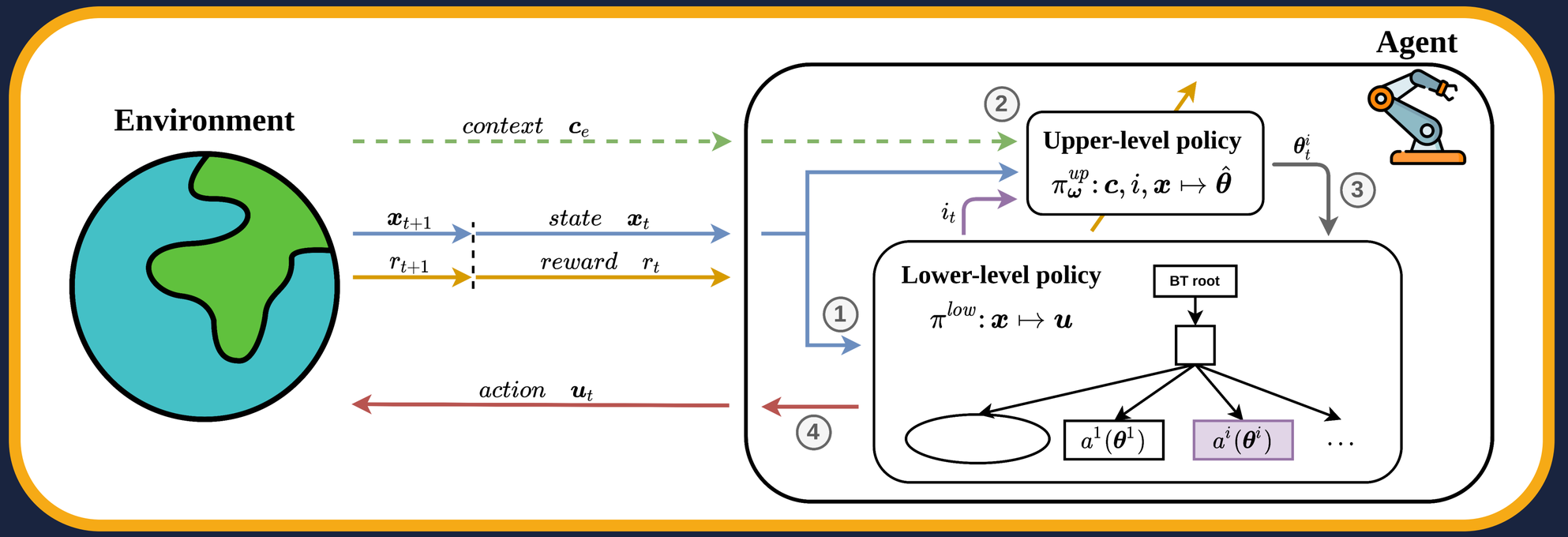
On the Fly Adaptation of Behavior Tree-Based Policies

Marco Iannotta

Örebro Universitet · Suzuki Garphyttan · WASP Wallenberg AI, Autonomous Systems and Software Program

Context
~~**Attention**~~ is all you need!



1. Introduction

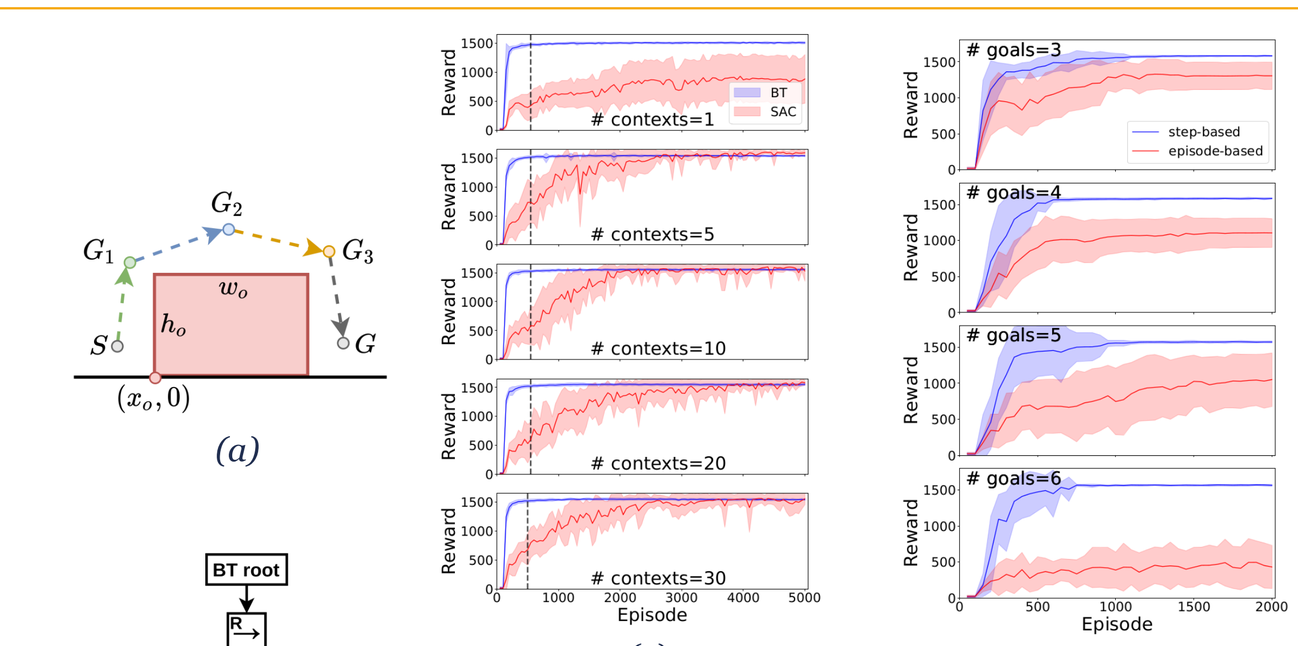
- Behavior Trees (BTs) are usually designed or learned with a single task in mind.
- **Adapting** BTs to novel *task variations* often require extensive re-programming efforts.

2. Method

- We train an **upper-level policy** to adapt the BT Action Nodes to different task variations.
- We condition the upper-level policy on a **context** vector that describes the possible task variations.
- We adopt online **Reinforcement Learning** to operate within the BT structure.

3. Experiments

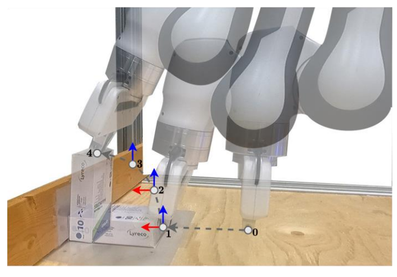
- **Obstacle avoidance** (sim): Figure 1.
  - Convergence training time does not depend on the number of training contexts (c).
  - No curse of dimensionality with step-based parameter selection (d).
  - Trained policies are able to zero-shot generalize to unseen contexts (e).
- **Pivoting** (real): Figure 2.
  - Each policy is trained for 1h directly on the real robot.
  - The policy trained on the broadest range of box sizes successfully completed the task for each dataset entry.



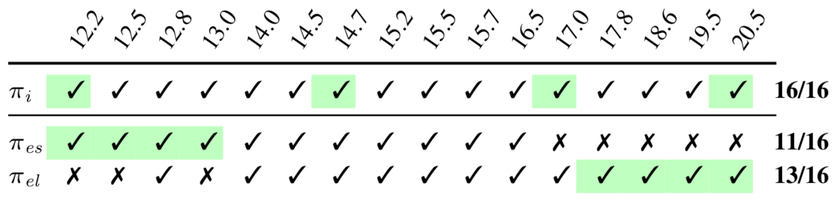
# contexts	Reward		# collisions	
	BT-based	SAC	BT-based	SAC
1	1163 ± 65	653 ± 185	5.4 ± 1.0	7.2 ± 6.3
5	1417 ± 122	1370 ± 251	2.2 ± 1.8	3.6 ± 3.5
10	1443 ± 142	1550 ± 130	1.8 ± 2.2	1.0 ± 1.7
20	1560 ± 21	1590 ± 73	0.0 ± 0.0	0.4 ± 0.8
30	1567 ± 16	1577 ± 98	0.0 ± 0.0	0.0 ± 0.0
Trained BT-based	*1559 ± 13*		*0.0 ± 0.0*	
Trained SAC	*1632 ± 31*		*0.0 ± 0.0*	

(e)

Figure 1. Obstacle avoidance task. (a) Task description; (b) BT design; (c,d) Training curves; (e) Zero-Shot Generalization to unseen contexts.



(a)



(b)

Figure 2. Pivoting task. (a) Task description; (b) Performance (highlighted cells indicate training boxes).