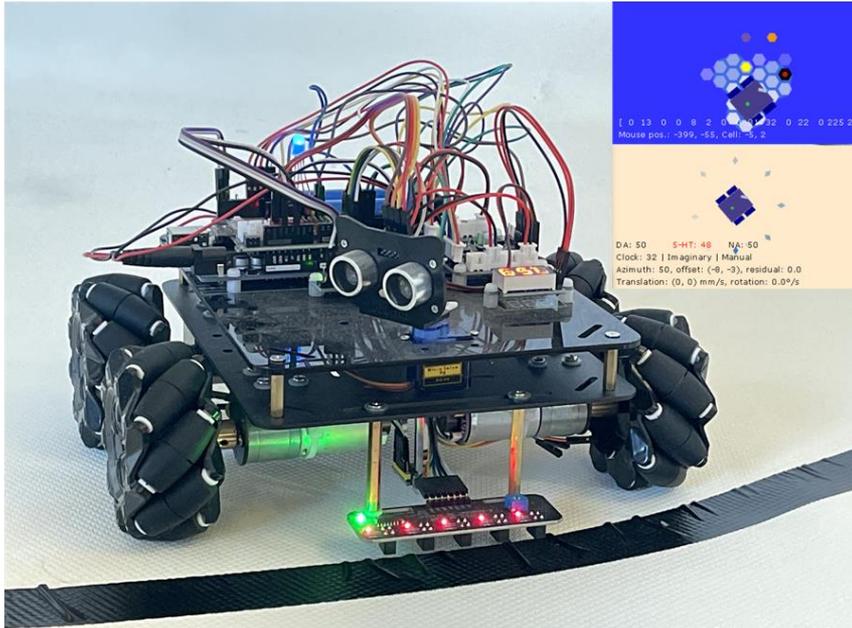


# Intelligence Artificielle Développementale



1 Octobre 2025  
olivier.georgeon@gmail.com  
<http://www.oliviergeorgeon.com>

# Déroulement du cours

UE « IA et Cognition » (30h): Marie Lefevre

Module « IA Développementale » (9h) Olivier Georgeon

- Mercredi 24 septembre: 3h
- **Mercredi 1 octobre: 3h**
- Mercredi 8 octobre: 3h

Contrôle des connaissances:

- TD par groupe de 2: 33% de le note de contrôle continu
- Rendu par mail le **vendredi 24 octobre**
- Examen final: 6 points sur 20

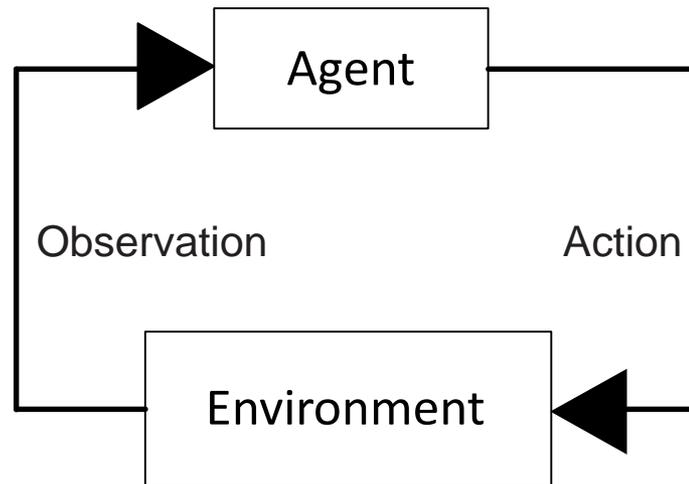
# Objectifs pédagogiques

## Après ce cours, vous serez capables de:

- Cours
  - Expliquer ce qu'est l'IA développementale
  - Différencier IA en domaine modélisé / non modélisé
  - Nommer quelques auteurs de référence dans ce domaine
- TD
  - Implémenter un agent minimaliste dans lequel on ne code pas a priori une ontologie du "monde".

# Modélisation

# Discrete Decision System (DDS)



AKA Markov Decision Process (MDP)

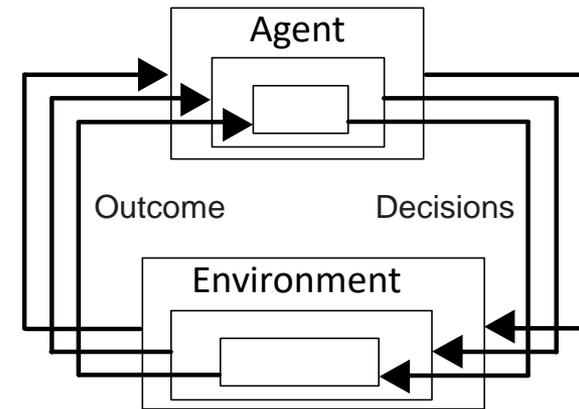
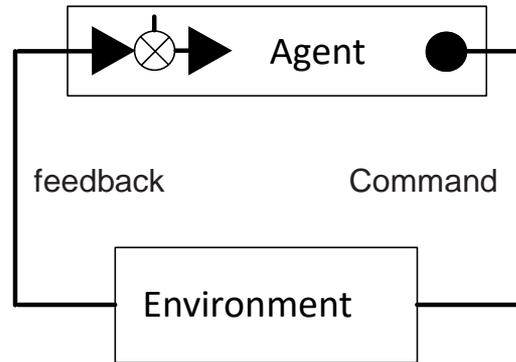
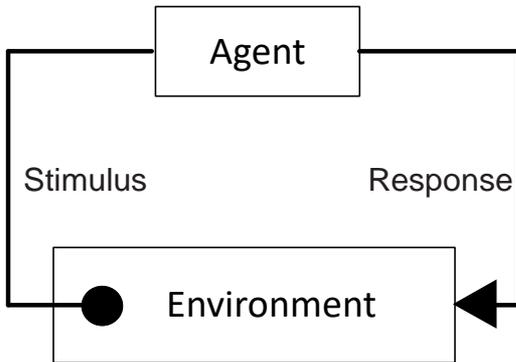
Goertzel (2021) The General Theory of General Intelligence

# Criticisms of the DDS model

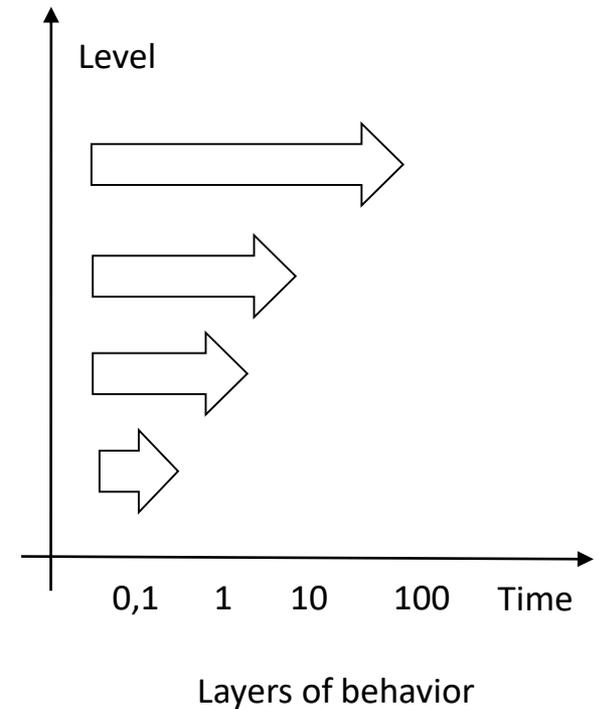
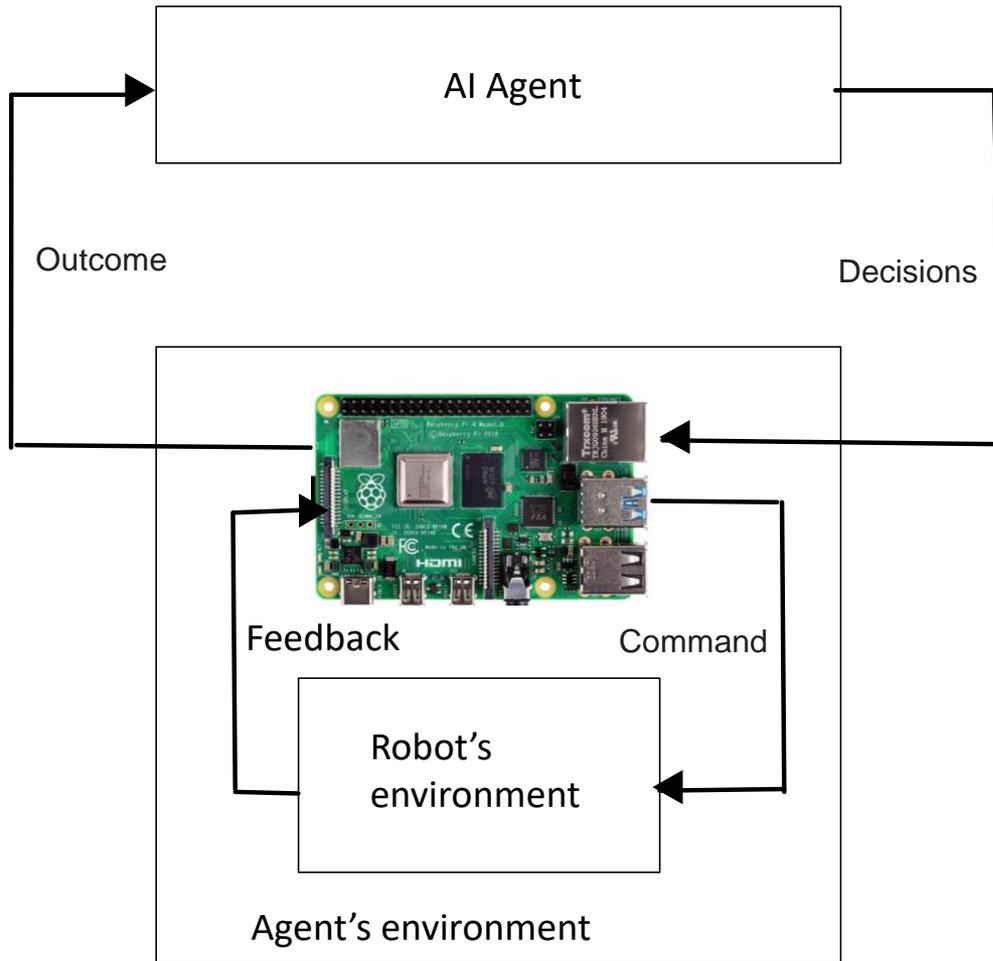
Weinbaum and Veitas (2015) Open-Ended intelligence: The individuation of intelligent agents

- Pre-given separation between agent and environment
  - Does not allow the study of individuation
- Dissociation of perception and action
  - Perception and action cannot always be separated

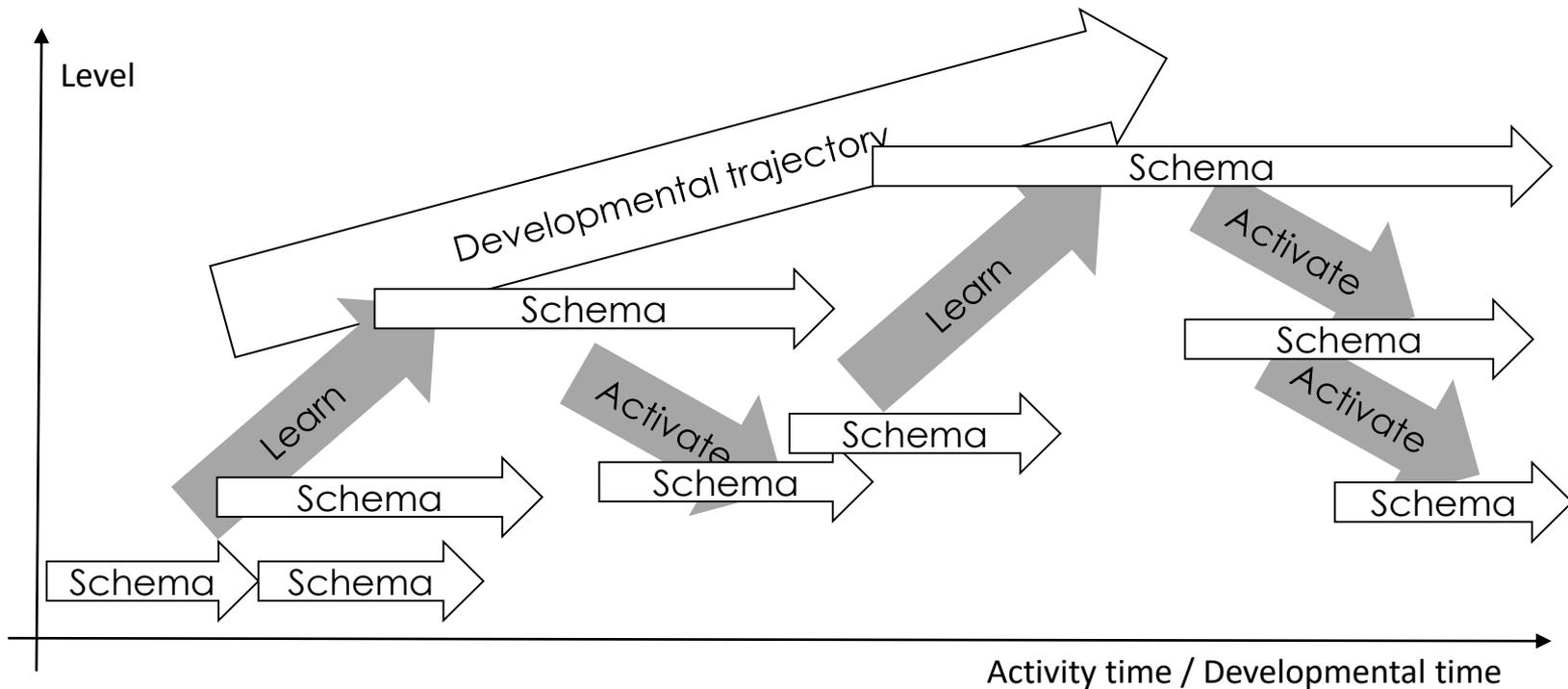
# Declinations of DDSs



# Nested loops in a DDS



# Developmental process



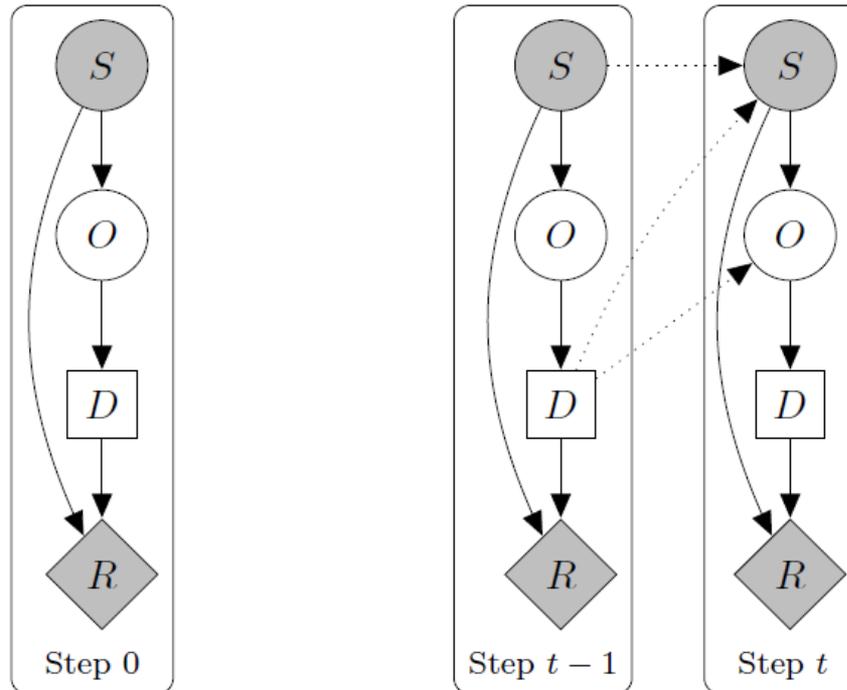
Open-ended process driven by self-motivation

# Inférence

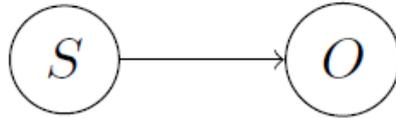
# Exemple d'inférence

- Poursuivre la série 2, 4, 6, 8
- Modèle inféré:
  - $o_t = 2t \rightarrow o_5 = 10$
  - $o_t = 2t^4 - 20t^3 + 70t^2 - 98t + 48 \rightarrow o_5 = 58$
- “Whatever experiences an organism or mechanism has had, there are infinitely many mutually contradictory generalizations that are consistent with those experiences. Therefore, **any induction apparatus must impose a choice** among those consistent generalizations” (The made up mind: a constructivist approach to artificial intelligence, Drescher, 1991)
- Les tests d'intelligence attendent que le participant infère le modèle “le plus simple” (principe du rasoir d'Ockham)

# Inference dans un Système fermé



# Inférence Bayésienne

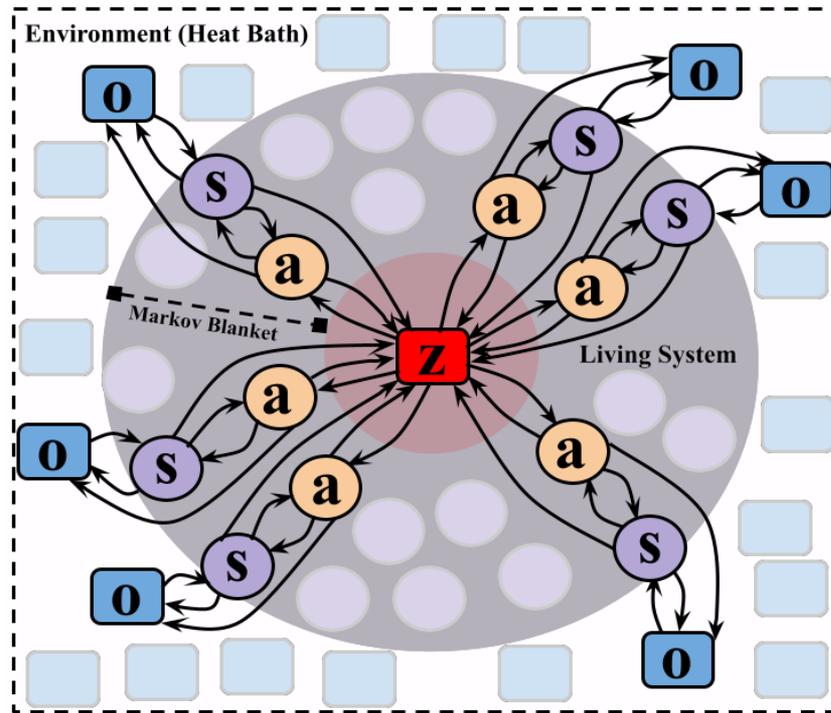


Réseau Bayésien représentant que  
« la variable S influence la variable O » ou « O dépend de S »

Théorème de Bayes:

$$P(O)P(S|O) = P(S)P(O|S)$$

# La couverture de Markov



Ororbian and Friston, Mortal computation: A foundation for bio-mimetic intelligence (2024)

Judea Pearl, The book of Why: The new science of Cause en Effects (2018)

# Travaux dirigés

# Consignes pour les TP

- Par groupe de 2.
- Rendre un seul rapport à la fin .PDF
- Indiquer bien le nom des deux membres du groupe
- Envoyer par mail à [olivier.georgeon@gmail.com](mailto:olivier.georgeon@gmail.com) pour le **vendredi 24 octobre 2024 23h59**
- Pour chaque agent 1 à 3
  - Décrire les principes de l'algorithme que vous avez implémenté
  - Inclure des captures d'écran des traces affichées à la console dans différents environnements
  - Expliquer les comportements obtenus en vous appuyant sur les traces.
- Conclure sur ce que vous retirez de cette expérience et suggestions de comment aller plus loin

# Agent 2

Deux actions possibles  $A = \{0, 1\}$

Deux outcome possibles  $O = \{0, 1\}$

Quatre interactions possibles  $I = A \times O = \{00, 01, 10, 11\}$

Environnement

- $env_1: 0 \rightarrow 0, 1 \rightarrow 1$  ( $01$  et  $10$  ne se produisent jamais)
- $env_2: 0 \rightarrow 1, 1 \rightarrow 0$  ( $00$  et  $11$  ne se produisent jamais)

Implémenter un agent qui sélectionne les interactions qui ont une valence positive (s'il en existe) sans connaître a priori son environnement ( $env_1$  ou  $env_2$ ).

Il change d'action quand il commence à s'ennuyer

Produire un rapport d'analyse de comportement basés sur les traces.

# Setup

Suivre la procédure écrite dans README.md:

<https://github.com/OlivierGeorgeon/Developmental-AI-Lab>

Vous avez deux méthode possibles :

1. Cloner (ou downloader) le repository
2. Ouvrir les notebook avec Google.colab ou un autre outil online

Méthode 1: Exécuter world.py et vérifiez que vous obtenez la trace d'interaction montrées dans les consignes