

Aide à la Décision

Introduction

© Hervé Barbot, 2005-2011

Intelligence Artificielle

Introduction

Cours « Aide à la Décision »
EFREI – 2010/2011 – L3

Intelligence ?

© Hervé BARBOT, 2005-2010 – www.proactitude.com

(C) Hervé Barbot, 2005-2010

2

Test de Turing

Qu'est-ce que l' « intelligence » ?

▪ Juge – Femme – Machine



- A. Turing
 - Distinction homme / machine
- C. Darwin
 - Survie du plus apte
- T. Edison
 - Fonctionne et produit plus de revenu

Définition selon le domaine d'activité ou de préoccupation de l'auteur

- Où est la femme ? Où est la machine ?
- L'ordinateur peut-il tromper le juge ???

(C) Hervé Barbot, 2005-2010

4 (C) Hervé Barbot, 2005-2010

7

IA ?

- Sur le net :
 - Simulation des processus intelligents
 - Reproduction des méthodes / résultats du raisonnement
 - Etude des facultés mentales
 - Méthodes calculatoires et informatiques
 - Imitation des humains

Qu'est-ce que l' IA ?

- « ...the science of making machines do things that would **require intelligence if done by humans** »
Marvin Minsky
- « AI is the part of computer science concerned with **designing intelligent computer systems** »
E. Feigenbaum
- « [The automation of] activities that we associate with **human thinking**, activities such as **decision-making**, problem solving, learning ... »
Bellman, 1978

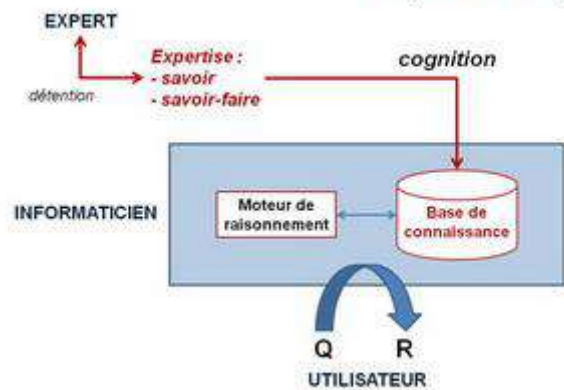
Penser puis Décider/Agir

	Humain	Machine
Penser	x	x
Agir	x	x
	« comme un humain »	« rationnellement »

Système Expert

- Système = « penser + agir »
- Connaissance :
 - Savoir
 - Capacité de raisonnement
- Rôles
 - Expert « métier »
 - Informaticien
 - Utilisateur
- Cognition
 - Analyse des processus mentaux

Système Expert



Données / Information / Connaissance

Base de connaissances

- **Données =**
 - Transport d'information
 - Signal
- **Information**
 - Interprétation d'une donnée
- **Connaissance**
 - Utilisation de l'information
 - Dans un but précis

(C) Hervé Barbot, 2005-2010

22 (C) Hervé Barbot, 2005-2010

23

Données / Information / Connaissance

Données = Transport d'information Signal	Ex.: ...-... C
Information Interprétation d'une donnée	Ex.: (. point d'exclamation) (...-... , SOS) (C . lettre) ou (C . note)
Connaissance Utilisation de l'information Dans un but précis	Ex.: exclamation en fin de phrase ...-... déclenche l'alerte et envoie les secours quand C apparaît dans une partition avec référence la gamme Do, alors jouer dans la gamme associée

La connaissance utilise l'information dans le cadre d'actions, dans un but précis. Les actions peuvent être la prise de décisions, la création de nouvelles informations, etc.

(C) Hervé Barbot, 2005-2010

27 (C) Hervé Barbot, 2005-2010

28

« Système à base de connaissance » ?

- **SBC**
 - Base de connaissance sur un sujet donné
 - Connaissance représentée **de façon explicite**
- **Système expert classique = SBC**

Transfert des connaissances vers la machine

- **Changement de**
 - Média
 - Forme
- **Modélisation linguistique**
 - Puissance expressive
 - Permet le raisonnement
 - Efficace



(C) Hervé Barbot, 2005-2010

29 (C) Hervé Barbot, 2005-2010

31

Exemples de représentation

- **Triplets < objet , attribut , valeur >**
 - < sujet , verbe , complément >
- **Réseaux sémantiques**
 - Nœuds = concepts / Arcs = relations
 - Représentation souple
- **Logique – Calcul des propositions / prédicats**
 - Représenter d'une situation à l'aide de prédicats
 - Définir de nouveaux prédicats
 - Donner des équivalences
 - Donner un but à atteindre
- **Règles**
 - Liens entre information connue / information à déduire

Exemples de représentation

▪ Triplets < objet , attribut , valeur >

▪ Syntaxe

- **Objet** (sujet) = la ressource (URI ou nœud local)
- **Attribut** (prédicat) = la propriété
- **Valeur** (objet)

▪ Exemple.

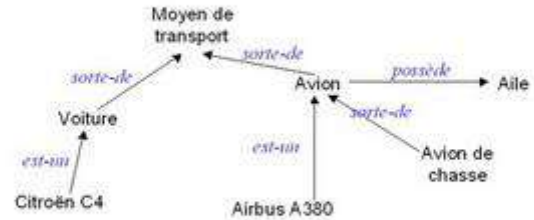
- la ressource <http://www.w3c.org/> a un propriété **titre** dont la valeur est **World Wide Web consortium**.



(C) Hervé Barbot, 2005-2010

Exemples de représentation

▪ Réseaux sémantiques



33 (C) Hervé Barbot, 2005-2010

38

Exemples de représentation

▪ Logique – Calcul des propositions / prédicats

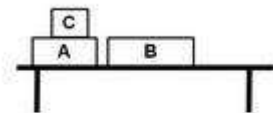
- Représenter d'une situation à l'aide de prédicats
- Définir de nouveaux prédicats
- Donner des équivalences
- Donner un but à atteindre

Exemples de représentation

▪ Logique

Savoir = situation actuelle

sur(C,A)
surtable(A) surtable(B)
libre(C) libre(b)



Savoir-faire = règles de (prise de) décision : quand / où

« prendre » et « poser »

$\text{libre}(x) \wedge \text{libre}(y) \wedge \text{poser}(x,y) \Rightarrow \text{sur}(x,y)$

$\text{libre}(x) \Leftrightarrow \neg (\exists y \mid \text{sur}(y,x))$

$\text{sur}(y,x) \wedge \text{prendre}(y,x) \Rightarrow \text{libre}(x) \wedge \neg \text{sur}(y,x)$

Objectif : « sur(A,B) »

(C) Hervé Barbot, 2005-2010

40 (C) Hervé Barbot, 2005-2010

44

Exemples de représentation

▪ Règles

- Liens entre :
Information connue
Information à déduire (inférer)
- Peut exécuter des procédures



Exemples de représentation

• Types de règles :

Relation

Si la batterie est morte ALORS l'auto ne démarre pas

Recommandation

Si l'auto ne démarre pas ALORS prendre un taxi

Directive

Si l'auto ne démarre pas ET
le système d'alimentation en essence est ok
ALORS vérifier le système électrique

Stratégie

Si l'auto ne démarre pas ALORS
vérifier le système d'alim essence PUIS le système électrique

Heuristique

Si l'auto ne démarre pas ET
l'auto est une Ford de 1962 ALORS vérifier le radiateur

(C) Hervé Barbot, 2005-2010

45 (C) Hervé Barbot, 2005-2010

51

Exemples de représentation

▪ Règles avec variables

Réaliser la même opération sur un ensemble d'objets.
Exemple.

Si ?x est employé & ?x âge > 65
alors ?x peut prendre sa retraite

▪ Règles incertaines

Traduisent des associations incertaines entre prémisses et conclusions.
Exemple.

Si inflation élevée
alors taux d'intérêt élevé CF=0.8

▪ Méta-règles

- Traduisent une connaissance sur l'utilisation et le contrôle de la connaissance du domaine.
- Disent comment utiliser les autres règles.
- Exemple.

Si auto ne démarre pas & système électrique normal
alors exploiter les règles concernant le système d'alim. en ess.

(C) Hervé Barbot, 2005-2010

55 (C) Hervé Barbot, 2005-2010

56

Raisonnement

Raisonnement / Inférence

▪ La plupart des systèmes experts existants reposent sur :

- la logique formelle
- le raisonnement déductif

▪ Syllogisme

- Raisonnement logique à deux propositions (également appelées prémisses) conduisant à une conclusion

Si (« P est vrai » un fait) ET (« P → Q » une règle)

ALORS « Q est vrai » nouveau fait : une conclusion

(C) Hervé Barbot, 2005-2010

57 (C) Hervé Barbot, 2005-2010

58

Raisonnement = processus

Savoir-faire

+

Savoir

+

Stratégies de résolution

→

Conclusions

Comprendre comment un expert humain raisonne lors de la résolution d'un problème.

Raisonnement à base de règles

▪ Composants :

- Ensemble de règles (savoir-faire)
- Ensemble de faits (savoir)
- Moteur d'inférence



Avantages

- Facile à comprendre
- Naturel
- Modulaire

(C) Hervé Barbot, 2005-2010

60 (C) Hervé Barbot, 2005-2010

61

Raisonnement à base de règles

▪ Base de règles

- Savoir-faire de l'expert
- Prémisses + Conclusions

▪ Base de faits

- Savoir
- Mémoire de travail du SBC
- Variables en cours d'exécution
- Base de faits initiaux
- Faits déduits ou demandés à l'utilisateur

Raisonnement à base de règles

- Principe de déduction
 - Trouver parmi les règles celles candidates
 - règles « déclenchables »
 - Choisir une de ces règles
 - stratégie / résolution de conflit
 - Exécuter la règle
- Critère d'arrêt (fin du processus d'inférence)
 - Aucune règle déclenchable
 - Solution acceptable trouvée
 - Impossibilité de trouver une solution
 - ...

(C) Hervé Barbot, 2005-2010

Raisonnement à base de règles

- Moteur d'inférence
 - Règle + faits = « ajouter » nouveaux faits
 - Différentes méthodes / catégories
 - Chaînage avant
 - Chaînage arrière
 - Méta-règles

64 (C) Hervé Barbot, 2005-2010

65

« Agent »

« Agent » : definition?

- ➔ No unique and standard definition
- ➔ Born from Artificial Intelligence and Networks
 - Distributed systems
 - Cooperating systems
- ➔ Simplest definition of an agent :
 - « a small software component specialized in one specific task »
 - ... « a component of a distributed application / system » !

(C) Hervé Barbot, 2005-2010

68 (C) Hervé Barbot, 2005-2010

69

Goals

- ➔ Solve multidisciplinary problems
 - ➔ Domains, constraints, ...
 - Need to have « specialized » systems for problem resolution : specific technics (algorithms), specific knowledge (what, how)*
- ➔ Taking advantages from problem specificity
 - Want to use the « best » system*
- ➔ Cooperation
 - Need inter-operation mechanisms between heterogeneous systems*
- ➔ Heterogeneous data
 - Required for each domain, shared between cooperating parts*

...

(C) Hervé Barbot, 2005-2010

Goals

- ...
- React to undefined situations
 - Ability to adapt, to learn*
- Use mobility to choose execution host
 - for example according to available ressources
 - but how to maintain accessibility from others?*
- Intelligent agent = software & hardware

70 (C) Hervé Barbot, 2005-2010

71

Some definitions

- « Un agent est une **entité autonome**, réelle ou abstraite,
 - qui est capable d'**agir sur elle-même et sur son environnement**,
 - qui, dans un univers multi-agent, peut **communiquer** avec d'autres agents,
 - et dont le **comportement est la conséquence de ses observations, de ses connaissances et des interactions** avec les autres agents »

Ferber

(C) Hervé Barbot, 2005-2010

Some definitions

- « An agent is a computer science system having
 - a specific environment,
 - a set of goals and
 - a set of abilities,
 - basically autonomy,
 - flexibility and
 - mobility »

Sycara & Wooldridge

72 (C) Hervé Barbot, 2005-2010

73

Some definitions

- An ideal, best and complete agent

includes

- Autonomy
- Mobility
- Intelligence

(C) Hervé Barbot, 2005-2010

What are agents ?

- Autonomous problem-solving entities
 - complex, dynamic environments (physical or software)
 - no permanent guidance from the user
- Intelligent Agents
 - Perceive and interpret 'sensor'-data
 - Reflect events in their environment
 - Take actions to achieve given goals

Source: FIPA (Foundation of Intelligent Physical Agents)

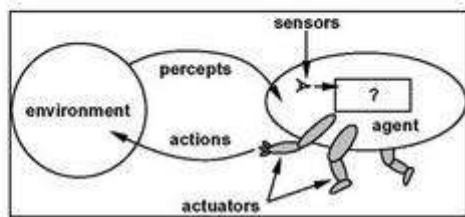
74 (C) Hervé Barbot, 2005-2010

75

Interactions

- Interactions with the environment

- Perception
 - Captors / Sensors
- Action
 - Effectors



(C) Hervé Barbot, 2005-2010

77 (C) Hervé Barbot, 2005-2010

Interaction means (examples)

	<i>Perception</i>	<i>Action</i>
Human	Eye - Skin - Ear - Nose	Hand - Leg - Mouth
Robot	Video camera - IrdA - Control board (human)	Wheels - Arm - Pliers
Software	Data - character stream	Data - Character stream

(C) Hervé Barbot, 2005-2010

78

Types of interaction

Agent = « Rational » entity

- Agent to Software
- Agent to Humans
- Agent to agent

- Overall goal / property / constraint : **maximize its performance**
 - External vision
 - Established as part of the objectives of the agent
 - Dependant of its specific tasks
- « Rational » means :
 - Able to explore different situations and alternatives
 - Able to learn
 - Be autonomous
- « Rational » does not mean :
 - All-knowing (omniscient)
 - Perceptive (clairvoyant)

(C) Hervé Barbot, 2005-2010

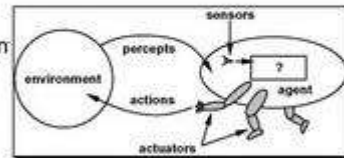
79 (C) Hervé Barbot, 2005-2010

82

Architecture

Architecture of an agent

- Agent =
 - Hardware (computer system)
 - Software
 - Sensor
 - Controlled by the agent
 - Effector
 - Used by the agent



- React to the environment :
 - series of percepts lead to actions
 - Percept 1, Percept 2, ... --> Action
 - $f: P^* \rightarrow A$

« P* » is the chronology of what has been observed

Some agents may take decision bases on the latest event, without memory of the previous ones.

« A » is the action that the agent performs

- Software : the « intelligence »
 - Learn – analyse – decide - react

(C) Hervé Barbot, 2005-2010

83 (C) Hervé Barbot, 2005-2010

84

Architecture of an agent

Types of agents

- Example :

```
function thermostat
  { temp }
if temp < desired -
  epsilon
  action= open tap
if temp > desired +
  epsilon
  action= close tap
else
  action = nothing
```

- General structure :

```
function agent
  { percept }
local state is L
L= update_local_state
  ( L , percept )
action= choose_best
  ( L )
L= update_local_state
  ( L , action )
```

- Table-driven

Has a table of possible actions :
Sequences of percepts x Actions

- Reflex

Has a set of rules :
Condition x Action

No memory of what happened or has been decided

- Reflex with states

Knowledge of the past states

- Based on goals

Knows what is to be reached... *The « ideal » situation*
Can envision future actions / events and states

- With usefulness function

Has the knowledge of how usefulness would be such action / such state

(C) Hervé Barbot, 2005-2010

85 (C) Hervé Barbot, 2005-2010

86

Table-driven agent

```
function td_agent ( percept )
  P* is a sequence of percepts,
  initially empty
  T is a table indexed by percept sequences,
  initially fully specified

  append percept to the end of P*
  action = lookup ( P* , T )
```

- Problems ?
 - Size of the table
Chess game = 35^{100} entries
 - How / how long to build the table
 - Autonomy : none if the table can not change

(C) Hervé Barbot, 2005-2010

Reflex agent

```
function reflex ( percept )
  R is a set of condition-action rules,
  initially fully specified

  state = interpret_input ( percept )
  rule = rule_match ( state , R )
  action = rule_action ( rule )
```

- No memory : only the latest « percept »
- Human examples :
 - Good : green light = go ; red light = stop
 - Bad : car drivers can cross a road after having a look on BOTH sides (...memory needed, even for a short time)

88 (C) Hervé Barbot, 2005-2010

90

Reflex agent with state

```
function reflex_with_state ( percept )
  state is the description of the
  current world state
  R is a set of condition-action rules

  state = update_state ( state , percept )
  rule = rule_match ( state , R )
  action = rule_action ( rule )
  state = update_state ( state , action )
```

- « Update-state » : how to implement ?
- How many states ?

(C) Hervé Barbot, 2005-2010

Agent based on its objectives

- Need to envision possible actions and their results
 - Evolution of the world : current state + action = new state
- Be able to determine which new state is the « closest » to the objective
 - Need of domain expertise
 - Need of heuristics
- A* ?...

94 (C) Hervé Barbot, 2005-2010

98

Agent based on usefulness

- « Goal » ?
 - Binary..... « ok » or « nok »
- « Usefulness » ?
 - A scale from less useful to most useful
u : state \rightarrow value
- « usefulness » instead of « goal » :
 - Conflict : speed / security
 - Several goals

(C) Hervé Barbot, 2005-2010

Learning agents : principles

- Why « learning » ?
 - Improve for decision making
 - Increase accuracy of the « percept / state X action » decisions
- How to decide what is « better » ?
 - Need to have feedback
 - Feedback needs « performance standard »

102 (C) Hervé Barbot, 2005-2010

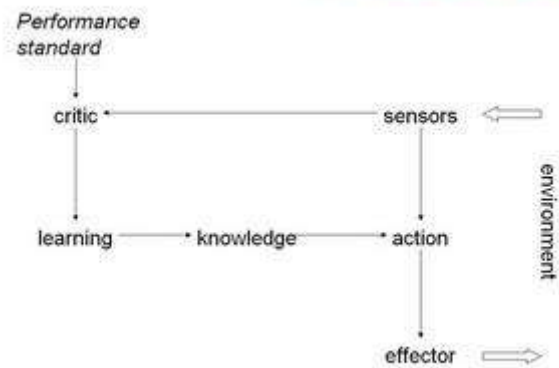
106

Learning - General architecture

- From « normal » agent :
 - Sensors – decision / action element – effectors
 - Based on knowledge
- Critic
 - Evaluation of performances
 - Based on external measurement standards
 - Determines what is « good », « bad » or « somewhere in between »
- Learning element
 - Implements the learning mechanism
 - Improves the efficiency of the decision / action element through the knowledge base

(C) Hervé Barbot, 2005-2010

General architecture



107 (C) Hervé Barbot, 2005-2010

108

Performance

▪ How an agent should act?

- Considérer toutes les options et faire le meilleur choix pour maximiser les chances de succès
- Mesure de performance :
 - Réussir la tâche ?
 - Quantité de ressources consommées ?
 - Temps mis pour réaliser l'action ?
- Exemples :
 - Véhicule automatique : destination atteinte, conduite sûre, parcours le plus court, ...
 - Aspirateur robotisé : quantité de poussière récoltée, quantité d'électricité consommée, bruit généré, propreté du sol, ...

(C) Hervé Barbot, 2005-2010

110 (C) Hervé Barbot, 2005-2010

112

Performance

▪ Mesure de la performance d'un agent

- Comment ?
 - Par mesures externes ou par auto-évaluation
- Quand ?
 - De manière continue, périodique, ou une seule fois ?

(C) Hervé Barbot, 2005-2010

Performance

▪ Mesures de performance extérieures

- Les critères de performance sont analysés
- L'environnement fournit un feedback à l'agent

$$V: S^* \rightarrow \mathcal{R}$$

- Feedback
 - après chaque action ou
 - périodiquement ou
 - en fin de traitement

113 (C) Hervé Barbot, 2005-2010

114

Performance

- Exemple du thermostat



- la température ambiante est échantillonnée à intervalles réguliers.

$$V(s_t) = 100 \text{ si } \text{température}(s_t) \approx \text{DÉSIRÉE} \\ = -100 \text{ sinon}$$

- objectif: maximiser $V(s_0) + \gamma V(s_1) + \dots + \gamma^n V(s_n) + \dots$
avec le facteur $0 < \gamma \leq 1$