

Reinforcement Symbolic Learning

Chloé Mercier¹ - Frédéric Alexandre¹ - Thierry Viéville^{1,2}

¹ Mnemosyne, Inria, Bordeaux, France

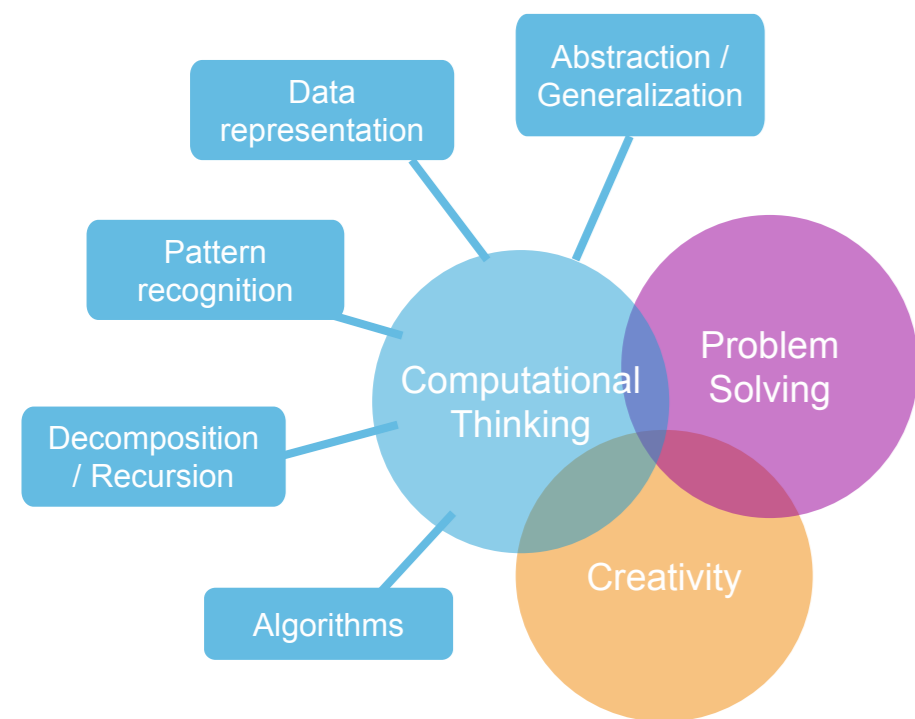
² Laboratoire d'Innovation et Numérique pour l'Éducation (LINE), Université Côte d'Azur, Nice, France

Supported by Inria Exploratory Action AIDE (Artificial Intelligence Devoted to Education)

Contact us: chloe.mercier@inria.fr - frederic.alexandre@inria.fr - thierry.vieville@inria.fr



Context



How to teach and assess "21st century skills"?

⇒ We model a creative problem-solving learning activity involving modular robotic artefacts.

CreaCube: a creative problem-solving task

- The state space is defined by **complex structured symbolic data** + inferences
- The learning paradigm corresponds to a **final rewarded task**
- ⇒ Make a reinforcement algorithm work directly on symbolic data.

Symbolic state space specification

Input structure

An **event sequence** of **semi-ordered hierarchical tuples** so that :

- the agent may potentially be hypermnesic (no Markov property restriction)
- each value has a predefined **"type"**
- literal values are taken among either
 - a finite enumeration of qualitative values (e.g., a color set), or
 - quantitative values (i.e., finite precision bounded values).

Comparing inputs

This **tree data structure** is equipped with a **partial semi-order** compatible with an **extended semi-distance**.

- Two values may be equal, indistinguishable, comparable or incomparable.
- An **edit distance** allows us to define a **metrizable** space

The minimal cost of editing sequences transforming one input into another, by adding, deleting or changing elements

- ✓ It yields a path of transformation of a data tree to another
- ✓ Thus allowing us to **interpolate** two data trees (geodesic)

quadratic complexity only 😊

Reinforcement learning on structured data

The vanilla **Q-learning** algorithm revisited:

$$Q[s, a_t] += \alpha e^{-d(s, s_t)/\rho} (r_{t+1} + \gamma \max_a Q[s_{t+1}, a] - Q[s, a_t])$$

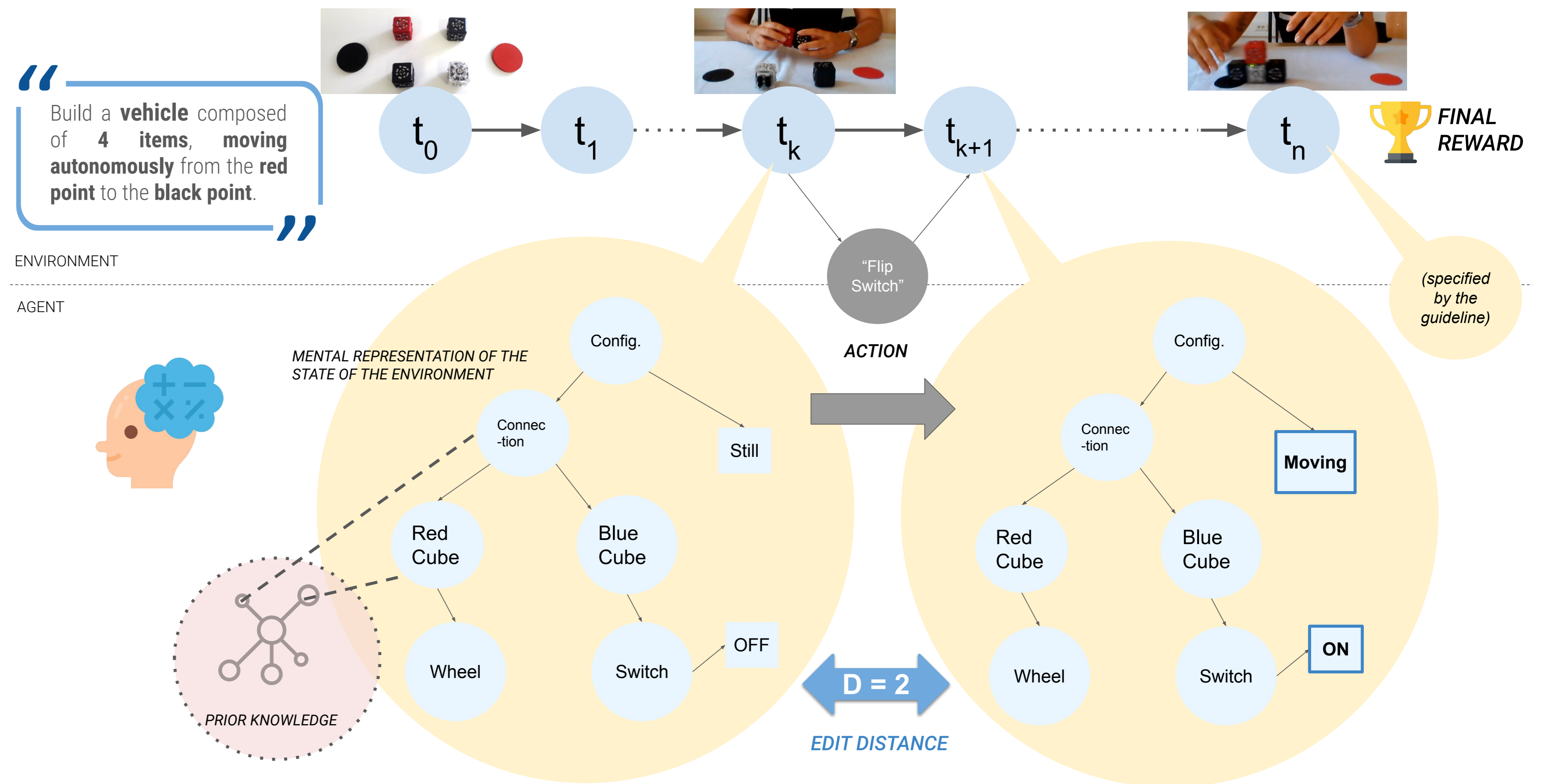
Annotations for the equation:

- exponential weighting radius
- tabulated value on stored state and action
- learning factor
- edit distance
- discount factor
- observed reward
- interpolated value on new state

⇒ The key technical point is that the algorithm requires to **interpolate values**

$$Q(s, a_k) = \sum_{s_t, a_t} e^{-(d(s, s_t) + d(a_k, a_t))/\rho} Q[s_t, a_t] / \sum_{s_t, a_t} e^{-(d(s, s_t) + d(a_k, a_t))/\rho}$$

thanks to the data space **geodesic structure**.



Discussion

An original approach, yet some further work is needed for :

- formalizing the A-box part of the knowledge base as hierarchical data structures
- experimenting the effective pertinence of edit distances
- embedding such mechanisms in a cognitive architecture

References

- Alexandre, F.: A global framework for a systemic view of brain modeling. 2021.
- Domenech, P., Koechlin, E.: Executive control and decision-making in the prefrontal cortex. 2015.
- Garnelo, M., Arulkumaran, K., Shanahan, M.: Towards Deep Symbolic Reinforcement Learning. 2016.
- Ma, Z., Zhuang, Y., Weng, P., Li, D., Shao, K., Liu, W., Zhuo, H.H., Hao, J.: Interpretable Reinforcement Learning With Neural Symbolic Logic. 2021.
- Mercier, C., Roux, L., Romero, M., Alexandre, F., Viéville, T.: Formalizing Problem-Solving in Computational Thinking : an Ontology approach. 2021.
- Quangraoua, A., Ferraro, P.: A Constrained Edit Distance Algorithm Between Semi-ordered Trees. 2009.
- Singh, S., Lewis, R.L., Barto, A.G., Sorg, J.: Intrinsically Motivated Reinforcement Learning: An Evolutionary Perspective. 2010.