

Université de Jijel
3^{ème} année Licence Informatique



Intelligence Artificielle

Chapitre III: Fonctionnement du moteur d'inférence

Boudebza Souâad
souad.boudebza@univ-jijel.dz

Plan

- 1 Cycle de base d'un MI
 - Phase d'évaluation
 - Phase d'exécution
- 2 Caractéristiques d'un MI
 - Mode d'invocation des règles
 - Stratégie de recherche
 - Régime de contrôle
 - Critère de monotonie
- 3 Exercices

Abréviations

MI : Moteur d'Inférence

BC : Base de Connaissance

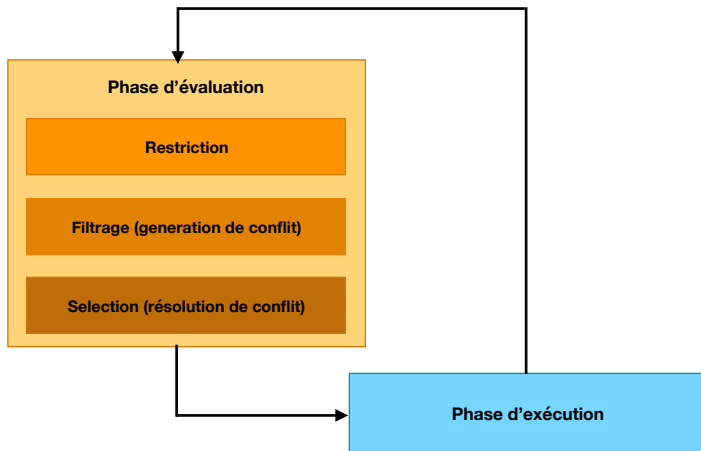
BF : Base de Faits

BR : Base de Règles

Le cycle de base d'un MI

- Au cours de son raisonnement, le moteur d'inférence (MI) enchaîne une séquence de cycles jusqu'à aboutir au résultat désiré ou jusqu'à saturation.
- Un cycle de travail comporte deux phases :
 - **Un phase d'évaluation**
 - **Une phase d'exécution**

Le cycle de base d'un MI



Le cycle de base d'un MI

Phase d'évaluation : est effectuée en 3 étapes :

- **La restriction** : C'est une étape optionnelle. Elle a pour objet de trier et de rassembler en un sous-ensemble, les faits et les règles de la BC qui méritent plus d'attention que d'autres. Elle permet de réduire le temps du traitement.
 - On appelle F1 le sous-ensemble BF, et R1 le sous-ensemble de BR.
 - F1 et R1 sont déterminés à partir de l'état présent (cas du chaînage avant) ou de l'état passé (cas chaînage arrière) de la BF et de la BR.
 - A ce stade, certaines stratégies peuvent être mises en œuvre pour sélectionner de préférence certains faits ou règles plutôt que d'autres
Exemple : répartir les faits et les règles en classes particulières, les indexer.

Le cycle de base d'un MI

Phase d'évaluation : est effectuée en 3 étapes :

- **Filtrage (pattern-matching)** : encore appelée génération de conflits. Dans cette étape, le MI sélectionne les règles qui peuvent effectivement être déclenchées ;
 - Un sous-ensemble R2 de R1 rassemble les règles jugées compatibles avec F1.
 - R2 est appelé l'ensemble de conflits.
 - R2 contient toutes les règles dont le ou les faits situés :
 - dans la partie prémisse correspond aux faits stockés dans la base des faits (cas chaînage avant).
 - dans la partie action correspondant aux faits (qui sont autant d'objectifs) stockés dans la base des faits (cas chaînage arrière).
 - dans la partie prémisse ou action correspondant aux faits stockés dans la base des faits (cas chaînage mixte).

Le cycle de base d'un MI

Phase d'évaluation : est effectuée en 3 étapes :

- **Sélection (Résolution de conflits)** : choisir la règle à déclencher.
 - La situation où plus d'une règle est déclenchable dans un même cycle.
 - Le moteur détermine l'ensemble R3 (une ou plusieurs règles parmi l'ensemble R2) pour les déclencher.
 - Si le moteur d'inférence ne trouve aucune règle à déclencher celui-ci s'arrêtera et demandera des informations supplémentaires à l'utilisateur (dans le cas de chaînage arrière).

Le cycle de base d'un MI

Phase d'évaluation : est effectuée en 3 étapes :

- **Sélection (Résolution de conflits)** : Plusieurs stratégies de sélection :
 - la première (selon l'ordre dans la BR) qui est filtrée ;
 - la règle la plus prioritaire où la priorité est définie par le programmeur ;
 - la règle dont la partie condition détaille le mieux la base des faits ;
 - une nouvelle règle qui vient d'être déclenchée, et qui n'était pas apparue précédemment ;
 - par un ordre tout simplement arbitraire ;
 - par l'exploration de toutes les règles applicables en parallèle ;
 - la règle se référant à l'élément le plus récent ajouté à la base des faits ;
 - la règle qui est jugée de par sa forme, la moins complexe ;
 - la règle qui semble d'emploi la moins coûteuse ;
 - ou une combinaison des approches précédentes.

Le cycle de base d'un MI

Phase d'exécution

- Déclenchement de l'ensemble de règles sélectionnées : Mise en œuvre les actions définies par les règles R3.
- Les nouveaux faits résultants seront ajoutés à la base de faits :
 - Les faits sont extraits de la partie action dans un contexte de chaînage avant, de la partie prémisses dans un contexte de chaînage arrière.
- Si le but est atteint la recherche est arrêtée, sinon, un nouveau cycle est déclenché

Le cycle de base d'un MI

Phase d'exécution : Les stratégies de mise en œuvre peuvent être différents. Généralement cette phase consiste à ajouter des faits à la base de fait, mais il est aussi possible d'en retirer d'autres ou de déclencher des réactions ou exécuter des tâches supplémentaires :

- si R3 est vide certains moteurs (avec régime irrévocable) s'arrêtent. certains d'autres (avec régime par tentative) opèrent des retours arrière pour remettre en cause certains déclenchements des règles.

Le cycle de base d'un MI

Le moteur d'inférence exécute ces différentes phases de façon cyclique jusqu'à une condition d'arrêt soit vérifiée :

- Un objectif atteint
- Étape de filtrage ne fournit aucune règles potentiellement applicable
- Épuisement normale des règles

Caractéristiques d'un MI

La programmation d'un MI nécessite la sélection d'un ensemble de critères :

- Mode d'invocation des règles
- La stratégie de recherche
- Régime de contrôle
- Critère de monotonie

Mode d'invocation des règles

Chaînage Avant (forward chaining)

Principe

- On part à partir des données initiales et on essaye d'en déduire de nouvelles
- On s'arrête quand aucune règle n'est applicable (ou si le but est atteint)

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

Mode d'invocation des règles

Chaînage Avant (forward chaining)

Raisonnement : dirigé par les faits, déductif

Paramètres d'entrée : la base de règle (BR), la base de faits (BF), [le but (F)]

Déclencheur : partie gauche de la règle (hypothèses)

Mode d'invocation des règles

Chaînage Avant - Algorithme

Donnée : BF, BR, F

Début

Phase de filtrage => Détermination des règles applicables

Tant que (ensemble de règles applicables n'est pas vide ET que le problème n'est pas résolu) Faire

Phase de choix => Résolution des conflits

Appliquer la règle choisie (exécution)

Modifier (éventuellement) l'ensemble des règles applicables

Fin faire

Fin

Mode d'invocation des règles

Chaînage Avant

Remarques Importantes :

- Une règle n'est déclenchée qu'une seule fois
- L'algorithme termine toujours
- Les conclusions des règles peuvent être des faits négatifs. Différentes situations en cas d'utilisation de faits négatifs :
 - $F \in BF$ fait établi
 - $\neg F \in BF$ la négation du fait est établi
 - $F \notin BF$ et $\neg F \notin BF$ Aucune information ni le fait ni sa négation est établi
 - $F \in BF$ et $\neg F \in BF$ la base de faits est incohérente

Mode d'invocation des règles

Chaînage Arrière (backward chaining)

Principe

- On cherche à prouver un but (idem 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

Avantage

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

Inconvénients

- Le risque de bouclage.

Mode d'invocation des règles

Chaînage Arrière (backward chaining)

Raisonnement : dirigé par les buts, inductif

Paramètres d'entrée : la base de règles, la base de faits, le but à prouver

Déclencheur : partie droite de la règle (conclusion)

Mode d'invocation des règles

Chaînage Arrière - Algorithme

Début

Phase de filtrage

Si l'ensemble des règles sélectionnées est vide

Alors questionner l'utilisateur

Sinon

Tant que (le but n'est pas résolu ET qu'il reste des règles
sélectionnées) Faire

Phase de choix

Ajouter les sous-buts (partie gauche de la règle choisie)

Si un sous-but n'est pas résolu

Alors mettre le sous-but en but à résoudre

Fin faire

Fin

Mode d'invocation des règles

Quel chaînage utiliser ?

Les Caractéristiques du problème vont conditionner le choix du chaînage.

- Les faits sont peu nombreux ou que le but est inconnu \Rightarrow chaînage avant.
- Les cas où les buts sont peu nombreux ou précis \Rightarrow chaînage arrière.

Mode d'invocation des règles

Chaînage Mixte

Principe

- Combiner les deux chaînages avant et arrière
 - Chaînage avant : obtenir les informations suffisantes pour émettre une hypothèse
 - Chaînage arrière : vérifier l'hypothèse émise lors du chaînage avant
- Une règle peut être utilisée en chaînage avant ou en chaînage arrière

Avantage

- Optimiser la recherche et améliorer le raisonnement

Mode d'invocation des règles

Chaînage Mixte - Algorithme

Données : F (à déduire)

DEBUT

 TANT QUE (F n'est pas déduit mais peut encore l'être) FAIRE

 Saturer la base de faits par chaînage AVANT (c'est-à-dire,
 déduire tout ce qui peut être déduit)

 Chercher quels sont les faits encore éventuellement déductibles

 Déterminer une question pertinente à poser à l'utilisateur et
 ajouter sa réponse à la base de faits

 FIN DU TANT QUE

FIN

La stratégie de recherche

Au cours des cycles de recherche d'un MI, on développe un arbre de recherche dans lequel chaque niveau correspond à l'ensemble des règles applicables (ensemble de conflits).

Chaque règle déclenchée crée une nouvelle situation et de nouvelles règles à invoquer.

Deux principales stratégies de recherche se présentent :

- **Stratégie en largeur d'abord**
- **Stratégie en profondeur d'abord**

La stratégie de recherche

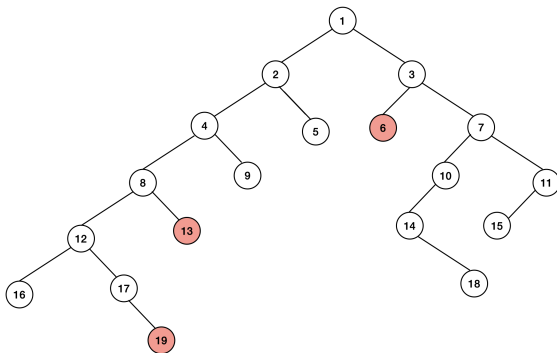
Largeur d'abord

- Le principe est de partir de la racine et d'énumérer l'ensemble de ses descendants directs puis réitérer le processus sur les descendants eux-mêmes. On effectue ainsi un parcours "de gauche à droite et de haut en bas".
- On parcourt à l'horizontal les nœuds d'un même niveau, avant de considérer les nœuds du niveau inférieur.

La stratégie de recherche

Largeur d'abord

- **Exemple** : soit l'arbre de recherche ci-dessous. Les noeuds buts sont colorés en rouge. Les étiquettes des noeuds correspondent à l'ordre de prise en considération.



- La recherche dans ce cas va se perpétuer jusqu'à ce que l'on obtienne un noeud but. Ainsi, on s'arrêtera après 6 itérations.

La stratégie de recherche

Largeur d'abord

• Avantages

- Si on trouve un chemin vers un but, on est certain qu'il s'agit du chemin le plus court en termes de noeuds.
- S'il existe une solution au problème, l'algorithme la trouvera.

• Inconvénients

- l'efficacité de l'algorithme dépend de la position du but dans l'arbre : Si le but se situe profondément dans l'espace de recherche, cette stratégie n'est pas payante.
- Les interactions avec un utilisateur sont difficiles en raison de l'exploration de chemins nombreux (manque de focalisation).

La stratégie de recherche

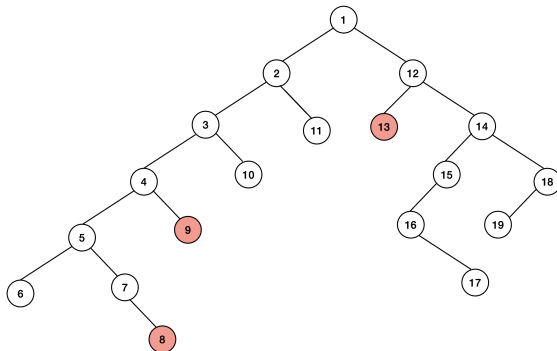
Profondeur d'abord

- Le principe est de descendre le plus profondément dans l'arbre avant de se déplacer en largeur. On effectue ainsi un parcours "de haut en bas et de gauche à droite".
- Le moteur tente de toujours vérifier complètement la première prémisse de la règle avant de commencer la vérification de la deuxième prémisse
- Le retour arrière dans le cas où la recherche en profondeur échoue sera appelé : **backtracking**

La stratégie de recherche

Profondeur d'abord

- **Exemple** : Prenant l'arbre de recherche précédente. Les noeuds buts sont colorés en rouge. Les étiquettes des noeuds correspondent à l'ordre de prise en considération par le mode de recherche en profondeur.



- La recherche dans ce cas s'arrêtera après 8 itérations.

La stratégie de recherche

Profondeur d'abord

• Avantages

- Lorsqu'il est connu que l'espace de recherche est profond, la stratégie en profondeur d'abord est un bon choix.
- De plus, cette stratégie se focalise sur une solution et est donc à même d'être comprise par un utilisateur qui interagit avec le système, car les questions sont relatives à un même chemin.

• Inconvénients

- L'efficacité de ce mode dépendant de la position des buts dans l'arbre. En effet si les buts se trouvent à droite, la recherche sera longue. De plus, s'il existe une branche infinie (sans solution et non terminante) avant une branche contenant un but, la recherche cycle indéfiniment.

Régime de contrôle

Le régime de contrôle d'un MI peut être :

- **Irrévocable**
- **Par tentative**

Régime de contrôle

Irrévocable

- Une stratégie est dite irrévocable si les applications des règles choisies lors de la résolution de conflits ne sont jamais remises en cause.
 - Le moteur d'inférence n'opère pas de retour-arrière (backtracking).
 - S'il n'y a plus de règles à appliquer. Le MI s'arrête et signale un échec sans faire retour en arrière (pas de backtracking).
- Le régime irrévocable convient aux systèmes modélisés en logique monotones. Par exemple, dans des situations impliquant des transitions irréversibles : une fois la valeur du fait changée, on ne peut pas revenir en arrière. Par exemple, vider la casserole pour un robot cuisinier.

Régime de contrôle

Par tentative :

- Une stratégie par tentative peut remettre en cause l'application d'une règle si ce choix débauche sur un échec.
- Le moteur d'inférence opère alors un retour en arrière (backtrating) pour tenter de vérifier une règle écartée au préalable.
- On doit pouvoir maintenir un arbre des états dépendant de l'ordre des règles appliquées.
- Il est à noter que la plupart des moteurs d'inférences procédant par chaînage arrière utilisent une stratégie par tentatives alors que leurs homologues, opérant en chaînage avant, optent plutôt pour une stratégie irrévocable.

Critère de monotonie

Un moteur d'inférence MI peut être :

- **Monotone**
- **Non monotone**

Critère de monotonie

Monotone :

- L'état de véracité de connaissance ne change pas (statique), ç.a.d, les faits produits ne sont pas remis en cause.
- En régime monotone, le MI ne fait qu'ajouter des faits à la BF et n'élimine jamais une règle de BR
- Les systèmes de diagnostic sont le plus souvent monotones.
- Exemple : La plupart des diagnostics médicaux appellent un fonctionnement monotone. En effet, à moins d'une maladie foudroyante les symptômes du patient avant la consultation du médecin sont les mêmes qu'après la consultation

Critère de monotonie

Non monotone :

- L'état de véracité de la connaissance change (dynamique).
- Le MI dispose de primitives pour modifier la base de connaissance.
- Le MI peut en cas de retour arrière par exemple retrancher de la BF un fait précédemment ajouté. Les situations rencontrées sont les suivantes :
 - En cas de backtracking : les faits déduits d'une règle qui n'a pas abouti seront supprimés à l'occasion de la remise en cause de la règle en question et le choix d'une autre en faisant un retour-arrière.
 - Certains faits peuvent aussi être retirés de la base de faits s'ils présentent des contradictions avec d'autres ou dans le cas des systèmes de diagnostic interactif.
- La monotonie concerne aussi les règles. Un système non-monotone peut retirer des règles en constatant des contradictions avec d'autres. Par exemple dans les situations où l'on exploite des règles expertes simplement plausibles dites "règle par défaut".
 - Exemple : "Les métaux à température normale sont solides sauf le mercure" ; "Les métaux sont solides par défaut".

Exercice 01

Soit la base de règle suivante :

R1 : $A, B \Rightarrow F$

R2 : $F, H \Rightarrow I$

R3 : $D, H, G \Rightarrow A$

R4 : $O, G \Rightarrow H$

R5 : $E, H \Rightarrow B$

R6 : $G, A \Rightarrow B$

R7 : $G, H \Rightarrow P$

R8 : $G, H \Rightarrow Q$

R9 : $D, O, G \Rightarrow J$

Soit la base de fait initial suivante : D, O, G.

Prouver le but I en utilisant :

1. un chaînage avant en profondeur d'abord monotone avec régime irrévocable et une stratégie de résolution de conflits qui privilège la règle ayant le plus de prémisses (en cas d'égalité la priorité est à la règle ayant le plus petit indice).
2. un chaînage avant en largeur d'abord monotone avec régime irrévocable. Utilisez une stratégie de résolution de conflit qui explore toutes les règles applicables en parallèle.
3. un chaînage arrière en profondeur d'abord non monotone avec un régime de contrôle par tentative. Utilisez la même stratégie de résolution de conflits précédente.

Corrigé Exercice 01

Corrigé sur le tableau.

Exercice 02

Soit la base de règle suivante :

R1 : $B, A \Rightarrow C$

R2 : $B, E \Rightarrow F$

R3 : $A, C, F \Rightarrow K$

R4 : $C, E \Rightarrow G$

R5 : $D \Rightarrow B$

R6 : $K, H, D \Rightarrow G$

R7 : $C \Rightarrow F$

R8 : $D \Rightarrow C$

R9 : $A, G, B \Rightarrow C$

R10 : $B, D \Rightarrow G$

Soit la base de fait initiale : A, D

1. Quels sont les faits déduits en :

1.1 Chaînage avant avec stratégie de recherche en largeur d'abord.

1.2 Chaînage avant avec stratégie de recherche en profondeur d'abord et une stratégie de résolution de conflits qui privilège la règle ayant le moins de prémisses (en cas d'égalité la priorité est à la règle ayant le plus petit indice).

2. Prouver que G se déduit par chaînage arrière avec stratégie de recherche en profondeur d'abord et un régime de contrôle par tentative. On prend comme critère de choix la règle ayant le plus petit indice.

Corrigé Exercice 02

Corrigé sur le tableau.

Questions ?