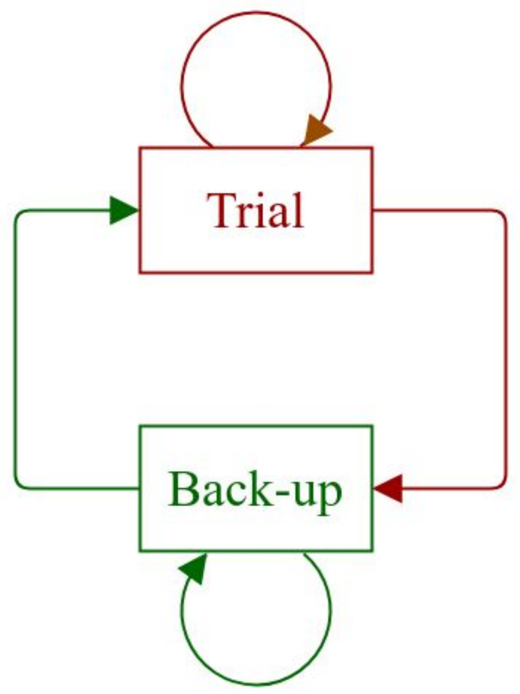
A Framework for Reinforcement Learning and Planning: Extended Abstract

Thomas Moerland, Joost Broekens and Catholijn Jonker
Delft University of Technology, The Netherlands

Main Idea
Reinforcement learning and MDP planning solve the same problem.
Can we identify one underlying algorithmic space, and can we disentangle it?

Main inspiration
Trial-based Heuristic Tree Search (THTS) (Keller & Helmert, 2013) -- Similar effort for tree search alone

Central concept:
Trials *A single call to the environment/transition model*
Back-ups *A single operation to back-up the obtained information from the trial*



Framework
<u>Six main questions</u>:
1) Where do we put our computational effort?
2) How do we select the next trial?
3) How do we estimate the cumulative return after a trial?
4) How do we back-up the new information to previous nodes?
5) How do we represent the solution?
6) How do we update the solution?

Full paper
Details all decisions, and presents a large table comparing well-known model-free RL, model-based RL, and planning methods.

Potential Benefits
- Bridge both fields (conceptually + terminology)
- Mutual inspiration
- In each separate field (e.g., no framework for RL itself)

Overview of framework

Table 1: Overview of dimensions in the Framework for Reinforcement learning and Planning (FRAP). For any planning or reinforcement learning algorithm, we should be able to identify the decision on each of the dimensions. The subconsiderations and possible options are shown in the right columns. IM = Intrinsic Motivation.

Dimension	Consideration	Choices
1. Comp. effort	- State set	All ↔ reachable ↔ relevant
2. Trial selection	- Candidate set	Step-wise ↔ frontier
	- Exploration	Random ↔ Value-based ↔ State-based -For value: mean value, uncertainty, priors -For state: ordered, priors (shaping), novelty, knowledge IM, competence IM
	- Phases	One-phase ↔ two-phase
	- Reverse trials	Yes ↔ No
3. Return estim.	- Sample depth	$1 \leftrightarrow n \leftrightarrow \infty$
	- Bootstrap func.	Learned ↔ heuristic ↔ none
4. Back-up	- Back-up policy	On-policy ↔ off-policy
	- Policy expec.	Expected ↔ sample
	- Dynamics expec.	Expected ↔ sample
5. Representation	- Function type	Value ↔ policy ↔ both (actor-critic) - For all: generalized ↔ not generalized
	- Function class	Tabular ↔ function approximation - For tabular: local ↔ global
6. Update	- Loss	- For value: e.g., squared -For policy: e.g., (det.) policy gradient ↔ value gradient ↔ cross-entropy, etc.
	- Update	Gradient-based ↔ gradient-free - For gradient-based, special cases: replace & average update

Full paper has a large table comparing well-known planning and RL algorithms on these dimensions

Email: T.M.Moerland@tudelft.nl

A Framework for Reinforcement Learning and Planning: Extended Abstract

Thomas Moerland, Joost Broekens and Catholijn Jonker
Delft University of Technology, The Netherlands

Paper	Environment	Learned model	Comp. effort	Trial selection					
				Candidate Set	Exploration	Sub-category	Phases	Reverse Trials	Description
Dynamic Programming (Bellman, 1966)	Reversible analytic		All	State set	State	Ordered	1		Sweep
Depth-first exh. search (Russell and Norvig, 2016)	Reversible analytic		Reach.	Step	State	Ordered	1		Sweep
Heuristic search (e.g., A^* (Hart et al., 1968))	Reversible analytic		Reach.	Frontier	Value	Prior	1		Greedy on heuristic
MCTS (Browne et al., 2012)	Reversible sample		Reach.	Step	Value	Uncertainty	2		Upper confidence bound
Real-time DP (Barto et al., 1995)	Reversible analytic		Reach.	Step	State	Ordered	1		Random starts
Q-learning (Watkins and Dayan, 1992)	Irreversible sample		Reach.	Step	State	Ordered	1		Random starts
SARSA + eligibility trace (Sutton and Barto, 2018)	Irreversible sample		Reach.	Step	Value	Mean values	1		e.g., Boltzmann
REINFORCE (Williams, 1992)	Irreversible sample		Reach.	Step	Random	-	1		Stochastic policy
DQN (Mnih et al., 2015)	Irreversible sample		Reach.	Step	Value	Random	1		ϵ-greedy
PPO (Schulman et al., 2017b)	Irreversible sample		Reach.	Step	Value	Mean values	1		Stochastic policy with entropy regularization
DDPG (Lillicrap et al., 2015)	Irreversible sample		Reach.	Step	Random	-	1		Noise process (Ornstein-Uhlenbeck)
Go-Explore[†] (Ecoffet et al., 2019)	Irreversible sample		Reach.	Frontier	State+ val+rand	Novelty+ prior+random	1		Frontier prior.: visit freq. + heuristics. On frontier: random perturbation.
AlphaStar (Vinyals et al., 2019)	Irreversible sample		Reach.	Step	State+ Value	Prior+mean values	1		Imitation learning + shaping rewards + entropy regularization
Dyna-Q (Sutton, 1990)	Irreversible sample	✓	Reach.	Step	State+ Value	Knowledge+ mean values	1		Novelty bonus + Boltzmann
Prioritized sweeping (Moore and Atkeson, 1993)	Irreversible sample	✓	Reach.	Step	State	Novelty	1	✓	Visitation frequency + Reverse trials
PILCO (Deisenroth and Rasmussen, 2011)	Irreversible sample	✓	Reach.	Step	Random	-	2		Stochastic policy on initialization
AlphaGo (Silver et al., 2017)	Reversible Sample		Reach.	Step	Value + random	Uncertainty	2		Upper confidence bound + noise
Knowledge, e.g., surprise (Achiam and Sastry, 2017)	Irreversible sample	✓	Reach.	Step	State	Knowledge	1		Intrinsic reward for surprise
Competence IM, e.g., (Péré et al., 2018)	Irreversible sample	✓	Reach.	Frontier	State	Competence	1		Sampling in learned goal space

Email: T.M.Moerland@tudelft.nl

A Framework for Reinforcement Learning and Planning: Extended Abstract

Thomas Moerland, Joost Broekens and Catholijn Jonker
Delft University of Technology, The Netherlands

Paper	Cumulative return			Back-up		Representation		Update	
	Sample depth	Bootstrap type	Back-up policy	Action expectation	Dynamics Expectation	Function type	Function class	Loss	Update type
Dynamic Programming (Bellman, 1966)	1	Learned	Off-policy	Max	Exp.	Value	Global table	(Squared)	Replace
Depth-first exh. search (Russell and Norvig, 2016)	∞	None	Off-policy	Max	Exp	Value	Global table	(Squared)	Replace
Heuristic search (e.g., A^* (Hart et al., 1968))	1	Heuristic	Off-policy	Max	Determ.	Value	Global table	(Squared)	Replace
MCTS (Browne et al., 2012)	∞	None	On-policy	Sample	Sample	Value	Local table	(Squared)	Average
Real-time DP (Barto et al., 1995)	1°	Learned	Off-policy	Max	Exp.	Value	Global table	(Squared)	Replace
Q-learning (Watkins and Dayan, 1992)	1	Learned	Off-policy	Max	Sample	Value	Global table	Squared	Gradient
SARSA + eligibility trace (Sutton and Barto, 2018)	$1-n$ (eligibility)	Learned	On-policy	Sample	Sample	Value	Global table	Squared	Gradient
REINFORCE (Williams, 1992)	∞	None	On-policy	Sample	Sample	Policy	Func.approx. (NN)	Policy gradient	Gradient
DQN (Mnih et al., 2015)	1	Learned	Off-policy	Max	Sample	Value	Func.approx. (NN)	Squared	Gradient
PPO (Schulman et al., 2017b)	$1-n$ (eligibility)	Learned	On-policy	Sample	Sample	Policy	Func.approx. (NN)	Policy gradient	Gradient (trust.reg.)
DDPG (Lillicrap et al., 2015)	1	Learned	Off-policy	Max	Sample	Policy+ value	Func.approx. (NN)	Determ. policy grad. + squared	Gradient
Go-Explore† (Ecoffet et al., 2019)	1	Heuristic	On-policy	Sample	Sample	Policy	Global table	(Squared)	Replace
AlphaStar (Vinyals et al., 2019)	1-n (importance weighted)	Learned	On-policy	Sample	Sample	Policy+ value	Func.approx. (NN)	Policy gradient + squared	Gradient
Dyna (Sutton, 1990)	1	Learned	On-policy	Sample	Sample	Value	Global table	Squared	Gradient
Prioritized sweeping (Moore and Atkeson, 1993)	1	Learned	Off-policy	Max	Exp.	Value	Global table	Squared	Gradient
PILCO (Deisenroth and Rasmussen, 2011)	∞	None	On-policy	Sample	Sample	Policy	Func.approx. (GP)	Value gradient	Gradient
AlphaGo (Silver et al., 2017)	MCTS: 1-n Value: ∞	Learned	On-policy	Sample	Sample	Policy+ value	Func.approx. (NN)+ local table	Cross-entropy+ Squared	Average+ Gradient
Knowledge, e.g., surprise (Achiam and Sastry, 2017)	∞	None	On-policy	Sample	Sample	Policy	Func.approx. (NN)	Policy gradient	Gradient
Competence IM, e.g., (Péré et al., 2018)	∞	None	On-policy	Sample	Sample	Generalized policy$^\$$	Func.approx. (k-NN)	k-NN loss	Gradient-free

Email: T.M.Moerland@tudelft.nl