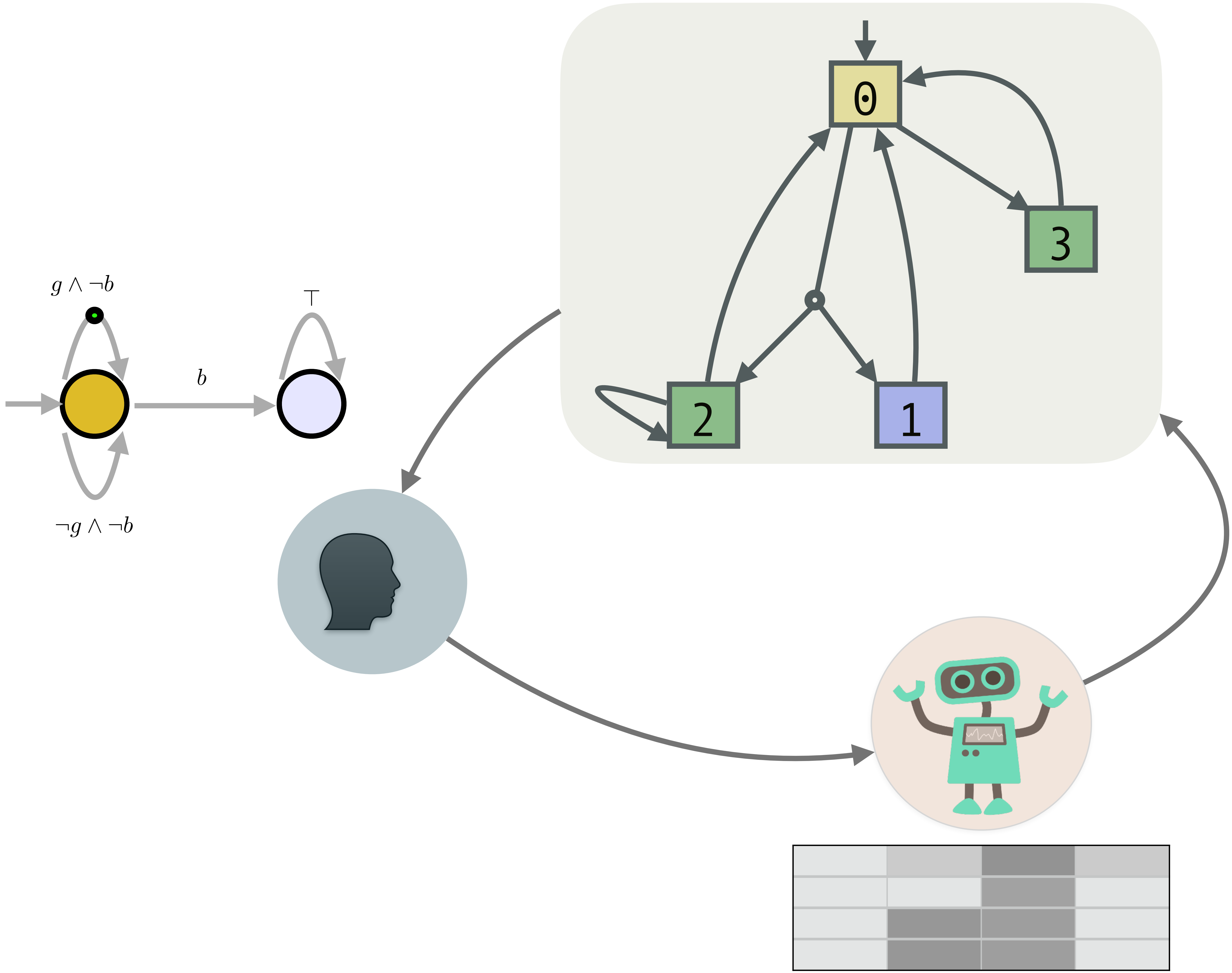
Probabilistic Model Checking

Model-Free Formal Reinforcement Learning

Dorsa Sadigh, Eric S. Kim, Samuel Coogan, S. Shankar Sastry, Sanjit A. Seshia:
A learning based approach to control synthesis of Markov decision processes for linear temporal logic specifications. CDC 2014: 1091-1096

Mohammadhosein Hasanbeig, Alessandro Abate, Daniel Kroening:
Logically-Correct Reinforcement Learning. CoRR abs/1801.08099 (2018)

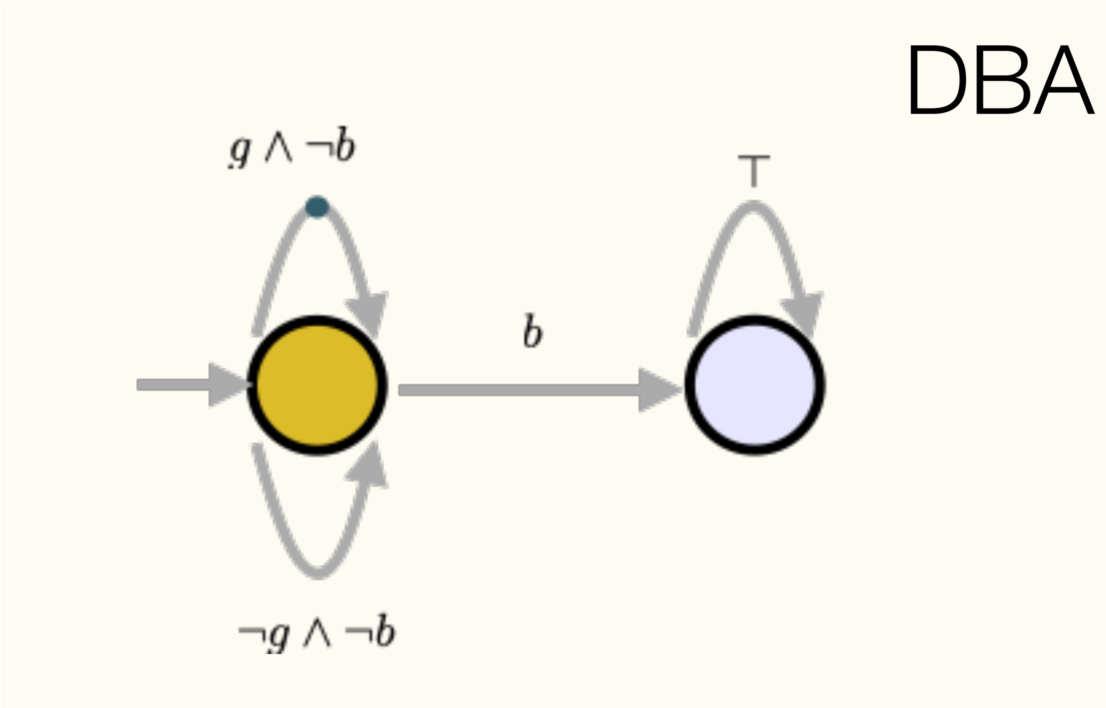
Masaki Hiromoto, Toshimitsu Ushio:
Learning an Optimal Control Policy for a Markov Decision Process Under Linear Temporal Logic Specifications. SSCI 2015: 548-555

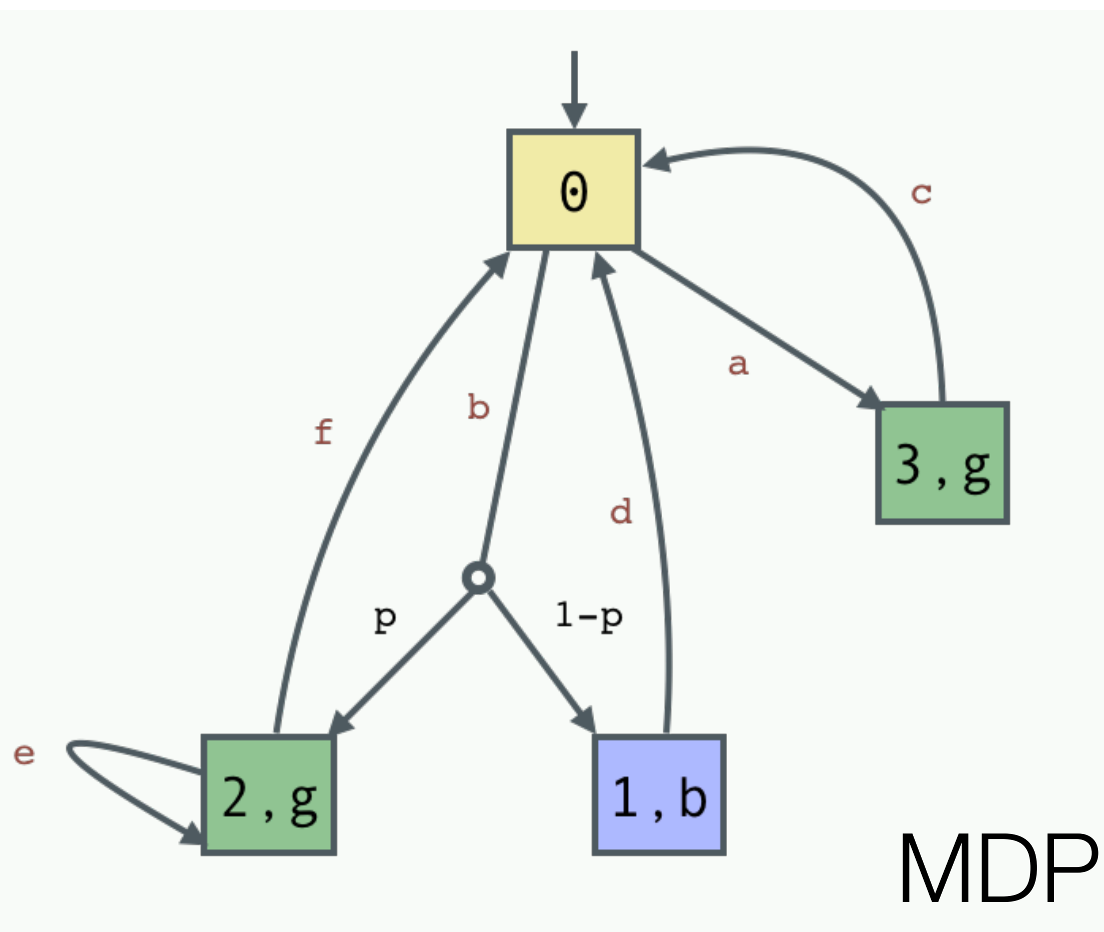


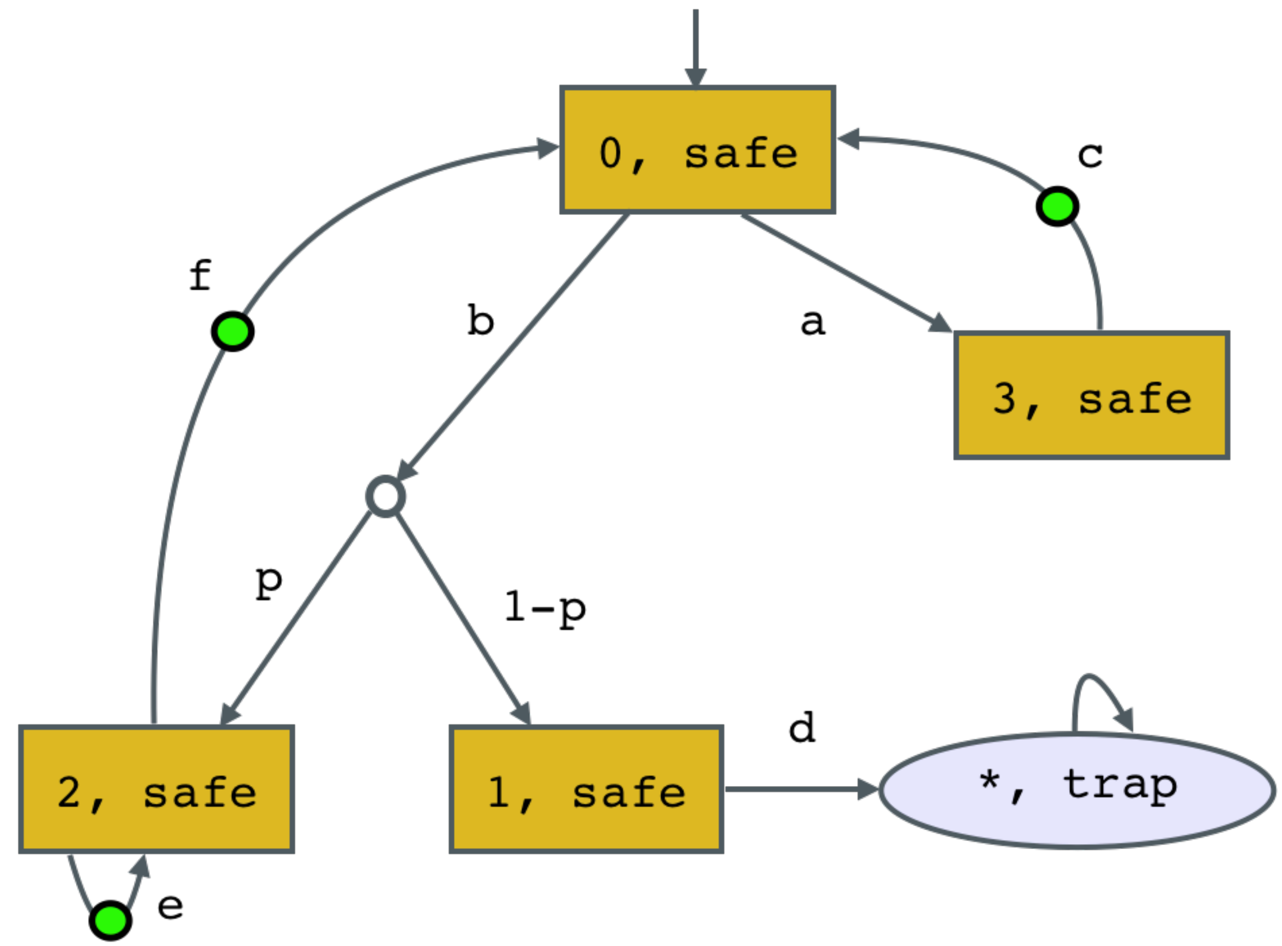
$g \wedge \neg b$
$\top$
b
$\neg g \wedge \neg b$
0
3
2
1

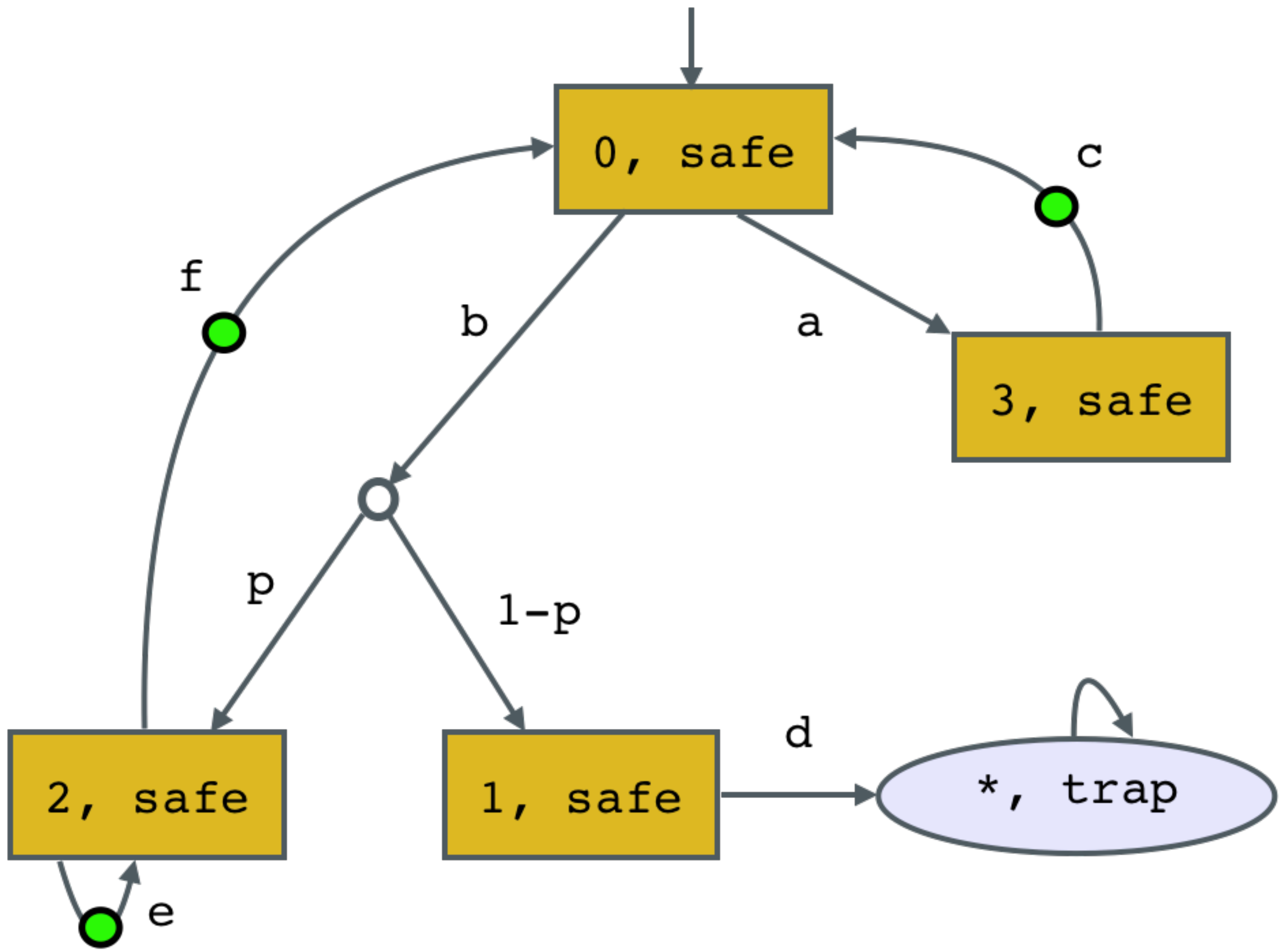
Unfortunately, this simple approach is incorrect.

Ernst Moritz Hahn, Mateo Perez, Sven Schewe, Fabio Somenzi, Ashutosh Trivedi, Dominik Wojtczak:
Omega-Regular Objectives in Model-Free Reinforcement Learning. TACAS (1) 2019: 395-412


DBA
$g \wedge \neg b$
$\top$
b
$\neg g \wedge \neg b$


0
c
a
f
b
d
3,g
p
1-p
e
2,g
1,b
MDP

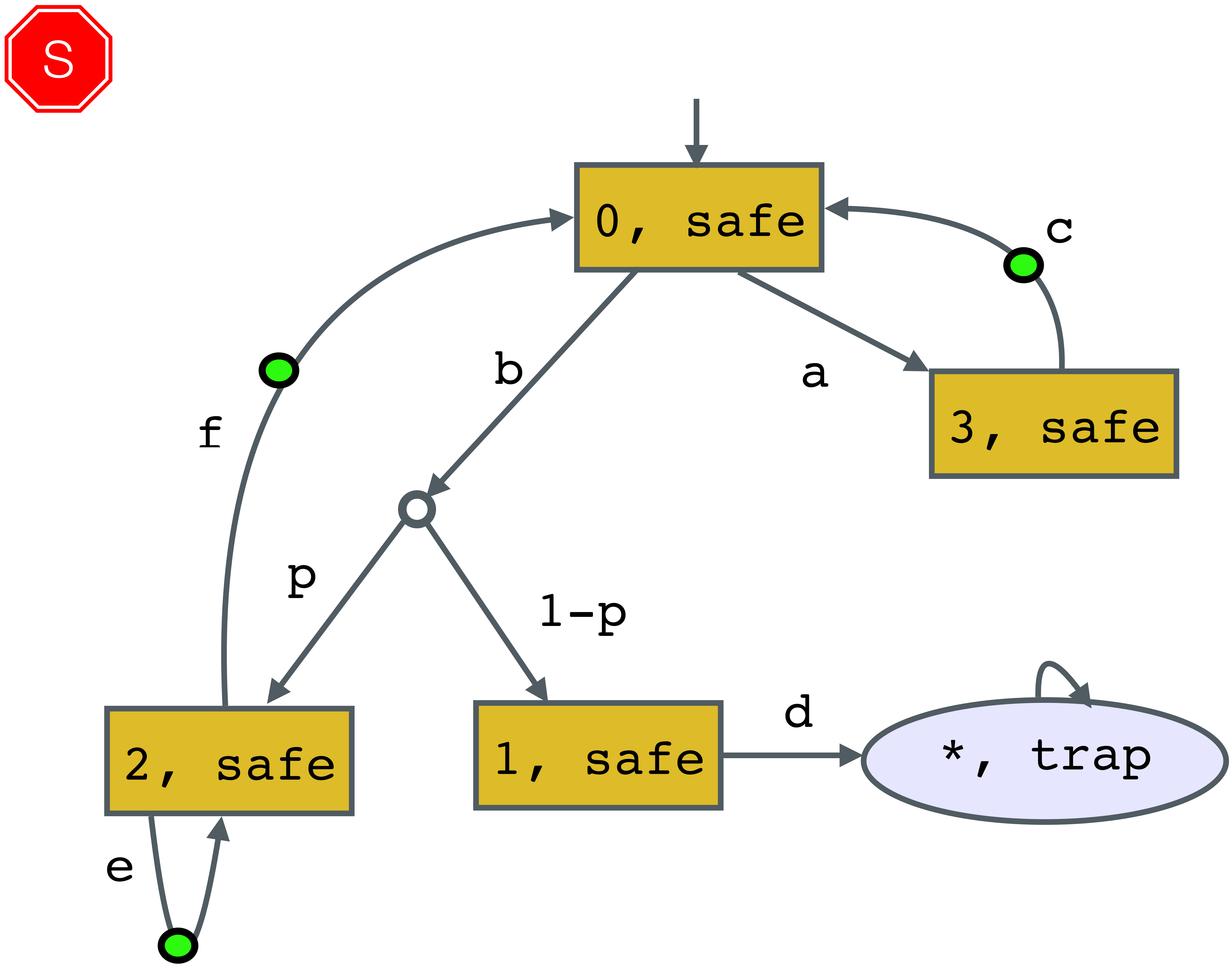

0, safe
c
f
b
a
3, safe
p
1-p
d
2, safe
1, safe
*, trap
e

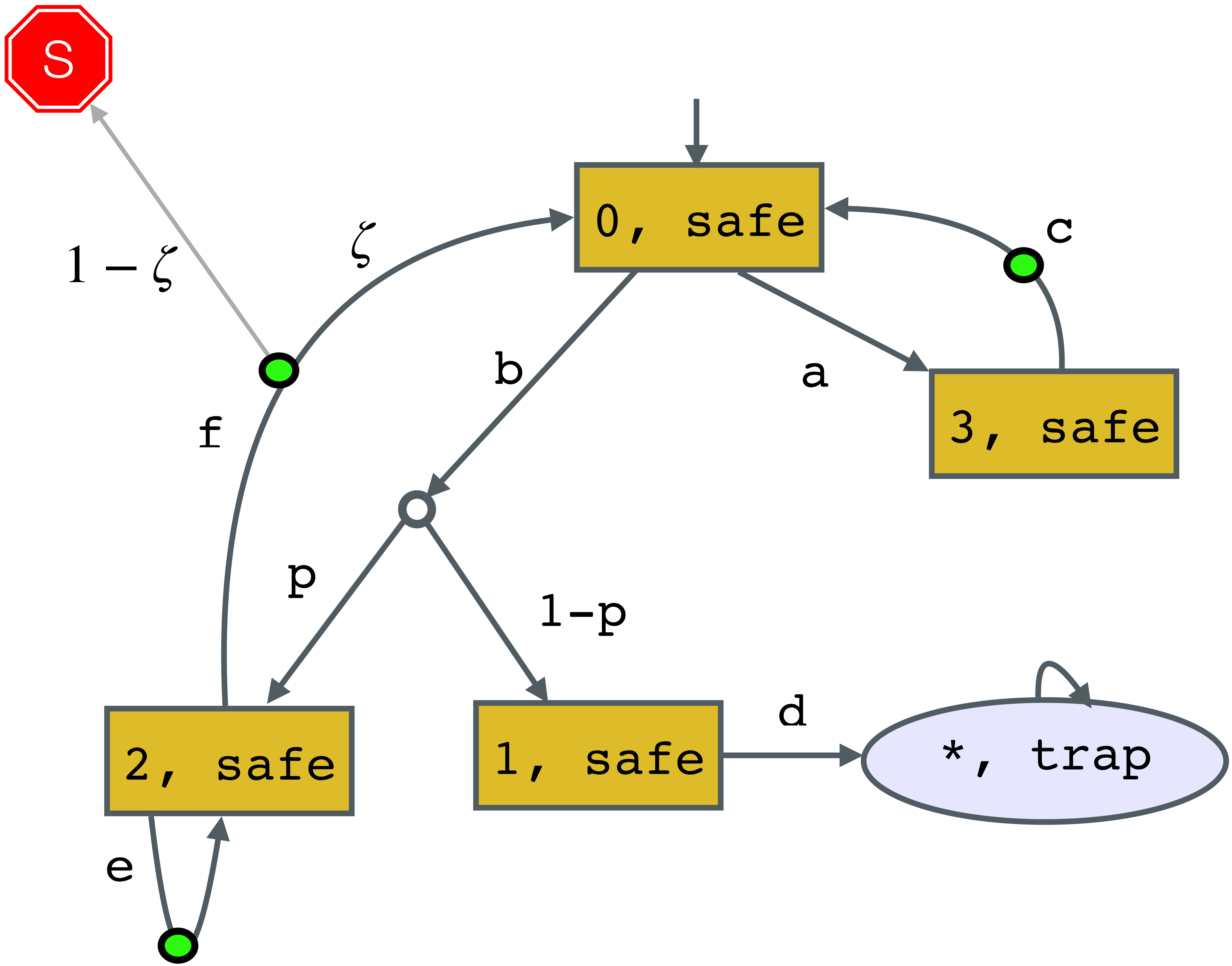


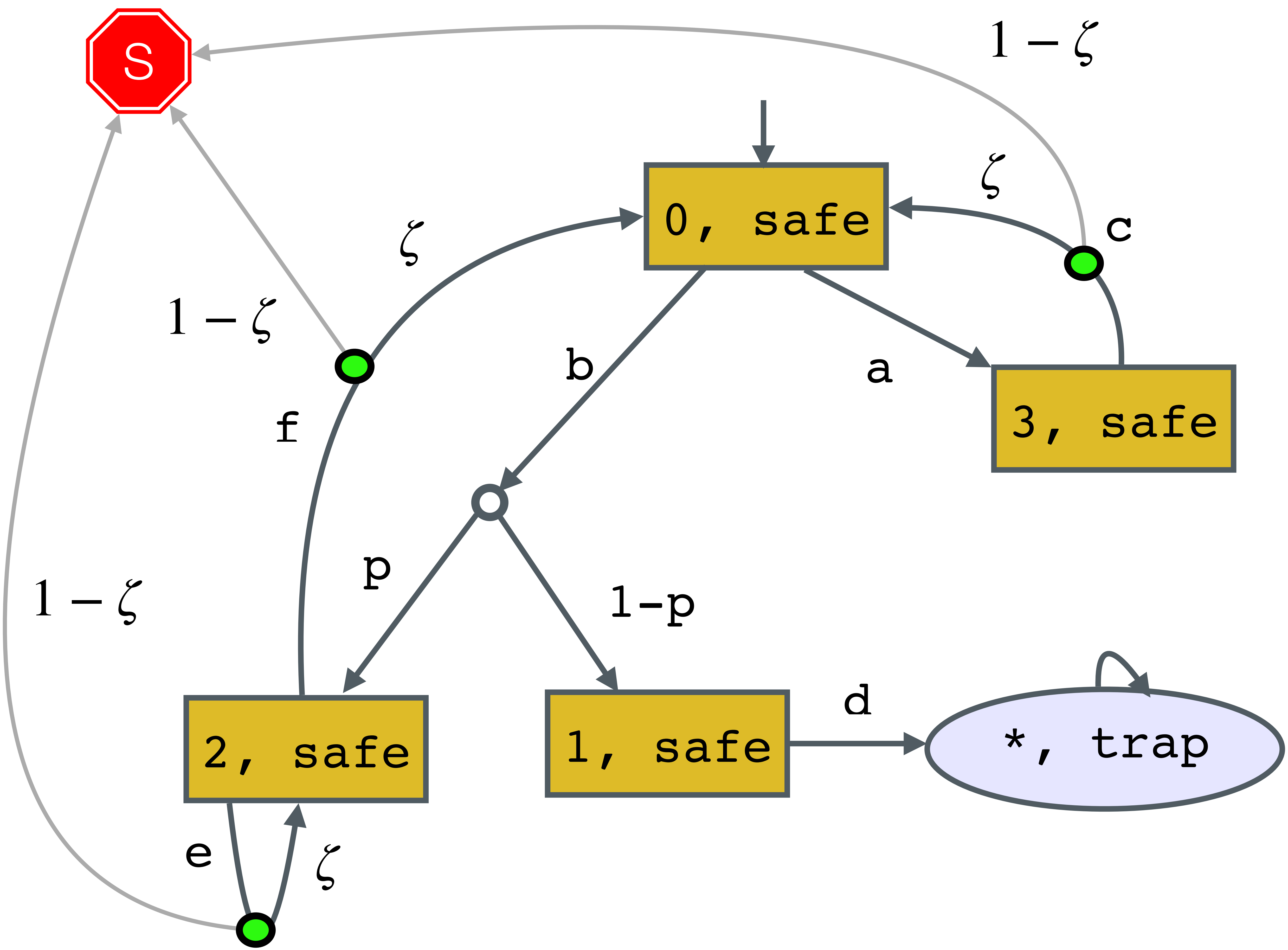
The optimal expected average policy may not always be the optimal policy for the LTL objective.

Limit Reachability Theorem

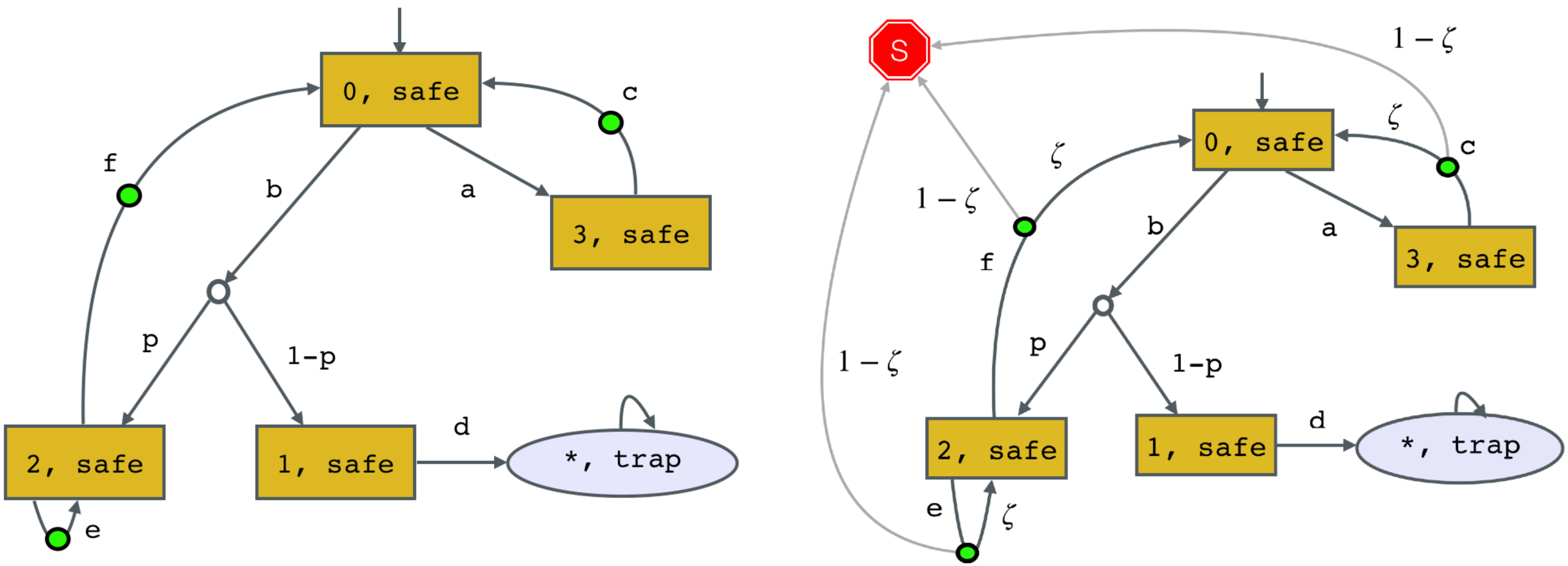
Ernst Moritz Hahn, Mateo Perez, Sven Schewe, Fabio Somenzi, Ashutosh Trivedi, Dominik Wojtczak: **Omega-Regular Objectives in Model-Free Reinforcement Learning.** TACAS (1) 2019: 395-412


S
0, safe
c
b
a
3, safe
f
p
1-p
d
2, safe
1, safe
*, trap
e

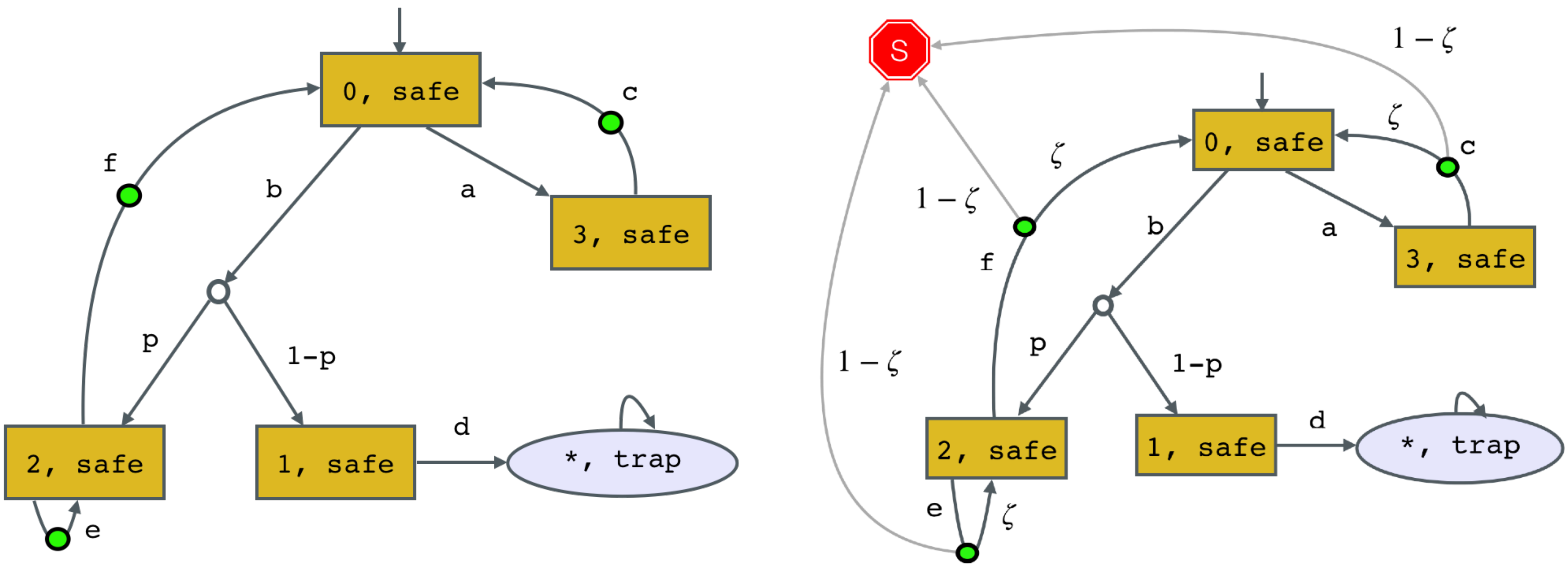

S
0, safe
3, safe
2, safe
1, safe
*, trap
$1-\zeta$
ζ
a
b
c
d
e
f
p
1-p



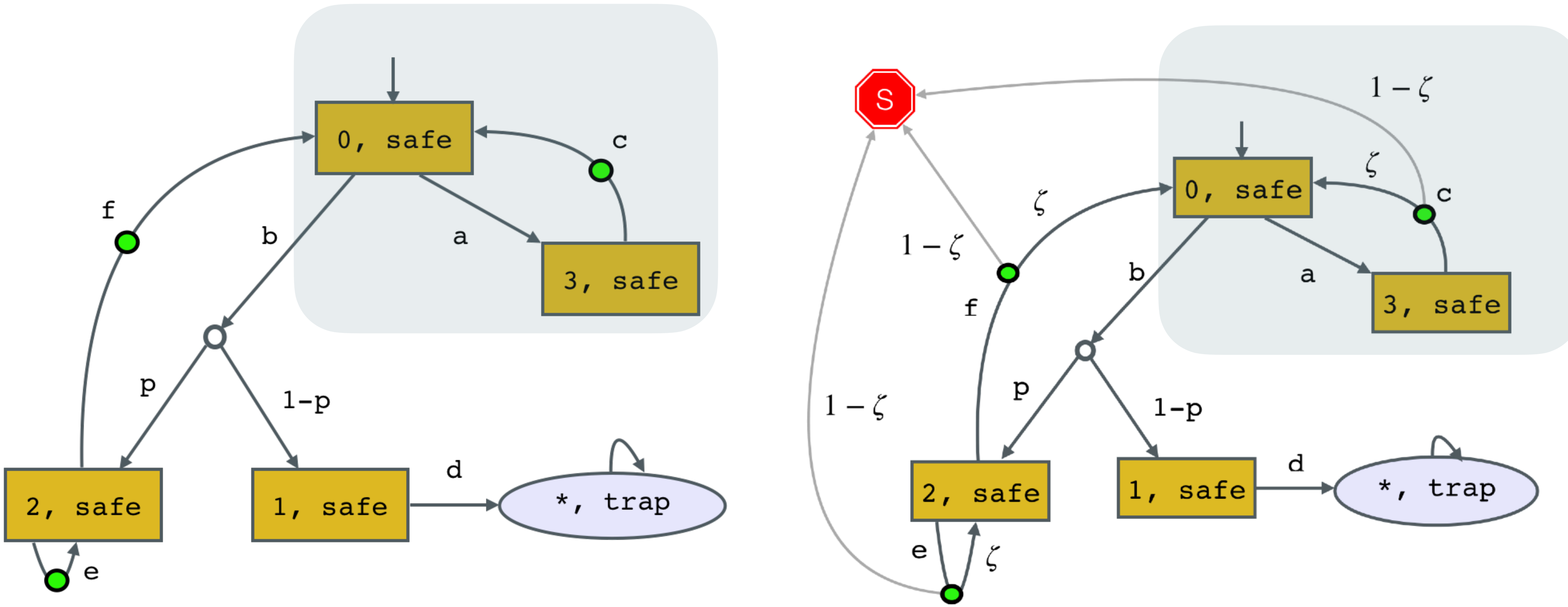
S
1 − ζ
ζ
c
0, safe
ζ
1 − ζ
b
a
3, safe
f
p
1-p
1 − ζ
d
2, safe
1, safe
*, trap
e
ζ



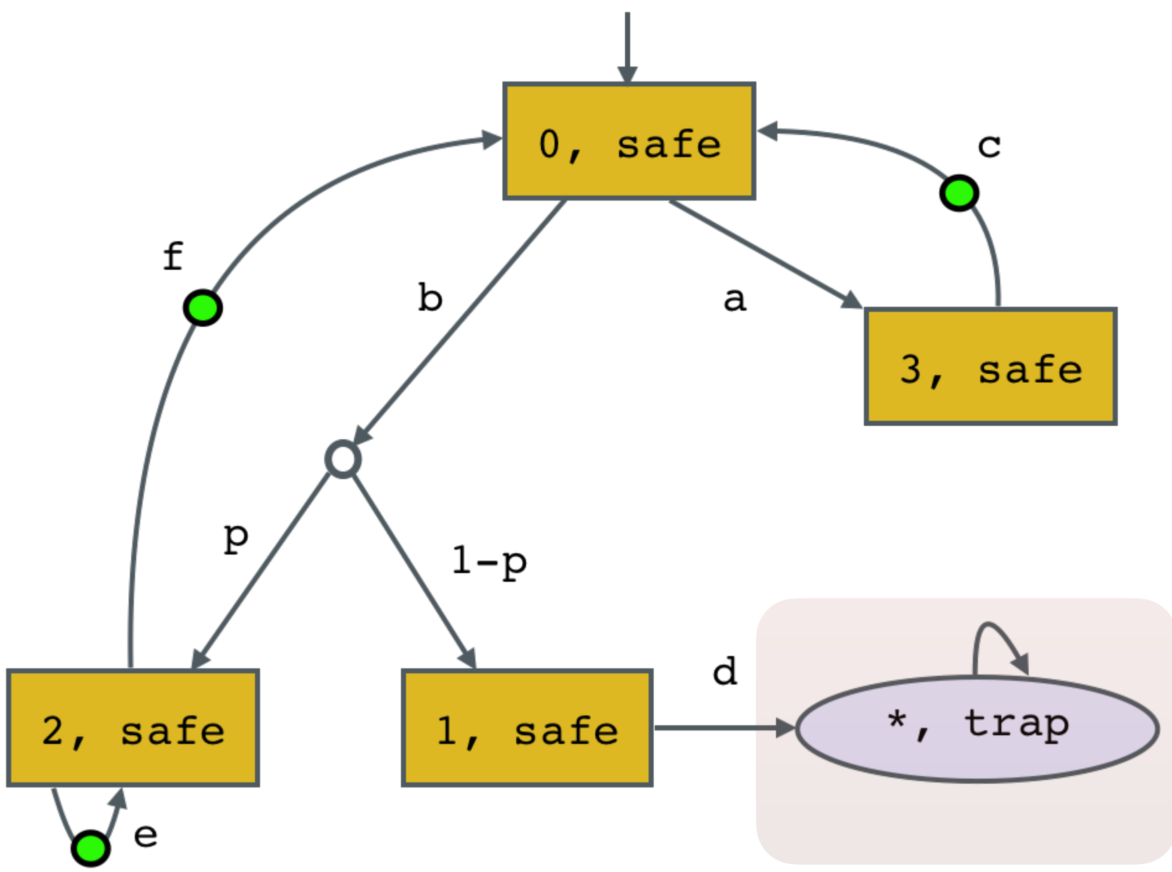
Theorem. For any value of ζ, the sink can be reached **almost-surely** iff the MDP satisfies LTL property **almost-surely**.

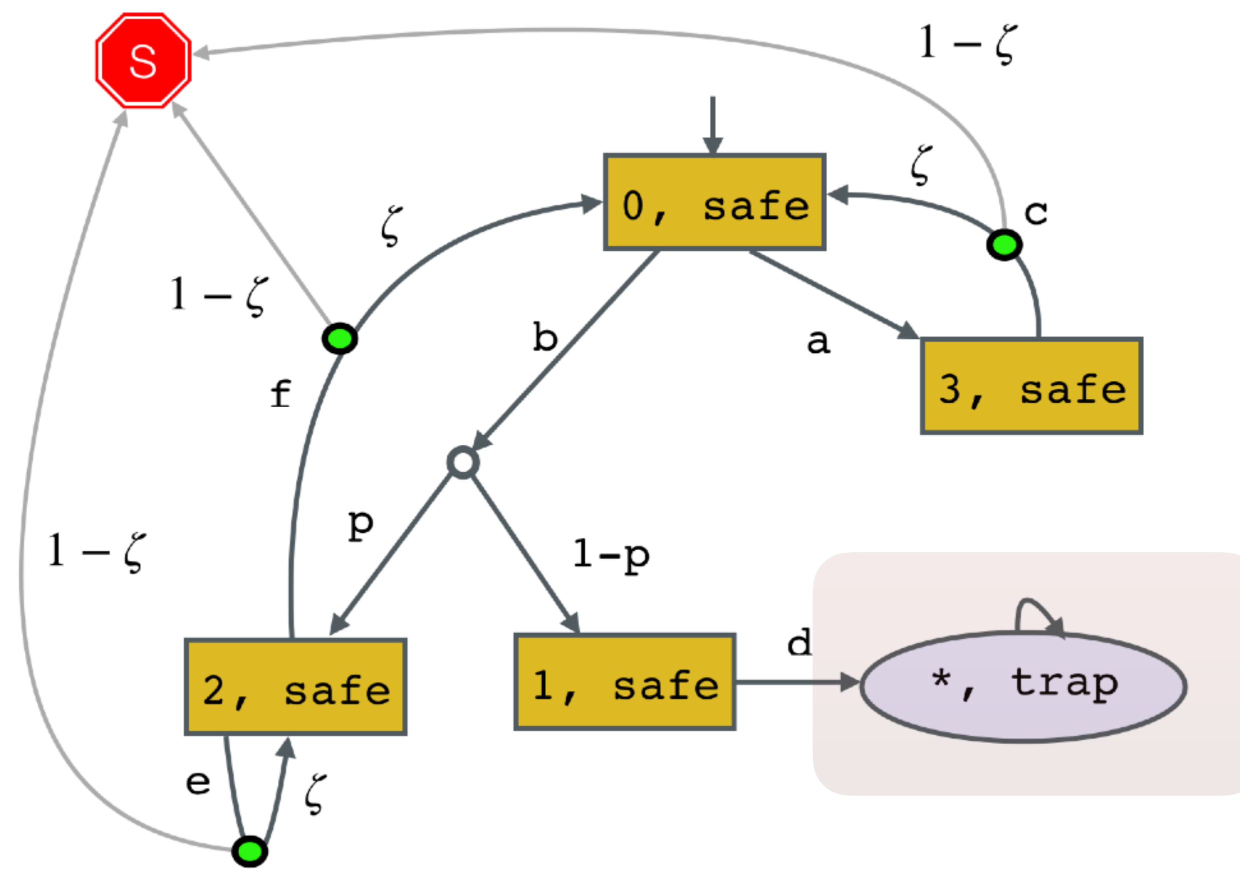


Theorem. The optimal probability of reaching the sink state as ζ tends to 1 equals the optimal probability of satisfying the LTL property.

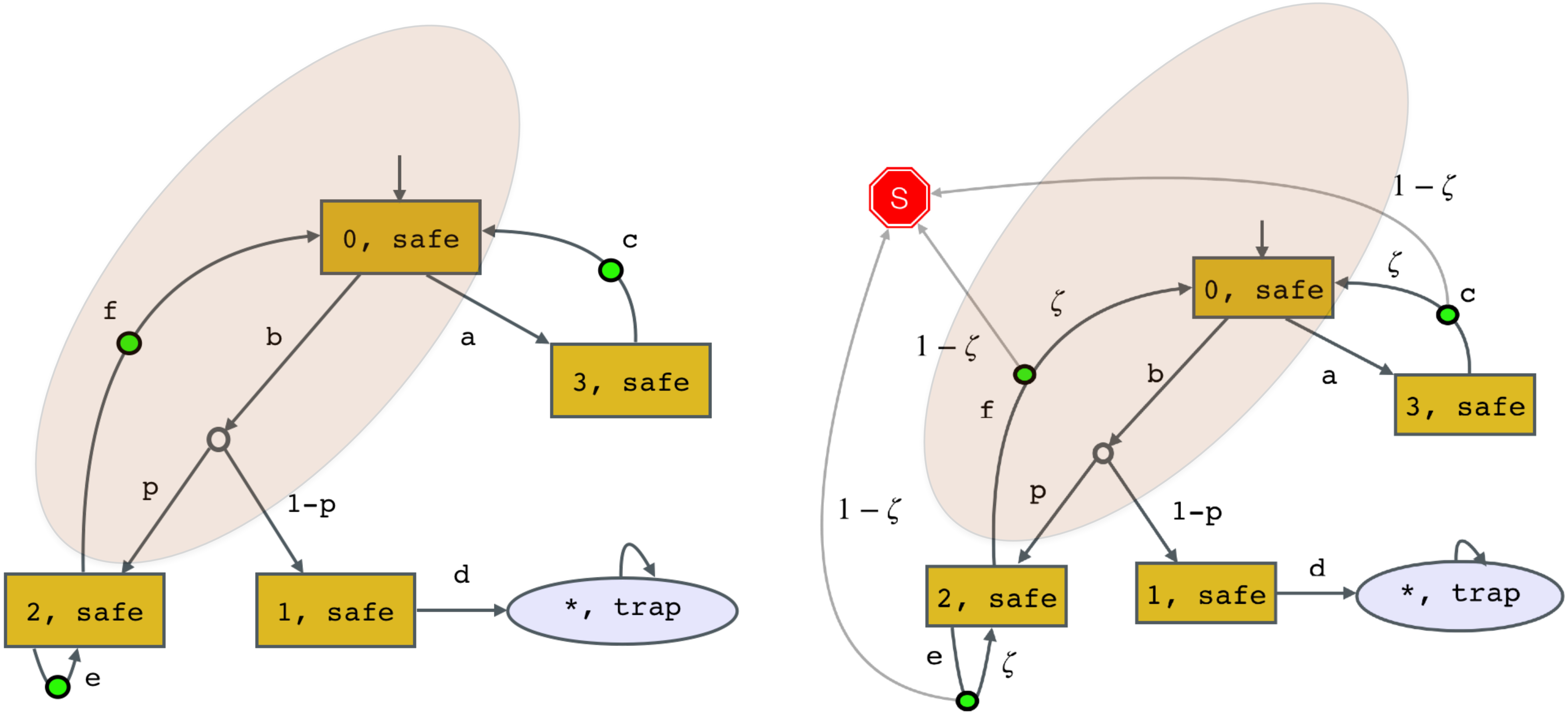


Theorem. The optimal probability of reaching the sink state as ζ tends to 1 equals the optimal probability of satisfying the LTL property.

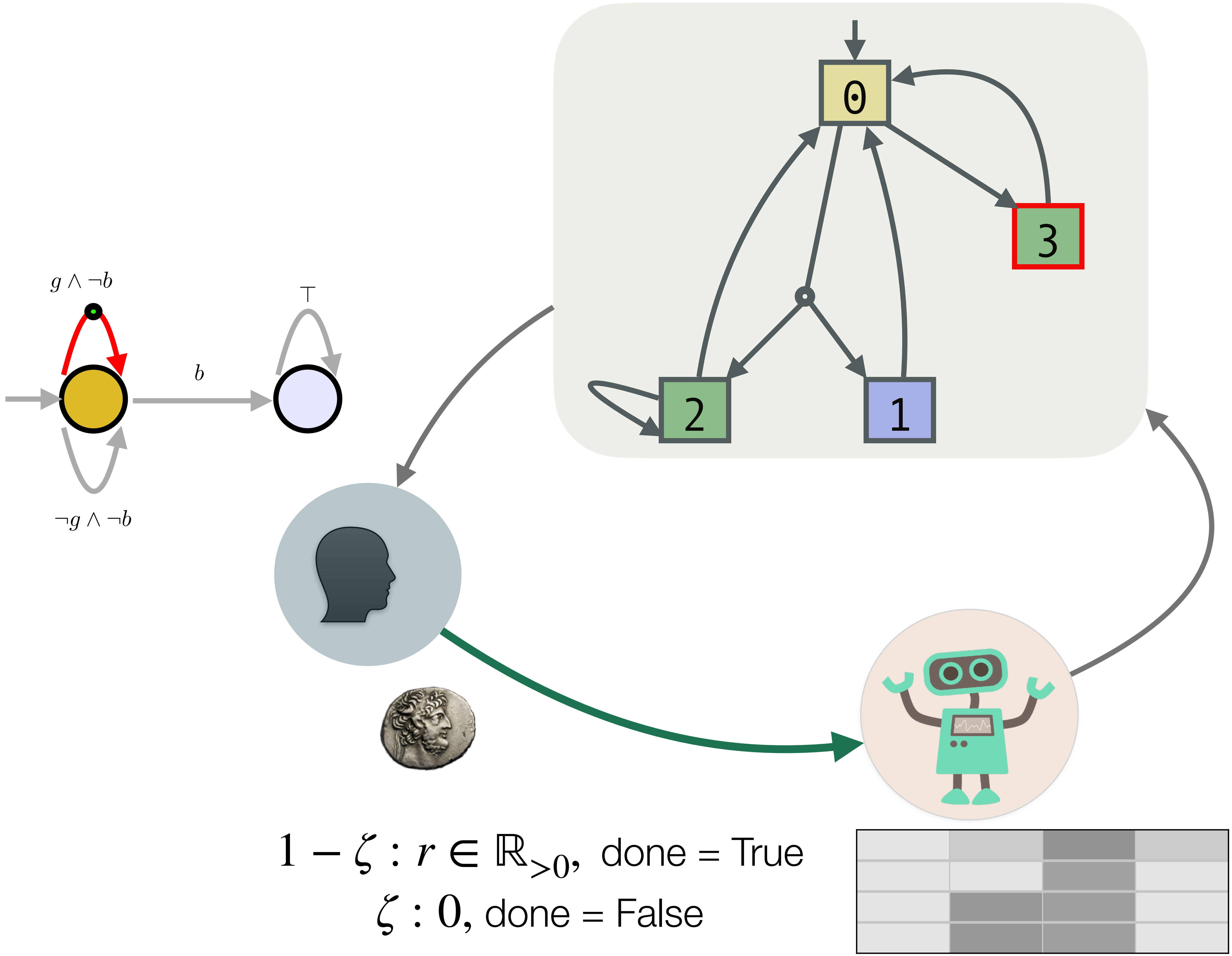


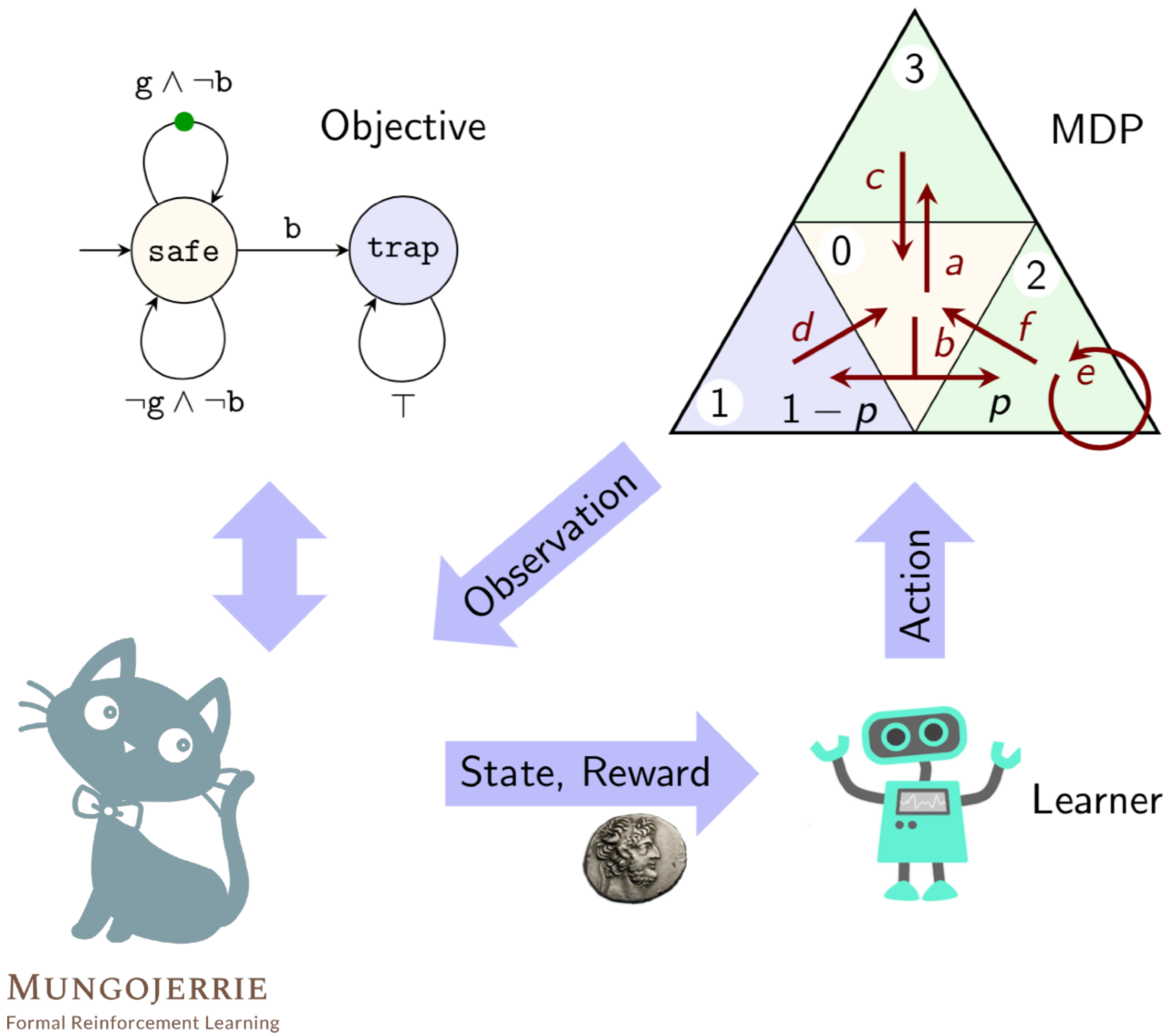


Theorem. The optimal probability of reaching the sink state as ζ tends to 1 equals the optimal probability of satisfying the LTL property.

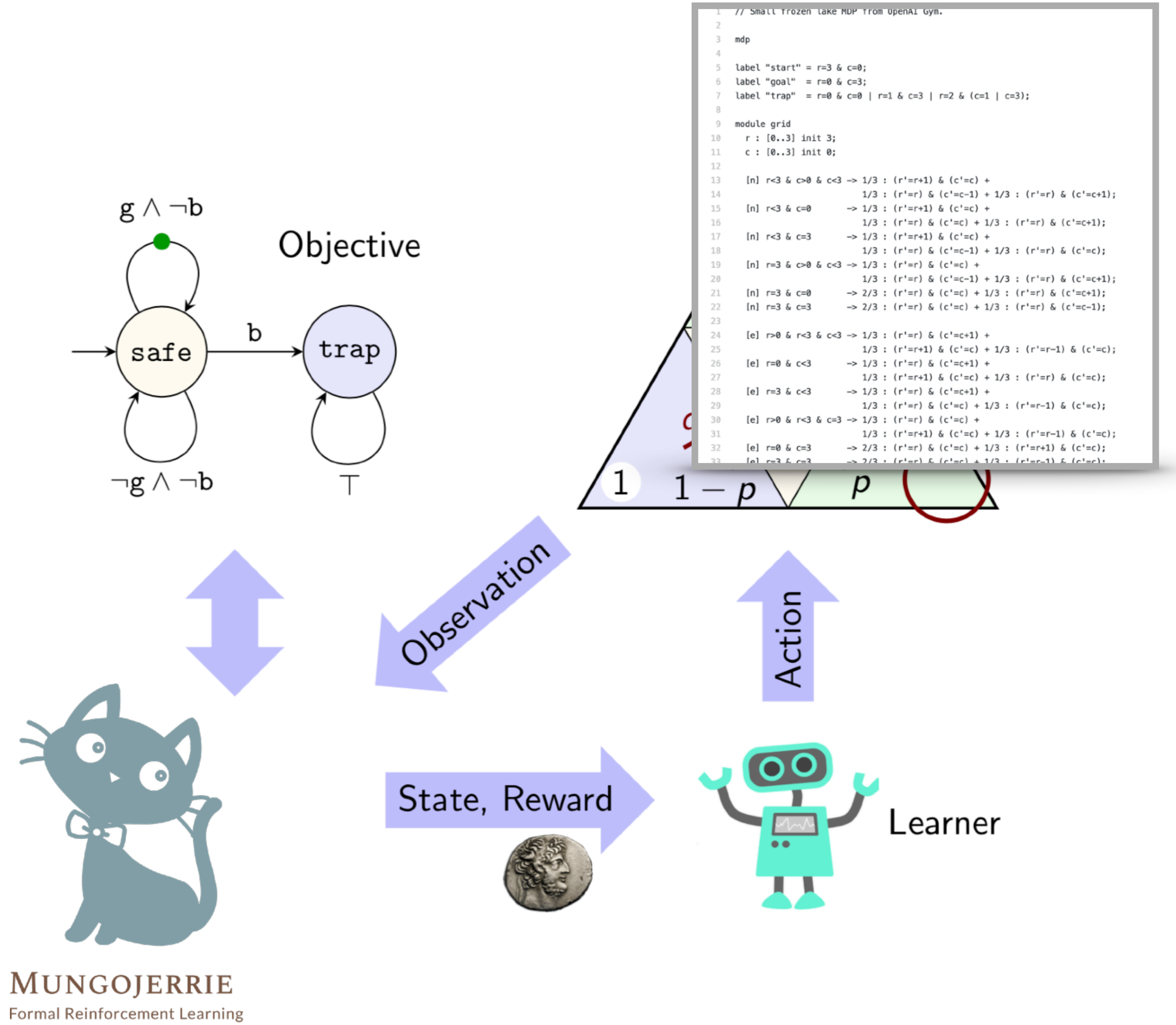


Theorem. The optimal probability of reaching the sink state as ζ tends to 1 equals the optimal probability of satisfying the LTL property.


$g \wedge \neg b$
b
$\top$
$\neg g \wedge \neg b$
0
3
2
1
$1 - \zeta : r \in \mathbb{R}_{>0}$, done = True
$\zeta : 0$, done = False



plv. colorado.edu/omega-regular-rl-benchmarks-2019

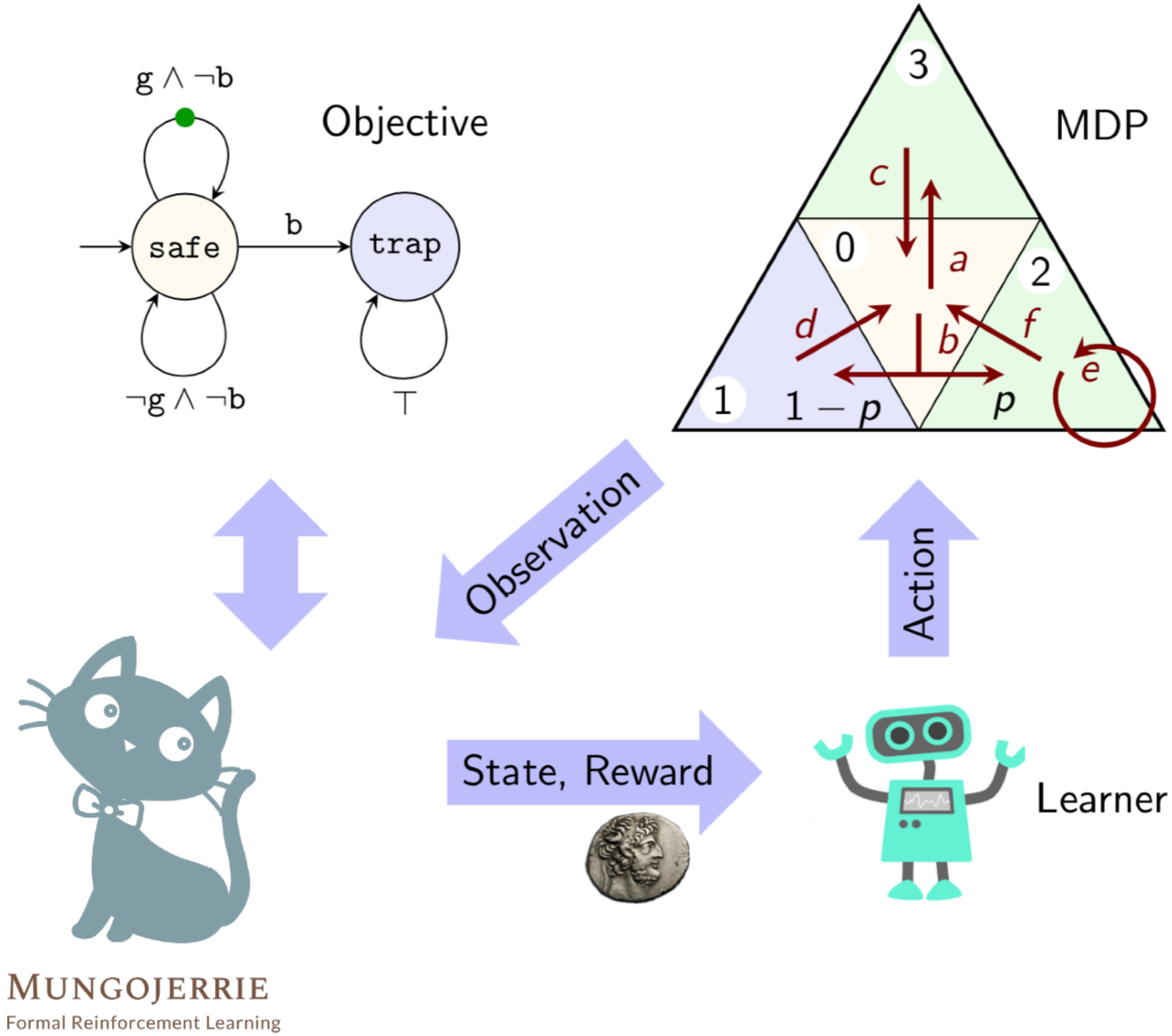



```
// Small frozen lake MDP from OpenAI Gym.

mdp

label "start" = r=3 & c=0;
label "goal"  = r=0 & c=3;
label "trap"  = r=0 & c=0 | r=1 & c=3 | r=2 & (c=1 | c=3);

module grid
  r : [0..3] init 3;
  c : [0..3] init 0;

  [n] r<3 & c>0 & c<3 -> 1/3 : (r'=r+1) & (c'=c) +
                         1/3 : (r'=r) & (c'=c-1) + 1/3 : (r'=r) & (c'=c+1);
  [n] r<3 & c=0       -> 1/3 : (r'=r+1) & (c'=c) +
                         1/3 : (r'=r) & (c'=c) + 1/3 : (r'=r) & (c'=c+1);
  [n] r<3 & c=3       -> 1/3 : (r'=r+1) & (c'=c) +
                         1/3 : (r'=r) & (c'=c-1) + 1/3 : (r'=r) & (c'=c);
  [n] r=3 & c>0 & c<3 -> 1/3 : (r'=r) & (c'=c) +
                         1/3 : (r'=r) & (c'=c-1) + 1/3 : (r'=r) & (c'=c+1);
  [n] r=3 & c=0       -> 2/3 : (r'=r) & (c'=c) + 1/3 : (r'=r) & (c'=c+1);
  [n] r=3 & c=3       -> 2/3 : (r'=r) & (c'=c) + 1/3 : (r'=r) & (c'=c-1);

  [e] r>0 & r<3 & c<3 -> 1/3 : (r'=r) & (c'=c+1) +
                         1/3 : (r'=r+1) & (c'=c) + 1/3 : (r'=r-1) & (c'=c);
  [e] r=0 & c<3       -> 1/3 : (r'=r) & (c'=c+1) +
                         1/3 : (r'=r+1) & (c'=c) + 1/3 : (r'=r) & (c'=c);
  [e] r=3 & c<3       -> 1/3 : (r'=r) & (c'=c+1) +
                         1/3 : (r'=r) & (c'=c) + 1/3 : (r'=r-1) & (c'=c);
  [e] r>0 & r<3 & c=3 -> 1/3 : (r'=r) & (c'=c) +
                         1/3 : (r'=r+1) & (c'=c) + 1/3 : (r'=r-1) & (c'=c);
  [e] r=0 & c=3       -> 2/3 : (r'=r) & (c'=c) + 1/3 : (r'=r+1) & (c'=c);
```

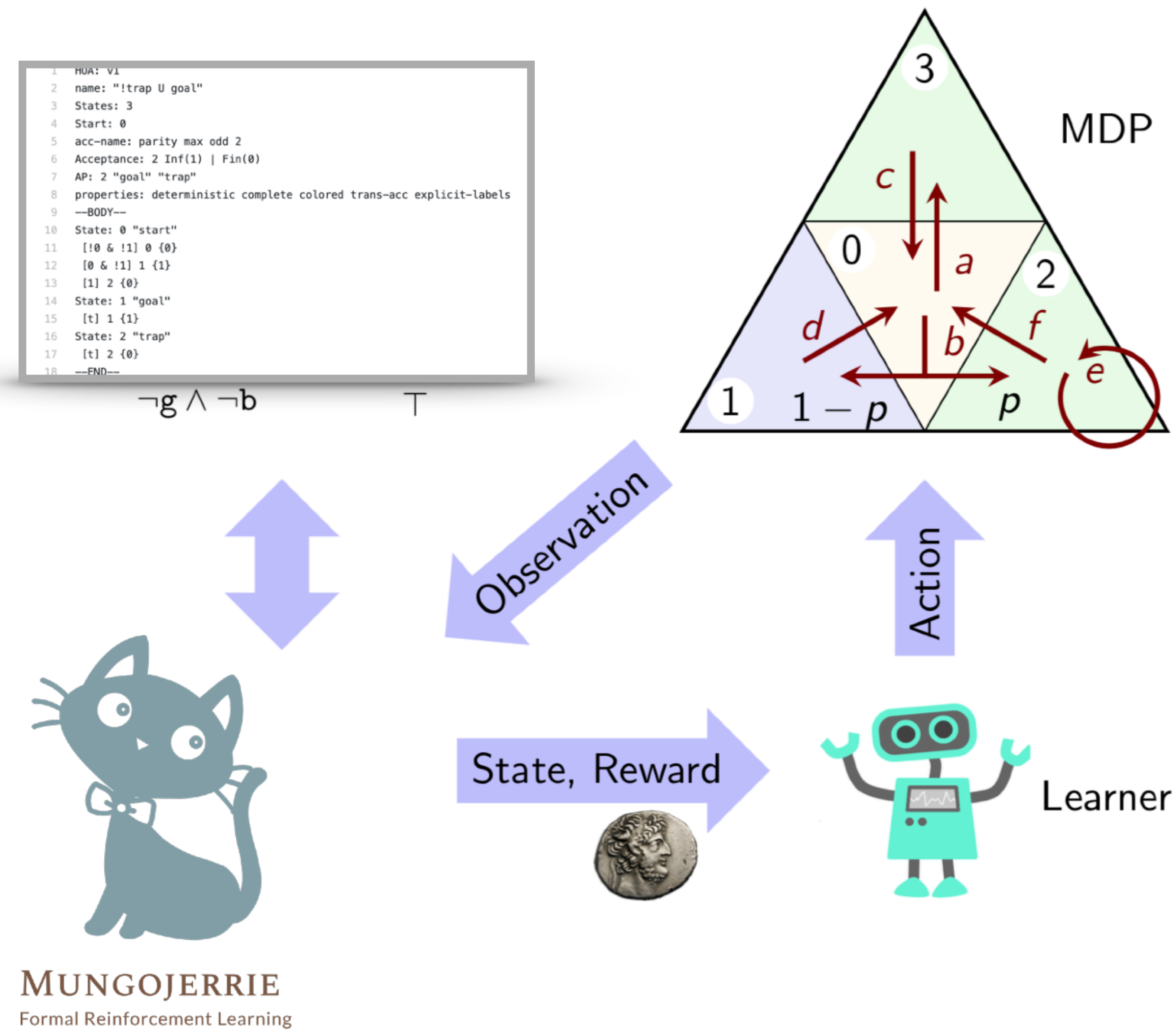
plv. colorado.edu/omega-regular-rl-benchmarks-2019

```
mdp

label "start" = r=3 & c=0;
label "goal"  = r=0 & c=3;
label "trap"  = r=0 & c=0 | r=1 & c=3 | r=2 & (c=1 | c=3);

module grid
  r : [0..3] init 3;
  c : [0..3] init 0;

  [n] r<3 & c>0 & c<3 -> 1/3 : (r'=r+1) & (c'=c) +
                         1/3 : (r'=r) & (c'=c-1) + 1/3 : (r'=r) & (c'=c+1);
  [n] r<3 & c=0       -> 1/3 : (r'=r+1) & (c'=c) +
                         1/3 : (r'=r) & (c'=c) + 1/3 : (r'=r) & (c'=c+1);
  [n] r<3 & c=3       -> 1/3 : (r'=r+1) & (c'=c) +
                         1/3 : (r'=r) & (c'=c-1) + 1/3 : (r'=r) & (c'=c);
  [n] r=3 & c>0 & c<3 -> 1/3 : (r'=r) & (c'=c) +
                         1/3 : (r'=r) & (c'=c-1) + 1/3 : (r'=r) & (c'=c+1);
  [n] r=3 & c=0       -> 2/3 : (r'=r) & (c'=c) + 1/3 : (r'=r) & (c'=c+1);
  [n] r=3 & c=3       -> 2/3 : (r'=r) & (c'=c) + 1/3 : (r'=r) & (c'=c-1);

  [e] r>0 & r<3 & c<3 -> 1/3 : (r'=r) & (c'=c+1) +
                         1/3 : (r'=r+1) & (c'=c) + 1/3 : (r'=r-1) & (c'=c);
  [e] r=0 & c<3       -> 1/3 : (r'=r) & (c'=c+1) +
                         1/3 : (r'=r+1) & (c'=c) + 1/3 : (r'=r) & (c'=c);
  [e] r=3 & c<3       -> 1/3 : (r'=r) & (c'=c+1) +
                         1/3 : (r'=r) & (c'=c) + 1/3 : (r'=r-1) & (c'=c);
  [e] r>0 & r<3 & c=3 -> 1/3 : (r'=r) & (c'=c) +
                         1/3 : (r'=r+1) & (c'=c) + 1/3 : (r'=r-1) & (c'=c);
  [e] r=0 & c=3       -> 2/3 : (r'=r) & (c'=c) + 1/3 : (r'=r+1) & (c'=c);
```

plv. colorado.edu/omega-regular-rl-benchmarks-2019



plv. colorado.edu/omega-regular-rl-benchmarks-2019

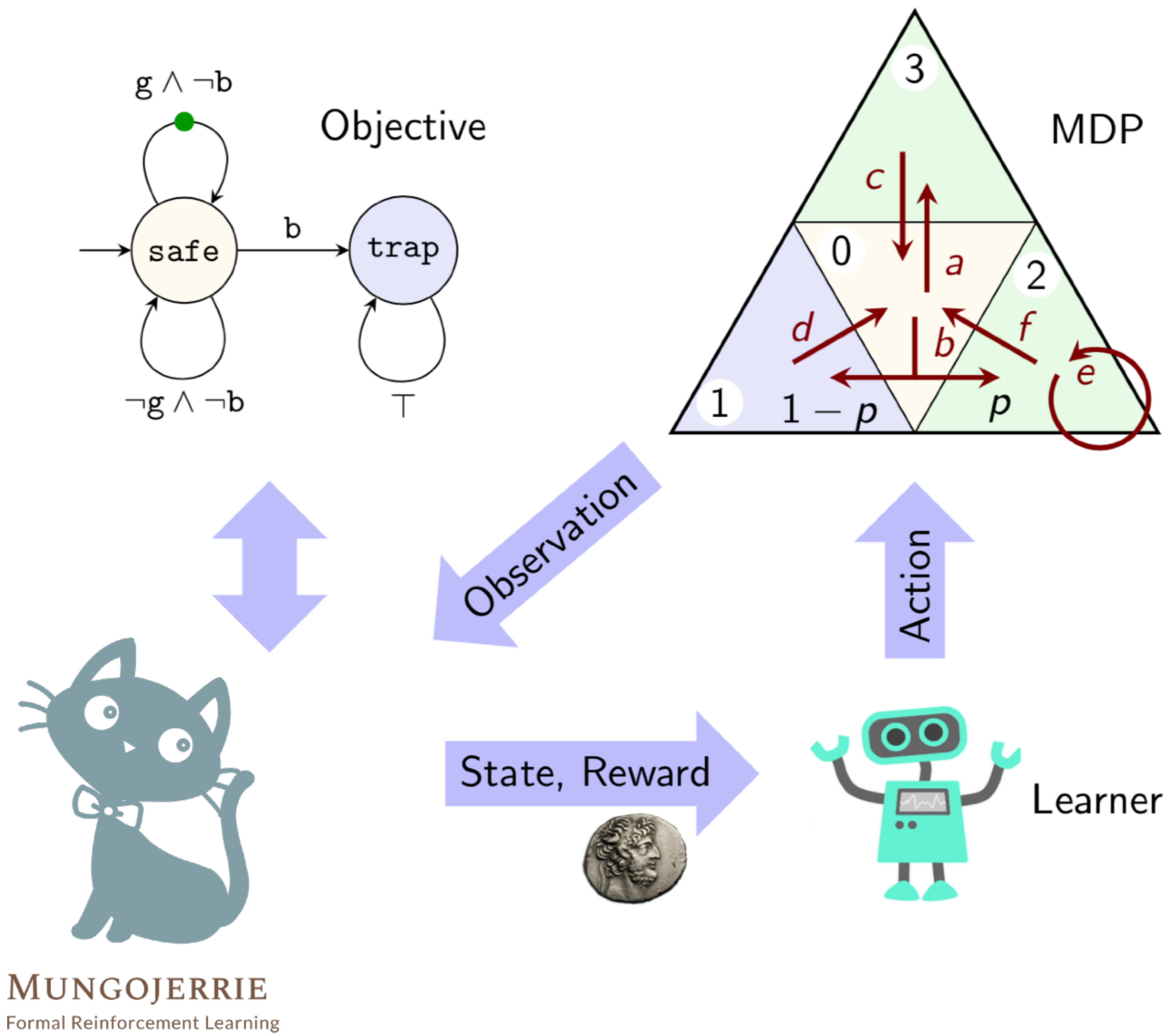


plv. colorado.edu/omega-regular-rl-benchmarks-2019

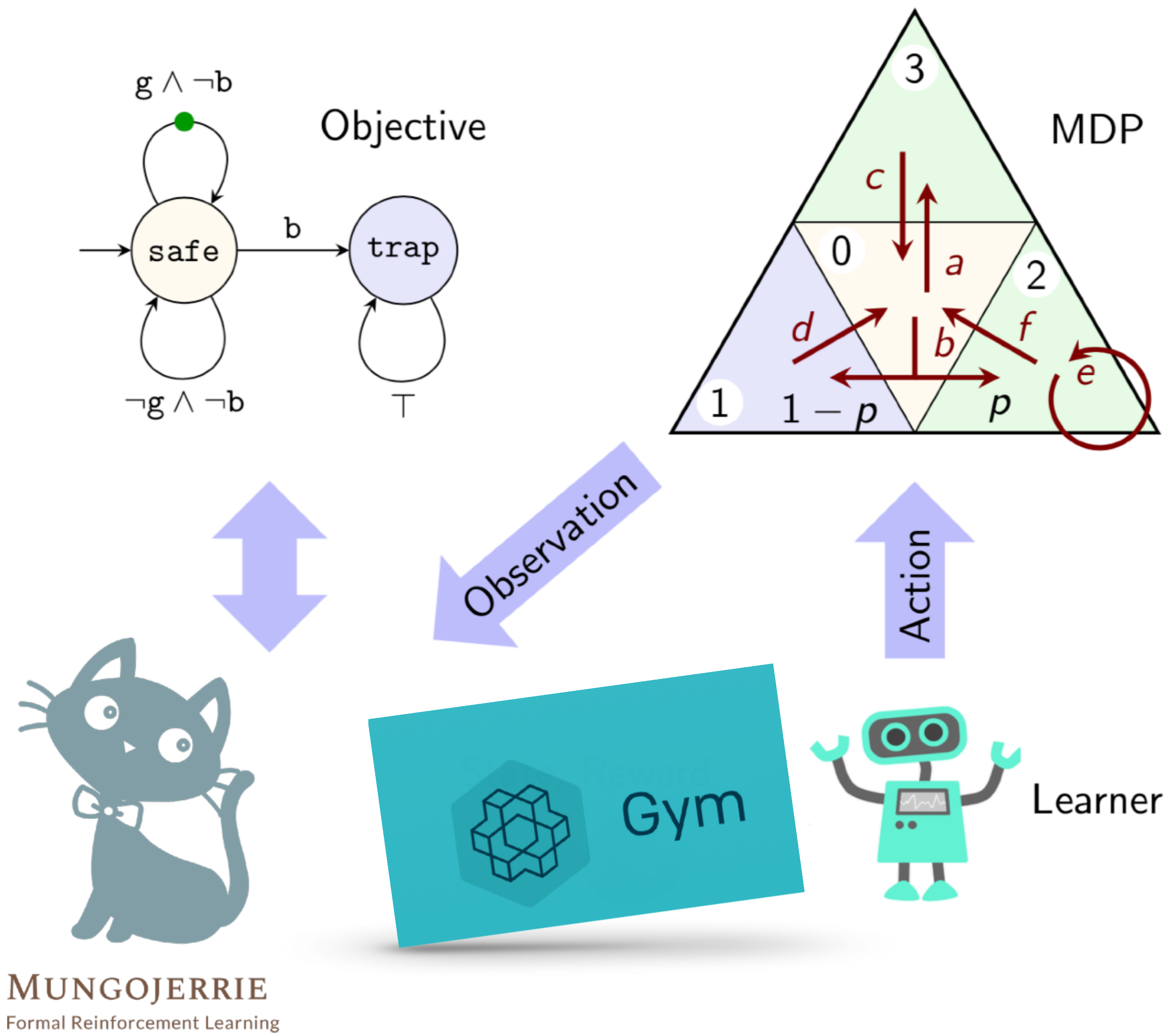
```
HOA: v1
name: "!trap U goal"
States: 3
Start: 0
acc-name: parity max odd 2
Acceptance: 2 Inf(1) | Fin(0)
AP: 2 "goal" "trap"
properties: deterministic complete colored trans-acc explicit-labels
--BODY--
State: 0 "start"
 [!0 & !1] 0 {0}
 [0 & !1] 1 {1}
 [1] 2 {0}
State: 1 "goal"
 [t] 1 {1}
State: 2 "trap"
 [t] 2 {0}
--END--
```

MUNGOJERRIE
Formal Reinforcement Learning

plv. colorado.edu/omega-regular-rl-benchmarks-2019



plv. colorado.edu/omega-regular-rl-benchmarks-2019



plv. colorado.edu/omega-regular-rl-benchmarks-2019

Name	states	aut.	prod.	prob.	est.	time	ζ	ε	α	tol	ep-l	ep-n
`twoPairs`	4	4	16	1	1	0.26						
`riskReward`	4	2	8	1	1	1.47						
`deferred`	41	1	41	1	1	1.01						
`grid5x5`	25	3	75	1	1	10.82		0.01	0.2		400	30k
`trafficNtk`	122	13	462	1	1	2.89						
`windy`	123	2	240	1	1	12.35	0.95	0.001	0.05	0	900	200k
`windyKing`	130	2	256	1	1	14.34	0.95	0.02	0.2	0	300	120k
`windyStoch`	130	2	260	1	1	47.70	0.95	0.02	0.2	0	300	200k
`frozenSmall`	16	3	48	0.823	0.83	0.51			0.05	0	200	
`frozenLarge`	64	3	192	1	1	1.81			0.05	0	700	
`othergrid6`	36	25	352	1	1	10.80				0	300	75k
`othergrid20`	400	25	3601	1	1	78.00	0.9999	0.07	0.2	0	5k	
`othergrid40`	1600	25	14401	1	0.99	87.90	0.9999	0.05	0.2	0	14k	25k
`doublegrid8`	4096	3	12287	1	1	45.50				0	3k	100k
`doublegrid12`	20736	3	62207	1	1	717.6				0	20k	300k
`slalom`	36	5	84	1	1	0.98						
`rps1`	121	2	130	0.768	0.76	5.21		0.12	0.006	0		500k
`dpenny`	52	2	65	0.5	0.5	1.99		0.001	0.2	0	50	120k
`devious`	11	1	11	1	1	0.81						
`arbiter2`	32	3	72	1	1	5.16			0.5	0.02	200	
`knuthYao`	13	3	39	1	1	0.31					100	
`threeWayDuel`	10	3	13	0.397	0.42	0.08						
`mutual4-14`	27600	128	384386	1	1	2.74						
`mutual4-15`	27600	527	780504	1	1	3.61						

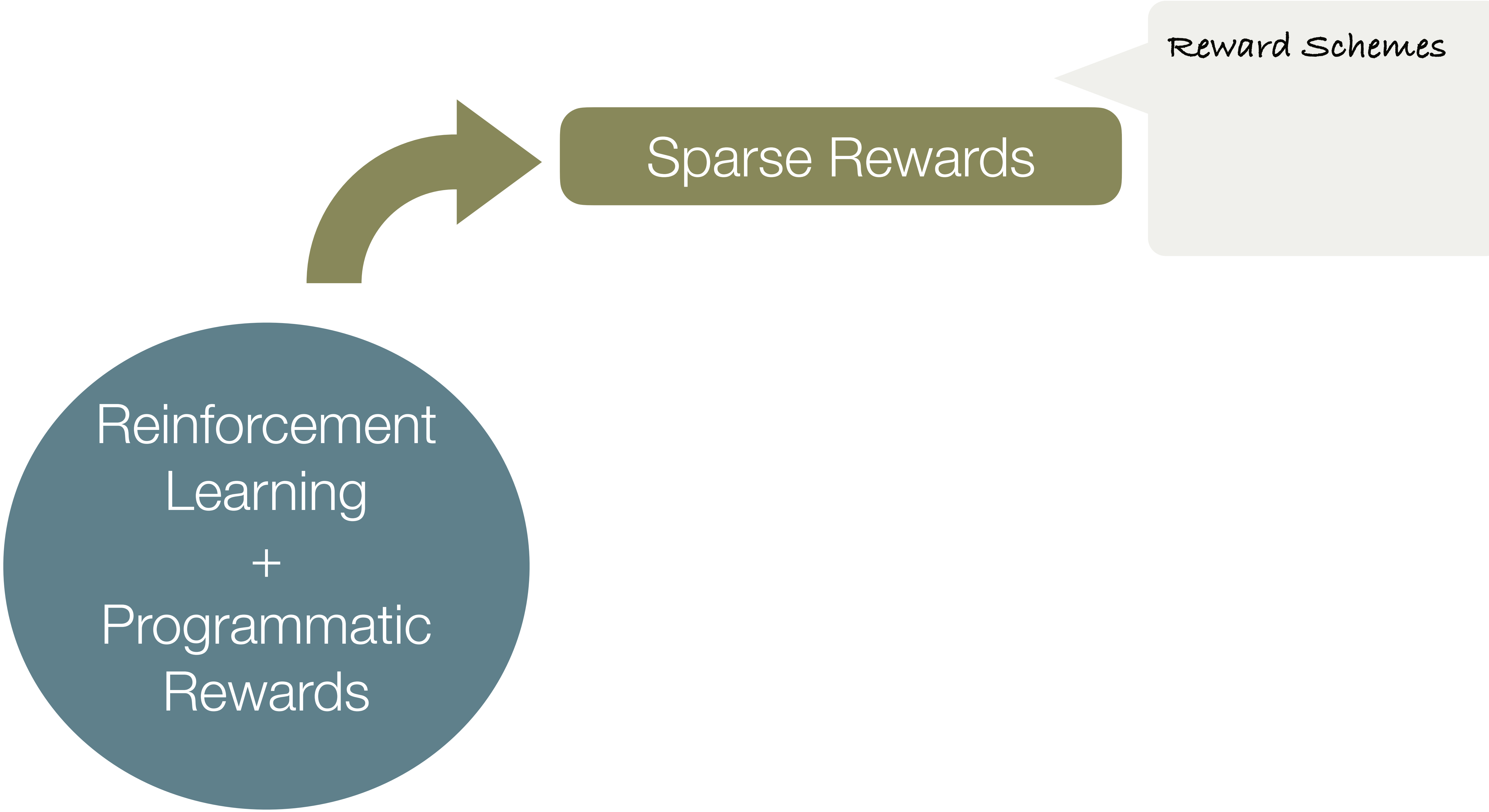
Reinforcement Learning

+

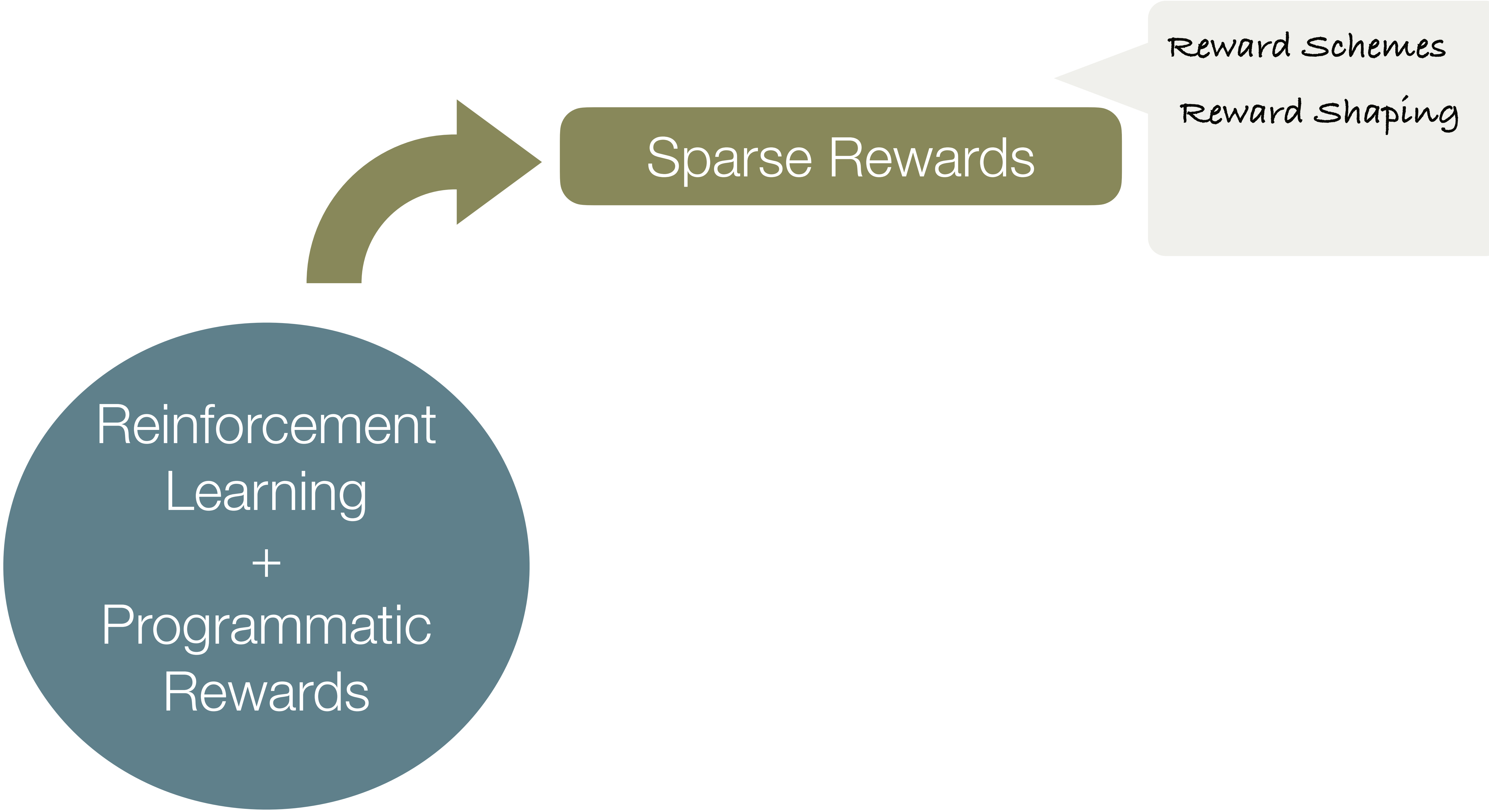
Programmatic Rewards

Reinforcement Learning
+
Programmatic Rewards

Sparse Rewards



Ernst Moritz Hahn, Mateo Perez, Sven Schewe, Fabio Somenzi, Ashutosh Trivedi, Dominik Wojtczak:
Faithful and Effective Reward Schemes for Model-Free Reinforcement Learning of Omega-Regular Objectives. ATVA 2020: 108-124



Ernst Moritz Hahn, Mateo Perez, Sven Schewe, Fabio Somenzi, Ashutosh Trivedi, Dominik Wojtczak:
Faithful and Effective Reward Schemes for Model-Free Reinforcement Learning of Omega-Regular Objectives. ATVA 2020: 108-124

Abolfazl Lavaei, Fabio Somenzi, Sadegh Soudjani, Ashutosh Trivedi, Majid Zamani:
Formal Controller Synthesis for Continuous-Space MDPs via Model-Free Reinforcement Learning. ICCPS 2020: 98-107

Sparse Rewards

Reward Schemes

Reward Shaping

Average-Reward RL

Reinforcement Learning
+
Programmatic Rewards

Ernst Moritz Hahn, Mateo Perez, Sven Schewe, Fabio Somenzi, Ashutosh Trivedi, Dominik Wojtczak:
Faithful and Effective Reward Schemes for Model-Free Reinforcement Learning of Omega-Regular Objectives. ATVA 2020: 108-124

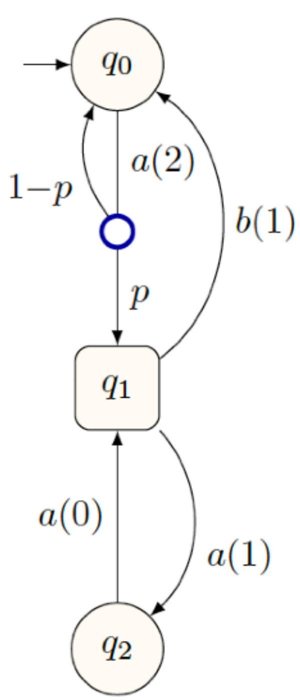
Abolfazl Lavaei, Fabio Somenzi, Sadegh Soudjani, Ashutosh Trivedi, Majid Zamani:
Formal Controller Synthesis for Continuous-Space MDPs via Model-Free Reinforcement Learning. ICCPS 2020: 98-107

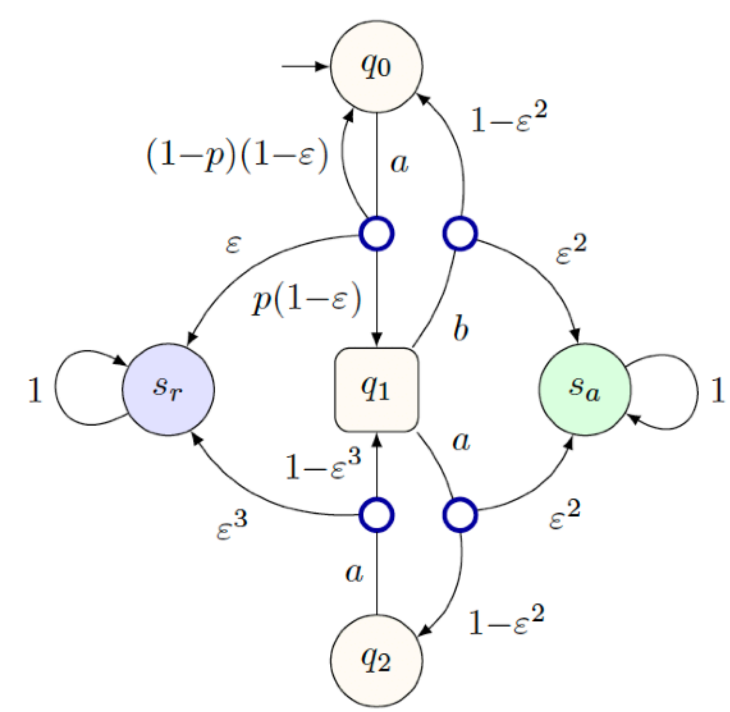
Milad Kazemi, Mateo Perez, Fabio Somenzi, Sadegh Soudjani, Ashutosh Trivedi, Alvaro Velasquez:
Translating Omega-Regular Specifications to Average Objectives for Model-Free Reinforcement Learning. AAMAS 2022: 732-741

Sparse Rewards

Multiplayer Games

Reinforcement Learning
+
Programmatic Rewards



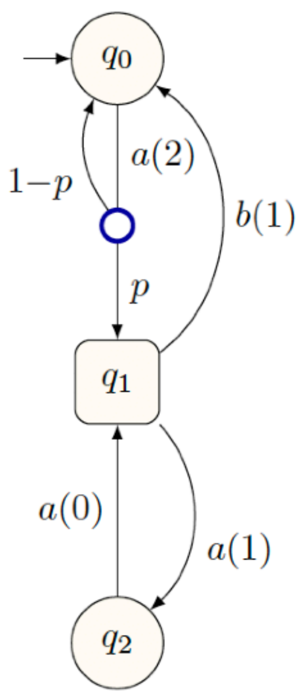


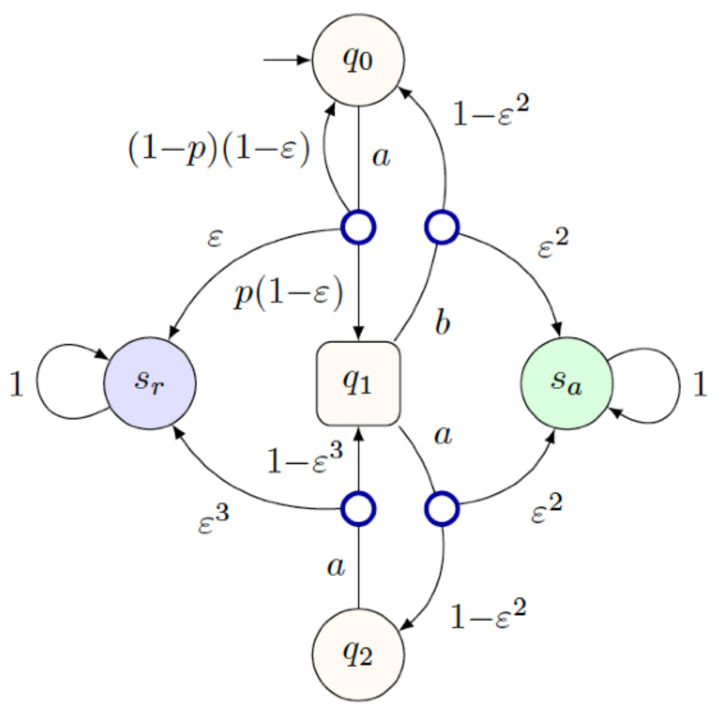
Sparse Rewards

Stochastic parity games

Multiplayer Games

Reinforcement Learning + Programmatic Rewards





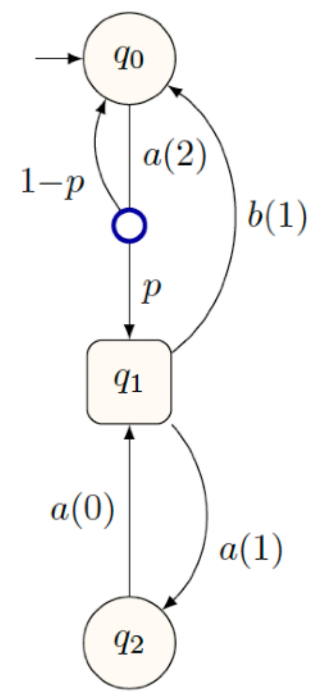
Ernst Moritz Hahn, Mateo Perez, Sven Schewe, Fabio Somenzi, Ashutosh Trivedi, Dominik Wojtczak: **Model-Free Reinforcement Learning for Stochastic Parity Games.** CONCUR 2020: 21:1-21:16

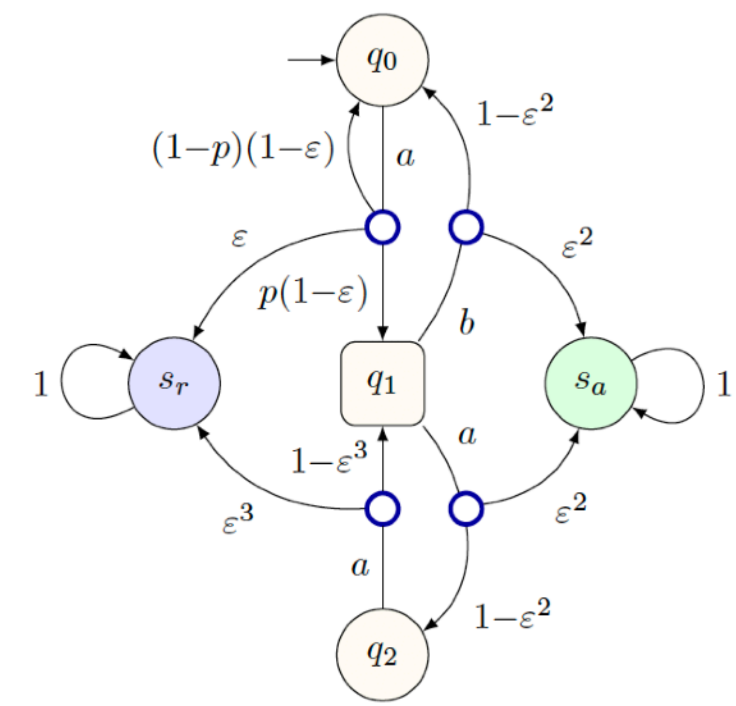
Sparse Rewards

Stochastic parity games
Multi-objective RL

Multiplayer Games

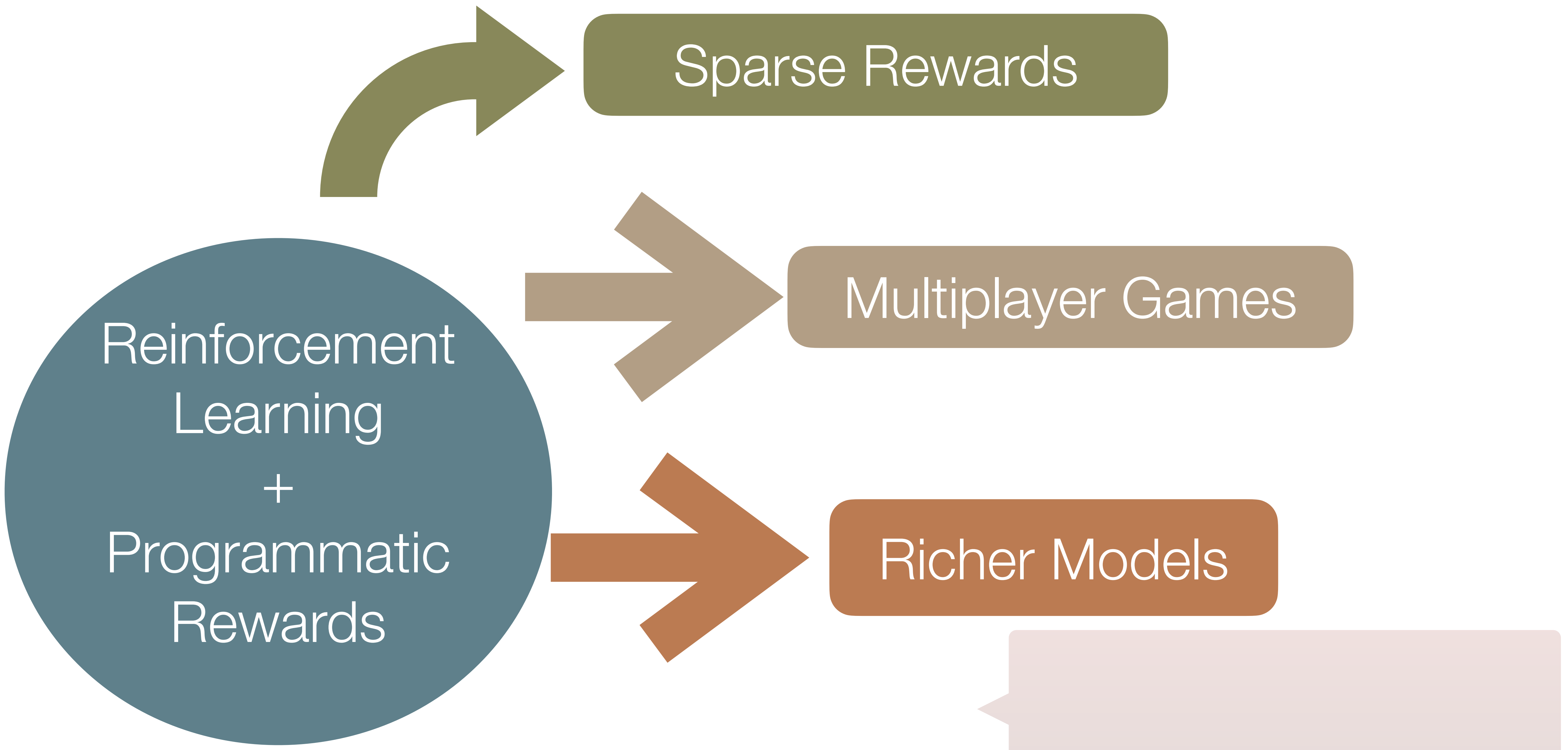
Reinforcement Learning + Programmatic Rewards

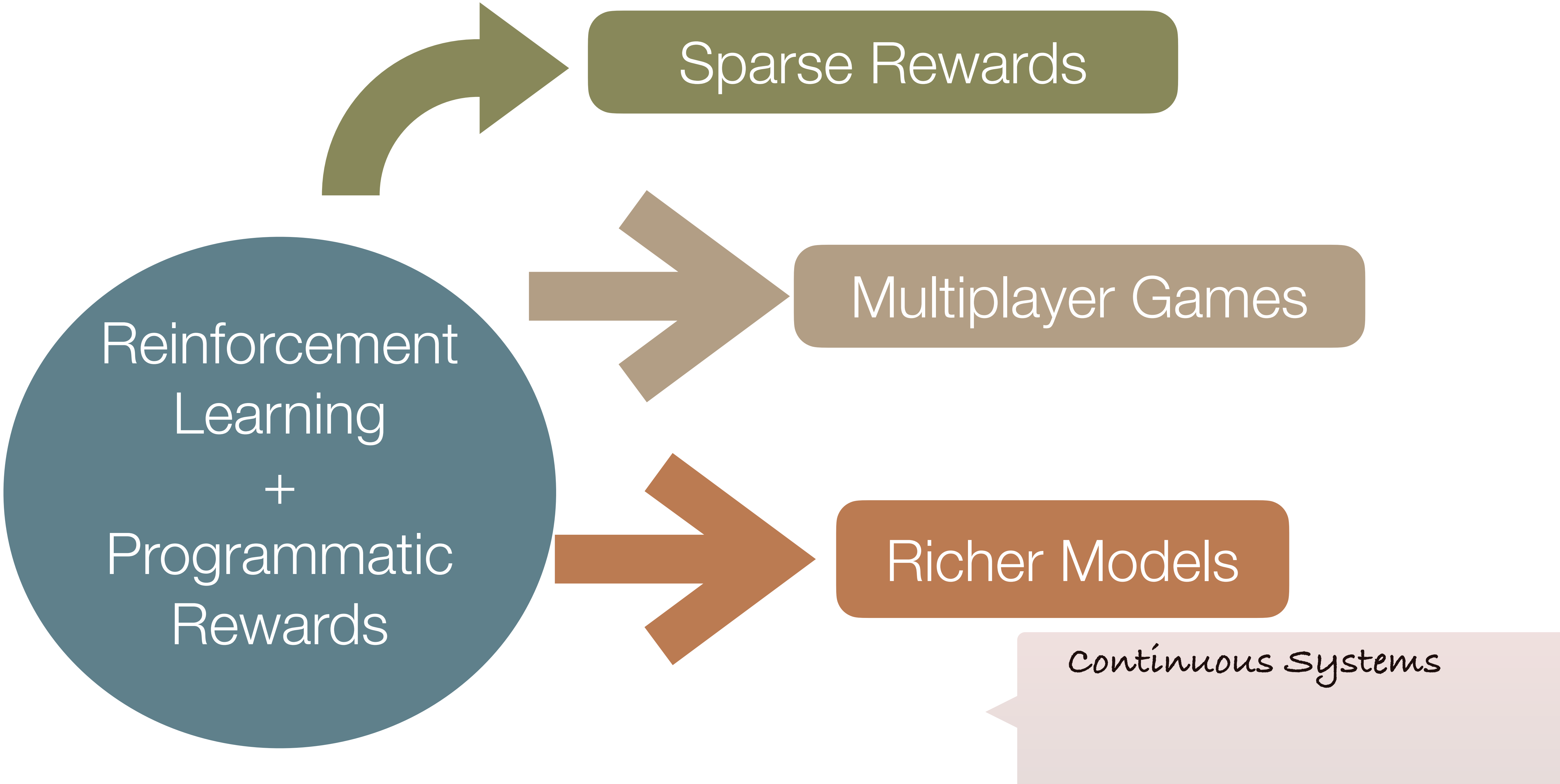




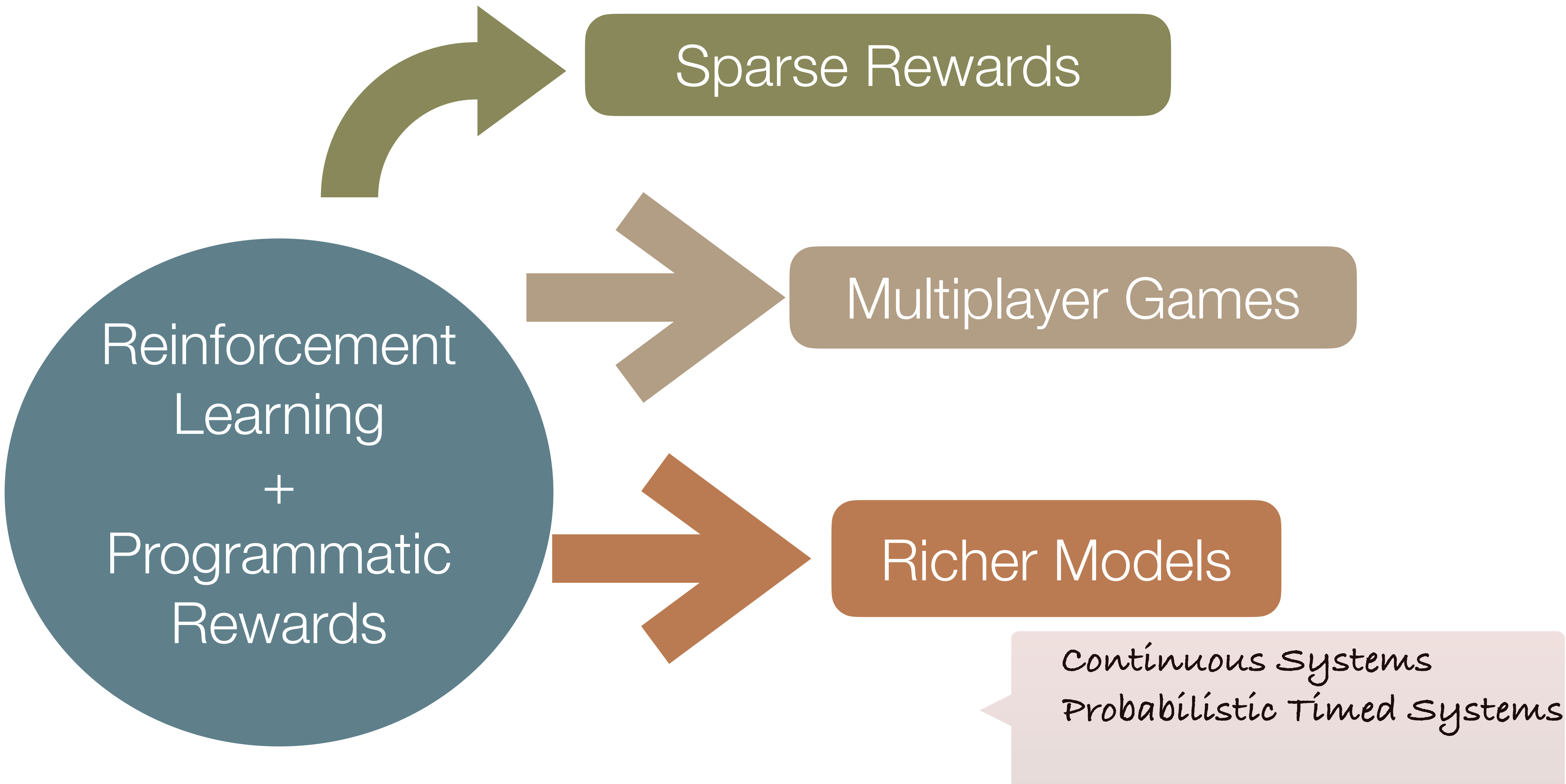
Ernst Moritz Hahn, Mateo Perez, Sven Schewe, Fabio Somenzi, Ashutosh Trivedi, Dominik Wojtczak:
Model-Free Reinforcement Learning for Stochastic Parity Games. CONCUR 2020: 21:1-21:16

Ernst Moritz Hahn, Mateo Perez, Sven Schewe, Fabio Somenzi, Ashutosh Trivedi, Dominik Wojtczak:
Model-Free Reinforcement Learning for Lexicographic Omega-Regular Objectives. FM 2021: 142-159


Sparse Rewards
Multiplayer Games
Reinforcement Learning + Programmatic Rewards
Richer Models

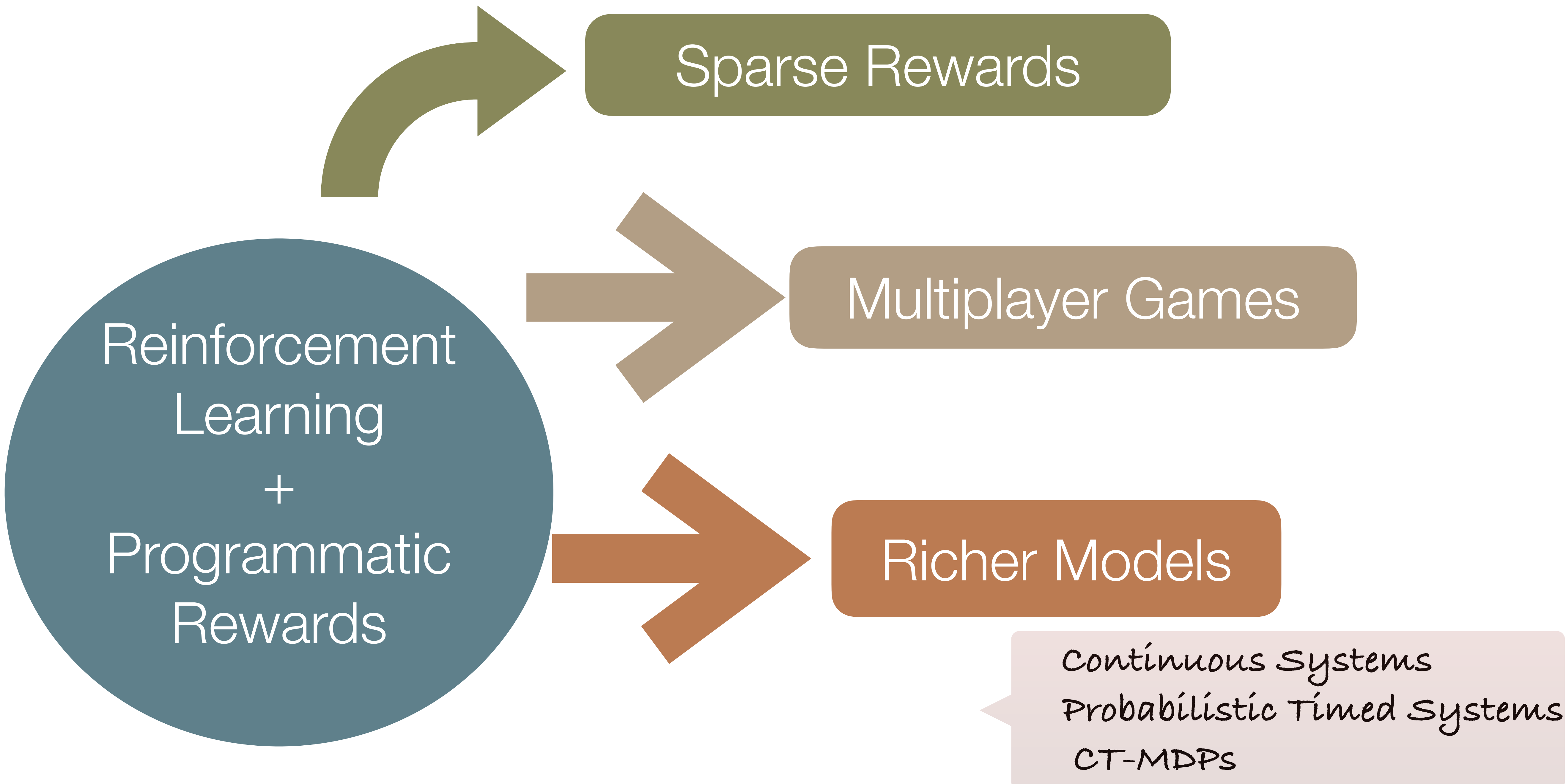


Abolfazl Lavaei, Fabio Somenzi, Sadegh Soudjani, Ashutosh Trivedi, Majid Zamani:
Formal Controller Synthesis for Continuous-Space MDPs via Model-Free Reinforcement Learning. ICCPS 2020: 98-107



Abolfazl Lavaei, Fabio Somenzi, Sadegh Soudjani, Ashutosh Trivedi, Majid Zamani:
Formal Controller Synthesis for Continuous-Space MDPs via Model-Free Reinforcement Learning. ICCPS 2020: 98-107

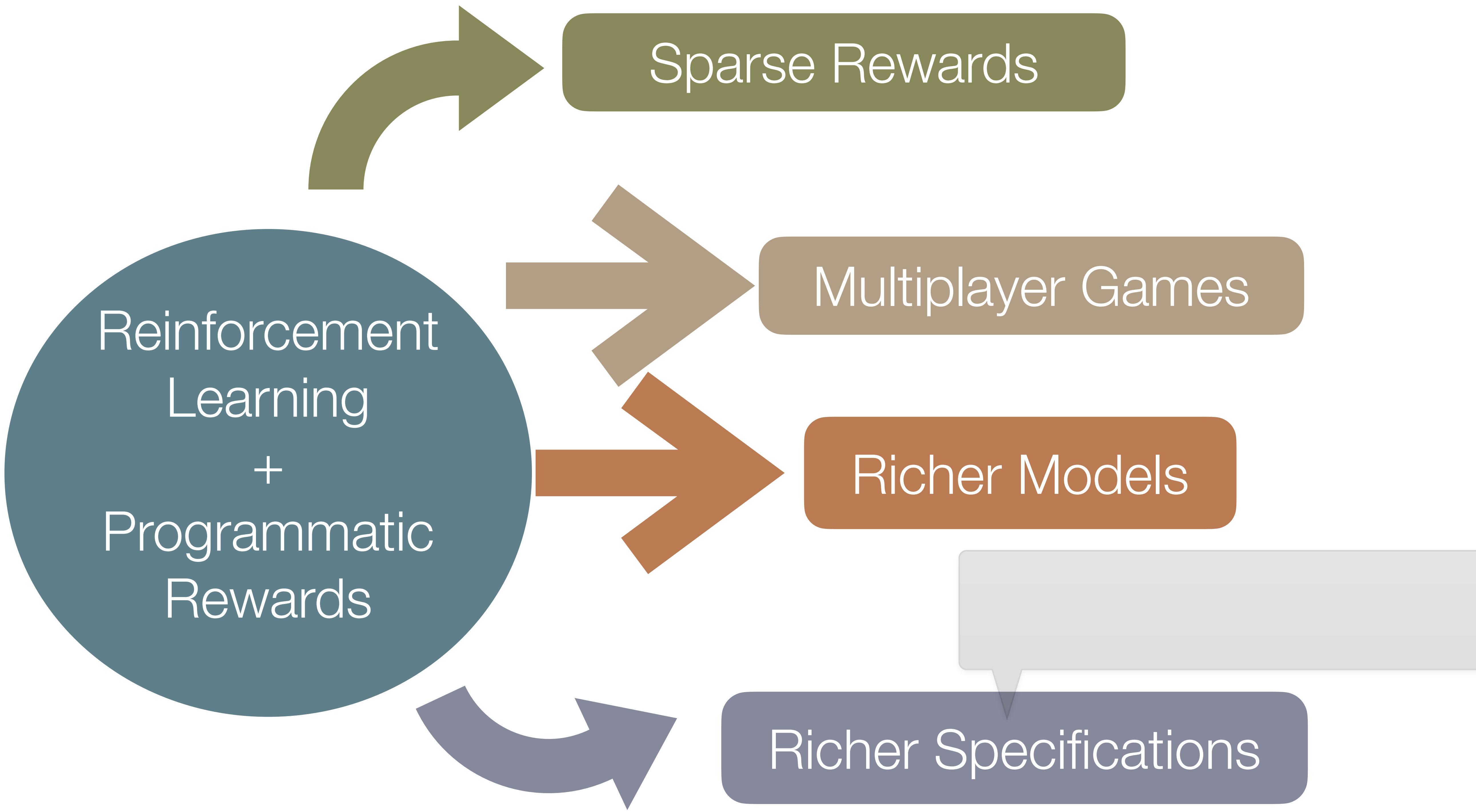
Kalyani Dole, Ashutosh Gupta, John Komp, Shankara Narayanan Krishna, Ashutosh Trivedi:
Event-Triggered and Time-Triggered Duration Calculus for Model-Free Reinforcement Learning. RTSS 2021: 240-252

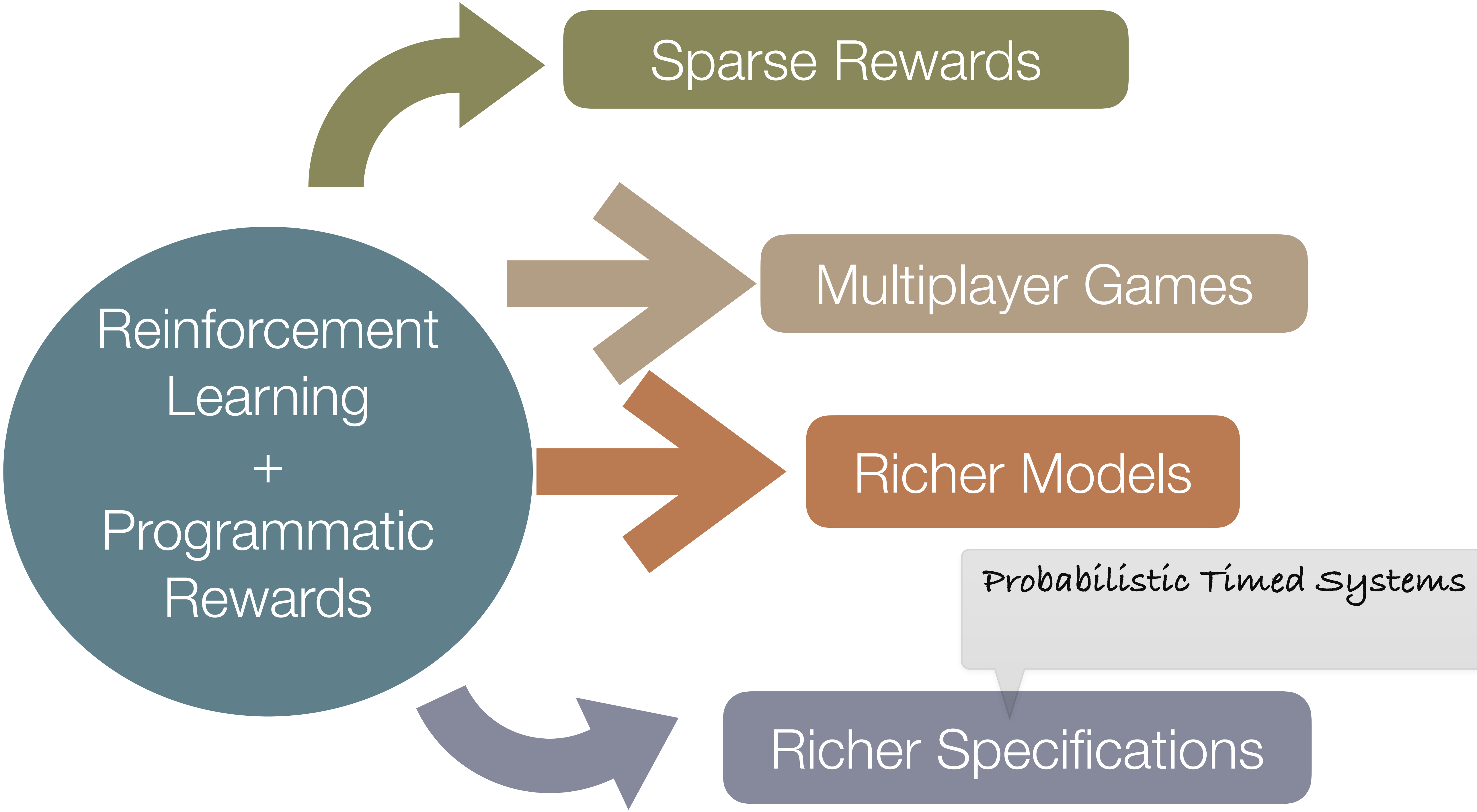


Abolfazl Lavaei, Fabio Somenzi, Sadegh Soudjani, Ashutosh Trivedi, Majid Zamani:
Formal Controller Synthesis for Continuous-Space MDPs via Model-Free Reinforcement Learning. ICCPS 2020: 98-107

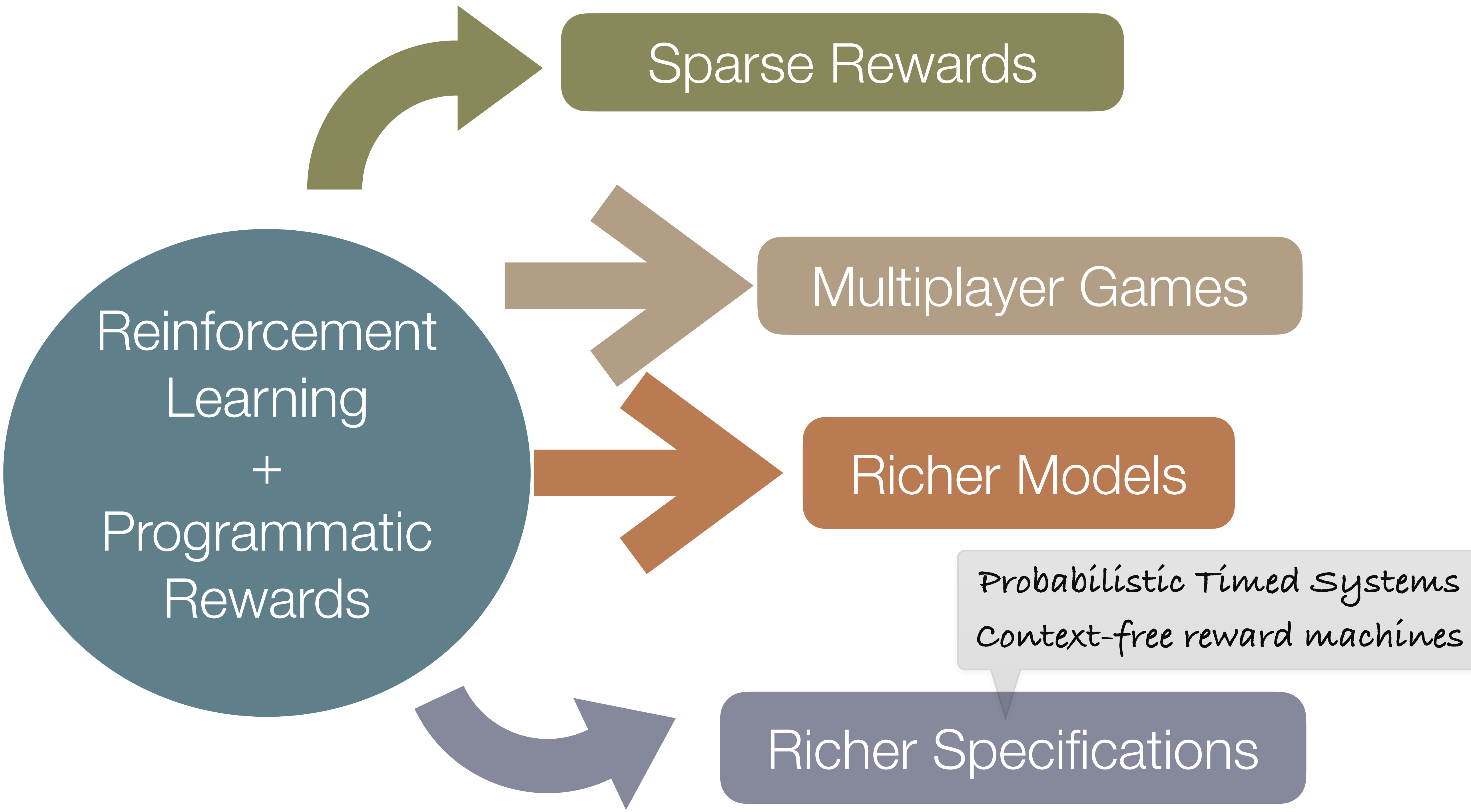
Kalyani Dole, Ashutosh Gupta, John Komp, Shankara Narayanan Krishna, Ashutosh Trivedi:
Event-Triggered and Time-Triggered Duration Calculus for Model-Free Reinforcement Learning. RTSS 2021: 240-252

Amin Falah, Shibashis Guha, and Ashutosh Trivedi:
Reinforcement Learning for Omega-Regular Specifications on Continuous-Time MDPs. Under Review.


Sparse Rewards
Multiplayer Games
Reinforcement Learning + Programmatic Rewards
Richer Models
Richer Specifications



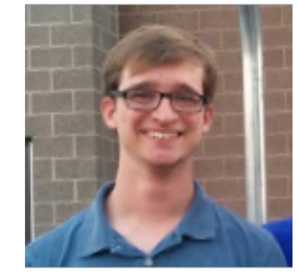
Kalyani Dole, Ashutosh Gupta, John Komp, Shankara Narayanan Krishna, Ashutosh Trivedi:
Event-Triggered and Time-Triggered Duration Calculus for Model-Free Reinforcement Learning. RTSS 2021: 240-252



Kalyani Dole, Ashutosh Gupta, John Komp, Shankara Narayanan Krishna, Ashutosh Trivedi:
Event-Triggered and Time-Triggered Duration Calculus for Model-Free Reinforcement Learning. RTSS 2021: 240-252

Ernst Moritz Hahn, Mateo Perez, Sven Schewe, Fabio Somenzi, Ashutosh Trivedi, Dominik Wojtczak
Recursive Reinforcement Learning. NeurIPS'22.

Improving Scalability

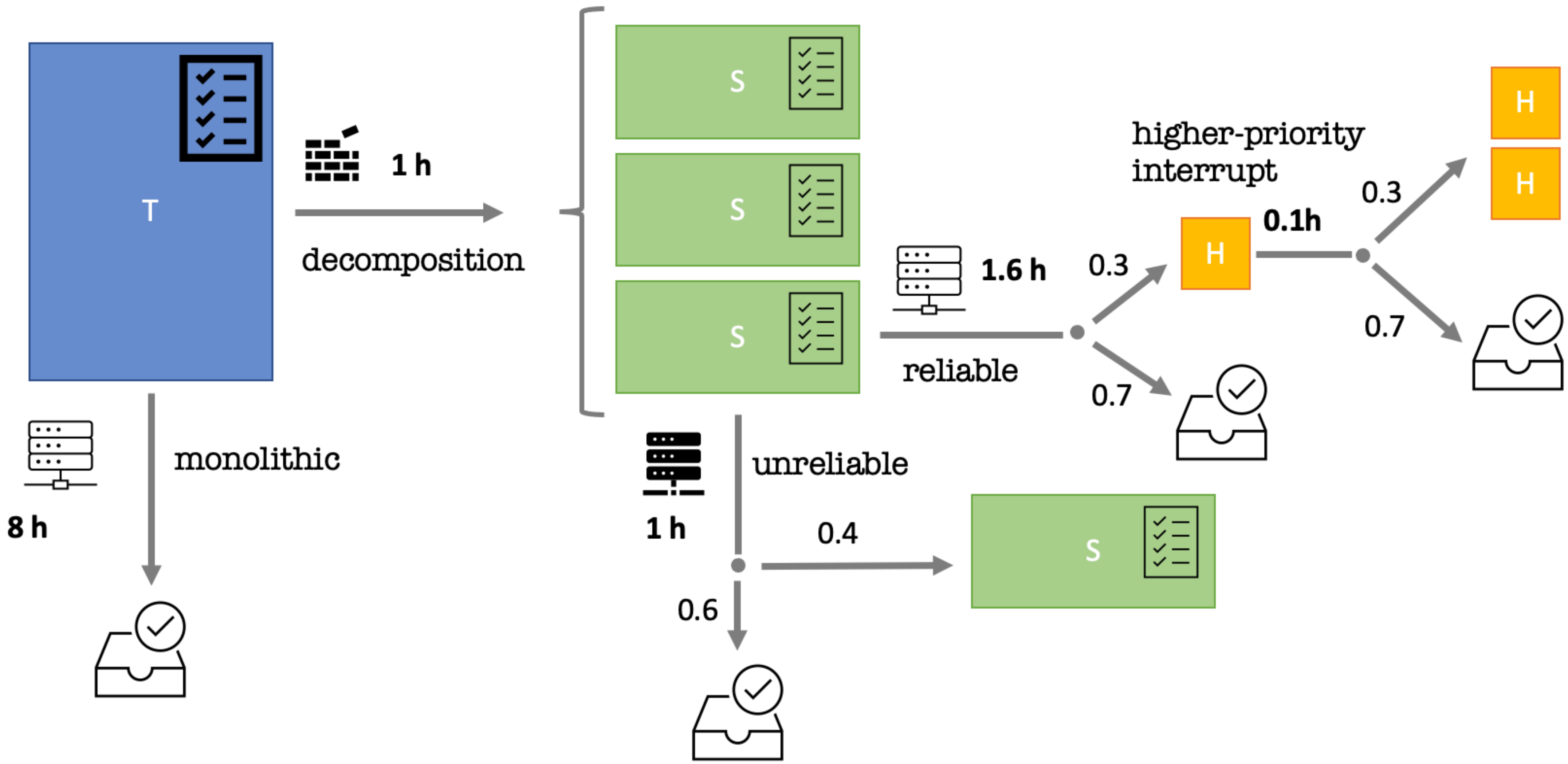



Recursive Reinforcement Learning

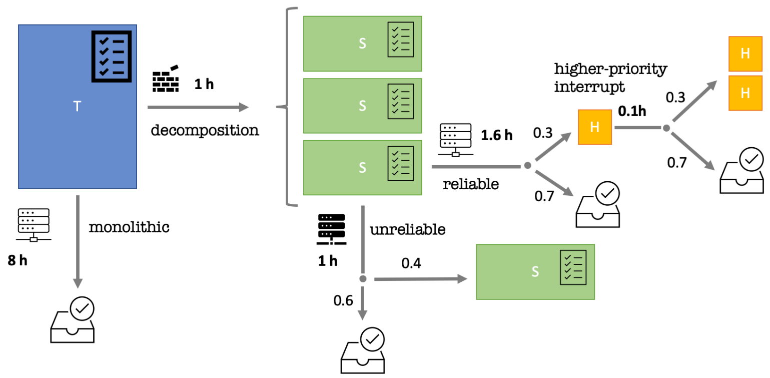
Ernst Moritz Hahn, Mateo Perez, Sven Schewe, Fabio Somenzi, Ashutosh Trivedi, Dominik Wojtczak:
Model-Free Reinforcement Learning for Branching Markov Decision Processes. CAV (2) 2021: 651-673

Ernst Moritz Hahn, Mateo Perez, Sven Schewe, Fabio Somenzi, Ashutosh Trivedi, Dominik Wojtczak
Recursive Reinforcement Learning. NeurIPS’22.

Taylor Dohmen, Noah Topper, George K. Atia, Andre Beckus, Ashutosh Trivedi, Alvaro Velasquez:
Inferring Probabilistic Reward Machines from Non-Markovian Reward Signals for Reinforcement Learning. ICAPS 2022: 574-582



T
1 h
decomposition
S
S
S
monolithic
8 h
1.6 h
reliable
0.3
0.7
higher-priority
interrupt
H
0.1h
0.3
0.7
H
H
unreliable
1 h
0.4
S
0.6

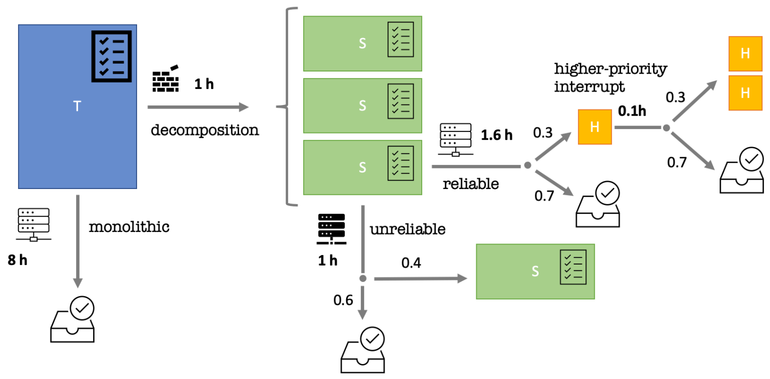


Branching Markov Decision Processes

$$T \overset{a_1}{\underset{1}{\rightarrow}} SSS \qquad S \overset{a_1}{\underset{1.6}{\rightarrow}} 0.3\colon H + 0.7\colon \varepsilon$$

$$T \overset{a_2}{\underset{8}{\rightarrow}} \varepsilon \qquad S \overset{a_2}{\underset{1}{\rightarrow}} 0.4\colon S + 0.6\colon \varepsilon$$

$$H \overset{a_1}{\underset{0.1}{\rightarrow}} 0.3\colon HH + 0.7\colon \varepsilon$$



Branching Markov Decision Processes

$$T \underset{1}{\overset{a_1}{\rightarrow}} SSS \qquad S \underset{1.6}{\overset{a_1}{\rightarrow}} 0.3\colon H + 0.7\colon\ \varepsilon$$

$$T \underset{8}{\overset{a_2}{\rightarrow}} \varepsilon \qquad S \underset{1}{\overset{a_2}{\rightarrow}} 0.4\colon S + 0.6\colon\ \varepsilon$$

$$H \underset{0.1}{\overset{a_1}{\rightarrow}} 0.3\colon HH + 0.7\colon \varepsilon$$

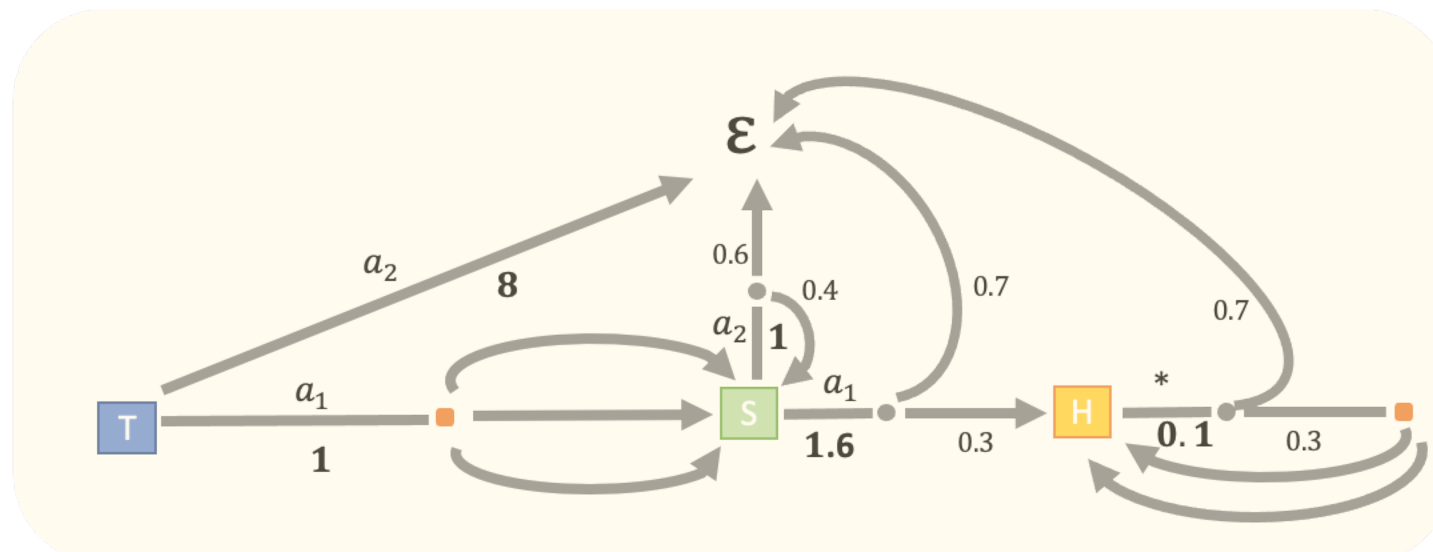
Harris, Theodore Edward, *The Theory of Branching Process*, Santa Monica, Calif.: RAND Corporation, R-381-PR, 1964. As of June 16, 2021: https://www.rand.org/pubs/reports/R381.html

Kousha Etessami, Mihalis Yannakakis:
Recursive Markov Decision Processes and Recursive Stochastic Games. J. ACM 62(2): 11:1-11:69 (2015)

$T \xrightarrow[1]{a_1} SSS$ $\quad$ $S \xrightarrow[1.6]{a_1} 0.3{:}\, H + 0.7{:}\ \varepsilon$

$T \xrightarrow[8]{a_2} \varepsilon$ $\quad$ $S \xrightarrow[1]{a_2} 0.4{:}\, S + 0.6{:}\ \varepsilon$

$H \xrightarrow[0.1]{*} 0.3{:}\, HH + 0.7{:}\, \varepsilon$



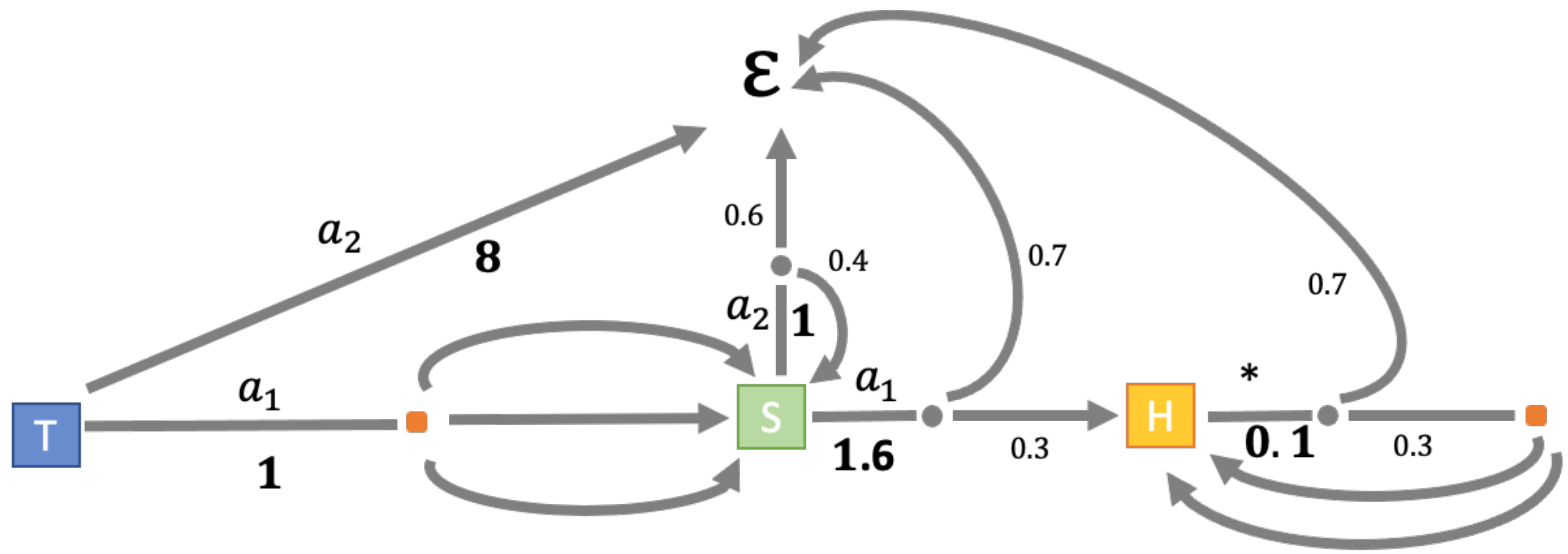
BMDP Optimization (When the model is completely known).

Given a Branching Markov decision process (BMDP) with **positive cost**, compute

- the minimum total expected cost (to termination) and
- an optimal strategy.

When the model is known.

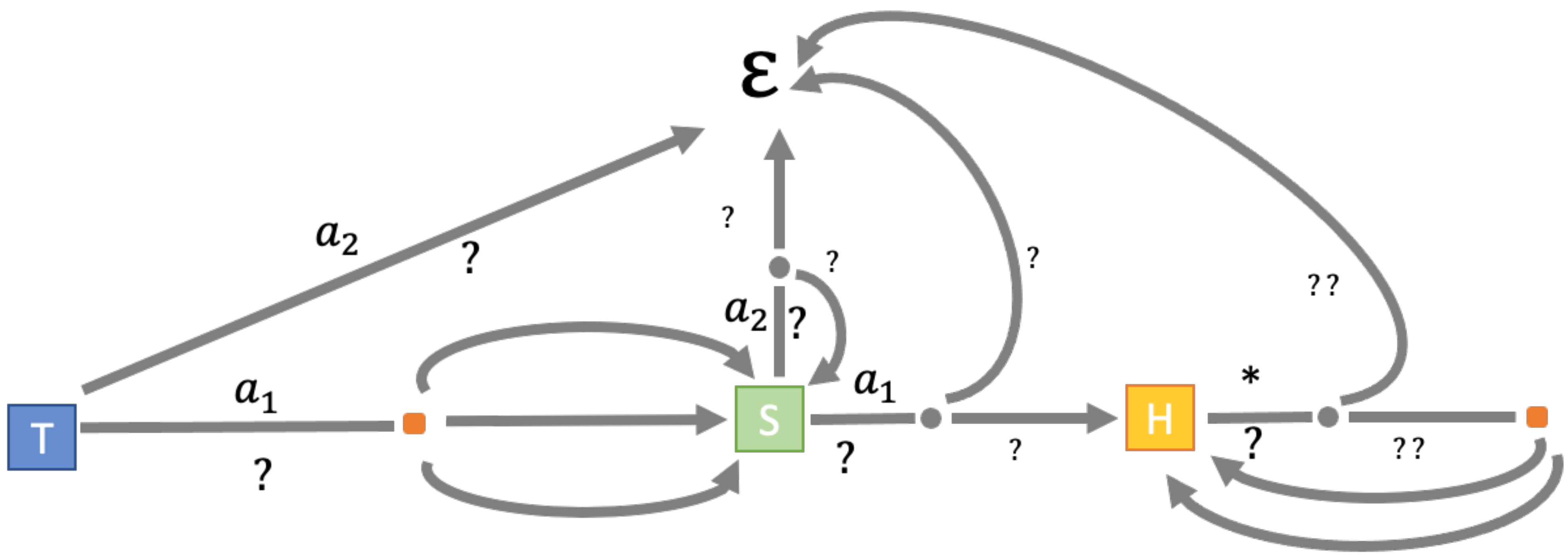
Kousha Etessami, Dominik Wojtczak, Mihalis Yannakakis:
Recursive Stochastic Games with Positive Rewards. ICALP (1) 2008: 711-723



$$x_T = \min \{1 + (x_S + x_S + x_S), \quad 8 + x_\varepsilon\}$$
$$x_S = \min \{1.6 + 0.3\, x_H + 0.7\, x_\varepsilon, 1 + 0.4\, x_S + 0.6\, x_\varepsilon\}$$
$$x_H = \min \{0.1 + 0.3\,(x_H + x_H) + 0.7\, x_\varepsilon\}$$
$$x_\varepsilon = 0$$

When the model is unknown.

Ernst Moritz Hahn, Mateo Perez, Sven Schewe, Fabio Somenzi, Ashutosh Trivedi, Dominik Wojtczak:
Model-Free Reinforcement Learning for Branching Markov Decision Processes. CAV (2) 2021: 651-673

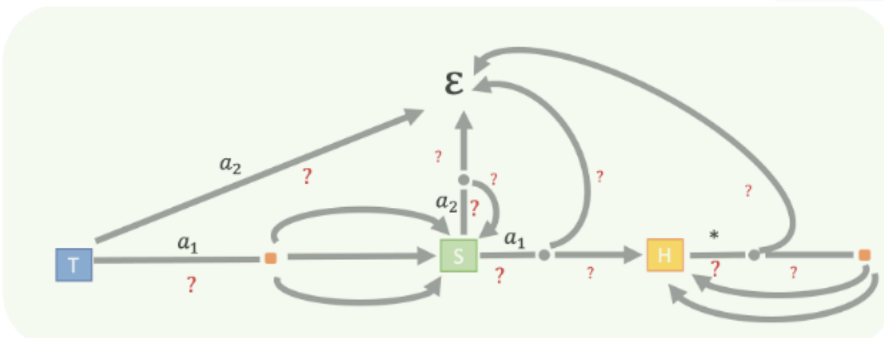


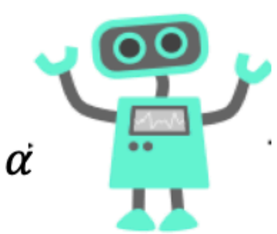
Challenges:

- The state-space of the underlying MDP is infinite
- One-step reward is unbounded
- Undiscounted rewards result in a lack of contraction

Q- Learning for BMDP

$$Q(q,a) = c_{qa} + \sum_{\alpha \in \{\alpha_1,\dots,\alpha_k\}} p_{qa}(\alpha)\cdot\left(\sum_{n\leq|\alpha|} \min_{a'} Q(\alpha(n),a')\right)$$





next population = α
cost = c

current type = q
action = a

Theorem (Convergence of Q-Learning for BMDPs).

For positive rewards r_i, and learning-rates satisfying

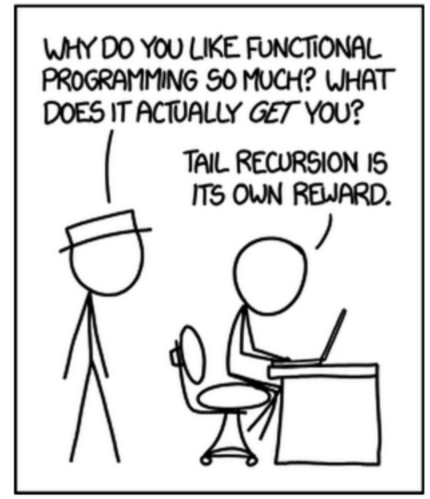
$$\sum_{i=0}^{\infty} \lambda_i = \infty \text{ and } \sum_{i=0}^{\infty} \lambda_i^2 < \infty$$

We have that $Q_i(x_q,a) \to Q(x_q,a)$ as i to ∞ if every (s,a) pair has a minimum probability p_{min} of being selected in every step.

Estimated Q

Q	a_1	a_2	a_3	a_4
q_1				
q_2				
q_3				
q_4				

$$\underbrace{Q_{i+1}(q,a)}_{\text{new estimate}} = (1-\lambda_i)\,\underbrace{Q_i(q,a)}_{\text{old estimate}} + \lambda_i\underbrace{\left(c + \sum_{n\leq|\alpha|} \min_{a'} Q(\alpha(n),a')\right)}_{\text{current sample}}$$



Beyond Branching MDPs: Recursion

- Probabilistic pushdown systems
- "Context-free" specifications
- Modular Reinforcement Learning
- Probabilistic Boolean programs

Ernst Moritz Hahn, Mateo Perez, Sven Schewe, Fabio Somenzi, Ashutosh Trivedi, Dominik Wojtczak
Recursive Reinforcement Learning. NeurIPS'22.

```
def T():
  a = ?({mono, divide});
  if a = mono:
    execute_mono() # −$8
  else :
    decompose() # −$0.5
    S()
    S()
    S()
    combine() # −$0.5
  return
```

(a) Task T

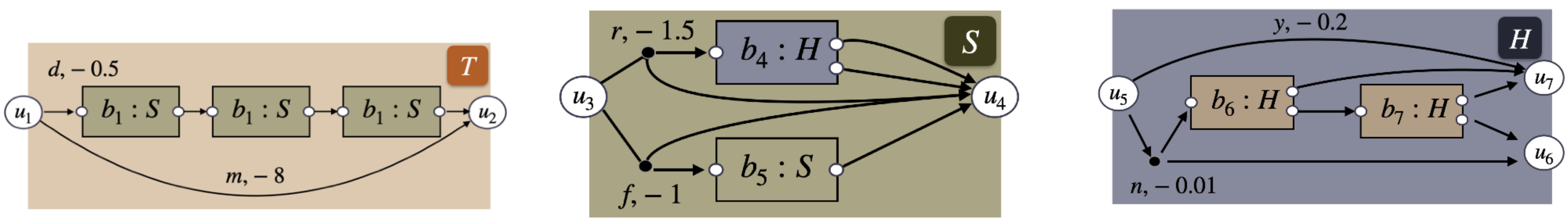
```
def S():
  a = ?({reliable, fast});
  if a = fast:
    try:
      execute_fast() # −$1
    except Crash: # prob = 0.4
      S()
  else:
    try:
      execute_reliable() #−$1.5
    except Interrupt: #prob = 0.3
      H() # +$0.2
  return
```

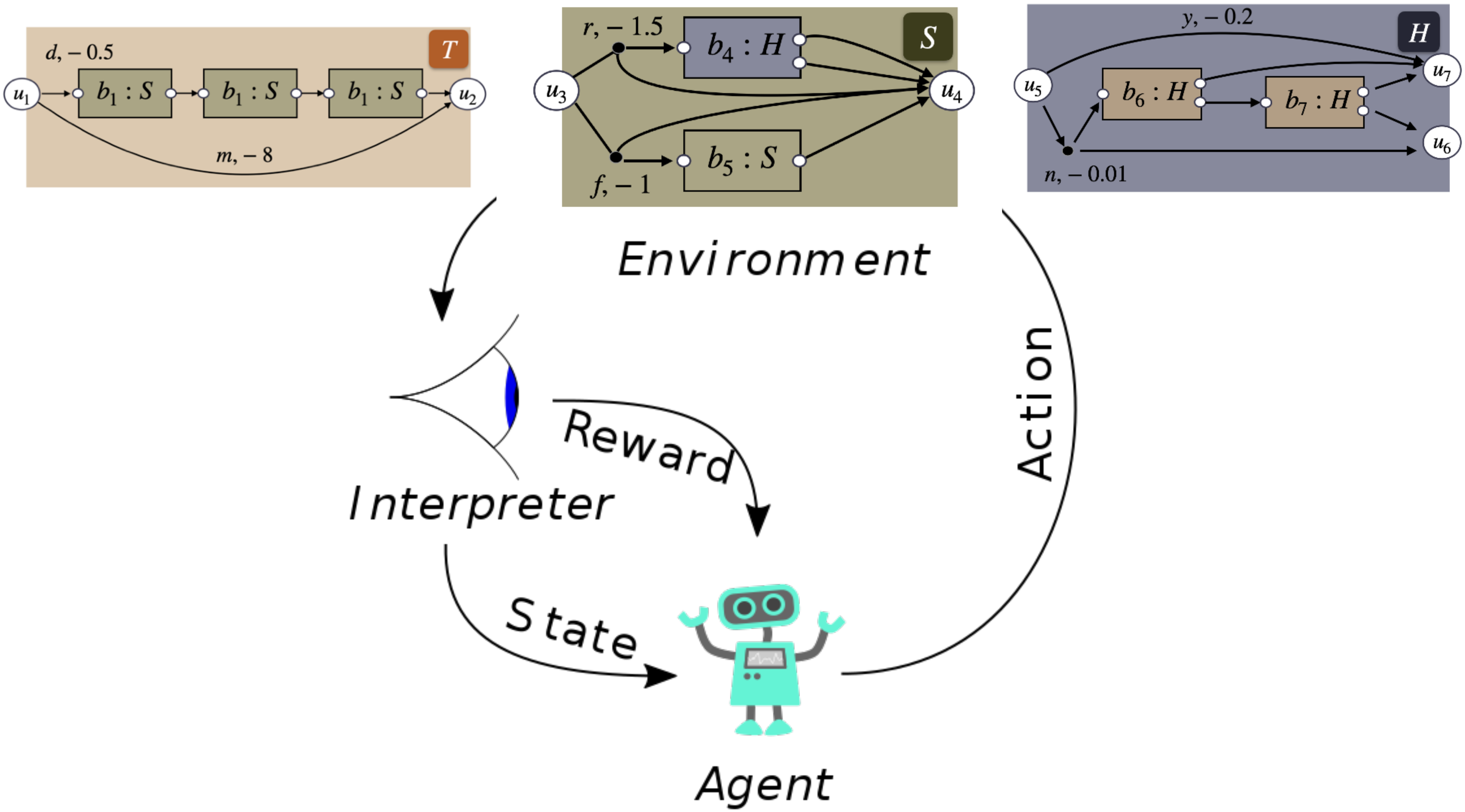
(b) Task S

```
def H():
  upgrade = ?({yes, no});
  if upgrade = no:
    try:
      ISR() # −$0.01
    except Interrupt: #prob = 0.3
      if (H()):
        return 1
      if (H())
        return 1
    return 0
  else : #−$0.2
    return 1
```

(c) Task H

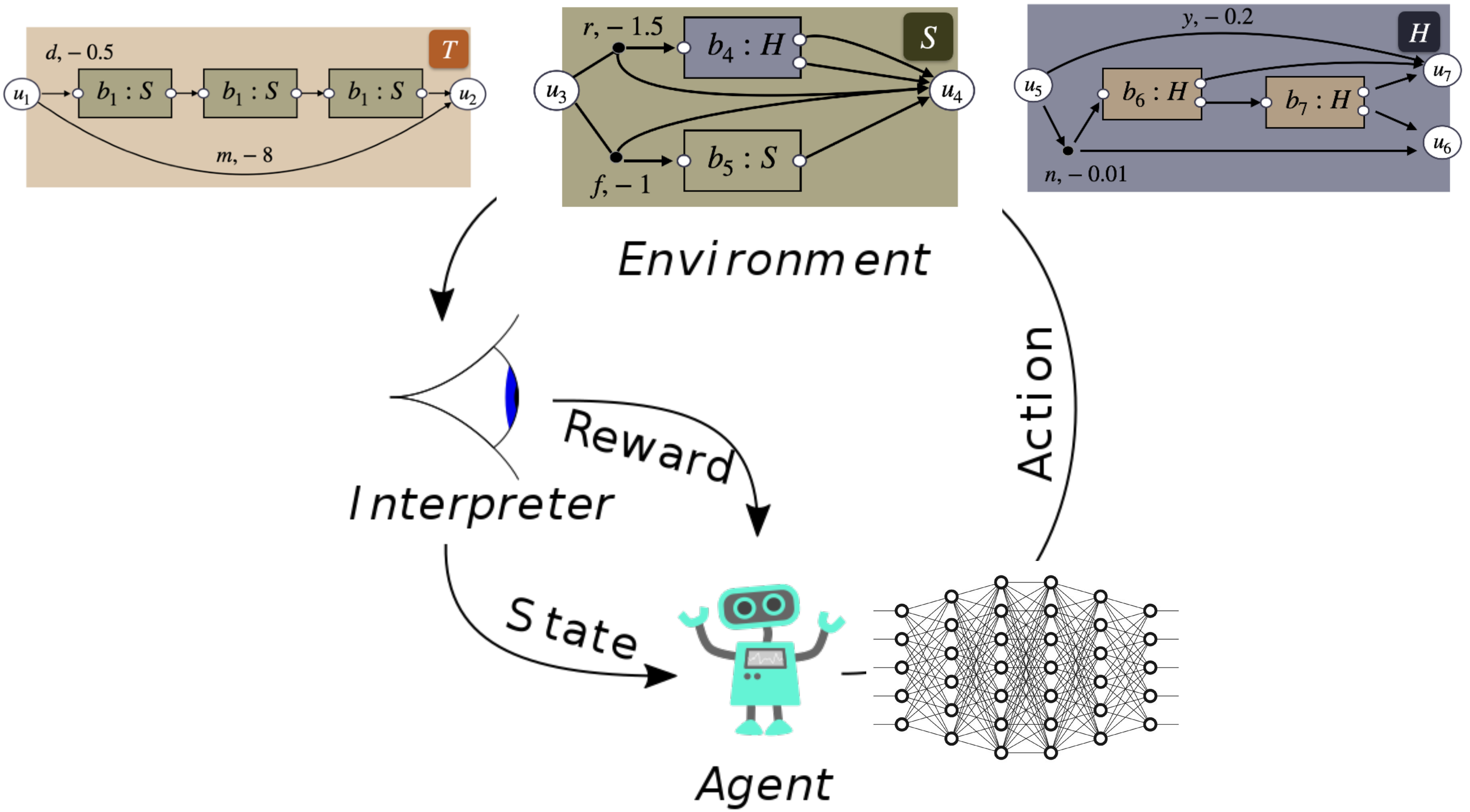
Figure 2: A probabilistic Boolean program sketch where the choice of the hole (?) is to be filled by RL agent.



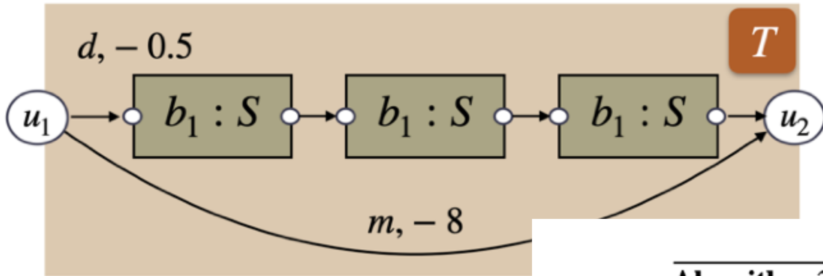


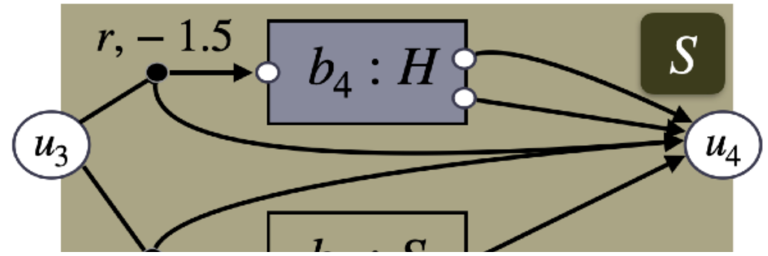
Theorem (Convergence)

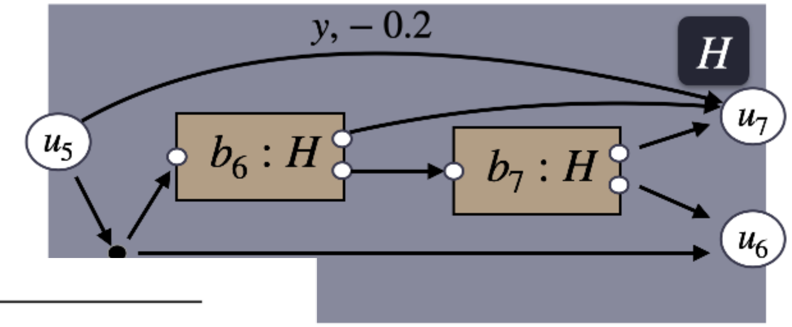
Tabular recursive Q-learning algorithm converges for deterministic multi-exit RMDPs and single-exit RMDPs.



d, − 0.5
T
u_1
$b_1 : S$
$b_1 : S$
$b_1 : S$
u_2
m, − 8
r, − 1.5
$b_4 : H$
S
u_3
u_4
$b_5 : S$
f, − 1
y, − 0.2
H
u_5
$b_6 : H$
$b_7 : H$
u_7
u_6
n, − 0.01
Environment
Action
Reward
Interpreter
State
Agent





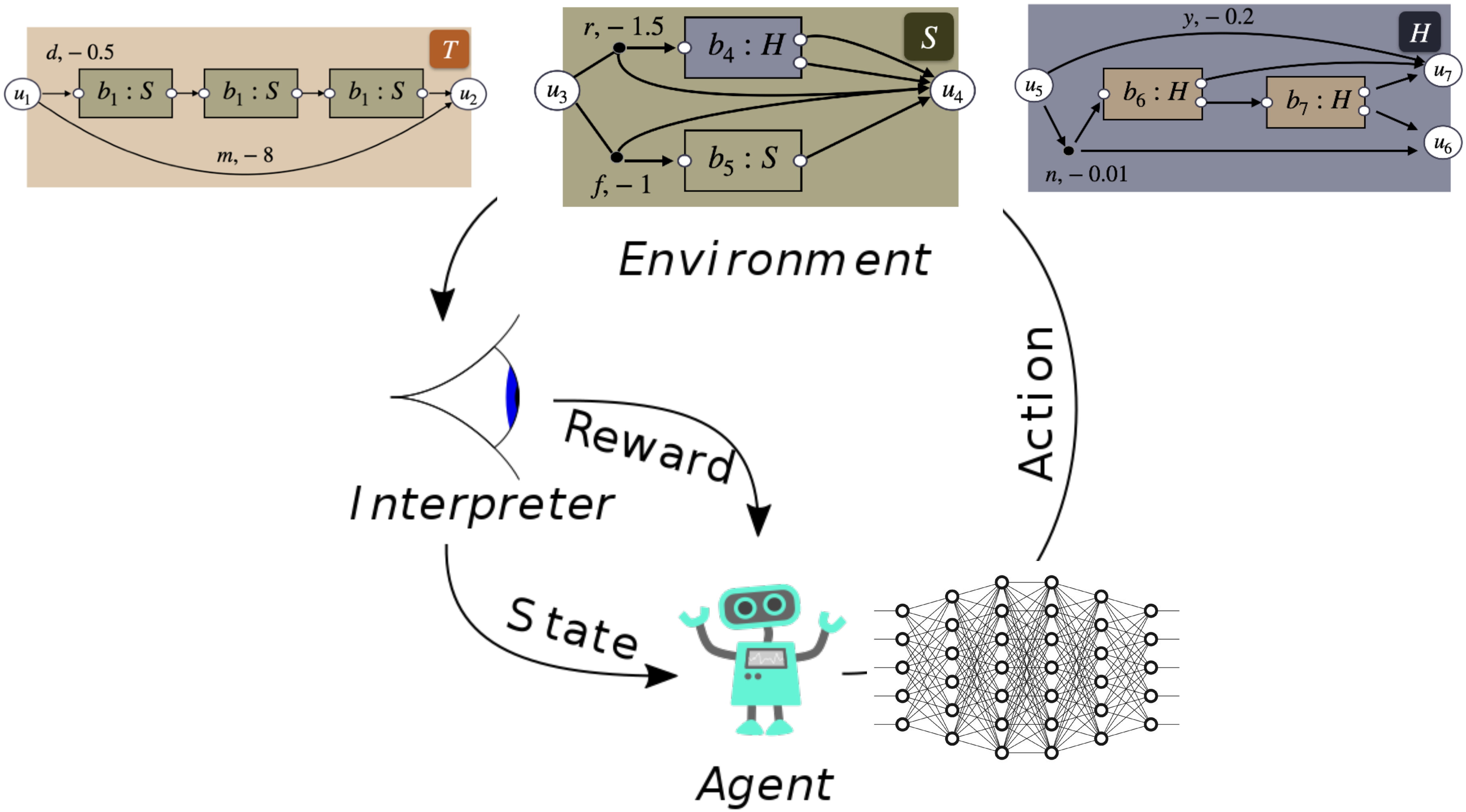


Algorithm 3: Deep Recursive Q-learning (DQN-style)

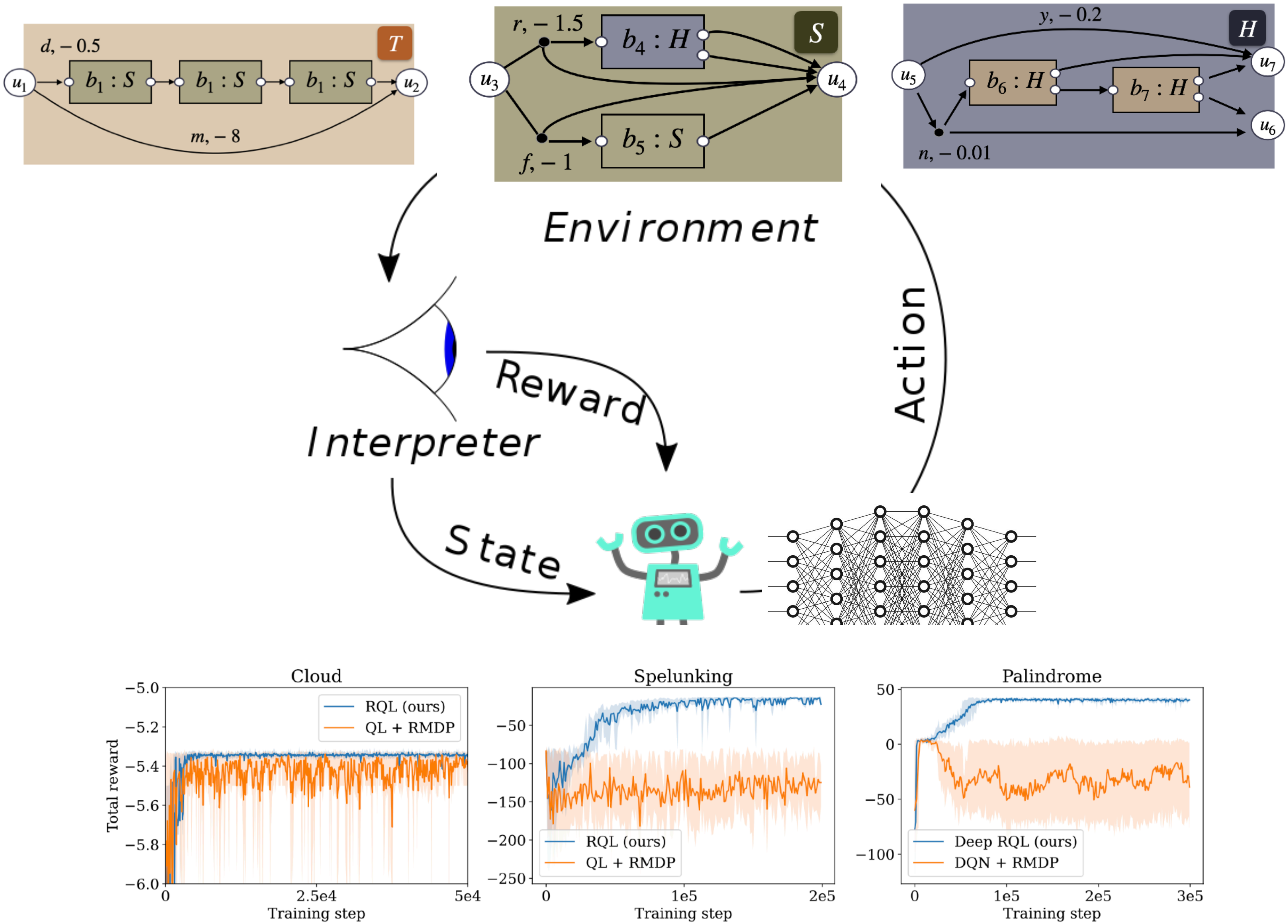
Buffer size N, update frequency C
Initialize network parameters θ
Set target network parameters $\theta^- \leftarrow \theta$
Initialize empty replay buffer
while *not converged* **do**
  $v \leftarrow \mathbf{0}$
  stack $\leftarrow \emptyset$
  Sample trajectory $\tau \sim \{(s, a, r, s'), \ldots\}$
  for s, a, r, s' *in* τ **do**
    `// Push update to replay buffer`
    if *entered box* **then**
      $\{s_{\text{exit}_1}, \ldots, s_{\text{exit}_n}\} \leftarrow \text{getExits}(s')$
      $v' \leftarrow [\max_{a' \in A(s_{\text{exit}_1})} Q(s_{\text{exit}_1}, v, a'; \theta), \ldots, \max_{a' \in A(s_{\text{exit}_n})} Q(s_{\text{exit}_n}, v, a'; \theta)]$
      $v'_{\text{min}} \leftarrow \min(v')$
      $v' \leftarrow v' - v'_{\text{min}}$
      buffer.add(*entered box*, (s, v, a, r, s', v'), v'_{min})
      stack.push(v)
      $v \leftarrow v'$
    else if *exited box* **then**
      $\{s_{\text{exit}_1}, \ldots, s_{\text{exit}_n}\} \leftarrow \text{getExits}(s)$
      Set k such that $s' = s_{\text{exit}_k}$
      buffer.add(*exited box*, (s, v, a, r, s', v), $v(k)$)
      $v \leftarrow$ stack.pop()
    else
      buffer.add(*normal*, $(s, v\ a, r, s', v)$, $\perp$)
    end
    `// Update network`
    Sample minibatch $\{type_j, (s_j, v_j, a_j, r_j, s'_j, v'_j), aux_j\}_{j=1}^{N}$ of size N from replay buffer
    for $j = 1, \ldots, N$ **do**

$$\text{targ}_j \leftarrow \begin{cases} r_j + \max_{a' \in A(s'_j)} Q(s'_j, v'_j, a'_j; \theta^-) + aux_j & type_j = \textit{entered box} \\ r_j + aux_j & type_j = \textit{exited box} \\ r_j + \max_{a \in A(s'_j)} Q(s'_j, v'_j, a'_j; \theta^-) & type_j = \textit{normal} \end{cases}$$

    end
    $\mathcal{L} = \frac{1}{N} \sum_{j=1}^{N} (Q(s_j, v_j, a_j) - \text{targ}_j)^2$
    Update θ with respect to loss $\mathcal{L}$ with gradient descent
    Set $\theta^- \leftarrow \theta$ every C steps
  end
end
return θ



d, − 0.5
T
u1
b1 : S
b1 : S
b1 : S
u2
m, − 8
r, − 1.5
b4 : H
S
u3
u4
b5 : S
f, − 1
y, − 0.2
H
u5
b6 : H
b7 : H
u7
u6
n, − 0.01
Environment
Action
Reward
Interpreter
State
Agent



$d, -0.5$
T
u_1
$b_1 : S$
$b_1 : S$
$b_1 : S$
u_2
$m, -8$
$r, -1.5$
$b_4 : H$
S
u_3
u_4
$b_5 : S$
$f, -1$
$y, -0.2$
H
u_5
u_7
$b_6 : H$
$b_7 : H$
u_6
$n, -0.01$
Environment
Action
Reward
Interpreter
State
Cloud
Spelunking
Palindrome
Total reward
Training step
RQL (ours)
QL + RMDP
Deep RQL (ours)
DQN + RMDP

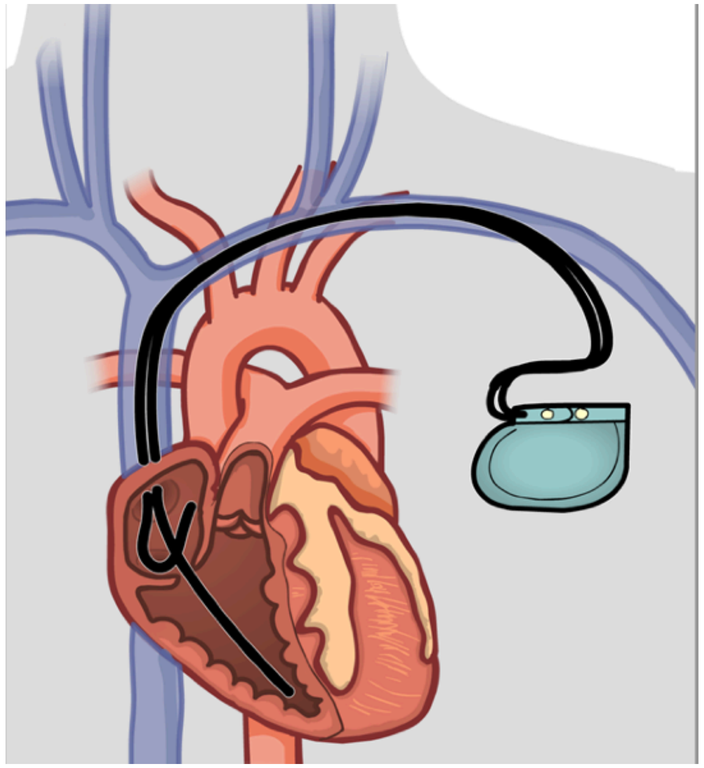
MUNGOJERRIE

Formal Reinforcement Learning

plv.colorado.edu/mungojerrie/

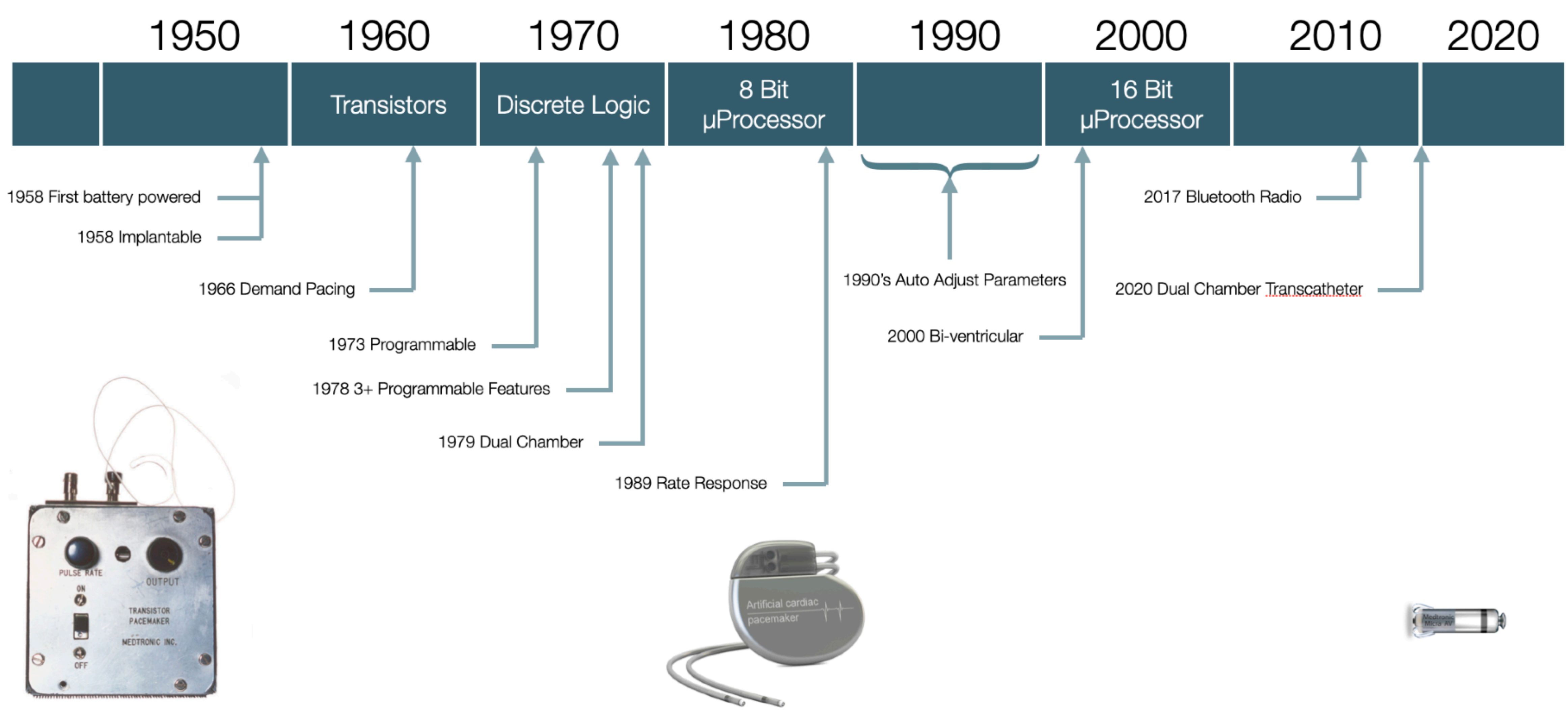
Improving Trust

Next-Generation Cardiac Pacemakers



Kalyani Dole, Ashutosh Gupta, John Komp, Shankara Narayanan Krishna, Ashutosh Trivedi:
Event-Triggered and Time-Triggered Duration Calculus for Model-Free Reinforcement Learning. RTSS 2021: 240-252

Kalyani Dole, Ashutosh Gupta, John Komp, S. Krishna, and Ashutosh Trivedi:
Correct-by-Construction Reinforcement Learning of Cardiac Pacemakers from Duration Calculus Requirement. Under Review.


1950
1960
1970
1980
1990
2000
2010
2020
Transistors
Discrete Logic
8 Bit µProcessor
16 Bit µProcessor
1958 First battery powered
1958 Implantable
1966 Demand Pacing
1973 Programmable
1978 3+ Programmable Features
1979 Dual Chamber
1989 Rate Response
1990's Auto Adjust Parameters
2000 Bi-ventricular
2017 Bluetooth Radio
2020 Dual Chamber Transcatheter
PULSE RATE
OUTPUT
ON
OFF
TRANSISTOR PACEMAKER
MEDTRONIC INC.
Artificial cardiac pacemaker

RL for creative design space exploration

RL for creative design space exploration

- Time-triggered and event-triggered Duration Calculus
- Elicitation of VVI and DDD Pacemaker Specifications in DC
- DC to Scalar Rewards
- Learning and Validation

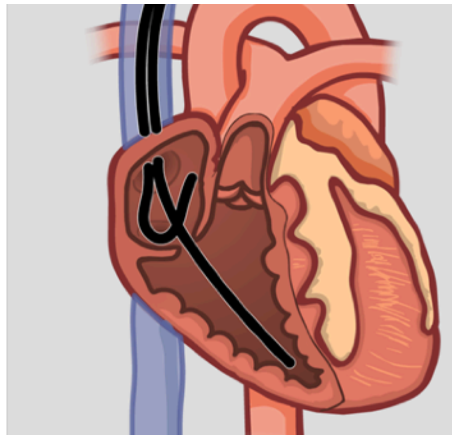
VVI Pacemaker Specification

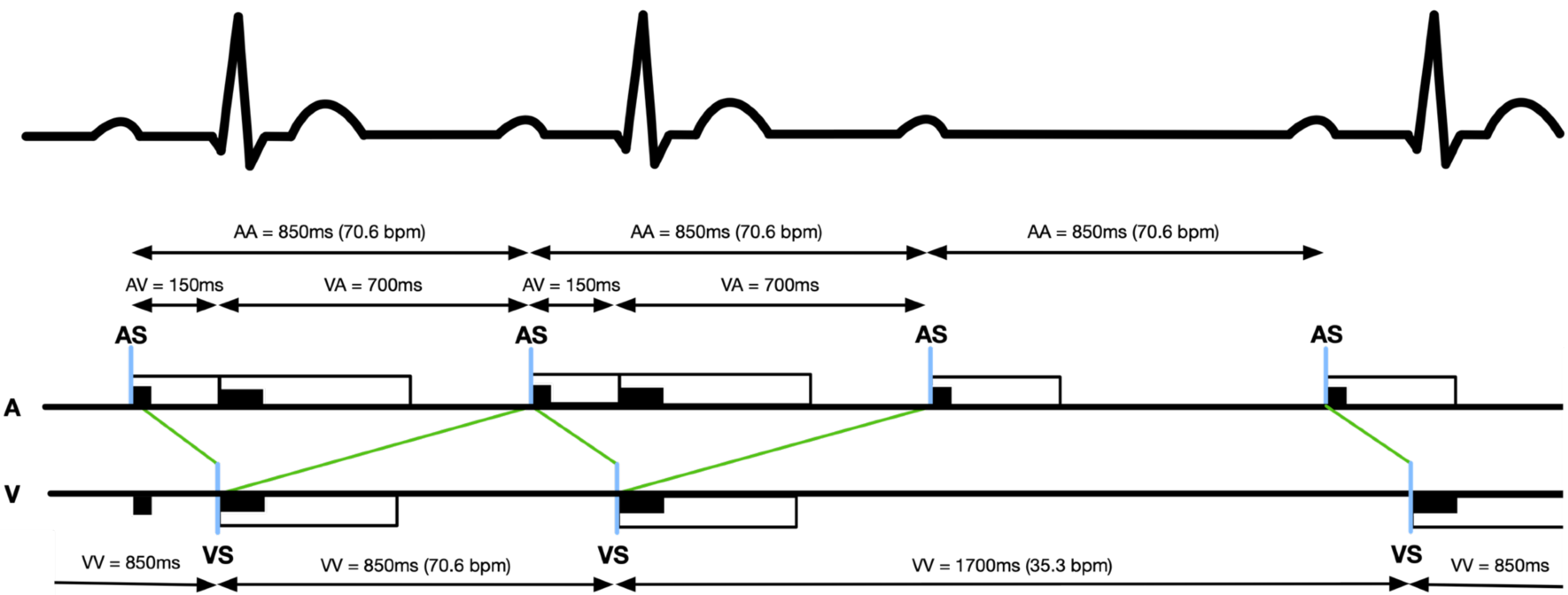
lowvp	$\lceil VP \rceil^{\bullet} \frown true \implies ((\ell =^{\bullet}_{VP} (pace - 1)) \wedge \Box \lceil \neg VP \rceil)$
onlyref	$(\ell =^{\bullet}_{tic} pace) \wedge \Box ((\ell =^{\bullet}_{tic} ref) \wedge \Box \lceil \neg VS \wedge \neg VP \rceil \implies ((\ell =^{\bullet}_{tic} (ref + 1)) \frown \lceil \neg VS \rceil^{\bullet}))$
pace_req	$(\ell =^{\bullet}_{tic} pace) \frown \lceil VP \rceil^{\bullet})$
no_pace_req	$(\ell =^{\bullet}_{tic} pace - 1) \wedge (\Box \lceil \neg VP \rceil)$
repeatvp	$\Box ((\lceil VP \rceil^{\bullet} \frown true) \wedge onlyref \implies pace\_reg)$
vstrigger	$[((\lceil VS \rceil^{\bullet} \frown true) \wedge onlyref) \implies pace\_req] \wedge (\lceil VS \rceil^{\bullet} \frown true \implies no\_pace\_req)$
repeatvs	$\Box [((\ell =^{\bullet}_{tic} ref) \wedge \Box (\lceil \neg VS \wedge \neg VP \rceil)) \implies ((\ell =^{\bullet}_{tic} ref + 1) \frown vstrigger)]$
boot	$((onlyref \implies pace\_req) \wedge (\ell =^{\bullet}_{tic} pace - 1 \wedge \Box \lceil \neg VP \rceil))$
Specification	$lowvp \wedge repeatvp \wedge repeatvs \wedge boot$

RL for creative design space exploration

- Time-triggered and event-triggered Duration Calculus
- Elicitation of VVI and DDD Pacemaker Specifications in DC
- DC to Scalar Rewards
- Learning and Validation

RL for adaptive cardiac pacemakers



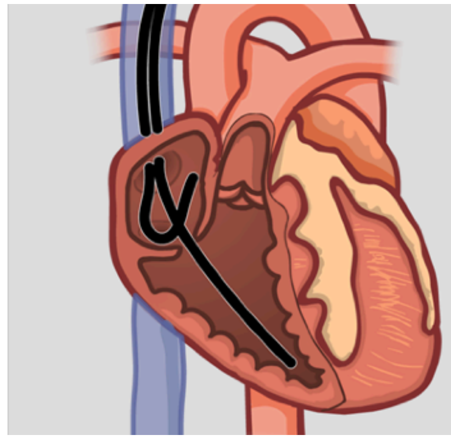


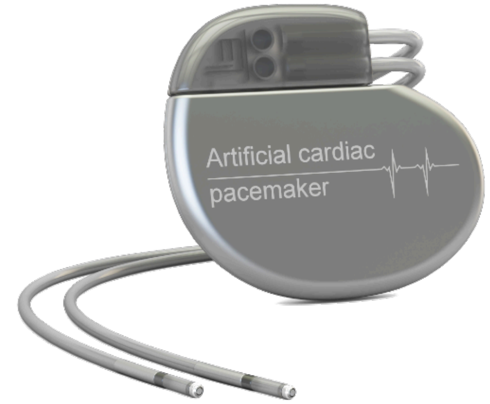
LRL = 60 bpm
URL = 180 bpm

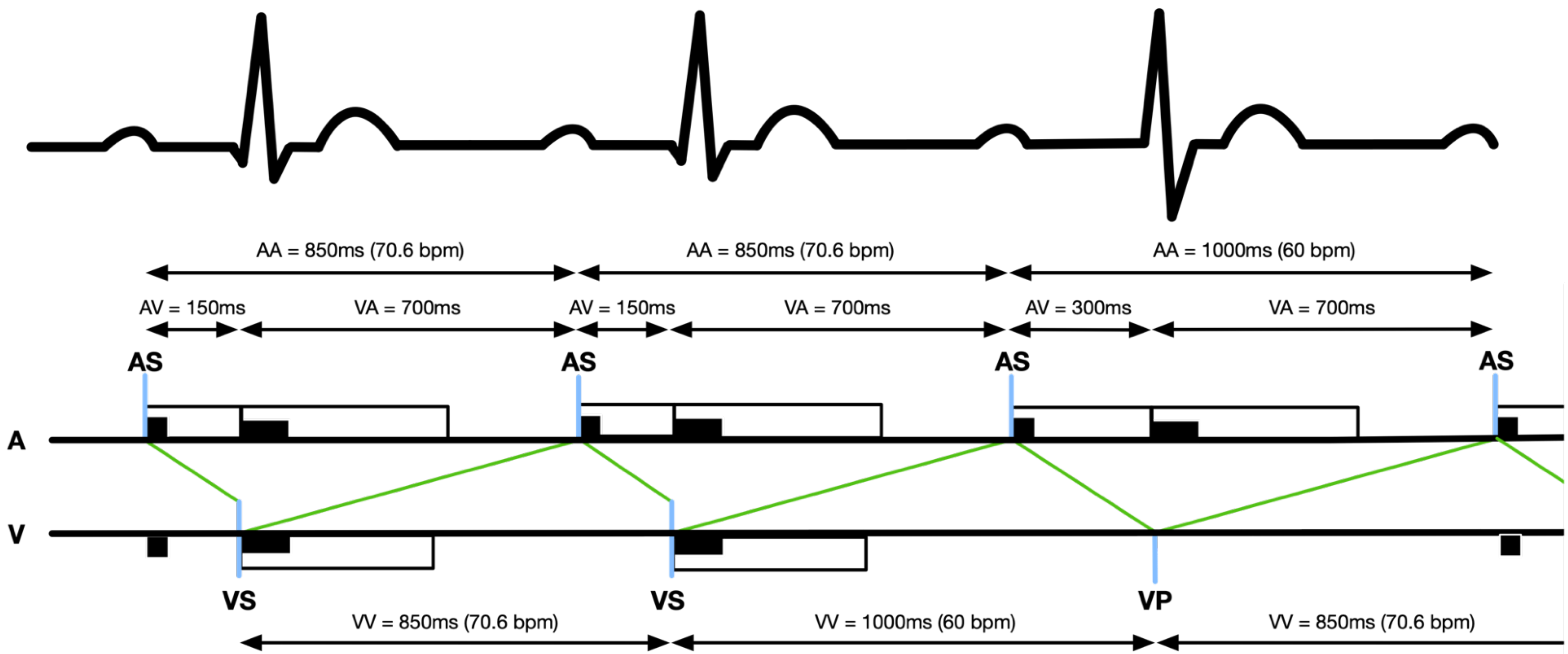
3:2 Heartblock without pacing

PAAB = 30ms
PAVB = 25ms
PAAR = PAV

PVAB = 75ms
PVVB = 75ms
PVARP = 300ms
PVVR = 275ms





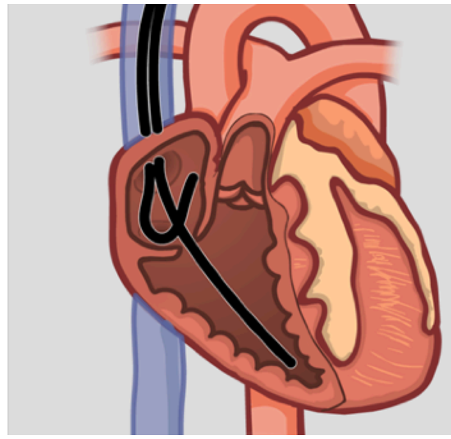


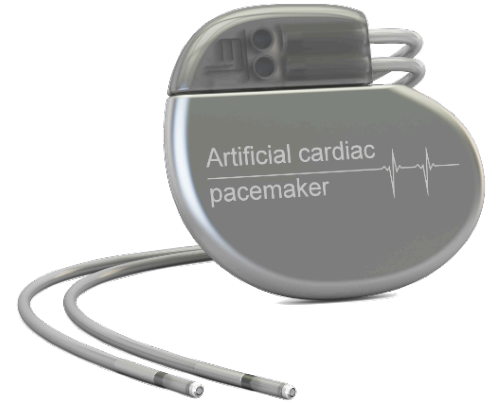
LRL = 60 bpm
URL = 180 bpm

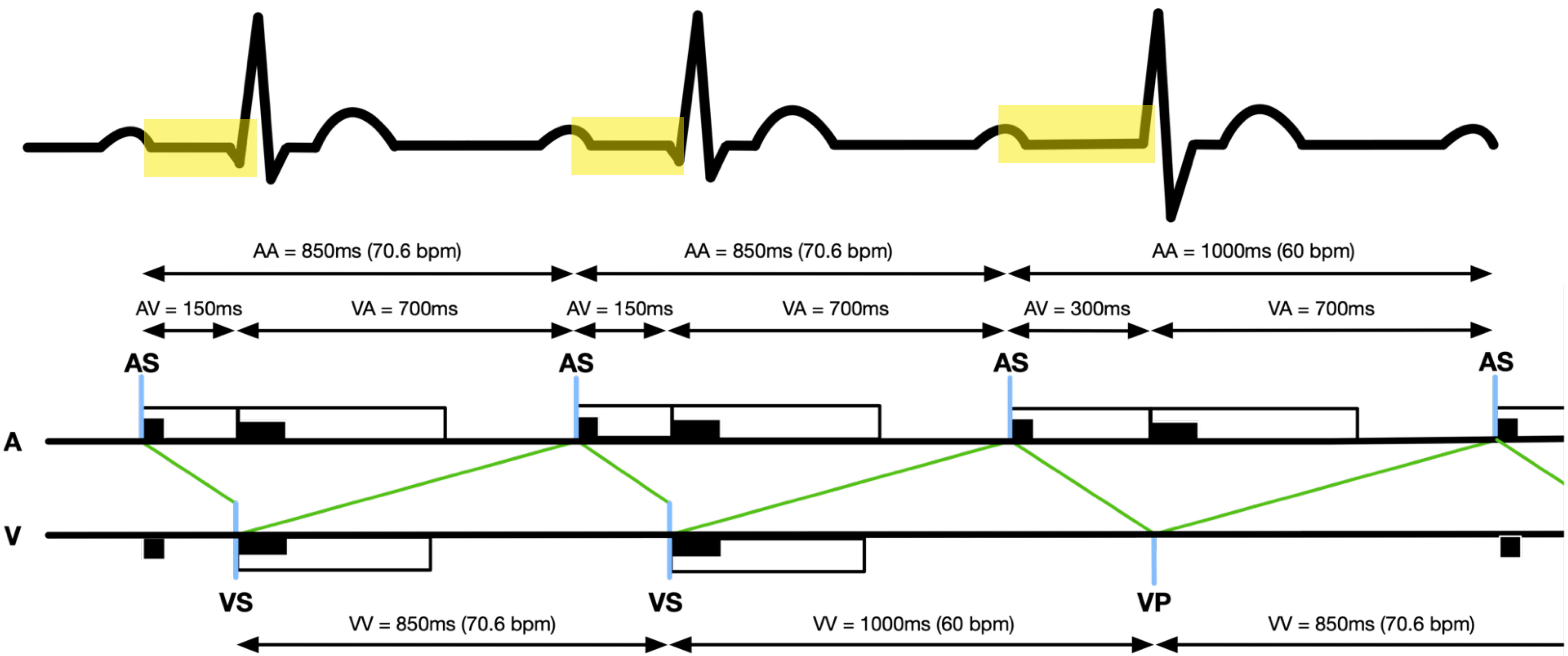
3:2 Heartblock with pacing

PAAB = 30ms
PAVB = 25ms
PAAR = PAV

PVAB = 75ms
PVVB = 75ms
PVARP = 300ms
PVVR = 275ms





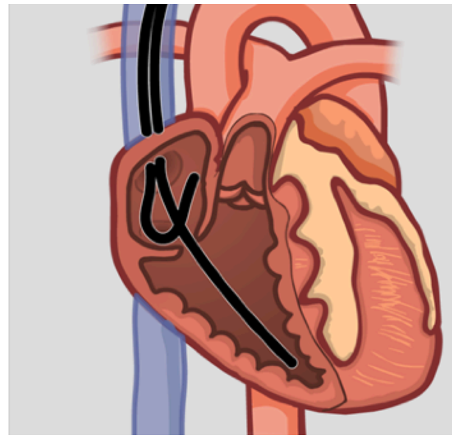


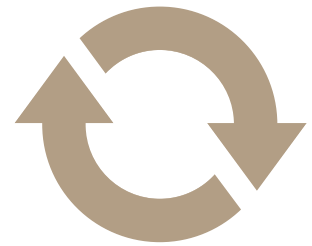
LRL = 60 bpm
URL = 180 bpm

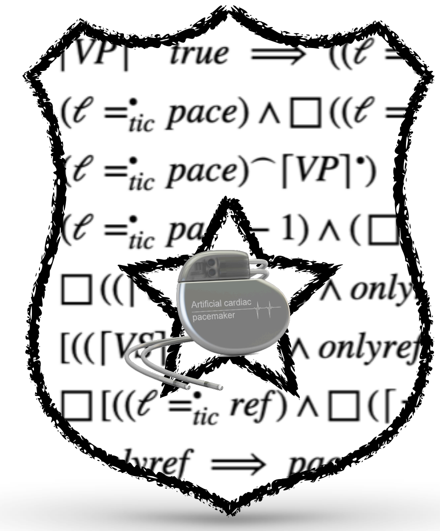
3:2 Heartblock with pacing

PAAB = 30ms
PAVB = 25ms
PAAR = PAV

PVAB = 75ms
PVVB = 75ms
PVARP = 300ms
PVVR = 275ms

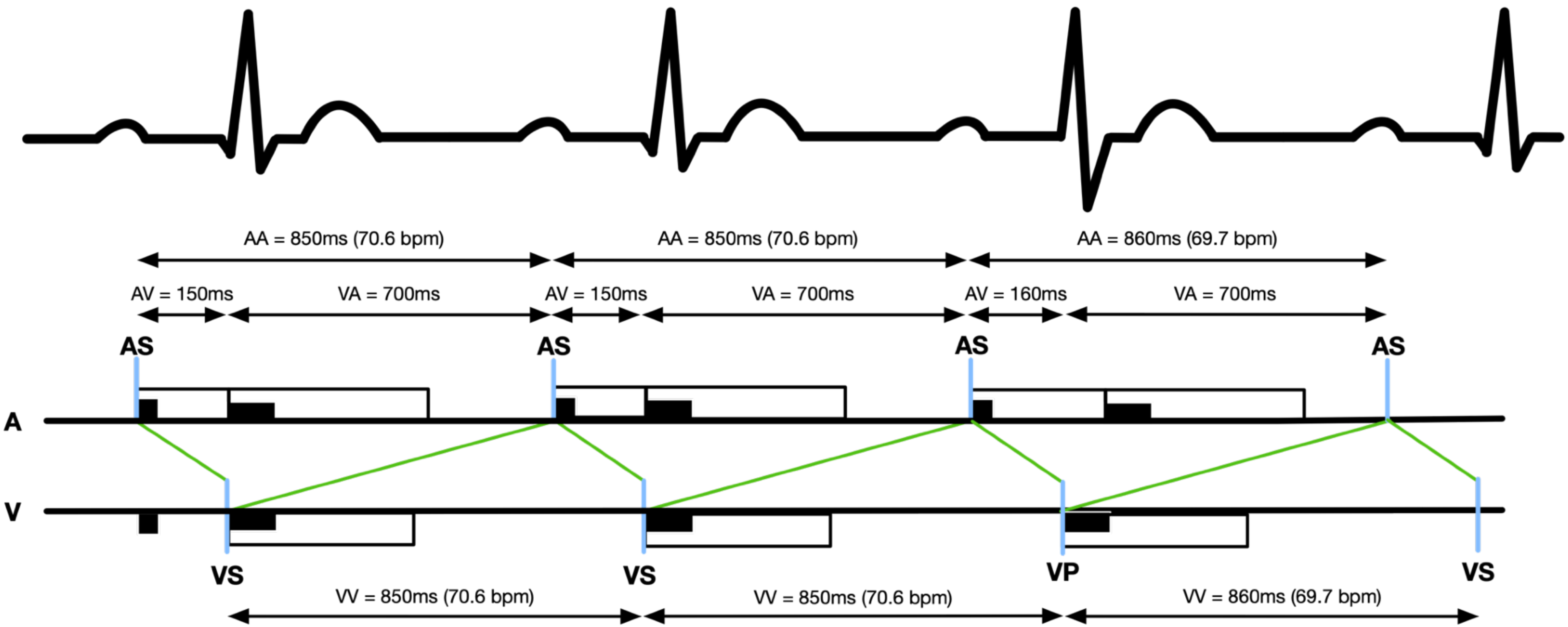


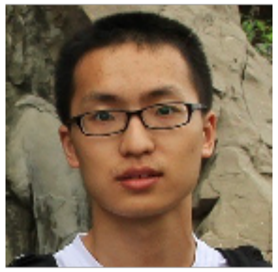




RL for adaptive cardiac pacemakers

- Safety-DC fragment
- Shield Synthesis from DC specifications
- Scalar Reward for adaptive behavior
- Adaptable-by-construction synthesis


AA = 850ms (70.6 bpm)
AA = 850ms (70.6 bpm)
AA = 860ms (69.7 bpm)
AV = 150ms
VA = 700ms
AV = 150ms
VA = 700ms
AV = 160ms
VA = 700ms
AS
AS
AS
AS
A
V
VS
VS
VP
VS
VV = 850ms (70.6 bpm)
VV = 850ms (70.6 bpm)
VV = 860ms (69.7 bpm)

Siyuan Liu, Ashutosh Trivedi, Xiang Yin, Majid Zamani:
Secure-by-construction synthesis of cyber-physical systems. Annu. Rev. Control. 53: 30-50 (2022)

Mahathi Anand, Vishnu Murali, Ashutosh Trivedi, Majid Zamani:
Formal verification of hyperproperties for control systems. CAADCPS@CPSIoTWeek 2021: 29-30

Saeid Tizpaz-Niari:
Differential Performance Debugging and its application to side-channel analysis. PhD Thesis. CU Boulder (2020).

Saeid Tizpaz-Niari, Pavol Cerný, Ashutosh Trivedi:
Data-Driven Debugging for Functional Side Channels. NDSS 2020

Tianhan Lu, Bor-Yuh Evan Chang, Ashutosh Trivedi:
Selectively-Amortized Resource Bounding. SAS 2021: 286-307

Beyond Safety

Security

Fairness/Discrimination

Accountability

Siyuan Liu, Ashutosh Trivedi, Xiang Yin, Majid Zamani:
Secure-by-construction synthesis of cyber-physical systems. Annu. Rev. Control. 53: 30-50 (2022)

Mahathi Anand, Vishnu Murali, Ashutosh Trivedi, Majid Zamani:
Formal verification of hyperproperties for control systems. CAADCPS@CPSIoTWeek 2021: 29-30

Saeid Tizpaz-Niari:
Differential Performance Debugging and its application to side-channel analysis. PhD Thesis. CU Boulder (2020).

Saeid Tizpaz-Niari, Pavol Cerný, Ashutosh Trivedi:
Data-Driven Debugging for Functional Side Channels. NDSS 2020

Tianhan Lu, Bor-Yuh Evan Chang, Ashutosh Trivedi:
Selectively-Amortized Resource Bounding. SAS 2021: 286-307

Beyond Safety

Security

Fairness/Discrimination

Accountability

Saeid Tizpaz-Niari, Ashish Kumar, Gang Tan, Ashutosh Trivedi:
Fairness-aware Configuration of Machine Learning Libraries. ICSE 2022: 909-920

Verya Monjezi, Ashutosh Trivedi, Gang Tan, Saeid Tizpaz-Niari:
Information-Theoretic Testing and Debugging of Fairness Defects in Deep Neural Networks. (Under Review)

Beyond Safety

Security

Fairness/Discrimination

Accountability

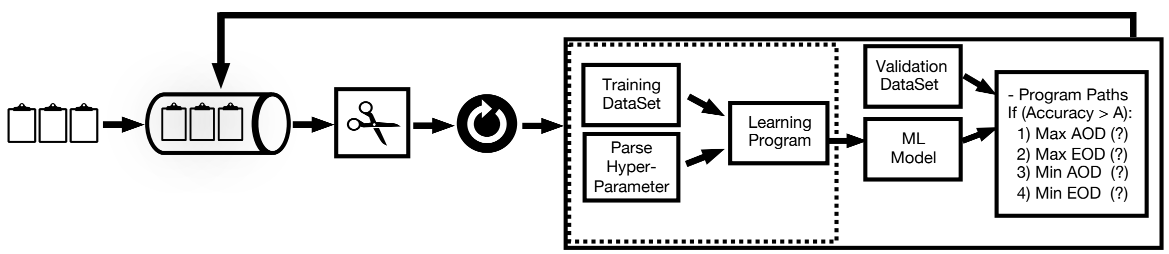

Saeid Tizpaz-Niari, Ashish Kumar, Gang Tan, Ashutosh Trivedi:
Fairness-aware Configuration of Machine Learning Libraries. ICSE 2022: 909-920

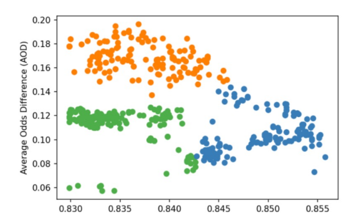
Verya Monjezi, Ashutosh Trivedi, Gang Tan, Saeid Tizpaz-Niari:
Information-Theoretic Testing and Debugging of Fairness Defects in Deep Neural Networks. (Under Review)

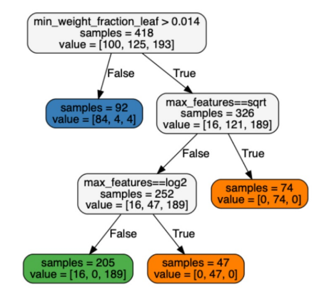
Fairness-aware Configuration of Machine Learning Libraries

https://arxiv.org/pdf/2202.06196.pdf

Boulder







Challenges:
- Testing individual models
- Modification of datasets and learning algorithms
- Low precision and Scalability.

RQ1: Hyperparameter can aggravate or suppress present biases in the dataset.
RQ2: Mutation-based Evolutionary algorithms are effective search technique.
RQ3: Some hyperparameters systematically introduce biases.
RQ4: Parfait-ML outperforms Exp. Gradients and Fairway mitigation techniques.

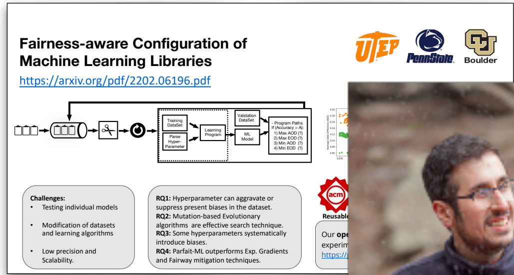
Reusable

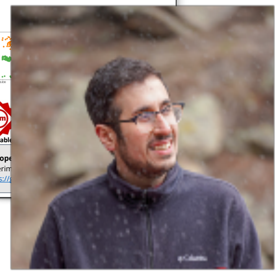
Available

Our **open-source** tool **Parfait-ML** and all experimental subjects are **publicly available**:
https://github.com/Tizpaz/Parfait-ML

Sec

ability





Saeid Tizpaz-Niari, Ashish Kumar, Gang Tan, Ashutosh Trivedi:
Fairness-aware Configuration of Machine Learning Libraries. ICSE 2022: 909-920

Verya Monjezi, Ashutosh Trivedi, Gang Tan, Saeid Tizpaz-Niari:
Information-Theoretic Testing and Debugging of Fairness Defects in Deep Neural Networks. (Under Review)

Beyond Safety

Security

Fairness/Discrimination

Accountability

Saeid Tizpaz-Niari, Morgan Wagner, Shiva Darian, Krystia Reed, Ashutosh Trivedi:
Metamorphic Testing and Debugging of Tax Preparation Software. CoRR abs/2205.04998 (2022)

Beyond Safety

Security

Fairness/Discrimination

Accountability

Form **1040** Department of the Treasury—Internal Revenue Service (99)
U.S. Individual Income Tax Return **2021** OMB No. 1545-0074 IRS Use Only—Do not write or staple in this space.

Filing Status Check only one box.
☐ Single ☐ Married filing jointly ☐ Married filing separately (MFS) ☐ Head of household (HOH) ☐ Qualifying widow(er) (QW)
If you checked the MFS box, enter the name of your spouse. If you checked the HOH or QW box, enter the child's name if the qualifying person is a child but not your dependent ▶

Your first name and middle initial	Last name	**Your social security number**
If joint return, spouse's first name and middle initial	Last name	**Spouse's social security number**

Home address (number and street). If you have a P.O. box, see instructions.		Apt. no.
City, town, or post office. If you have a foreign address, also complete spaces below.	State	ZIP code
Foreign country name	Foreign province/state/county	Foreign postal code

Presidential Election Campaign Check here if you, or your spouse if filing jointly, want $3 to go to this fund. Checking a box below will not change your tax or refund. ☐ **You** ☐ **Spouse**

At any time during 2021, did you receive, sell, exchange, or otherwise dispose of any financial interest in any virtual currency? ☐ **Yes** ☐ **No**

Standard Deduction **Someone can claim:** ☐ You as a dependent ☐ Your spouse as a dependent
☐ Spouse itemizes on a separate return or you were a dual-status alien

Age/Blindness **You:** ☐ Were born before January 2, 1957 ☐ Are blind **Spouse:** ☐ Was born before January 2, 1957 ☐ Is blind

Dependents (see instructions): If more than four dependents, see instructions and check here ▶ ☐

(1) First name Last name	(2) Social security number	(3) Relationship to you	(4) ✔ if qualifies for (see instructions): Child tax credit	Credit for other dependents
			☐	☐
			☐	☐
			☐	☐
			☐	☐

Attach Sch. B if required.

1	Wages, salaries, tips, etc. Attach Form(s) W-2			1	
2a	Tax-exempt interest	2a	b Taxable interest	2b	
3a	Qualified dividends	3a	b Ordinary dividends	3b	
4a	IRA distributions	4a	b Taxable amount	4b	
5a	Pensions and annuities	5a	b Taxable amount	5b	
6a	Social security benefits	6a	b Taxable amount	6b	
7	Capital gain or (loss). Attach Schedule D if required. If not required, check here ▶ ☐			7	
8	Other income from Schedule 1, line 10			8	
9	Add lines 1, 2b, 3b, 4b, 5b, 6b, 7, and 8. This is your **total income** ▶			9	
10	Adjustments to income from Schedule 1, line 26			10	
11	Subtract line 10 from line 9. This is your **adjusted gross income** ▶			11	
12a	**Standard deduction or itemized deductions** (from Schedule A)	12a			
b	Charitable contributions if you take the standard deduction (see instructions)	12b			
c	Add lines 12a and 12b			12c	
13	Qualified business income deduction from Form 8995 or Form 8995-A			13	
14	Add lines 12c and 13			14	
15	**Taxable income.** Subtract line 14 from line 11. If zero or less, enter -0-			15	

Standard Deduction for—
- Single or Married filing separately, $12,550
- Married filing jointly or Qualifying widow(er), $25,100
- Head of household, $18,800
- If you checked any box under *Standard Deduction,* see instructions.

Form 1040
Best Tax Software
TaxAct
H&R BLOCK
intuit
turbotax

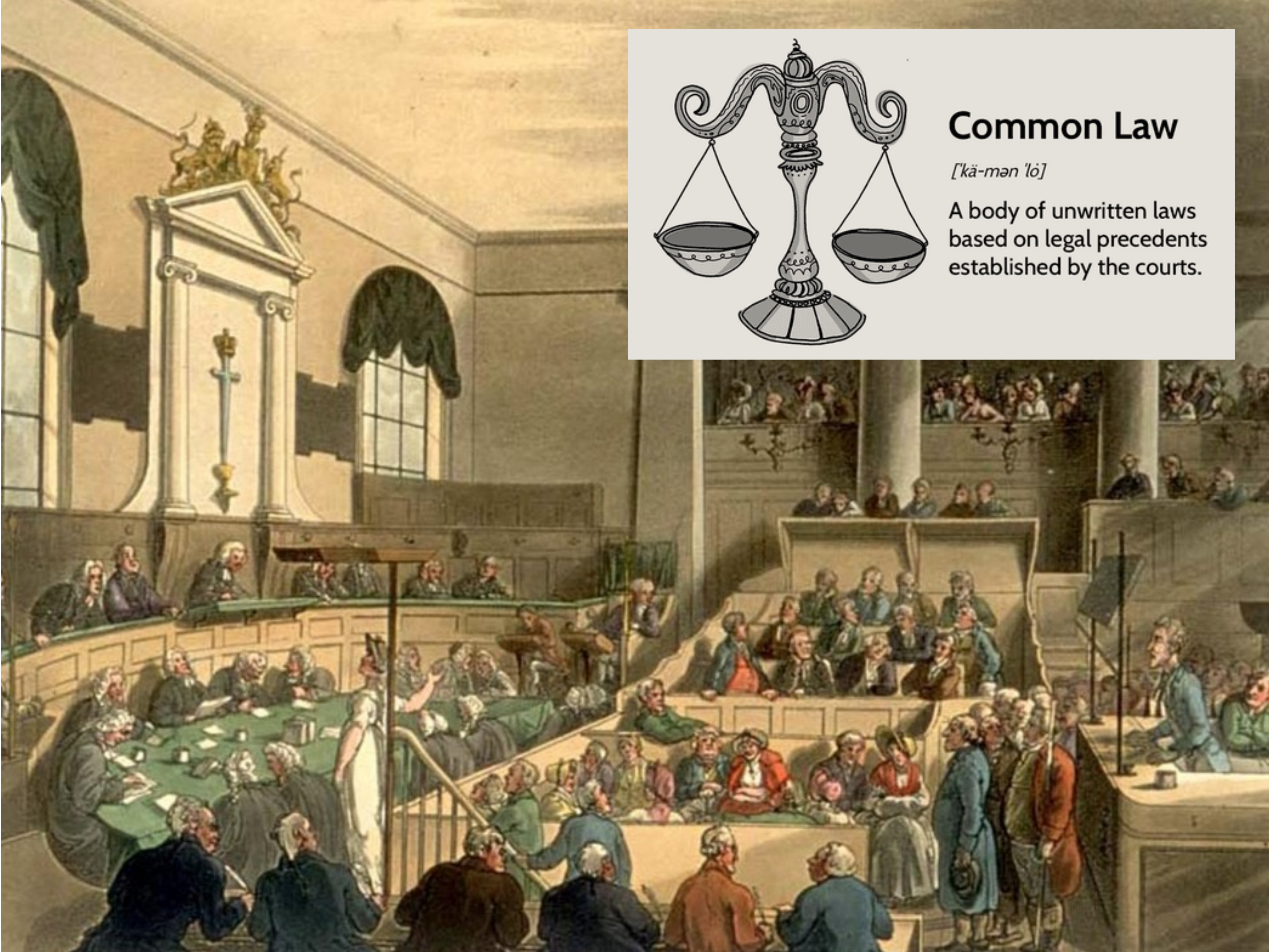

Form 1040

The misuse of tax preparation software, even if unintentional or accidental, is no defense to accuracy-related penalties under section 6662.

— Langley v. Comm'r, T.C. Memo. 2013-22

Challenges

- Absence of oracle
- Lack of trustworthy dataset
- Computationally infeasible



Common Law

['kä-mən 'lȯ]

A body of unwritten laws based on legal precedents established by the courts.

Stare decisis: similar cases must follow similar rulings

Stare decisis: similar cases must follow similar rulings

Corollary. Correctness properties for legal software can be defined in terms of relating similar cases!

Stare decisis: similar cases must follow similar rulings

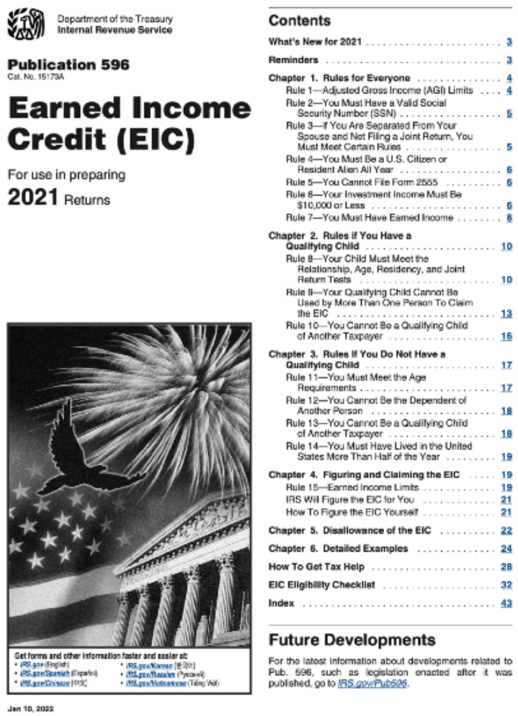
Corollary. Correctness properties for legal software can be defined in terms of relating similar cases!

Krystia Reed, JD, PhD (UTEP)

Morgan Wagner (UTEP)

Shiva Darian (CU)

Saeid Tizpaz-Niari (UTEP)


Department of the Treasury
Internal Revenue Service

Publication 596
Cat. No. 15173A

Earned Income Credit (EIC)

For use in preparing
2021 Returns

Get forms and other information faster and easier at:
- IRS.gov (English)
- IRS.gov/Spanish (Español)
- IRS.gov/Chinese (中文)
- IRS.gov/Korean (한국어)
- IRS.gov/Russian (Pусский)
- IRS.gov/Vietnamese (Tiếng Việt)

Jan 10, 2022

Contents

Future Developments

For the latest information about developments related to Pub. 596, such as legislation enacted after it was published, go to IRS.gov/Pub596.

Stare decisis: similar cases must follow similar rulings

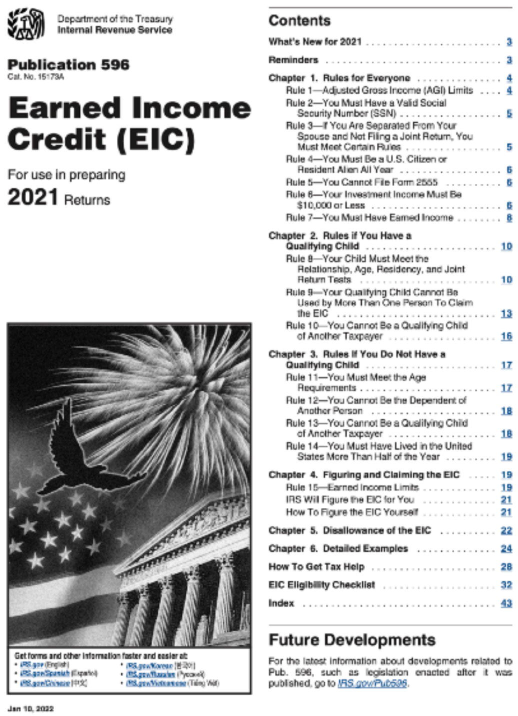
Corollary. Correctness properties for legal software can be defined in terms of relating similar cases!

Krystia Reed, JD, PhD (UTEP)

Morgan Wagner (UTEP)

Shiva Darian (CU)

Saeid Tizpaz-Niari (UTEP)


Department of the Treasury
Internal Revenue Service

Publication 596
Cat. No. 15173A

Earned Income Credit (EIC)

For use in preparing
2021 Returns

Get forms and other information faster and easier at:
• IRS.gov (English)
• IRS.gov/Spanish (Español)
• IRS.gov/Chinese (中文)
• IRS.gov/Korean (한국어)
• IRS.gov/Russian (Pусский)
• IRS.gov/Vietnamese (Tiếng Việt)

Jan 10, 2022

Contents

Future Developments

For the latest information about developments related to Pub. 596, such as legislation enacted after it was published, go to IRS.gov/Pub596.

Id	Domain	Metamorphic Property
1	Disability	$\forall \mathbf{x},\mathbf{y}((\mathbf{x}\equiv_{age}\mathbf{y}) \wedge (\mathbf{x}.age\geq 65) \wedge (\mathbf{y}.age<65)) \vee ((\mathbf{x}\equiv_{blind}\mathbf{y}) \wedge (\mathbf{x}.blind \wedge \neg\mathbf{y}.blind)) \implies \mathcal{F}(\mathbf{x}) \geq \mathcal{F}(\mathbf{y})$
2	Disability	$\forall \mathbf{x}(\mathbf{x}.sts = MFJ) \implies \forall \mathbf{y}((\mathbf{x} \equiv_{s\_age} \mathbf{y}) \wedge (\mathbf{x}.s\_age \geq 65) \wedge (\mathbf{y}.s\_age < 65)) \vee ((\mathbf{x} \equiv_{s\_blind} \mathbf{y}) \wedge (\mathbf{x}.s\_blind \wedge \neg\mathbf{y}.s\_blind)) \implies \mathcal{F}(\mathbf{x}) \geq \mathcal{F}(\mathbf{y})$
3	EITC	$\forall \mathbf{x}(\mathbf{x}.sts = MFS) \implies \forall \mathbf{y}(\mathbf{x}\equiv_{L27}\mathbf{y} \wedge \mathbf{x}.L27 > 0.0 \wedge \mathbf{y}.L27 = 0.0) \implies \mathcal{F}(\mathbf{x})=\mathcal{F}(\mathbf{y})$
4	EITC	$\forall \mathbf{x}(\mathbf{x}.sts=MFJ) \wedge (\mathbf{x}.AGI>56,844) \implies \forall \mathbf{y}(\mathbf{x}\equiv_{L27}\mathbf{y} \wedge \mathbf{x}.L27>0.0 \wedge \mathbf{y}.L27=0.0) \implies \mathcal{F}(\mathbf{x})=\mathcal{F}(\mathbf{y})$
5	EITC	$\forall \mathbf{x}(\mathbf{x}.sts=MFJ) \implies \forall \mathbf{y}(\mathbf{x}\equiv_{AGI}\mathbf{y} \wedge \mathbf{x}.AGI\leq 56,844 \wedge \mathbf{y}.AGI>56,844) \vee (\mathbf{x}\equiv_{L27}\mathbf{y} \wedge \mathbf{x}.L27>0.0 \wedge \mathbf{y}.L27=0.0) \vee (\mathbf{x}\equiv_{QC}\mathbf{y} \wedge \mathbf{x}.QC\geq 1 \wedge \mathbf{y}.QC=0) \implies \mathcal{F}(\mathbf{x})\geq\mathcal{F}(\mathbf{y})$
6	EITC	$\forall \mathbf{x}(\mathbf{x}.sts=MFJ)\wedge(\mathbf{x}.AGI\leq 56,844)\wedge(\mathbf{x}.QC\leq 3) \implies \forall \mathbf{y}((\mathbf{x}\equiv_{L27}\mathbf{y})\wedge\mathbf{x}.L27\geq\mathbf{y}.L27) \implies \mathcal{F}(\mathbf{x})\geq\mathcal{F}(\mathbf{y})$
7	CTC	$\forall \mathbf{x}(\mathbf{x}.sts=MFS)\wedge(\mathbf{x}.AGI\leq 200k)\forall \mathbf{y}((\mathbf{x}\equiv_{L19}\mathbf{y})\wedge(\mathbf{x}.L19\geq\mathbf{y}.L19)) \implies \mathcal{F}(\mathbf{x})\geq\mathcal{F}(\mathbf{y}))$
8	CTC	$\forall \mathbf{x},\mathbf{x}'(\mathbf{x}.sts=\mathbf{x}'.sts=MFJ)\wedge(\mathbf{x}.AGI<400k)\wedge(\mathbf{x}'.AGI\geq 400k)\wedge\lceil \mathbf{x}'.AGI-400k\rceil_{1k} * 0.05<\mathbf{x}'.QC * 2k + \mathbf{x}.OD * 0.5k \implies \forall \mathbf{y},\mathbf{y}'(\mathbf{x}\equiv_{\{QC,OD\}}\mathbf{y})\wedge(\mathbf{x}'\equiv_{\{QC,OD\}}\mathbf{y}') \wedge (0\leq\mathbf{y}.QC=\mathbf{y}'.QC\leq\mathbf{x}.QC=\mathbf{x}'.QC \leq 10) \wedge (0\leq\mathbf{y}.OD=\mathbf{y}'.OD \leq \mathbf{x}.OD=\mathbf{x}'.OD \leq 10) \implies (\mathcal{F}(\mathbf{x}) - \mathcal{F}(y)) \geq (\mathcal{F}(x') - \mathcal{F}(y'))$
9	ETC	$\forall \mathbf{x}(\mathbf{x}.sts = MFS) \implies \forall \mathbf{y}(\mathbf{x}\equiv_{L29}\mathbf{y} \wedge \mathbf{x}.L29 > 0.0 \wedge \mathbf{y}.L29 = 0.0) \implies \mathcal{F}(\mathbf{x})=\mathcal{F}(\mathbf{y})$
10	ETC	$\forall \mathbf{x}(\mathbf{x}.sts=MFJ) \wedge (\mathbf{x}.AGI\geq 180k) \implies \forall \mathbf{y}(\mathbf{x}\equiv_{L29}\mathbf{y} \wedge \mathbf{x}.L29>0.0 \wedge \mathbf{y}.L29=0.0) \implies \mathcal{F}(\mathbf{x})=\mathcal{F}(\mathbf{y})$
11	ETC	$\forall \mathbf{x}(\mathbf{x}.sts=MFJ) \wedge (\mathbf{x}.AGI\leq 160k) \implies \forall \mathbf{y}(\mathbf{x}\equiv_{L29}\mathbf{y} \wedge \mathbf{x}.L29\geq\mathbf{y}.L29) \implies \mathcal{F}(\mathbf{x})\geq\mathcal{F}(\mathbf{y})$
12	ETC	$\forall \mathbf{x},\mathbf{x}'(\mathbf{x}.sts=\mathbf{x}'.sts=MFJ) \wedge (\mathbf{x}.AGI\leq 160k) \wedge (160k<\mathbf{x}'.AGI<180k) \implies \forall \mathbf{y},\mathbf{y}'((\mathbf{x} \equiv_{L29} \mathbf{y}) \wedge (\mathbf{x}' \equiv_{L29} \mathbf{y}') \wedge (\mathbf{x}.L29 = \mathbf{x}'.L29 \geq \mathbf{y}.L29 = \mathbf{y}'.L29)) \implies (\mathcal{F}(\mathbf{x}) - \mathcal{F}(y)) \geq (\mathcal{F}(x') - \mathcal{F}(y'))$
13	ID	$\forall \mathbf{x},\mathbf{y}(\mathbf{x}\equiv_{MDE}\mathbf{y}) \wedge (\mathbf{x}.MDE\leq\mathbf{x}.AGI * 7.5\%) \wedge (\mathbf{y}.MDE=0.0) \implies \mathcal{F}(\mathbf{x})=\mathcal{F}(\mathbf{y})$
14	ID	$\forall \mathbf{x}(\neg\mathbf{x}.iz) \implies \forall \mathbf{y}(\mathbf{x}\equiv_{MDE}\mathbf{y} \wedge \mathbf{x}.MDE>0.0 \wedge \mathbf{y}.MDE=0.0) \implies \mathcal{F}(\mathbf{x})=\mathcal{F}(\mathbf{y})$
15	ID	$\forall \mathbf{x}(\mathbf{x}.sts=MFJ) \implies \forall \mathbf{y}((\mathbf{x}\equiv_{iz,L12}\mathbf{y})\wedge(\mathbf{x}.iz\wedge\neg\mathbf{y}.iz)\wedge(\mathbf{x}.L12\leq 24.8k\wedge\mathbf{y}.L12=0.0)) \implies \mathcal{F}(\mathbf{x})\leq\mathcal{F}(\mathbf{y})$
16	ID	$\forall \mathbf{x}(\mathbf{x}.sts=MFJ) \implies \forall \mathbf{y}((\mathbf{x}\equiv_{iz,L12}\mathbf{y})\wedge(\mathbf{x}.iz\wedge\neg\mathbf{y}.iz)\wedge(\mathbf{x}.L12>24.8k\wedge\mathbf{y}.L12=0.0)) \implies \mathcal{F}(\mathbf{x})\geq\mathcal{F}(\mathbf{y})$

Stare decisis: similar cases must follow similar rulings

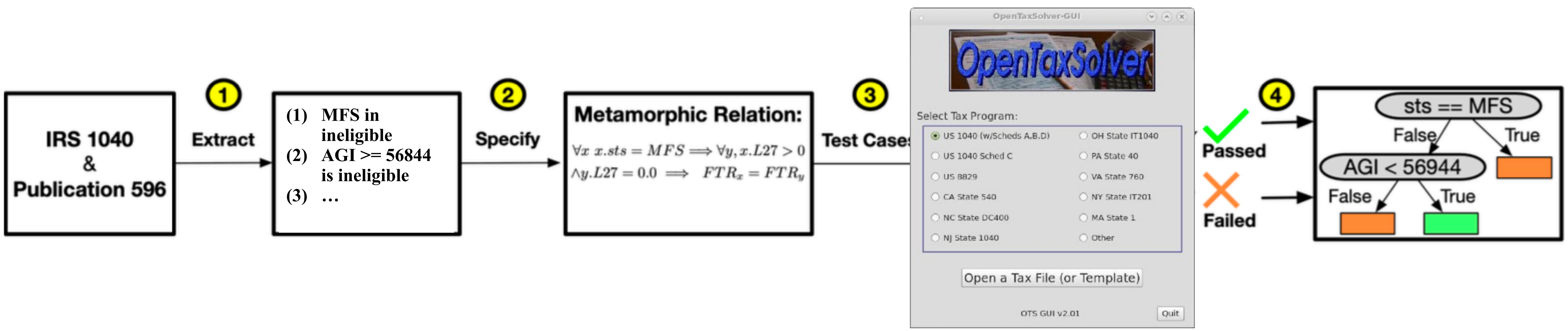
Corollary. Correctness properties for legal software can be defined in terms of relating similar cases!

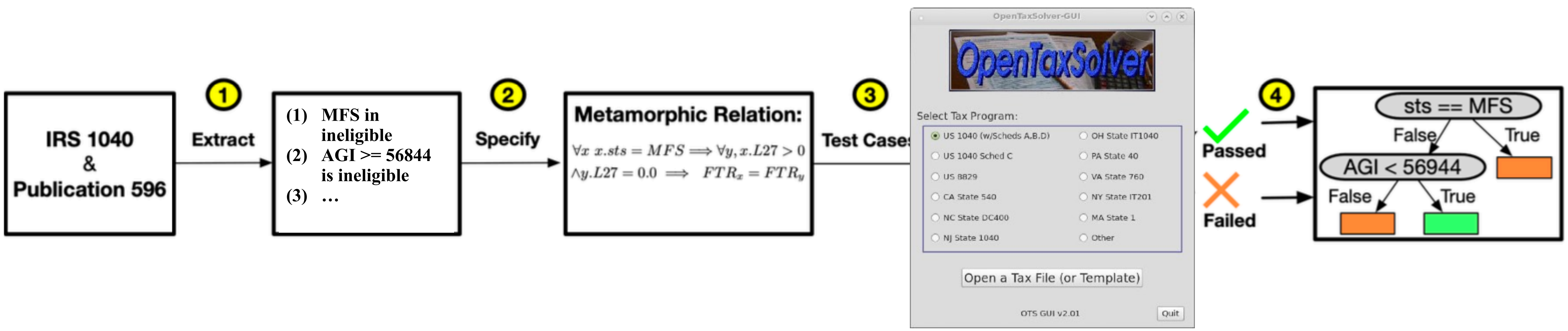
Krystia Reed, JD, PhD (UTEP)

Morgan Wagner (UTEP)

Shiva Darian (CU)

Saeid Tizpaz-Niari (UTEP)


IRS 1040
&
Publication 596
1
Extract
(1) MFS in ineligible
(2) AGI >= 56844 is ineligible
(3) …
2
Specify
Metamorphic Relation:
$\forall x\ x.sts = MFS \implies \forall y, x.L27 > 0$
$\wedge y.L27 = 0.0 \implies FTR_x = FTR_y$
3
Test Cases
OpenTaxSolver-GUI
OpenTaxSolver
Select Tax Program:
US 1040 (w/Scheds A,B,D)
OH State IT1040
US 1040 Sched C
PA State 40
US 8829
VA State 760
CA State 540
NY State IT201
NC State DC400
MA State 1
NJ State 1040
Other
Open a Tax File (or Template)
OTS GUI v2.01
Quit
Passed
Failed
4
sts == MFS
False
True
AGI < 56944
False
True



i) missing eligibility conditions completely (properties #3-#4 and #9-#10);

ii) unexpected outcomes when the federal tax returns get close to zero (property #11);

iii) applying itemized deductions when expenses are less than 7.5% of AGI (property #13);

iv) itemized deductions slightly exceed a standard deduction, but result in a lower tax return (property #16).

CUPLV

Programming Languages and Verification
University of Colorado Boulder

http://ashutoshtrivedi
plv.colorado.edu