

Agents intelligents

IA - Chapitre 2

Master STIC - Université d'Orléans - Janvier 2012

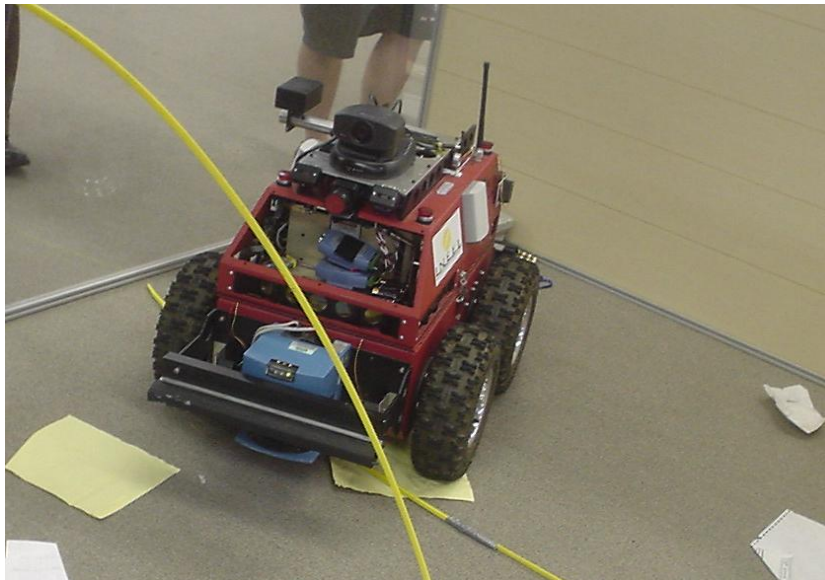
- Agents intelligents
- Mesure de performance, environnement, actions, capteurs : *PEAS (Performance measure, Environment, Actions, Sensors)*
- Types d'environnements
- Fonctions et programmes d'un agent
- Types d'agents

Navigation icons: back, forward, search, etc.

IA - Chapitre 2

Agents intelligents

Exemple



Navigation icons: back, forward, search, etc.

IA - Chapitre 2

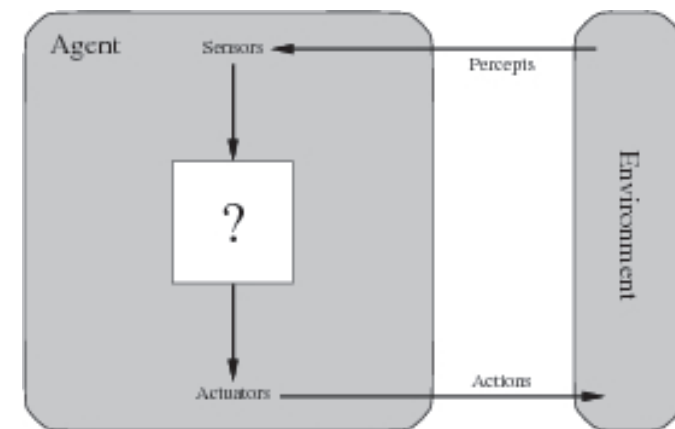
Agents intelligents

Navigation icons: back, forward, search, etc.

IA - Chapitre 2

Agents intelligents

Qu'est-ce qu'un agent ?



Navigation icons: back, forward, search, etc.

IA - Chapitre 2

Agents intelligents

- C'est une entité capable de percevoir son environnement par des capteurs (senseurs) et d'agir sur son environnement à l'aide d'effecteurs (actionneurs)
- Agent : humains, robots, thermostats, logiciel, etc.
- Pour un humain :
 - capteurs : oreilles, yeux, nez, peau, ...
 - actionneurs : mains, bras, jambes, bouches, ...
- Pour un robot
 - capteurs : caméra vidéo, laser, capteur infra-rouge, tactile, ...
 - actionneurs : bras robotisé, roues, pinces, ...
- Pour un agent logiciel (softbot) :
 - capteurs : données de clavier, contenu de fichier, paquets réseau, ...
 - actionneurs : affichage écran, fichiers, paquets réseau, ...

- Un agent est une *entité logicielle persistante* dédiée à une *tâche spécifique* [Smith, Cypher et Spohrer].
- Un agent effectue continuellement trois fonctions :
 - perception des conditions dynamiques de l'environnement,
 - interprétation de ces perceptions par raisonnement, résolution de problèmes,
 - détermination des actions possibles [Hayes-Roth, 1995].
- Un agent *intelligent* est une entité logicielle qui réalise des opérations pour le compte d'un usager (ou d'un autre programme) avec un certain *degré de liberté* et *d'autonomie* et qui, pour ce faire, exploite des *connaissances* ou des *représentations* des désirs et des objectifs de l'usager [IBM White paper, 1995].
- Un agent *autonome* est un *système calculatoire* qui, placé dans un *environnement complexe et dynamique*, *perçoit* et *agit* de manière *autonome* dans cet environnement et, ce faisant, réalise les *objectifs* ou des *tâches* pour lesquels il est conçu [P. Maes, 1995].

Spécification d'un agent

- Le choix de l'action d'un agent à l'instant t dépend des séquences perceptives (cas général)
 - Spécifier l'action qu'un agent devrait produire en réponse à une séquence perceptive donnée constitue le cadre abstrait d'un agent
- Fonction d'agent* (spécification abstraite) :

$$f : P^* \mapsto A$$

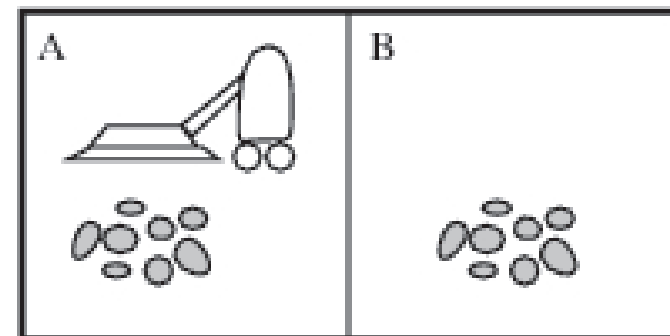
avec

P^* : chronologie des séquences perceptives

A : action

- *Programme d'agent* : implantation concrète de la fonction d'agent sur une architecture physique

Exemple : Aspirateur



- Perceptions : lieu et contenu, e.g. $[A, \text{Dirty}]$
- Action : *Left*, *Right*, *Suck*, *NoOp*

Séquence perceptive	Action
<i>[A, Clean]</i>	<i>Right</i>
<i>[A, Dirty]</i>	<i>Suck</i>
<i>[B, Clean]</i>	<i>Left</i>
<i>[B, Dirty]</i>	<i>Suck</i>
<i>[A, Clean], [A, Clean]</i>	<i>Right</i>
<i>[A, Clean], [A, Dirty]</i>	<i>Suck</i>
...	...

- Quelle est la fonction correcte ?
- Peut-elle être implantée dans un petit programme d'agent ?

```

function Reflex-Vacuum-Agent([location, status]) returns an action
  if status = Dirty then return Suck
  else if location = A then return Right
  else if location = B then return Left

```

“For each possible percept sequence, an ideal rational agent should do whatever action is expected to maximize its performance measure, on the basis of the evidence provided by the percept sequence and whatever built-in knowledge the agent has”

Russell, Norvig (2003), page 36

“Pour chaque séquence perceptive possible, un agent rationnel idéal devrait effectuer l'action capable de maximiser sa mesure de performance, sur la base des informations fournies par la séquence perceptive et compte tenu de la connaissance dont l'agent peut disposer”

- Suppositions :
 - Offrir 1 point pour chaque case propre à chaque moment pendant un “temps de vie” de 1000 pas
 - L'environnement géographique est à priori connu (les cases)
 - Les seules actions possibles sont *Left*, *Right*, *Suck* et *NoOp*
 - L'agent perçoit correctement sa localisation ainsi que la propreté de celle-ci
- Avec ces suppositions, l'agent donné dans cet exemple est rationnel.

- Agent rationnel = basé sur le raisonnement
- Toutes options considérées, faire le meilleur choix pour maximiser les chances de “succès”
- Mesure de performances
 - réussir la tâche ?
 - quantité de ressources consommées ?
 - temps mis pour réaliser l'action ?
- Rationnel \neq omniscient
 Rationnel \neq succès
 Rationnel \Rightarrow exploration, apprentissage, autonome

- Spécification de l'environnement = “problème”
- Agent rationnel = “solution”
- pour chaque agent il faut spécifier son environnement : sa mesure de performance, son environnement, ses actions, ses capteurs
Performance measure, Environment, Actuators, Sensors =
PEAS

PEAS

Exemple : conduite automatique, agent “taxi automatique”

- **P** (mesure de performance) ??
- **E** (environnement) ??
- **A** (actions) ??
- **S** (capteurs) ??

PEAS

Exemple : conduite automatique, agent “taxi automatique”

- **P** (mesure de performance) ??
sécurité, atteindre la destination, respecter les règles de circulation, confort du passager, ...
- **E** (environnement) ??
rues citadines, autoroutes, piétons, conditions météo, clients, ...
- **A** (actions) ??
conduire, accélérer, freiner, klaxonner, discuter, ...
- **S** (capteurs) ??
caméra vidéo, accéléromètres, capteurs moteur, GPS, ...

agent = architecture + programme

- **architecture** :
 - organisation matérielle et logicielle
 - contrôle les capteurs et produits les séquences perceptives
 - exécute les programmes et les actions (contrôle les actionneurs)
- **programme** : une procédure qui met en correspondance les séquences perceptives et les actions

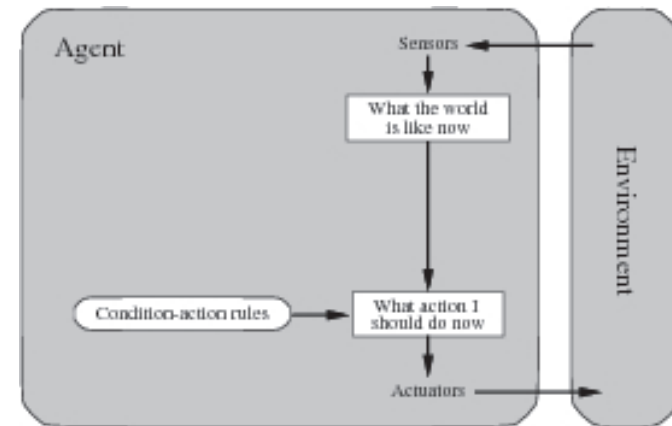
- *Entièrement observable / partiellement observable* : l'agent a, à tout instant, accès à l'*état complet* de l'environnement. Pas besoin de conserver une trace des changements
- *Déterministe / non-déterministe* : un environnement est déterministe si son *prochain état* est totalement *déterminé par son état courant* et par l'action effectuée par l'agent.

- *Episodique / séquentiel* : les *épisodes futurs ne dépendent pas des actions effectuées* lors des épisodes précédents. Pas besoin de planifier à l'avance
- *Statique / dynamique* : un environnement qui *ne change pas* pendant que l'agent "réfléchit" est dit *statique*. Pas besoin de prendre en compte le paramètre "temps".
Si l'environnement ne change pas avec le temps mais que la performance de l'agent change, l'environnement est *semidynamique*.
- *Discret / continu* : un environnement est discret si le *nombre de séquences perceptives et d'actions est fini*.
- Simple agent / multi-agent

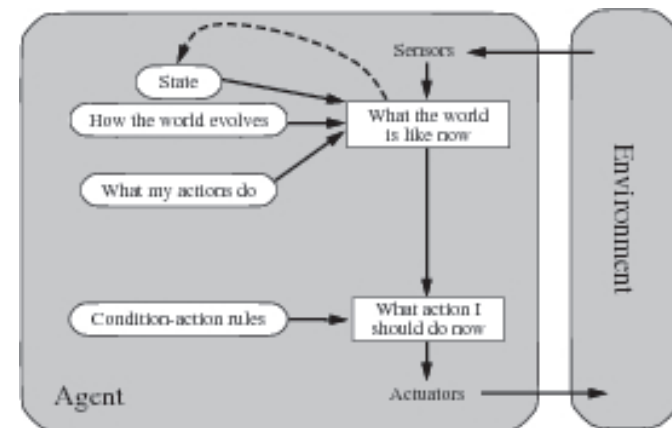
	Mots-croisés	Analyse d'images	Internet shopping	Taxi
Observable	entier	entier	partiel	partiel
Déterministe	oui	oui	partiel	non
Episodique	non	oui	non	non
Statique	oui	semi	semi	non
Discret	oui	non	oui	non
Agent	single	single	multi	multi

- Le type d'environnement détermine la conception d'agent
- Le monde réel est (bien sur) partiellement observable, non-déterministe, séquentiel, dynamique, continu et multi-agents.

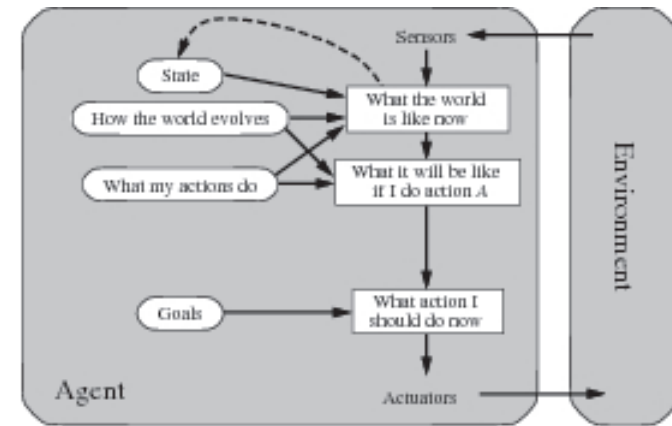
- Quatre types en ordre croissant de généralité
 - **agent simple-reflex** : utilise un système de règles condition-action (système de production), ne possède pas de mémoire des états passés
 - **agent reflex avec états** : possède une mémoire interne conservant les états antérieurs
 - **agent avec objectif** : possède une information explicite décrivant l'objectif à atteindre (situations souhaitables), il est capable de considérer des événements futurs
 - **agent avec fonction d'utilité** : fonde ses décisions rationnelles sur une mesure d'utilité
- Ces agents peuvent être évolués en agents d'apprentissage



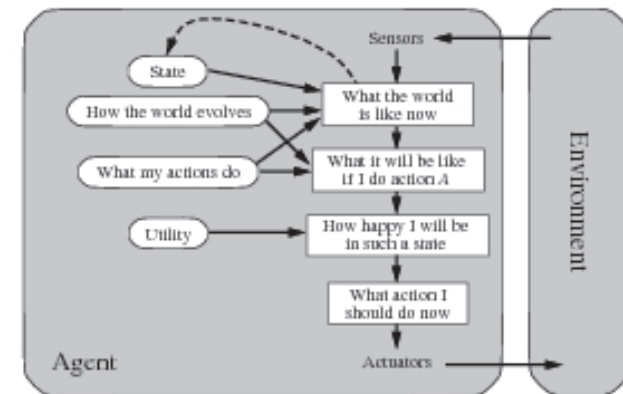
- règles condition-action (chez l'humain à la fois apprises et innées)
 - si le feu est vert alors passer
 - si le feu est rouge alors s'arrêter
- décision prise uniquement sur la base de la séquence perceptive courante
- contre-exemple :
la conduite automatique d'un véhicule n'est pas (toujours) reflex, par exemple il faut se souvenir de ce qu'on a vu dans le rétroviseur, pour cela il faut une variable d'état



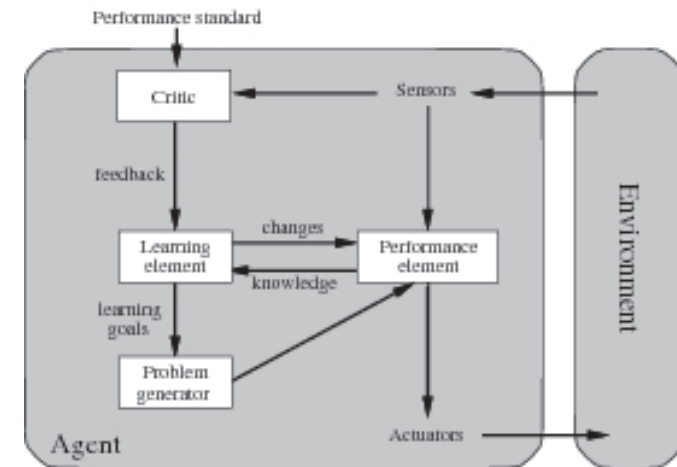
- Etat interne pour garder la trace des séquences perceptives
- Mettre à jour l'état interne : comment l'environnement change et comment les actions affectent l'environnement



- L'agent a un objectif qui décrit les situations désirées
- L'agent combine ces informations avec le résultat des actions possibles pour choisir l'action permettant de s'approcher de l'objectif
- L'agent devrait être capable de considérer des longues séquences pour atteindre l'objectif : **recherche** et **planification**
- Taxi : plusieurs séquences d'actions différentes permettant d'arriver à la destination, mais certaines sont plus rapides, ou plus sécurisée, ou moins chères,...



- Utiliser un modèle de l'environnement et une fonction qui mesure la préférence des états de l'environnement
- Choisir l'action qui mène à la meilleure utilité espérée



Conclusion

- Un agent **perçoit** et **agit** dans un **environnement**. Il a une **architecture** et est réalisé par un **programme**.
- Un agent **rationnel** choisit toujours l'action qui **optimise sa performance attendue** (étant donné la séquence perceptive reçue).
- La spécification d'**environnement** inclut la mesure de performance, l'environnement extérieur, actions, capteurs.
- Un agent **fait correspondre** une **action** à la **séquence perceptive** reçue et met à jour son **état interne**.
- Les agents **réflex répondent immédiatement** aux séquences perceptives.
- Les agents avec **objectif** agissent pour réaliser leur **propre objectif**.
- Les agents avec **utilité** optimisent leur propre **fonction d'utilité**.
- Certains environnements sont plus difficiles que d'autres, les **plus difficiles** sont partiellement accessibles, non déterministes, séquentiels, dynamiques, continus et multi-agents.