

Cours Agent Intelligent Partie 2

Jimmy Vogel

19 mars 2013

Table des matières

1 Rappel PEAS

1.1 Exemple

P(mesure de performance)

- trouver l'or +1000
- mort -1000
- -1 par déplacement
- -10 pour chaque flèche tiré

E(environnement)

- odeur désagréable dans les cases adjacentes au Wumpus
- courant d'air dans les cases adjacentes au puits
- éclat dans la case contenant l'or

A(Actions)

- se déplacer a gauche, droite, en haut, en bas
- tirer devant soit

S(Sensors)

- sentir l'air
- sentir l'odeur du Wumpus

1.2 Cas différents

Dans l'environnement il faut préciser si on est dans un environnement mono-agent. Ici on aurait pu préciser que le monstre ne peut se déplacer. Il faut aussi préciser que la taille de la matrice est limité.

2 Systèmes experts

2.1 Definition

- Un système expert utilise des connaissances spécifiques à un domaine pour fournir des conseils ou des solutions à des problèmes.
- Des connaissances d'un domaine sont présentées dans une base de connaissances (par exemple base de règles).
- Simuler le raisonnement de l'expert humain : moteur d'inférence
- La performance d'un système expert dépend essentiellement de ses connaissances, moins du moteur d'inférence
- Un système expert doit être riche en connaissances : Knowledge is power
- Comme un expert humain, un système expert
 - se spécialise dans un domaine
 - enrichit ses connaissances avec des expériences

2.2 Domaines

- Interprétation : former des conclusion de haut niveau à partir de données brutes
- Prédiction : trouver des consséquences probables des situations données
- Diagnostique : déterminer la cause du mal fonctionnement dans des situations complexes à partir des symptômes observables.
- Configuration : construire une configuratino des composants pour atteindre des objectifs de performance tout en satisfaisant des contraintes de configuration.
- Planification : déterminer une suite d'actions pour arriver à un ensemble d'objectifs.
- Surveillance : comparer des comportements d'un système par rapport au comportement souhaité.
- Contrôle : contrôler le comportement d'un environnement complexe.
- etc.

2.3 Architecture

Base de connaissances

- Les connaissances générales et spécifiques du domaine.
- connaissance sous forme de règle Si alors pour des systèmes à base de règles.
- des éditeurs à base de connaissances permet de représenter les connaissances sous forme facile à accéder.

Moteur inférence

- effectuer le raisonnement pour tirer les conséquences impliquées par la connaissance incluse dans le système.

Interface utilisateur Le mieux est d’avoir la possibilité pour l’utilisateur d’utiliser une langue naturel. A partir du traitement du langage naturel, obtenir des connaissances ou vérifier des connaissances. Donne l’impression que l’utilisateur parle avec un humain.

Exigences

- Raisonnement correct
- Raisonnement ouvert à l’inspection
- Capacité d’explication des choix et des décisions pris

Construction Aussi des modules pour la création et la gestion d’un système expert peuvent être fournis ou réutilisés :

- Jess en Java.
- CLIPS de la Nasa en C.

La construction de la base de connaissance est la partie difficile. Elle dépend des domaines.

3 Logique propositionnelle

3.1 Definition

La logique des propositions permet d'exprimer

- des faits sur le monde
- des négations
- des conjonctions et des disjonctions
- des phrases avec conséquence logique

Une proposition est une expression (phrase) à propos du monde qui est soit vraie soit fausse.

Elements de base :

- symboles de propositions
- phrases spéciales : vrai , faux
- opérateurs

3.2 Exemple

Nous allons réutiliser le problème du Wumpus :

- Si la case (2,3) a un puits alors il y a un courant d'air dans les cases adjacentes. $P_{23} \rightarrow C_{13} \wedge C_{33} \wedge C_{22} \wedge C_{24}$
- S'il y a du courant d'air dans la case (1,2) alors il doit y avoir un puits dans une case adjacente. $C_{12} \rightarrow P_{11} \vee P_{22} \vee P_{13}$
- 1er ordre : $P(i, j) \wedge adj(i_1, j_1, i, j) \rightarrow C(i_1, j_1)$

3.3 Rappel

Voir les lois de négation, d'implication, de DeMorgan ...etc
Voir l'algorithme de résolution.

3.4 Soit

- Résolution : connaissances sous formes de clauses
- Chainages avant et arrière : connaissances sous forme de clauses de Horn (règles).

3.5