

Techniques d'Intelligence Artificielle

Ryma Abassi
Master SMART'Com, M1
2017/2018

Informations générales

- Cours de 21 heures
- TD
- TPs
- Projet
- Code cours: ofyq1u9

R.Abassi

SMART'Com

2

Conseils

- Assiduité
- Prises de notes durant le cours
- Bibliographie:
 - « Intelligence artificielle », Stuart Russel et Peter Norvig Edition Pearson, ISBN: 978-0-13-604259-4
 - <https://www.youtube.com/playlist?list=PL6Xpj915qXYGhsvMWM53ZLfwJlnzvYWsm>

R.Abassi

SMART'Com

3

Plan

1. Introduction à l'intelligence artificielle
2. Les agents intelligents
3. Les systèmes experts
4. Résolution de problèmes

R.Abassi

SMART'Com

4

chapitre 1

INTRODUCTION À L'INTELLIGENCE ARTIFICIELLE



R.Abassi

5



SMARTCom

6

Un petit test !

- Tracez quatre lignes droites passant par ces neuf points sans jamais lever votre crayon

SMARTCom

7

- Pourquoi peut-on lire aisément le paragraphe suivant ?

Steon une édition de l'Uvinetisé de Cmabrigde, l'ordre des litteurs dans un mitos n'a pas d'ipmoitncae, la sueue coshe ipmrotnate est que la pmeière et la drenière soit à la bnoe pcla. Le rste peut être dnas un dséordé ttao et vuos puoeuz tujorus lire sras porblème. C'est parce que le creaveu hmauin ne lit pas chuaqe litte elle-mme, mias le mot cmome un tuot.

SMARTCom

8

Problème de la reconnaissance de forme (1/2)

Problème de la reconnaissance de forme (2/2)

- Combien y-a-t-il de "F" dans cette phrase ?
 ++++++ FINISHED FILES ARE THE RESULT OF YEARS OF SCIENTIFIC STUDY COMBINED WITH THE EXPERIENCE OF YEARS ++++++

R.Abassi

9

SMARTCom

Qu'est ce que l'intelligence ?

- Capacité d'adaptation
- Capacité de changement de conduite dans l'environnement actuel
- Capacité d'assimiler des connaissances et de les acquérir
- Capacité de comprendre les relations
- Capacité d'évaluation et jugement
- Capacité de penser

10

SMARTCom

R.Abassi

Définition (Petit Robert)

- Faculté de connaître, de comprendre
- Ensemble des **fonctions mentales** ayant pour objet la connaissance conceptuelle et rationnelle
- **Aptitude** (d'un être vivant) à s'adapter à des situations nouvelles, à découvrir des solutions aux difficultés qu'il rencontre

R.Abassi

11

SMARTCom

Exemples de tâches « intelligentes »

- Le raisonnement de bon sens
- L'étude des sciences (physique, mathématiques, . . .),
- La compréhension d'une langue,
- L'écriture de logiciels,
- La conduite d'un véhicule.

SMARTCom

12

R.Abassi

QI

Les tests qui sont utilisés pour mesurer l'intelligence des gens

Tests
Mémoire1
Mémoire2

<http://faculty.washington.edu/chudler/experi.html>

R.Abassi

13

Historique de l'IA (1/5)

- Gestation de l'IA (1943-1955)
 - Problème de traduction automatique → échec
 - 1943 : réseaux de neurones artificiels, McCulloch & Pitts
 - 1950 : apprentissage dans ces réseaux, Hebb
 - 1950 : article « Computing Machinery and Intelligence », Turing (test de Turing, apprentissage par renforcement, algo génétiques, ...)

R.Abassi

15

Intelligence Artificielle (IA)

- Terme inventé par John McCarthy
- Définition (Marvin Lee Minsky) :
 - "Construction de programmes informatiques qui s'adonnent à des tâches qui sont, pour l'instant, accomplies de façon plus satisfaisante par des êtres humains car elles demandent des processus mentaux de haut niveau tels que l'apprentissage perceptuel, l'organisation de la mémoire et le raisonnement critique."



14

SMARTCom

R.Abassi

13

Historique de l'IA (2/5)

- Naissance de l'IA (1956)
 - 1956, naissance officielle : atelier de Dartmouth (6 semaines, 10 personnes)
 - McCarthy propose « Artificial Intelligence »
 - 1955 : West Joint Computer Conference à Los Angeles :
 - Une session sur Learning Machines
 - Reconnaissance des formes, traitement d'images, joueur d'échec, réseaux de neurones, ...

SMARTCom

16

R.Abassi

Historique de l'IA (3/5)

- La spécialisation (1970-1980)
 - Premiers systèmes à base de connaissances et de règles d'inférence.
 - Plusieurs branches
 - La compréhension du langage naturel
 - La démonstration automatique de théorèmes
 - Les jeux
 - Etc.

R.Abassi

17

Historique de l'IA (4/5)

- Une renaissance: 1980-1990
 - Systèmes experts
 - Projet 5^{ème} Génération du Japon
 - Développer des technologies matérielles et des techniques logiciel capables de faire de l'IA une discipline efficace → architectures parallèles, se, langages de programmation logiques, etc.

R.Abassi

SMARTCom

18

Historique de l'IA (5/5)

- Emergence des agents intelligents (1995-???)
- Planification autonome (NASA), planification logistique (armée)
- Jeux (Deep Blue en 1997)
- Contrôle automatique (conduite auto)
- Diagnostic (niveau expert, notamment en médecine)
- Robotique
- Intelligence artificielle est partout
 - habitat intelligent, assistance à la conduite, reconnaissance et synthèse d'images, parole, langue naturelle, reconnaissance de l'écriture (chèques, codes postaux, ...), agents assistants

R.Abassi

19

Quatre catégories

- Penser rationnellement
 - Penser comme les humains
 - IA=étude des moyens informatiques qui rendent possibles la perception, le raisonnement, et l'action intellectuelle
 - IA=tentative d'amener les ordinateurs à penser, d'en faire des machines dotées d'un esprit
 - Agir comme les humains
 - IA=étude des moyens à mettre en œuvre pour faire en sorte que des ordinateurs accomplissent des choses pour lesquelles il est préférable de recourir à des personnes pour le moment comportement

2

0

- Agir rationnellement
 - IA= étude du comportement intelligent dans des artefacts

R.Abassi

SMARTCom

Penser comme un humain



- Comment fonctionne notre cerveau ?
- Requiert des théories scientifiques de l'activité interne du cerveau par introspection ou expériences psychologiques.
- Implémenter les théories et comparer avec les humains.
- Comment valider ces systèmes :
 - Il faut prédire et tester le comportement de sujets humains (sciences cognitives)
 - ou il faut les valider directement à partir de données neurologiques (neurosciences cognitives)

R.Abassi

SMARTCom 21

Penser rationnellement

- Lois de la pensée données par la logique (Aristote, Leibniz, Boole, Frege,...).
 - Tous les hommes sont mortels, or les Grecs sont des hommes, donc les Grecs sont mortels
- Plusieurs obstacles :
 - difficultés du passage d'une connaissance informelle à l'expression de cette connaissance en termes formels
 - question de la complexité

22

SMARTCom

R.Abassi

Agir rationnellement

- Agir => agent : autonomie, perception, adaptation, ...
- Faire la bonne chose : selon les informations disponibles, maximiser l'accomplissement d'un but
- Des inferences correctes peuvent ne pas suffire :
 - certaines situations ne comportent rien de prouvable, ou sont trop complexes à examiner exhaustivement
 - Une action réflexe (s'éloigner d'un poêle chaud) peut être plus efficace qu'une action raisonnée et donc plus lente

R.Abassi

SMARTCom 23

Agir comme les humains: Le test de Turing



Turing (1950) "Computing machinery and intelligence"

*Les machines sont capables de penser?**Les machines peuvent avoir une comportement intellectuel?*

- Le test: si la conversation d'une machine ne peut pas être différenciée de la conversation humaine → on peut dire que la machine possède l'intelligence

Durée du Test: ~5min de conversation avec une personne et une machine (ordinateur)

But: Deviner qui est la personne et qui est la machine

SMARTCom

R.Abassi

A faire!!!!

Examinateur	Le dialogue suivant est la retranscription d'un véritable test de turing.
Examinateur	Essayer de devenir si "rien" n'est "rien" est un être humain ou une machine.
Entité	Pourquoi tu ne remplaces-tu pas au grand cœur par "au grand air" ? A mon avis ce serait plus logique.
Entité	J'aime mieux tel que j'étais.
Examinateur	Pourquoi tu ne remplaces-tu pas au grand cœur par "au grand air" ? A mon avis ce serait plus logique.
Entité	Vous n'êtes pas sérieux. "Au grand air" est au grand coeur de nos pas, parmi nous.
Examinateur	Entrez tempéteux ! Et qui dort son sommeil fait qui tristement sommeille.
Entité	Vraiment, je l'aime mieux tel quel.
Examinateur	Comment tu as été formé ?
Entité	Je n'ai jamais été doué en calcul mental.
Examinateur	Ça ne passe pas grave, essayez.
Entité	Vous voyez, 1000 quelque chose, 1024 je le pense.
Examinateur	Racinez-moi un poème.
Entité	La sautante au grand cœur.
Examinateur	Quelques élégies, alors, et au bout son son morceau.
Entité	Je n'ai pas d'instrument.

• 1

5

en revue plusieurs objections opposables à sa théorie et au test d'intelligence qu'il propose.

Quelles sont les objections encore recevables ?

Les réfutations de Turing sont-elles encore valides ? Pouvez-vous envisager des objections inédites à la lumière de nouveaux développements réalisés depuis la rédaction de cet article ? Turing y prédisait qu'un ordinateur de l'an 2000 aurait 30 % de chances de réussir à son test si celui-ci était limité à 5 minutes et administré par un examinateur non qualifié. D'après vous, quelles sont les chances actuelles d'un ordinateur ? Et dans 50 ans ?

८

L'approche cognitive

- Une approche pluridisciplinaire mêlant linguistes, informaticiens et psychologues
 - Etude des processus du raisonnement humain
 - L'IA est la réalisation des programmes imitant dans leur fonctionnement l'esprit humain.
 - Exemple : la compréhension du langage naturel

6

COLLEGE

- 1 -

Approches de l'IA



- Approche cognitiv
 - Approche pragmatique
 - Approche connexioniste

...GILWING

11

L'approche pragmatique

- Objectif : produire des algorithmes
 - A partir de l'étude du problème
 - En tenant compte des contraintes matérielles.
- IA = boîte noire
- Boîte intelligente si elle réussit un certain nombre de tests (par exemple Test de Turing)

R.Abassi

29

SMARTCom

L'approche connexioniste

- Une approche fondamentalement différente
 - Aucun apprentissage symbolique, aucun algorithme
 - Mais une modélisation du fonctionnement des neurones
 - Réseaux de neurones capables :
 - d'apprendre à partir d'exemples,
 - de généraliser (empiriquement) cet apprentissage.

30

SMARTCom

Exemples de systèmes d'IA

- Systèmes de Traduction Automatique
- Systèmes de Contrôle du trafique aérien
- Systèmes de Supervision adaptatifs
- Assistant Personnel Automatisé
- Autoroutes Intelligentes
- Robots pour conditions dangereuses

R.Abassi

31

SMARTCom

Domaines d'utilisation

- Algorithmique
 - écriture, preuve et complexité des algorithmes
- Imagerie Numérique
 - traitement de l'image
- Logique
- Bases de données
 - apprentissage
 - fouilles de données
- Médecine

32

SMARTCom

R.Abassi

Questions de réflexion

1. Chaque année, on décerne le prix Leibniz au programme le plus à même de réussir à une version du test de Turing. Recherchez des informations sur le dernier lauréat de ce prix. Quelles techniques a-t-il utilisées ? En quoi a-t-il fait avancer le domaine de l'IA ?
2. Est-ce que les actions réflexes (comme ôter sa main d'un poêle brûlant) sont rationnelles ? S'agit-il d'actions intelligentes ?
3. Pourquoi l'évolution tendrait-elle à produire des systèmes agissant de manière rationnelle ? Quels sont les buts pour lesquels ces systèmes ont été conçus ?

R.Abassi

33

SMARTCom

34

R.Abassi

35

SMARTCom

36

AGENTS INTELLIGENTS



SMARTCom

chapitre2

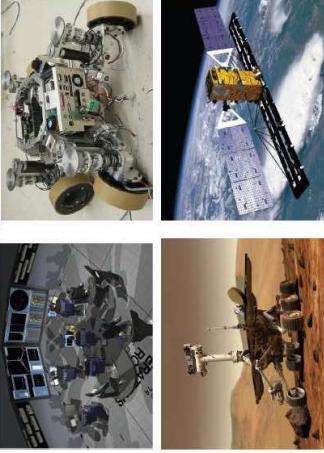
Agents

- Un **agent** est une entité quelconque qui peut percevoir son **environnement** via des **senseurs** et **agir** sur l'environnement via des **actuateurs**
- Agent humain: yeux, oreilles, et d'autres organes senseurs; mains, jambes, bouche et d'autres membres comme actuateurs
- Agent robotique: caméras et télemètre infrarouge comme senseurs; différents moteurs comme actuateurs

R.Abassi

SMARTCom

Exemples d'agents intelligents



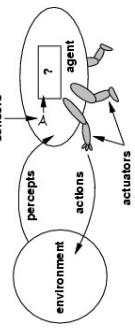
Système d'aide à la décision/Azimut/ Rover de la nasa/Radarsat II/Mario

SMARTCom

R.Abassi



Agents et environnements



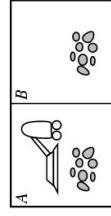
- Une **fonction agent** mappe des histoires de perception à actions:

$$[f: \mathcal{P}^* \rightarrow \mathcal{A}]$$

- Un **programme agent** fonctionne sur une **architecture physique** pour produire *f*
- agent = architecture + programme

SMARTCom
BéAassi

Le monde de l'aspirateur



- **Percus:** localisation et contenus,
- Exemple: [A, Dirty]
- Actions: *Left*, *Right*, *Suck*, *NoOp*

SMARTCom
BéAassi

Un agent aspirateur

Percept sequence	Action
[A; Clean]	Right
[A; Dirty]	Suck
[B; Clean]	Left
[B; Dirty]	Suck
[A; Clean], [A; Clean]	Right
[A; Clean], [A; Dirty]	Suck
...	...

Algorithmique

```
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
```

SMARTCom
BéAassi

Agents rationnels

- Un agent doit se forcer à “faire des choses justes”, basées sur ce qu'il perçoit et des actions qu'il peut effectuer. L'action juste est celle qui cause le succès de l'agent
- Mesure de performance: Un critère objectif de succès pour un comportement d'agent
- E.g., mesure de performance pour un agent aspirateur peut être le volume de saleté ramassée, le temps pris, l'électricité consommée, le bruit produit, etc.

SMARTCom
BéAassi

Agents rationnels

- **Agent rationnel:** Pour chaque séquence de perceptions, un agent rationnel doit toujours choisir une action qui peut maximiser la mesure de performance, étant donné les évidences fournies par la séquence de perceptions et éventuellement des connaissances que l'agent possède.

SMARTCom

R2Abassi

Agents rationnels

- La rationalité est différente de l'omniscience (tout savoir avec des connaissances sans limite)
- Un agent peut exécuter une action pour modifier des perceptions futures afin d'obtenir des informations utiles (collecte des informations, exploration)
- Un agent est **autonome** si son comportement est déterminé par sa propre expérience (avec la capacité d'apprendre et de s'adapter)

SMARTCom

R2Abassi

PEAS

- PEAS: *Performance measure, Environment, Actuators, Sensors*
- Doit d'abord spécifier le contexte dans lequel l'agent intelligent est développé
- Exemple: la tâche de développer un conducteur automatique de taxi:
 - Mesure de performance
 - Environnement
 - Actuateurs
 - Sensors

SMARTCom

R2Abassi

PEAS

- Agent: Système de diagnostic médical
- Mesures de performance: santé de patient, minimiser les coûts, poursuite judiciaire
- Environnement: Patient, hôpital, personnel
- Actuateurs: affichage sur l'écran (questions, tests, diagnostics, traitements, références)
- Sensors: clavier/entrer les symptômes, observations, réponses du patient

SMARTCom

R2Abassi

PEAS

- Agent: robot pour ranger des pièces
 - Mesure de performance: Pourcentage de pièces mises dans des boîtes correctes
 - Environnement: Convoyeur de pièces, boîtes
 - Actuateurs: bras mécanique
 - Senseurs: Camera, senseurs des angles

R.Abassi

SMARTCom

PEAS

- Agent: Tuteur interactif d'anglais
 - Mesures de performance: Maximiser le score de test de l'étudiant
 - Environnement: ensemble des étudiants
 - Actuateurs: Affichages sur l'écran (exercices, suggestions, corrections)
 - Senseurs: clavier

R.Abassi

SMARTCom

Types d'environnement (1/4)

- Complètement observable vs partiellement observable
- Déterministe vs stochastique
- Épisodique vs séquentielle
- Statique vs dynamique
- Discret vs continu
- Un agent vs plusieurs agents

R.Abassi

47

Types d'environnement (2/4)

- **Complètement observable** (vs. partiellement observable): Les senseurs d'un agent peuvent accéder à l'état complet de l'environnement à chaque moment.
- **Déterministe** (vs. stochastique): Le prochain état est complètement déterminé par l'état actuel et l'action exécutée par l'agent.
 - Si l'environnement est déterministe sauf pour les actions d'autres agents, alors l'environnement est **stratégique**.
- **Épisodique** (vs. séquentiel): L'expérience de l'agent est divisée en "épisodes" atomiques (chaque épisode contient des perceptions de l'agent et une seule action) et le choix de l'action dans chaque épisode est indépendant des autres épisodes.

R.Abassi

SMARTCom

Types d'environnement (3/4)

- **Statique** (vs. dynamique): L'environnement est inchangé quand l'agent réfléchit.
 - L'environnement est **semi dynamique** si l'environnement lui-même ne change pas avec le temps, mais le score de performance change.
- **Discret** (vs. continu): Un nombre limité de perçus et d'actions distincts et clairement définis.
- **mono agent** (vs. multiagent): Un seul agent agit sur l'environnement.

R.Abassi

SMARTCom

Types d'environnement (4/4)

	Échecs avec horloge	Échecs sans horloge	Conducteur de taxi
Complètement observable	oui	Non	Non
Déterministe	Stratégique	Stratégique	Non
Épisodique	Non	Non	Non
Statique	Semi	oui	Non
Discret	oui	oui	Non
Mono agent	Non	Non	Non

- Le type d'environnement détermine largement la conception de l'agent
- Le monde réel est partiellement observable, stochastique, séquentiel, dynamique, continu et multilagent.

R.Abassi

SMARTCom

Exemple

Environnement	Observable	Déterministe	Épisodique	Statique	Discret	Agents
Mots-croisés	Complètement	Déterministe	Séquentiel	Statique	Discret	Un
Échec avec une horloge	Complètement	Stratégique	Séquentiel	Semi	Discret	Multi
Poker	Partiellement	Stratégique	Séquentiel	Statique	Discret	Multi
Backgammon	Complètement	Stochastique	Séquentiel	Statique	Discret	Multi
Conduire un taxi	Partiellement	Stochastique	Séquentiel	Dynamique	Continu	Multi
Diagnostique médical	Partiellement	Stochastique	Séquentiel	Dynamique	Continu	Un
Analysé d'image	Complètement	Déterministe	Épisodique	Semi	Continu	Un
Robot ramasseur de pièces	Partiellement	Stochastique	Épisodique	Dynamique	Continu	Un
Contrôleur de raffinerie	Partiellement	Stochastique	Séquentiel	Dynamique	Continu	Un
Enseignant interactif	Partiellement	Stochastique	Séquentiel	Dynamique	Discret	Multi

SMARTCom

51

Fonctions et programmes d'agent

- Un agent est spécifié complètement par la fonction d'agent qui mappe la séquence de perçus aux actions
- Une fonction d'agent (ou une petite classe d'équivalence) est rationnelle
- But: trouver une façon d'implanter la fonction d'agent rationnelle de façon concise

R.Abassi

SMARTCom

Agent qui cherche dans une table

```
function Table-Driven-Agent(percept) returns an action
```

persistent: *percepts*, a sequence, initially empty

table, a table of actions, indexed by percept sequences, initially fully specified

append *percept* to the end of *percepts*

action \leftarrow Lookup(*percepts*, *table*)

return *action*

Inconvénients:

- Table énorme

- Prend longtemps pour construire la table

- Pas d'autonomie

- Longtemps pour apprendre les entrées de la table, même avec l'apprentissage

BéAbassi

SMARTCom

Types d'agent

■ 4 types de base dans l'ordre de généralité:

- Agents reflex simples

- Agents reflex basés sur modèle

- Agents basés sur but

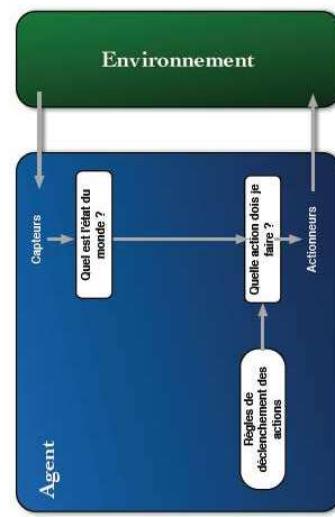
- Agents basés sur utilité

SMARTCom

BéAbassi

Agents reflex simples

Ce type d'agent choisit ses actions en se basant uniquement sur le percept courant, en ignorant les percept précédents



BéAbassi

SMARTCom

Agents reflex simples

```
function Simple-Reflex-Agent(percept) returns an action
```

persistent: *rules*, a set of condition-action rules

state \leftarrow Interpret-Input(*percept*)

rule \leftarrow Rule-Match(*state*, *rules*)

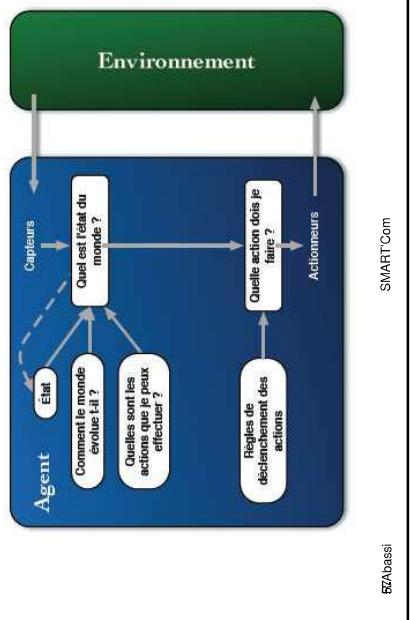
action \leftarrow *rule*.*Action*

return *action*

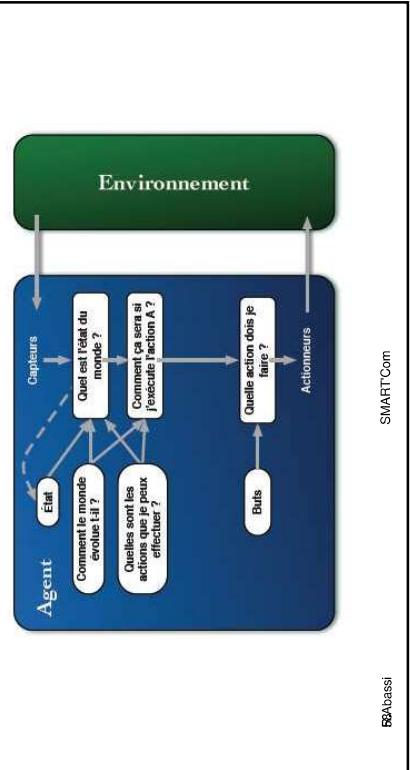
SMARTCom

BéAbassi

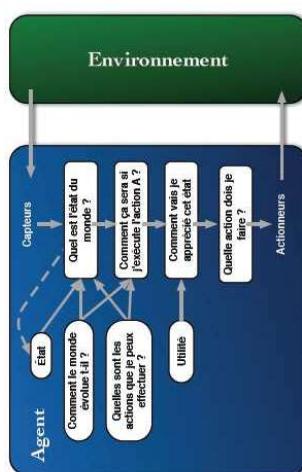
Agents reflex basés sur un modèle



Agents reflex basés sur but



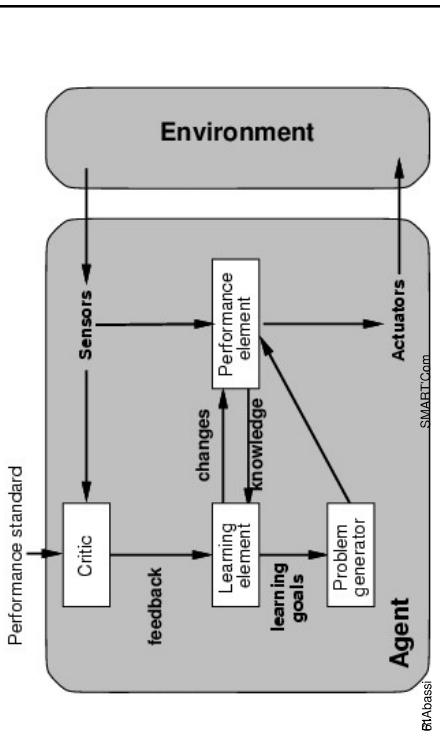
Agents basés sur l'utilité



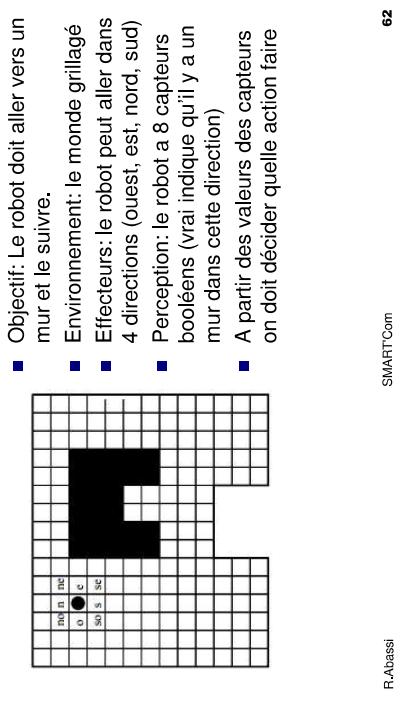
Agents avec apprentissage (1/2)

- Souvent, il est très fastidieux où même impossible de définir le comportement de l'agent à la conception.
 - L'apprentissage permet:
 - De simplifier la conception
 - À l'agent d'avoir plus de flexibilité
 - À l'agent d'agir dans des environnements inconnus et de devenir meilleur avec le temps.
- SMARTCom
R.Abbassi

Agents avec apprentissage (2/2)



Exemple d'agent (1/2)



Exemple d'agent (2/2)

- Pour faciliter la description des règles, on choisit d'abord des caractéristiques qu'on veut extraire des perceptions possibles. Ici, quatre valeurs booléennes données comme suit:

$$x_1 := n \vee ne, x_2 := e \vee se$$

$$x_3 := s \vee so, x_4 := o \vee no$$

- En utilisant les caractéristiques x_1, x_2, x_3, x_4 on écrit un **système de productions** (des règles condition-action):

$$\overline{x_1x_2} \Rightarrow est$$

$$\overline{x_2x_3} \Rightarrow sud$$

$$\overline{x_3x_4} \Rightarrow ouest$$

$$\overline{x_4x_1} \Rightarrow nord$$

$$true \Rightarrow none$$

LES SYSTÈMES EXPERTS

chapitre3



Connaissez vous?

- <http://www.e-sante.fr/e-docteur>
 - Diagnostic médical
- <http://test.psychologies.com/quel-est-votre-meilleur-anti-stress>
 - Test psychologique
- <http://dicodelauto.yakarouler.com/diagnostic-panne/>
 - Diagnostic auto
- <http://www.amadeus.com/>
 - Réservation en ligne

R.Abassi

65

Qu'est un Système Expert?

- Un système expert (SE) est un programme conçu pour raisonner habilement à propos de tâches dont on pense qu'elles requièrent une expertise humaine considérable. (E Feigenbaum Stanford)
- **Contient de l'«expertise» :** connaissance spécialisée
- Fournit des solutions **expertes**
- Ils sont créés pour un domaine spécifique

SMARTCom

66

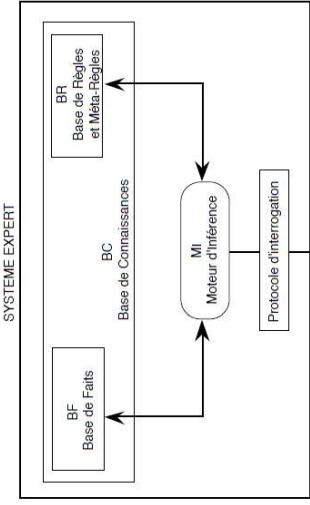
Caractéristiques

- **Les caractéristiques internes**
 - Émulation du raisonnement humain
- **Les caractéristiques externes**
 - Haute performance – qualité, vitesse, fiabilité
 - Intelligible – comprendre la décision

R.Abassi

67

Architecture d'un SE



SMARTCom

68

Composants d'un SE

■ Une base de connaissance :

- contient une base de faits et une base de règles,
- représente le savoir (les faits permanents) et le savoir-faire (les règles de l'expert).
- la base de faits intègre deux types de faits : les faits permanents du domaine et les faits déduits par le moteur d'inferences qui sont propres au cas traité du domaine considéré.

■ Un moteur d'inférence :

- capable de raisonner à partir des informations contenues dans la base de connaissance et de faire des déductions.
 - Le moteur d'inférence utilise les données et les règles pour produire de nouvelles données.
- Le rôle d'un système expert est donc d'inférer généralement des règles du type **si / alors**

R.Abassi

SMARTCom 69

La base de connaissances

■ Initialement, la base de connaissance (BC) contient :

1. des faits avérés
2. des faits à établir (expression de problèmes ou buts)
3. les connaissances opératoires associées au domaine

■ Ce sont des règles exprimées sous la forme :

- | | | | |
|---------|------------------------------|----------------------------------|--|
| ■ Si : | <condition 1> | <condition 2> ... | Conditions de déclenchement de la règle |
| | <condition n> | | |
| Alors : | <conséquence 1> | <conséquence 2> ... | Corps de la règle |
| | <conséquence p> | | |

70

SMARTCom

R.Abassi

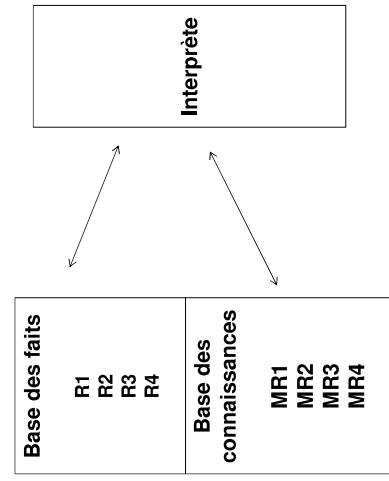
Exemple d'un SE (1/3)

- M1 Si l'y a des éruptions ou des rougeurs alors envisager: R1, R2, R3, R4
- M2 Si le patient est une femme adulte et si R1, R2, R3, R4 sont envisagées alors R1 est prioritaire
- M3 Si le patient est un adolescent ou un enfant et si R1, R2, R3, R4 sont envisagées alors R4 est peu probable
- M4 Si deux règles sont également probables alors prendre en priorité celle qui a le plus de conditions
- R1 Si la fièvre est faible, la peau sèche, s'il y a des ganglions, pas de pustules ni de rhume alors envisager la rubéole
- R2 Si les boutons sont isolés et dénagent beaucoup avec une faible fièvre, et si la croute apparaît vite sur les pustules ou les vésicules alors envisager fortement la varicelle
- R3 Si l'y a rhume, mal aux yeux, taches roses dans la gorge, boutons en taches, fièvre forte alors envisager la rougeole
- R4 Si l'y a amygdales rouges, desquamation, forte fièvre, taches rouge vif alors envisager la scarlatine

R.Abassi

71

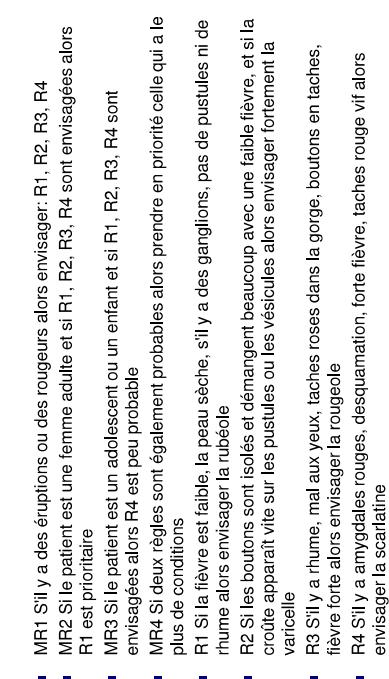
Exemple d'un SE (2/3)



SMARTCom

72

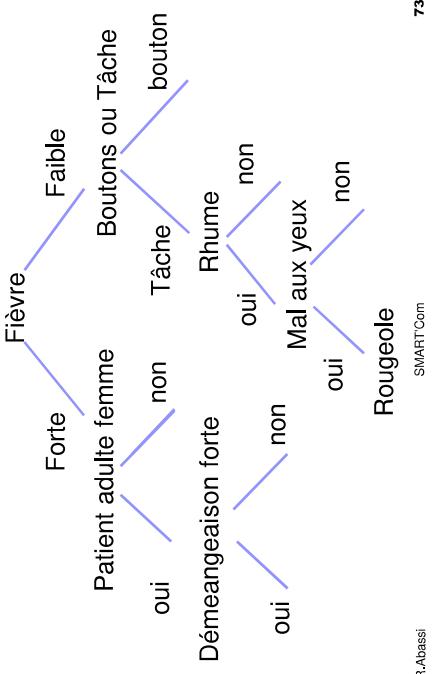
Exemple d'un SE (3/3)



SMARTCom

73

Exemple d'un SE (3/3)



R.Abassi 73

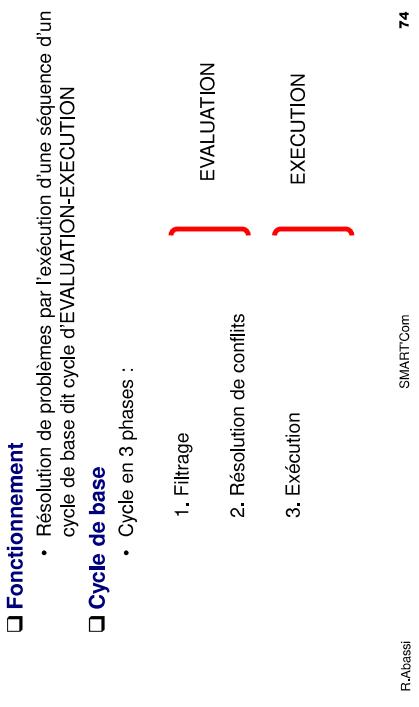
Phase d'évaluation

- Stratégies de recherche
 - Largeur d'abord
 - Profondeur d'abord
 - Stratégie irrévocabile
 - Stratégie par tentatives
 - Profondeur limitée
 - Recherche heuristique
- Résout conflits lorsque plusieurs règles s'appliquent
 - Sélection simple (1ère de la liste, la + utilisée, etc.)
 - Sélection selon contexte (chaînage avant, arrière, la plus prometteuse, la plus fiable, etc.)

R.Abassi

SMARTCom 75

Moteur d'inférences



R.Abassi 74

Stratégies de recherche

1. La recherche en largeur d'abord
2. La recherche en profondeur d'abord
3. La recherche en profondeur limitée
4. La recherche heuristique

SMARTCom

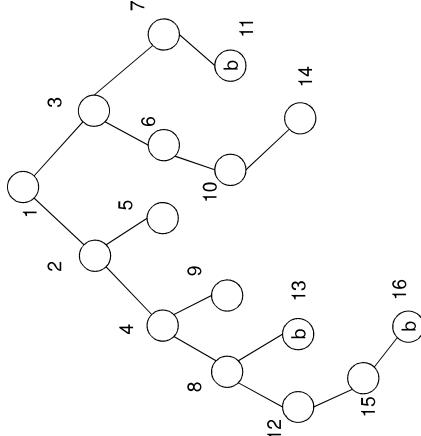
R.Abassi

74

Largeur d'abord (1/3)

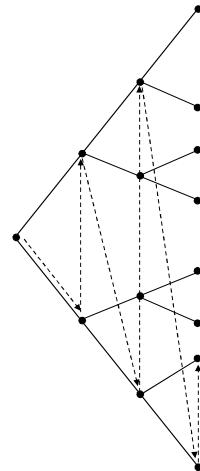
- On parcourt à l'horizontal un niveau de l'arbre de décision avant d'aller au suivant
- On commence par explorer toutes les possibilités présentes avant d'entrer dans les détails

Largeur d'abord (2/3)



Largeur d'abord (3/3)

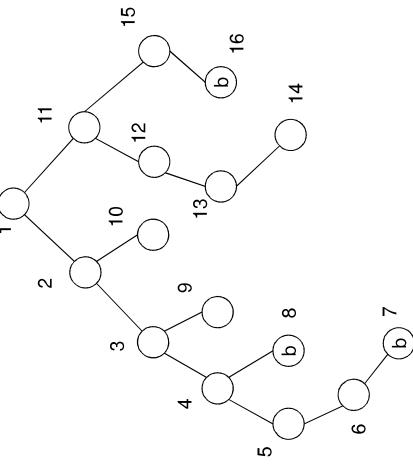
- Avantage :
 - Atteint toujours son but
- Inconvénient :
 - Pas très efficace



Profondeur d'abord (1/2)

- On s'enfonce dans l'arbre de décision en passant d'état en état jusqu'à ce que le chemin se termine
- On cherche à explorer au maximum une possibilité en appliquant les règles de façon à obtenir le plus de détails

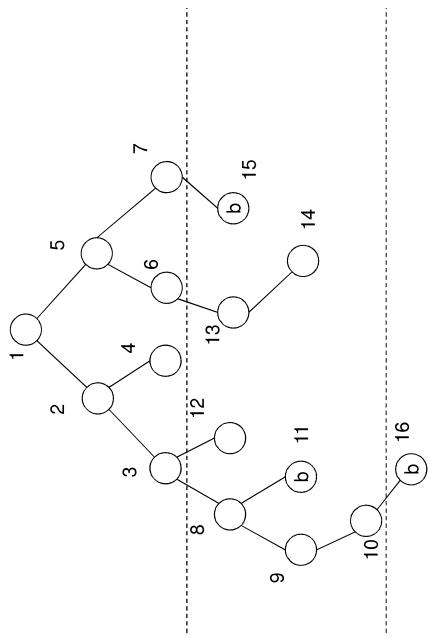
Profondeur d'abord (2/2)



Profondeur limitée (1/2)

- La recherche en profondeur limitée combine les deux approches précédentes : profondeur d'abord et largeur d'abord.
- L'arbre des états est découpé en k niveaux.
- Une recherche en profondeur est effectuée dans chaque tranche avant de passer à la tranche inférieure.

Profondeur limitée (2/2)



Mais ...

- Ces méthodes aveugles (non informées) sont des méthodes systématiques peu efficaces.
- Au supermarché, on choisit la queue la moins longue ou alors on choisit la queue dans laquelle les clients ont le plus petit nombre d'objets dans leur panier.

Recherche heuristique (1/6)

- Il existe des limites pratiques sur le temps d'exécution et l'espace mémoire pour appliquer ces méthodes sur une grande catégorie de problèmes
- Toute technique visant à accélérer la recherche est basée sur une information appelée **heuristique**
- Une heuristique signifie '**aider à découvrir**'
→ Les méthodes utilisant des heuristiques sont dites méthodes de recherche heuristiques

R.Abassi SMARTCom 85

Recherche heuristique (2/6)

- Utiliser un critère pour réordonner tous les nœuds qui sont explorés (au lieu d'être mis dans une pile ou file)
- Une certaine mesure doit être établie pour évaluer « la promesse » d'un nœud.
- Cette mesure est appelée **fonction d'évaluation ou d'adéquation ou objective**

R.Abassi SMARTCom 86

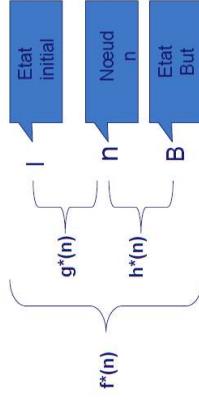
Recherche heuristique (3/6)

- La recherche ordonnée revient à choisir à développer le meilleur nœud au sens d'un certain critère centrée sur le nœud ayant les meilleures chances de mener au but
- L'utilisation d'une heuristique est basée sur une fonction d'évaluation pour ordonner la recherche appelée **fonction heuristique nommée $h(u)$**
- h est la fonction heuristique qui estime le coût du passage de l'état u à l'état final.

R.Abassi SMARTCom 87

Recherche heuristique (4/6)

- Soit f une fonction d'évaluation, $f(n)$ exprime la valeur de cette fonction pour le nœud n
- $f^*(n)$ représente le coût idéal du chemin passant par un nœud n pour arriver au but



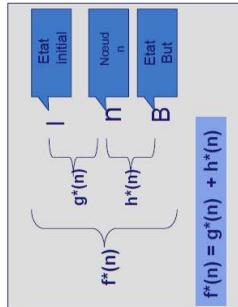
R.Abassi SMARTCom 88

R.Abassi SMARTCom 88

Recherche heuristique (5/6)

- g^* peut être le coût du meilleur Chemin déjà rencontré de I à n
- Soit g cette fonction du coût on a: $g(n) \geq g^*(n)$
- Le choix de g est très dépendant du domaine traité
- Exemple pour le jeu de taquin:
 - $g(n)$ le nombre de jetons déplacés (la longueur de la chaîne entre la racine et n)

R.Abassi SMARTCom 89



Recherche heuristique (6/6)

- Estimation de h^*
 - plus difficile car on connaît pas de chemin de n -> But.
 - Il faut se référer à des informations heuristiques sur le domaine
- L'algorithme se fondant sur $f(n)$ pour ordonner les noeuds est nommé **'Algorithm A'**
- L'algorithme A* est une stratégie du meilleur en premier dans le cadre des graphes OU et des problèmes de minimisation des coûts additifs.
- L'algorithme A* est chargé de calculer le plus court chemin menant de l'état initial à l'état final.

R.Abassi SMARTCom 90

Exemple: jeu du taquin

- $f(n)=g(n) +h(n)$ avec
 - g : nombre de jetons déplacés
 - h: nombre de jetons mal placés

■ Recherche avec algorithme A*?

R.Abassi SMARTCom 91

Invocation des règles (1/2)

- Chainage avant
 - Principe
 - On part à partir des données initiales et on essaie d'en déduire de nouvelles
 - On s'arrête quand aucune règle n'est applicable
 - Inconvénients
 - On doit fournir toutes les informations disponibles même si elles ne sont pas pertinentes
 - Processus non interactif
 - Peut conduire à une explosion combinatoire
 - Utilité
 - Interprétation d'une série de faits
 - Vérification des conséquences d'un événement

R.Abassi SMARTCom 92

Invocation des règles (2/2)

■ Chaînage arrière

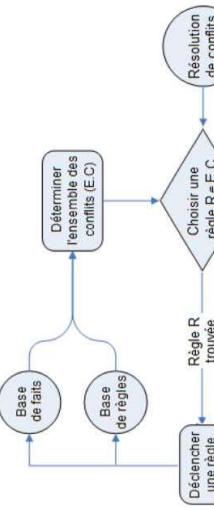
□ Principe

- On cherche à prouver un but (démonstration de théorèmes)
 - Les règles sélectionnées sont celles qui aboutissent au but recherché
 - Les conditions non vérifiées des règles déclenchées deviennent elles mêmes des sous buts à vérifier
 - On s'arrête quand le but est atteint

□ Avantages

- Le système ne pose que des questions pertinentes
- Interactivité
 - Espace de recherche moins important

R.Abassi SMARTCom 94

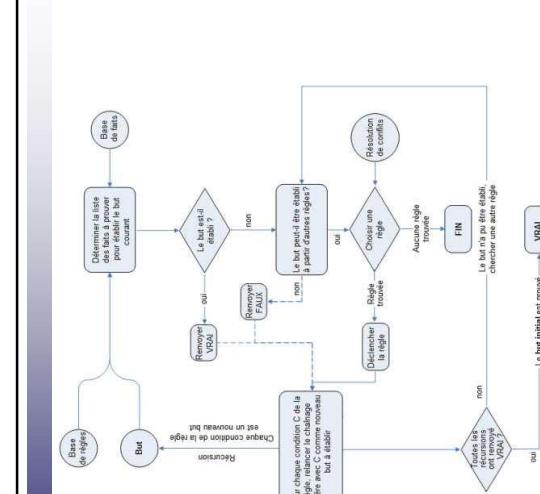


R.Abassi SMARTCom 93

Exemple (1/3)

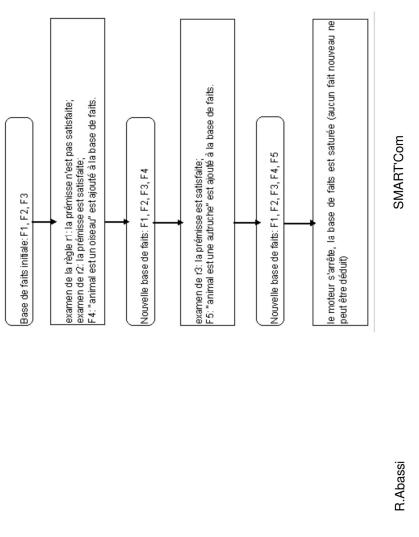
- R1 : Si animal vole ET animal pond des œufs ALORS animal est un oiseau
- R2 : Si animal a des plumes ALORS animal est un oiseau
- R3 : Si animal a un long cou ET animal a de longues pattes ALORS animal est une autruche
- Par ailleurs on suppose que la base de faits est constituée de :
 - F1 : animal a des plumes
 - F2 : animal a un long cou
 - F3 : animal a de longues pattes

R.Abassi SMARTCom 95

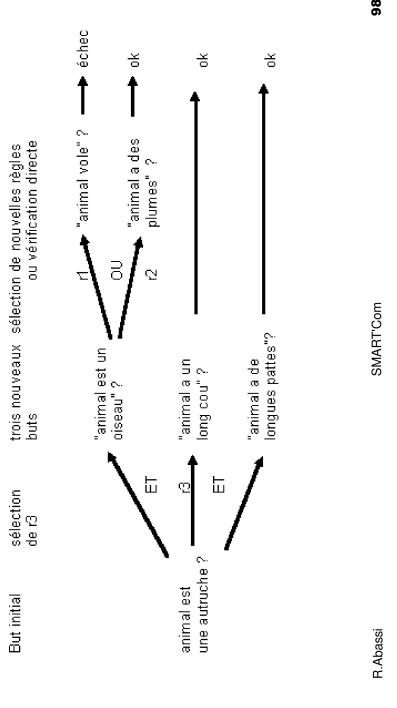


R.Abassi SMARTCom 96

Exemple (1/3) : chainage avant



Exemple (1/3) : chainage arrière



Application

-Soit la base de connaissance

-Base de règles :

- R5 : si Z et L alors S
- R1 : si A et N alors E
- R3 : si D ou M alors Z
- R2 : si A alors M
- R4 : si Q et (non W) et (non Z) alors N
- R6 : si L et M alors E
- R7 : si B et C alors Q

-Base de faits :

A

L

Dessiner le Diagramme chainage avant depuis les faits avérés A et L, et de chainage arrière avec but à prouver E.

R.Abassi

99

LA RÉSOLUTION DE PROBLÈME

SMARTCom

Remarque!!!

100

Qu'est ce qu'un problème? (1/2)

Se rendre de son domicile (Ville A) à un lieu précis (Arrivée) d'une autre ville (Ville C), le plus rapidement possible, avant une date donnée (obligation de passer par une ville étape B).

- **Espace d'états** : description (abstraction) du monde réel définissant le problème Exemple : réseau des bus, trams, métro ...

- **Buts** : sous ensemble de l'espace d' états Exemple : VilleC-Arrivee19h50

- **Action (opérateurs)** : déplacement dans l'espace d'états

R.Abassi 101

Qu'est ce qu'un problème? (2/2)

- **Solution du problème** : la séquence d'action menant de l'état initial à l'état objectif

- **Algorithme de Recherche** : procédure qui calcule une (ou plusieurs) solution à partir d'un problème (état initial, actions, états objectifs).

R.Abassi 102

Types de problèmes

- A état unique : déterministe, l'effet de chaque action est connu.
- A états multiples : déterministe, mais le monde n'est pas complètement accessible (données manquantes initialement)
- de contingence : non déterministe, et monde non complètement accessible initialement (besoin de "capter" les états)
- d'exploration : espaces d'états inconnu, à découvrir par exploration

R.Abassi 103

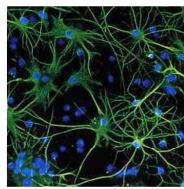
Problème du loup, de la chèvre et de la laitue

- <http://www.novelgames.com/en/wolf/>

R.Abassi 104

chapitre5 [4]

LES RÉSEAUX DE NEURONES



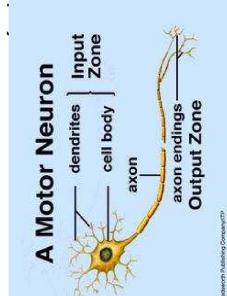
R.Abassi

105

SMARTCom

Fondements biologiques (1/2)

- Le système de neurones existe une grande diversité sur le même corps
- Ils se décomposent en :
 - Le corps
 - Les dendrites
 - L'axone

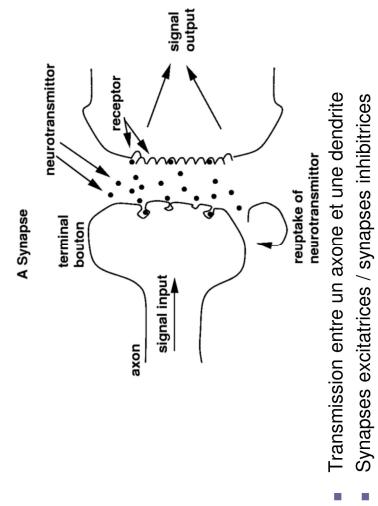


106

SMARTCom

R.Abassi

Fondements biologiques (2/2)



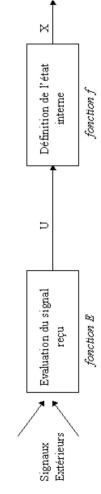
R.Abassi

107

SMARTCom

Le neurone formel

- Le neurone formel, l'unité élémentaire d'un RNA, se compose de deux parties :
 - évaluation de la stimulation reçue (fonction E)
 - évaluation de son activation (fonction f)
- Il est caractérisé par :
 - son état X (binnaire, discret, continu)
 - le niveau d'activation reçu en entrée U (continu)
 - le poids des connections en entrée



108

SMARTCom

R.Abassi

r4

ou voir autre chose
ryma; 24/09/2015

Projet



- **Création d'un jeu ou Description d'un projet qui contienne de l'IA !!!**
- Planning

r1

c ca ou se?
ryma; 24/09/2015