

Chapitre 4 : Systèmes experts

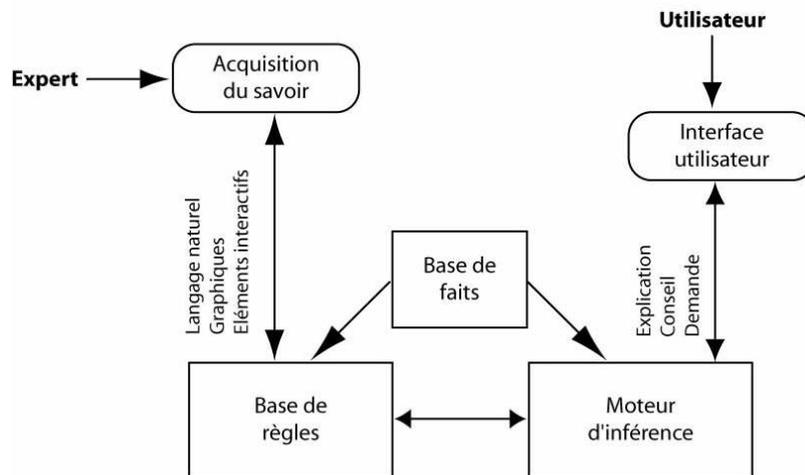
1. Introduction

La notion de systèmes experts est apparue dans les années 70 avec l'apparition du système expert célèbre MYCIN dont le but était d'aider les médecins à effectuer le diagnostic des maladies infectieuses du sang. La version de base contenait 200 règles ensuite 300 règles concernant les méningites ont été ajoutées.

Un système expert est un outil d'intelligence artificielle, capable de simuler le savoir-faire d'un expert humain dans un domaine précis, en exploitant des bases de données qui sont spécifiques à ce domaine, et en recourant à un moteur d'inférence pour simuler différents raisonnements déductifs.

2. Structure et fonctionnement d'un système expert

L'architecture d'un système expert typique est constitué de plusieurs modules qui s'interagissent :



- ✓ **L'interface utilisateur** sert à simplifier la communication, elle peut utiliser la forme question-réponse, le menu, le langage naturel etc.
- ✓ **La base de connaissances** contient les connaissances concernant la résolution du problème (base de faits et bases de règles).
- ✓ **Le moteur d'inférence** applique une stratégie de résolution en utilisant les connaissances et ceci pour en dériver une nouvelle information.
- ✓ **La base de faits** contient les données spécifiques liées à l'application traitée. Elle peut contenir aussi les solutions intermédiaires ou les conclusions partielles trouvées lors de l'inférence.
- ✓ **Le module d'explication** permet au système expert d'expliquer son raisonnement.
- ✓ **L'éditeur** permet l'édition des connaissances dans la base. Il est très important de remarquer la séparation faite entre les connaissances et l'inférence.

- **Base de faits** : Mémoire de travail, par exemple : dans le domaine médical = liste de symptômes en début de session et un diagnostic lorsque celle-ci se terminera. Par exemple
- **Base des règles** : SI condition ALORS action

SI l'animal pond des œufs et l'animal vole ALORS l'animal est un oiseau. Appelée inférence

Prémisse

Conclusion

Exemple d'un système expert animaux :

Base des règles "ANIMAUX" :	Base des faits "ANIMAUX" :
R1 : SI l'animal est oiseau et l'animal pèse moins de 10 gr ALORS l'animal est colibri.	Animal vole
R2 : SI l'animal vole et l'animal pond des œufs ALORS l'animal est oiseau	Poids animal est inférieur à 10 gr
	Animal pond des œufs

Le moteur d'inférence prend une règle et regarde dans la base de faits si les faits stockés vont rendre la prémisse de la règle vraie.

3. Domaines d'application

Les systèmes experts ont été conçus pour résoudre certains types de problèmes comme en médecine, en droit, en chimie, en éducation etc.

- Le prédiction des conséquences à partir de situations données.
- Le diagnostic d'une défaillance à partir d'un ensemble d'observations.
- La conception d'une configuration de composants à partir d'un ensemble de contraintes.
- le contrôle du comportement d'un environnement complexe.

4. Moteur d'inférences

On considère le système expert " pâtisserie "

Base de règles :	Base de faits
R1 : Si farine et beurre et œufs et sel alors pâte	• Pommes
R2 : Si pommes et sucre alors pommes sucrées	• Poires
R3 : Si pommes sucrées et pâte alors tarte aux pommes	• Abricots
R4 : Si abricots et pâte alors tarte aux abricots	• Farine
R5 : Si poires et pâte alors tarte aux poires	• Beurre
R6 : Si cerises et pâte alors tarte aux cerises	• Sucre
	• Sel

Comment exploiter ce système expert ?

- Démarche 1 : Avec les ingrédients (les faits) qu'est-ce que je peux faire ?
- Démarche 2 : Est-ce que je peux faire une tarte aux abricots ?

• **Chaînage avant : raisonnement guidé par les données**

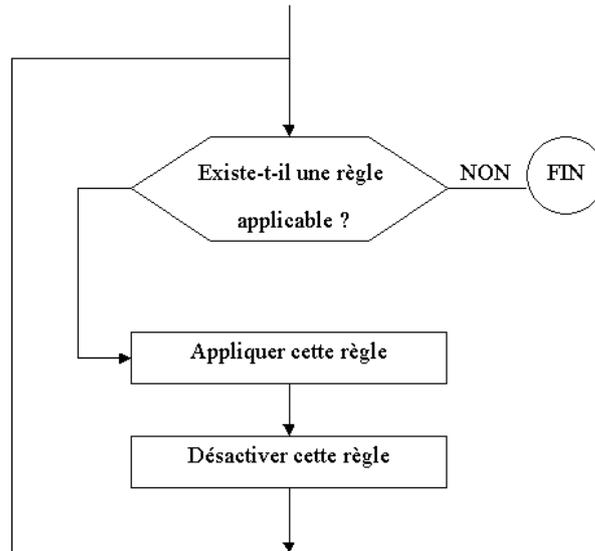
Un moteur d'inférence fonctionne dans ce mode lorsque les faits représentent des informations dont la vérité a été prouvée. Ce mode de fonctionnement va des faits vers les buts :

On part de la base de faits initiale ; on déclenche toutes les règles dont les prémisses sont satisfaites ; on ajoute les faits ainsi obtenus ; on poursuit jusqu'à "saturation" ou arrivée au but.

C'est un mécanisme d'exploitation des règles guidé par les faits. C'est la traduction du modus ponens : **si f1 est vrai et f1 -> f2 alors f2 est vrai.**

- Détecter les règles dont les prémisses sont vérifiées (filtrage)
- Sélectionner la règle à appliquer

- Appliquer la règle
- Recommencer jusqu'à ce qu'il n'y ait plus de règle applicable



- Existe-t-il une règle applicable ? : ceci consiste à trouver parmi toutes les règles celles dont la partie condition est vraie et à en choisir une à l'aide d'une fonction de choix.
- Appliquer cette règle : une fois la règle choisie, le programme exécute sa partie action ou conclusion.
- Désactiver cette règle : en logique des propositions, il est inutile d'appliquer plus d'une fois la même règle. Ainsi, les règles utilisées sont rendues inactives.
- Le but souhaité est-il démontré ? : cet algorithme suppose que l'utilisateur du moteur veut obtenir une proposition particulière, but du problème. Si ce fait vient d'être obtenu, il est inutile de poursuivre le travail.
- Dans le cas où aucun but particulier n'est demandé, le moteur fonctionne jusqu'au moment où aucune règle n'est applicable (condition d'arrêt). On dit alors que le moteur fonctionne par saturation.

Application :

Règle 1 : est-elle applicable ?

Oui (tous les faits sont dans la base des faits)

Le fait pâte est donc rajouté dans la base des faits et la règle 1 est désactivée.

Nouvelle base des faits :	Base des règles utilisables :
<ul style="list-style-type: none"> ○ Pommes ○ Poires ○ Abricots ○ Farine ○ Beurre ○ Sucre ○ Sel ○ Pâte 	<ul style="list-style-type: none"> ○ R2 ○ R3 ○ R4 ○ R5 ○ R6

Règle 2 : est-elle applicable ?

Oui

Le fait pommes sucrées est ajouté dans la base des faits et la règle 2 est désactivée.

Nouvelle base des faits :	base des règles utilisables
<ul style="list-style-type: none"> ○ Pommes ○ Poires ○ Abricots ○ Farine ○ Beurre ○ Sucre ○ Sel ○ Pâte ○ Pommes sucrées 	<ul style="list-style-type: none"> ○ R3 ○ R4 ○ R5 ○ R6

De même, les règles 3, 4 et 5 sont sélectionnées et exécutées.

Nouvelle base des faits :	base des règles utilisables
<ul style="list-style-type: none"> ○ Pommes ○ Poires ○ Abricots ○ Farine ○ Beurre ○ Sucre ○ Sel ○ Pâte ○ Pommes sucrées ○ Tarte aux pommes ○ Tarte aux abricots ○ Tarte aux poires 	<ul style="list-style-type: none"> ○ R6

Règle 6 : elle ne peut pas être exécutée.

Le fait "cerises" n'existe pas dans la base des faits.

Le mécanisme ne peut plus effectuer de déductions et le moteur s'arrête.

• **Chaînage arrière : raisonnement guidé par le but**

Le chaînage arrière est un mécanisme d'exploitation guidé par les buts. Il traduit la règle du **modus tollens** : **Si est non vrai et si p implique q alors p est non vrai.**

Examiniions l'exemple du cours : On part d'un but qu'on cherche à vérifier.

But : Tarte aux abricots.

- Considérons la règle 4 : les prémisses sont-elles vérifiées dans la base des faits ?
La première prémissse abricot est vérifiée.
La deuxième prémissse doit être vérifiée ; on vérifie s'il existe une règle qui a pour but pâte :
Oui, la règle 1.
- On considère maintenant la règle 1 : les prémisses sont vérifiées
- Donc, on peut conclure que le but **tarte aux abricots** est vérifié.

	Chaînage avant	Chaînage arrière
Points forts	Fonctionne bien lorsque le problème se présente « naturellement » avec des faits initiaux ; Produit une grande quantité de faits à partir de faits initiaux très peu nombreux ; Adapté à la planification, le contrôle, l'interprétation.	Fonctionne parfaitement lorsque le problème consiste à prouver une hypothèse ; Il est focalisé sur le but à prouver et pose donc des questions pertinentes, qui ne déroutent pas l'utilisateur ; Contrairement au chaînage avant, il recherche dans la base de connaissances les informations intéressantes pour le problème courant ; Adapté au diagnostic et à la prescription.
Points faibles	Souvent ne perçoit pas certaines évidences ; Le système peut poser de nombreuses questions, qui parfois s'avèrent non pertinentes.	Poursuit une ligne de raisonnement même s'il s'avère qu'il devrait l'abandonner pour une autre. Les facteurs de croyance et les métarègles peuvent aider à résoudre ce problème.

- **Chaînage mixte**

L'algorithme de chaînage mixte combine, comme son nom l'indique, les algorithmes de chaînage avant et de chaînage arrière.

Exercice : Vu au cours

Résoudre le problème suivant par chaînage avant et chaînage arrière :

Base de règles :	Base de faits initiale
<ul style="list-style-type: none"> • R1 : Si A et B alors C • R2 : Si F et D alors A • R3 : Si D et E alors B • R4 : Si B et D alors F • R5 : Si E et F alors D 	<ul style="list-style-type: none"> • E • F

On cherche à prouver C

Un moteur d'inférence à chaînage avant en profondeur d'abord monotone avec régime irrévocable :

Soit la BF = {B, C} et soit H le fait à établir en utilisant le chaînage avant.

Soit BR composée des règles suivantes :

- R1** : Si B, D, E alors F
- R2** : Si G, D alors A
- R3** : Si C, F alors A
- R4** : Si B alors X
- R5** : Si D alors E
- R6** : Si X, A alors H
- R7** : Si C alors D
- R8** : Si X, C alors A
- R9** : Si X, B alors D

Solution

1. Conflit = {R4, R7}, règle choisie R4 ; BF = {B, C, X} ; désactiver R4 ;
2. Conflit = {~~R4~~, R7, R8, R9}, règle choisie : R7, BF = {B, C, X, D} ;
3. Conflit = {~~R4~~, ~~R7~~, R5, R8, R9, }, règle choisie : R5, BF = {B, C, X, D, E} ;
4. Conflit = {~~R4~~, ~~R7~~, ~~R5~~, R1, R8, R9}, règle choisie : R1, BF = {B, C, X, E, F} ;
5. Conflit = {~~R4~~, ~~R7~~, ~~R5~~, ~~R1~~, R3, R8, R9}, règle choisie : R3, BF = {B, C, X, E, F, A} ;
6. Conflit = {~~R4~~, ~~R7~~, ~~R5~~, ~~R1~~, ~~R3~~, R6, R8, R9}, règle choisie : R6, BF = {B, C, X, E, F, A, H} ;

Le but **H** est établi, donc on s'arrête.

Un moteur d'inférence à chaînage avant en largeur d'abord monotone avec régime irrévocable :

Soit la BF = {B, C} et soit H le fait à établir en utilisant le chaînage avant.

Soit BR composée des règles suivantes :

R1 : Si B, D, E alors F

R2 : Si G, D alors A

R3 : Si C, F alors A

R4 : Si B alors X

R5 : Si D alors E

R6 : Si X, A alors H

R7 : Si C alors D

R8 : Si X, C alors A

R9 : Si X, B alors D

Solution

1. Conflit = {R4, R7}, on applique R4 puis R7, BF = {B, C, X, D} ;
2. Conflit = {R5, R8, R9}, on applique R5 puis R8 puis R9, BF = {B, C, X, D, E, A} ;
3. Conflit = {R1, R6}, on applique R1 puis R6, BF = {B, C, X, D, E, A, F, H} ;

Le fait **H** est établi, donc on s'arrête.

Solution du Chainage arrière

⌋ : et
— : ou

