Open-ended learning: DREAM project approach

S. Doncieux
Open-ended learning & development in robotics

What actions, what states to deal with an unknown environment?
Open-ended learning: self-built MDPs

Bootstrapping without action and state spaces: How to build an MDP?

Deferred Restructuring of Experience in Autonomous Machines
H2020 FET Proactive « Knowing, doing, being » 01/2015-12/2018

Collective scale

No initial policy
No single task
Motivations:
- curiosity
- satisfying humans
- global mission

Knowledge sharing between robots:
- better generalization
- faster learning

Daytime
Sequence of learning episodes driven by motivations
Behavior exploration
Knowledge improvement
Knowledge adaptation
Knowledge validation

Small batch
Skill

Consolidated knowledge
- task-relevant features
- task contexts
- abstract knowledge
- new motivations

Individual scale

New situation:
- no reprogramming
- fast adaptation

Daytime experience (large batch)

Knowledge restructuring
Transfer from STM to LTM

Nighttime
Dream

Collective scale

Knowledge sharing between robots:
- better generalization
- faster learning

Consolidated knowledge
- task-relevant features
- task contexts
- abstract knowledge
- new motivations

Individual scale

New situation:
- no reprogramming
- fast adaptation

Daytime experience (large batch)
DREAM cognitive architecture

Goal: build the knowledge required by the decision processes (MDP)

How to decompose it?

Decision
- Decision process 1
- Decision process 2
- Decision process 3

Cognitive strategy
- Motivational Engine
- Episodic Memory
- Short Term Memory
- Associative Long Term Memory

ACTUATORS
Building a task-specific MDP

Goal of open-ended learning: building the state and action spaces to make a robot solve the task it is facing with a given decision/learning process.

Approach:
1. Development: knowledge acquisition process (may be long)
   1. Building state spaces
      1. Object centered
      2. Generic
   2. Building action spaces

2. Decision: knowledge exploitation process (needs to be fast)
Building a task-specific MDP

Goal of open-ended learning: building the state and action spaces to make a robot solve the task it is facing with a given decision/learning process.

Approach:
1. Development: knowledge acquisition process (may be long)
   1. Building state spaces
      • Object centered: discovering objects
      • Generic: see David Filliat’s talk!
2. Building action spaces: learning a skill to move in the state space

2. Decision: knowledge exploitation process (needs to be fast)
Building state spaces: discovering objects

- State space = object pose and orientation
- What are the objects the robot can interact with?

Challenges:
- collecting experimental data without task or environment specific knowledge
- discovering objects on which the robot can generate a particular effect

Proposed approach:
1. structuring the environment through *affordances* (Montesano et al. 2008)
2. learning the structure through *interactive perception* (Fitzpatrick & Metta 2003)
Building state spaces: discovering objects

- RGBD images
- IK control + push primitive

Interaction with environment

- Observation of movement
- Online classification

Update of the saliency map

Autonomous Exploration of unknown environments

Hypothesis

- Interesting objects are moveable objects.
- Inverse Kinematic Model
- Push Primitive
- Over-segmentation of the scene
Building state spaces: discovering objects

Segmenting objects through an autonomous agnostic exploration conducted by a robot

Léni K. Le Goff, Pierre-Henri Le Fur, Ghanim Mukhtar and Stéphane Doncieux

Building state spaces: discovering objects

Learning skills to move in the state space

\[ \xi : S \times S \rightarrow A \]
\[ a = \xi(s_i, s_g) \] action to reach \( s_g \) from \( s_i \)

Proposed approach: decomposition in 2 steps

\[ \xi_1 : S \rightarrow SA \quad \xi_2 : SA \times S \rightarrow A \]
\[ \xi_1(s) = \{(a_1, s_1), (a_2, s_2), \ldots, (a_n, s_n)\} \quad \xi_2(SA_s, s_d) = a_d' \]

• 1. Behavior repertoire acquisition
  • Goal: reach all attainable states
  • Slow step, done in simulation

• 2. Repertoire exploitation
  • Goals:
    • cross the reality gap
    • adapt to \( s_d \) (sampling effect)
  • Fast step, done in reality
Behavior repertoire generation with Evolutionary Algorithms

Looking for the optimal solution
Looking for diverse solutions


Learning skills to move in the state space
1. Behavior repertoire acquisition

Repertoire exploitation

 Acquisition and adaptation of a robot behavior repertoire for ball throwinexperiment

 Seungsu Kim and Stéphane Doncieux

 Institute of Intelligent Systems and Robotics (ISIR)
 Sorbonne University

Another example of behavior repertoire
Next developmental steps...

Cognitive strategy
- Effect reproduction and generalization
  - Learning object features affording an effect
  - Using human feedback
- Learning state representations
- Transferring models
- Long term memory management
- Value function acquisition
- Social learning
- Factoring models

Decision
- Behavior Selection
- Decision process 3

SENSeRS

ACTUAToRS
Questions ?

stephane.doncieux@upmc.fr
https://twitter.com/SDoncieux
http://people.isir.upmc.fr/doncieux
http://www.robotsthatarompt.eu/
https://twitter.com/robotsthatarompt