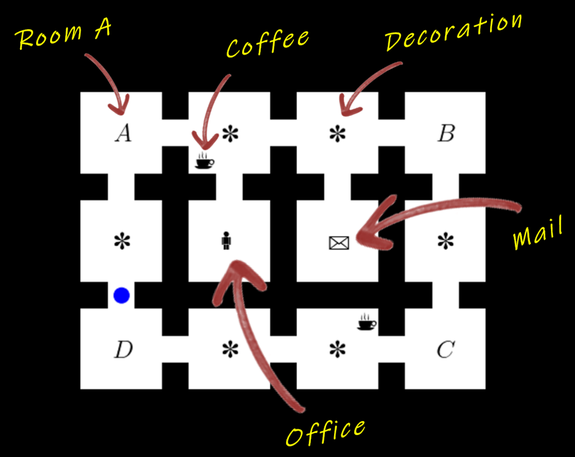
Skill Machines: Temporal Logic Composition in Reinforcement Learning

Geraud Nangue Tasse*, Devon Jarvis, Steven James and Benjamin Rosman

University of the Witwatersrand, Johannesburg, South Africa

A framework for solving any task specified by reward machines without further learning by composing skill primitives learned in a reward-free environment

Introduction



- Consider an agent in an environment where it needs to solve multiple tasks specified by Reward Machines (RM) [1] (can also be obtained from LTL).
  - An RM is a Finite state machine
  - Edges are over expressions over $\mathcal{P} = \{A, B, C, D, ✻, ☕, ✉, 🚹, ✉^+, 🚹^+\}$
  - Labelling function (sensors) $L : \mathcal{S} \times \mathcal{A} \times \mathcal{S} \to 2^{\mathcal{P}}$
- How to solve those tasks efficiently? i.e without costly learning every time

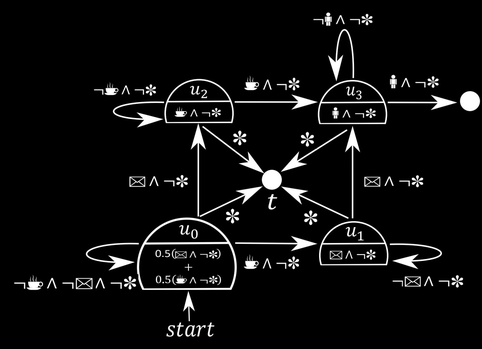
[1] R. T. Icarte et al., "Using reward machines for high-level task specification and decomposition in reinforcement learning," ICML, 2018
[2] G. Nangue Tasse et al., "World value functions: Knowledge representation for multitask reinforcement learning," RLDM, 2022.
[3] G. Nangue Tasse et al., "A Boolean task algebra for reinforcement learning" NeurIPS, 2020.

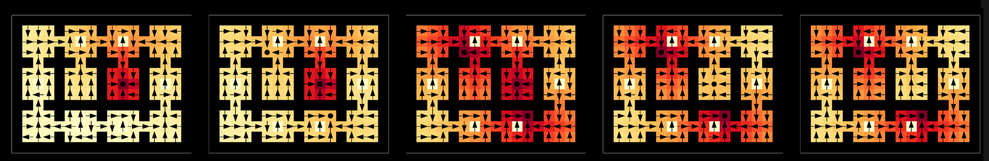
Skill Machines

An SM is a Finite state machine
Environment modified to learn primitives:
- Terminating action
- Goal space: $\mathcal{G} = 2^{\mathcal{P}}$
- Constraints: $C \subseteq \mathcal{P}$

Skill primitives [2] for each $\mathcal{P} = \{A, B, C, D, ✻, ☕, ✉, 🚹, ✉^+, 🚹^+\}$

Skill per nodes
- Boolean composition of skill primitives [3]
- Extended to linear compositions



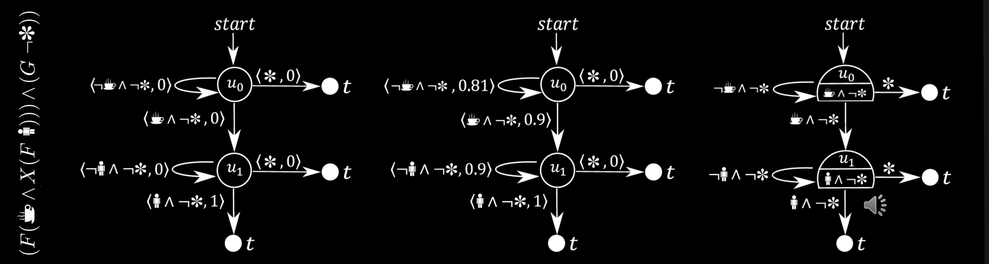


$1(✉ \wedge \neg ✻) + 0(☕ \wedge \neg ✻)$ | $0.75(✉ \wedge \neg ✻) + 0.25(☕ \wedge \neg ✻)$ | $0.5(✉ \wedge \neg ✻) + 0.5(☕ \wedge \neg ✻)$ | $0.25(✉ \wedge \neg ✻) + 0.75(☕ \wedge \neg ✻)$ | $0(✉ \wedge \neg ✻) + 1(☕ \wedge \neg ✻)$

Learning Skill Machines

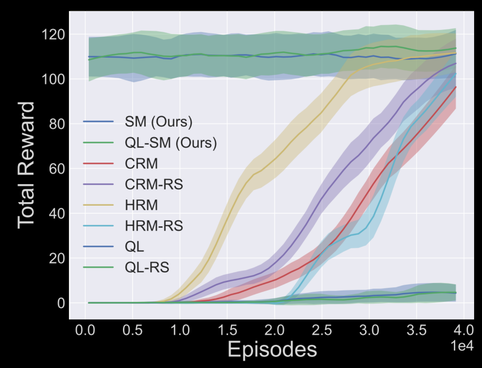
LTL → Reward Machine → Value iteration → Skill Machine

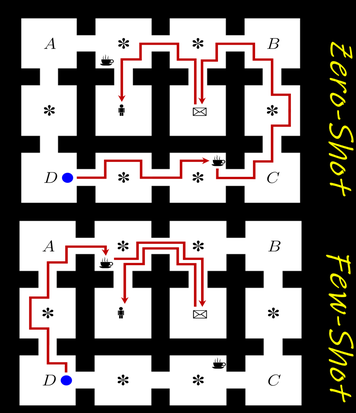
$(F(☕ \wedge X(F(🚹))) \wedge (G\ \neg ✻))$



Experiments: Office Gridworld

Task	Description | LTL
3	Deliver coffee and mail to the office without breaking any decoration $\mid ((F(☕ \wedge X(F(✉ \wedge X(F(🚹)))))) \mid\mid (F(✉ \wedge X(F(\wedge X(F(🚹)))))))) \wedge (G \neg ✻)$





Experiments: Moving Targets

Task	Description | LTL
2	Pick up blue then purple objects, then objects that are neither blue nor purple. Repeat this forever. $\mid F(■ \wedge X(F(■ \wedge X(F((○ \vee ☒) \wedge \neg(■ \vee ■))))))$