

SIT796 Reinforcement Learning

Introduction to Multi-Objective Reinforcement Learning

Presented by:
Thommen George Karimpanal
School of Information Technology



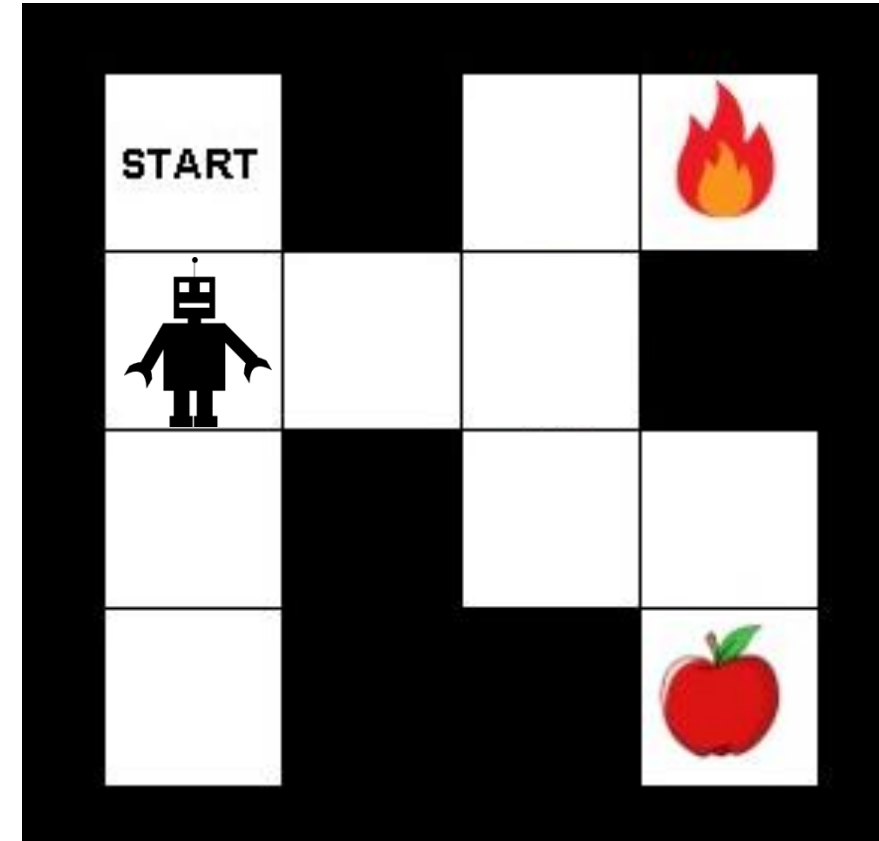
DEAKIN
UNIVERSITY

RL so far:

Agent takes actions in its environment

Maximise sum of rewards $\sum_{k=0}^{\infty} \gamma^k R_{t+k+1}$

Can all problems be expressed in the form of rewards?



Are rewards enough?



Sutton's Reward Hypothesis: "All of what we mean by goals and purposes can be well thought of as maximization of the expected value of the cumulative sum of a received *scalar* signal (reward)."

But how to compress all objectives into such a scalar reward? – Not easy!

A more pragmatic approach: we often care about multiple objectives, which need to be optimised together

Multi-Objective Reinforcement Learning



Artificial Intelligence
Volume 299, October 2021, 103535



Reward is enough

[David Silver](#)  , [Satinder Singh](#), [Doina Precup](#), [Richard S. Sutton](#)


[Show more](#) 

[+](#) Add to Mendeley [🔗](#) Share [📄](#) Cite

<https://doi.org/10.1016/j.artint.2021.103535> 

[Get rights and content](#) 

Under a Creative Commons [license](#) 

 open access

Multiobjective RL



How to deal with multiple objectives?

One reward function for each objective r_1, r_2

But how deal with these?

Combine them into a scalar value: $r = w_1 r_1 + w_2 r_2$

w specifies the preferences

Multiobjective RL



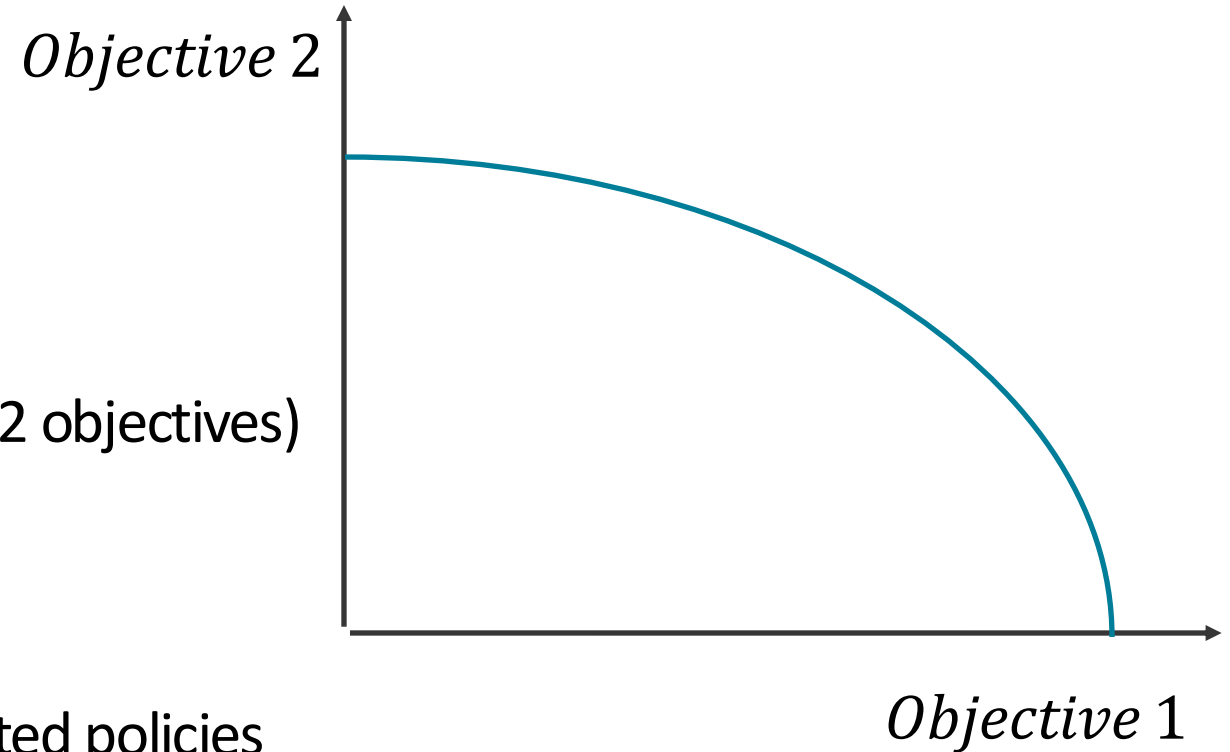
Single objective: only 1 solution

MultiObjective: multiple solutions (even with 2 objectives)

Optimal Solutions (Pareto front): non-dominated policies

-same performance in terms of the scalarised rewards

In general, $n_{\text{objectives}} > 2$



Specifying Preferences



At the end, we care about agent behaviours

Pick a w , check if any of the solutions correspond to the desired behaviour

If not, pick a different w

Semi-blind process

Problems with Scalarisation



Undue burden on engineers/designers

Linear model cannot encompass complex preferences we may have

Preferences change over time

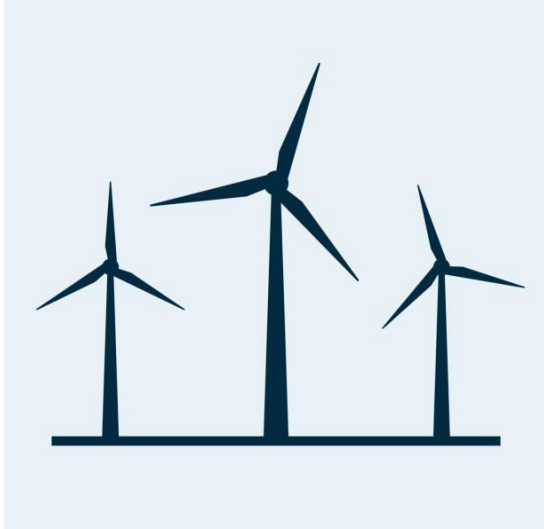
Power production application –

$$r = w_p r_p + w_c r_c$$

for double the power, cannot just double w_p

The solutions are not explainable

MORL Examples



Wind farms: maximise power, minimise wear

Other non-linear factors



Transport: minimise travel time, also minimise cost



Can prepare yourself
with a set of policies
– useful when trains
are cancelled etc.,

SIT796 Reinforcement Learning

MORL: Problem setting and Formulation

Presented by:
Thommen George Karimpanal
School of Information Technology



DEAKIN
UNIVERSITY

MultiObjective MDP $\langle S, A, T, \gamma, \mu, \mathbf{R} \rangle$

S is the state space

A is the action space

$T : S \times A \times S \rightarrow [0, 1]$ is a probabilistic transition function

$\gamma \in [0, 1)$ is a discount factor

$\mu : S \rightarrow [0, 1]$ is a probability distribution over initial states

$\mathbf{R} : S \times A \times S \rightarrow \mathbb{R}^d$ is a vector-valued reward function, specifying the immediate reward for each of the considered $d \geq 2$ objectives

$$\mathbf{V}^\pi = \mathbb{E} \left[\sum_{k=0}^{\infty} \gamma^k \mathbf{r}_{k+1} \mid \pi, \mu \right]$$

MOMDP: Utility Function



In MOMDPS, the value function is a vector

Utility functions scalarise the multiobjective value vector to a scalar

$$V_u^\pi = u(\mathbf{V}^\pi) \qquad u : \mathbb{R}^d \rightarrow \mathbb{R}$$

The optimal policy is not clearly defined unless we know how the objectives are prioritised

However, scalarisation has its own problems, as discussed earlier

SIT796 Reinforcement Learning

MORL: Taxonomy

Presented by:
Thommen George Karimpanal
School of Information Technology



DEAKIN
UNIVERSITY

Single

vs

Multiple policies

Single policy- If utility is known at the time of planning

Multiple policies- If utility is unknown

Linear utility

vs

Non-linear Utility policies

User preferences may not be adequately expressed

May better express user preferences

Deterministic policies

vs

Stochastic policies

When utility is linear, the optimal policy is deterministic and stationary

In some cases, stochastic policies should never be permitted

Scalarised Expected Returns (SER)

$$V_u^\pi = u\left(\mathbb{E}\left[\sum_{i=0}^{\infty} \gamma^i \mathbf{r}_i \mid \pi, s_0\right]\right).$$

Expected Scalarised Returns (ESR)

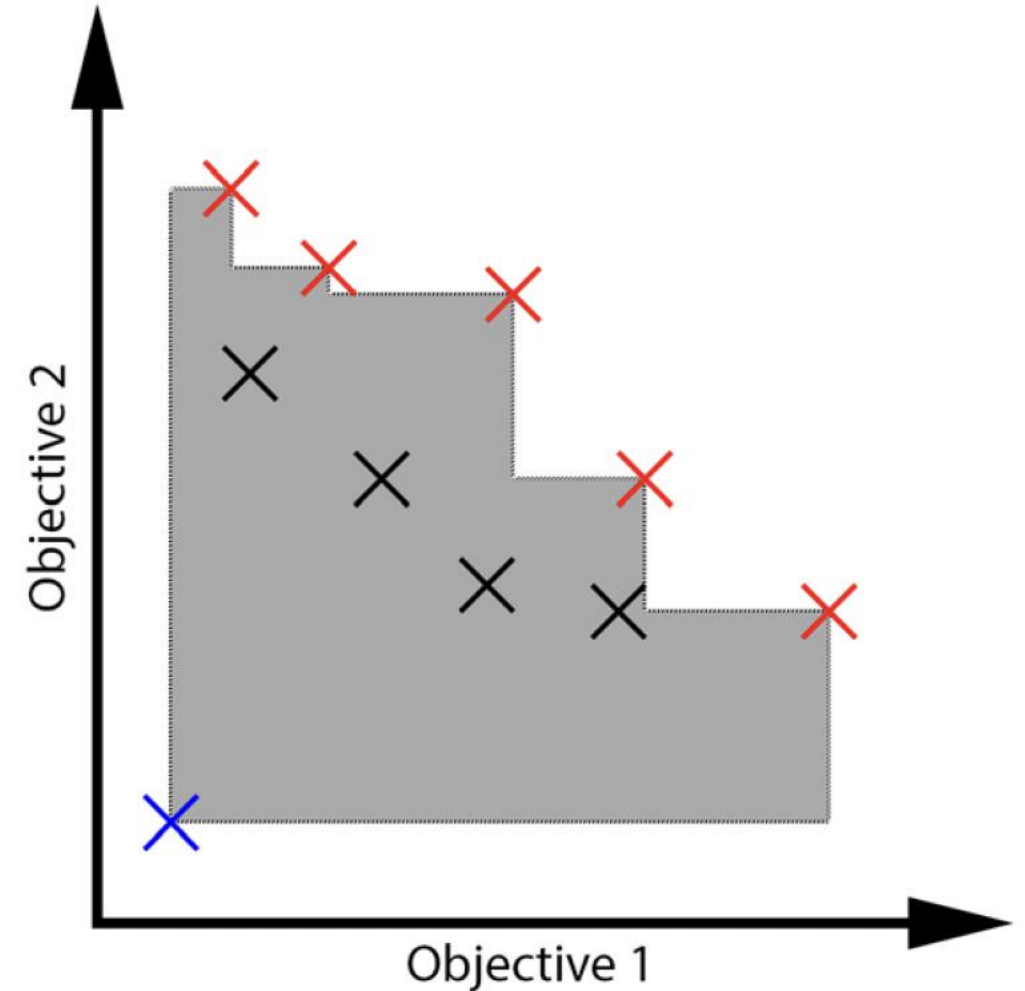
$$V_u^\pi = \mathbb{E}\left[u\left(\sum_{i=0}^{\infty} \gamma^i \mathbf{r}_i\right) \mid \pi, s_0\right]$$

These lead to different solutions when the utility is non-linear

Hypervolume

- ✗ Part of non-dominated set
- ✗ Dominated

Larger the hypervolume the better



When the hypervolume is similar, choose the solution that is most spread out in the space

$$Sp(\mathcal{S}) = \frac{1}{|\mathcal{S}| - 1} \sum_{j=1}^m \sum_{i=1}^{|\mathcal{S}|-1} (\tilde{\mathcal{S}}_j(i) - \tilde{\mathcal{S}}_j(i+1))^2$$

Sparsity metric for m objectives. S is the pareto front approximation

SIT796 Reinforcement Learning

MORL: Algorithms

Presented by:
Thommen George Karimpanal
School of Information Technology



DEAKIN
UNIVERSITY

MORL: Single Policy Algorithms



Adaptation of Q learning

Q vectors instead of Q values

Scalarisation function is needed to for action selection

*May fail to converge if transitions are stochastic

MORL: Multi-Policy Algorithms



Pareto Q learning: based on dynamic programming variant that returned pareto dominating policies

$$\hat{Q}_{set}(s, a) = \mathbf{R}(s, a) \oplus \gamma \sum_{s' \in S} T(s'|s, a) V^{ND}(s')$$

$$V^{ND}(s') = ND(\cup_{a'} \hat{Q}_{set}(s', a'))$$

Episodic problems with terminal state

Model-free

Produces deterministic non-stationary policies

Set Evaluation Mechanisms. (based on hypervolume, cardinality etc.,)

Pareto Q learning

- 1: Initialize $\hat{Q}_{set}(s, a)$'s as empty sets
- 2: **for** each episode t **do**
- 3: Initialize state s
- 4: **repeat**
- 5: Choose action a from s using a policy derived from the \hat{Q}_{set} 's
- 6: Take action a and observe state $s' \in S$ and reward vector $\mathbf{r} \in \mathbb{R}^m$
- 7:
- 8: $ND_t(s, a) \leftarrow ND(\cup_{a'} \hat{Q}_{set}(s', a'))$ ▷ Update ND policies of s' in s
- 9: $\bar{\mathcal{R}}(s, a) \leftarrow \bar{\mathcal{R}}(s, a) + \frac{\mathbf{r} - \bar{\mathcal{R}}(s, a)}{n(s, a)}$ ▷ Update average immediate rewards
- 10: $s \leftarrow s'$ ▷ Proceed to next state
- 11: **until** s is terminal
- 12: **end for**

Set Evaluation Mechanisms

Hypervolume Set Evaluation

```
1: Retrieve current state  $s$ 
2: evaluations = {}
3: for each action  $a$  do
4:    $hv_a \leftarrow HV(\hat{Q}_{set}(s, a))$ 
5:   Append  $hv_a$  to evaluations
6: end for
7: return evaluations
```

▷ Store hypervolume of the $\hat{Q}_{set}(s, a)$

Cardinality Set Evaluation: based on number of Pareto dominating \hat{Q} -vectors of the Qset of each action

```
1: Retrieve current state  $s$ 
2: allQs = {}
3: for each action  $a$  in  $s$  do
4:   for each  $\hat{Q}$  in  $\hat{Q}_{set}(s, a)$  do
5:     Append  $[a, \hat{Q}]$  to allQs
6:   end for
7: end for
8:  $NDQs \leftarrow ND(\text{allQs})$ 
9: return NDQs
```

▷ Store for each \hat{Q} -vector a reference to a

▷ Keep only the non-dominating solutions

MultiObjective Gymnasium

<https://mo-gymnasium.farama.org/index.html>

SIT796 Reinforcement Learning

MORL: Related Topics and Open Questions

Presented by:
Thommen George Karimpanal
School of Information Technology



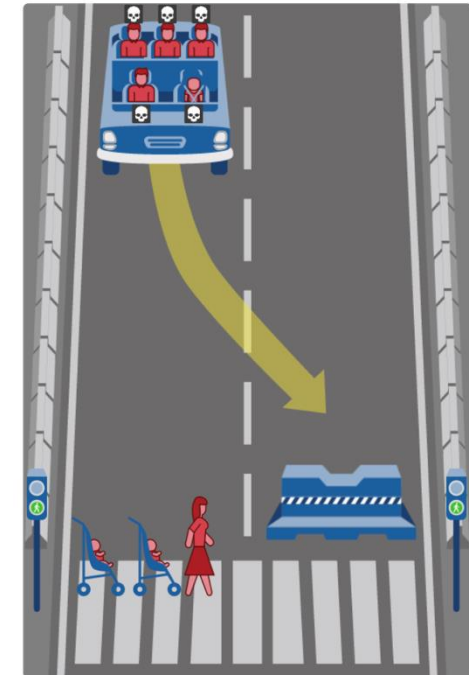
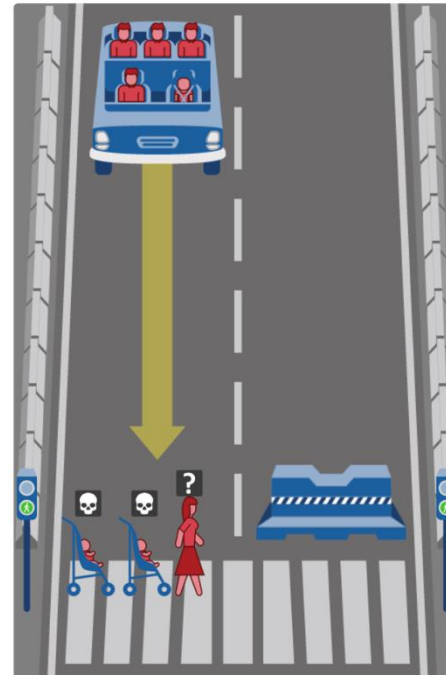
DEAKIN
UNIVERSITY

Human-alignment- how to take humans' preferences into account

RL Safety- Training RL algorithms efficiently, but at the same time, avoiding unsafe actions

Explainable RL, Moral decision making

<https://www.moralmachine.net/>



Many-Objective Problems ($n_{\text{objectives}} > 4$)

MultiAgent RL Problems (MOMADM). - several challenging problems

How to dynamically identify and add objectives?

This lecture focused on introducing Multi-objective RL.

For more detailed information see:

- Hayes, Conor F., et al. "A practical guide to multi-objective reinforcement learning and planning." *Autonomous Agents and Multi-Agent Systems* 36.1 (2022): 26.
- Van Moffaert, Kristof, and Ann Nowé. "Multi-objective reinforcement learning using sets of pareto dominating policies." *The Journal of Machine Learning Research* 15.1 (2014): 3483-3512.
- Miguel Terra-Neves, Ines Lynce, Vasco Manquinho — Stratification for Constraint-Based Multi-Objective Combinatorial Optimization
- Diederik M. Roijers, Luisa M. Zintgraf, Pieter Libin, Ann Nowé — Interactive Multi-Objective Reinforcement Learning in Multi-Armed Bandits for Any Utility Function
- Felten, Florian, et al. "A toolkit for reliable benchmarking and research in multi-objective reinforcement learning." *Advances in Neural Information Processing Systems* 36 (2024).