

MICREXP

Sujet d'intelligence artificielle
3ème année
Christian JOUSSELIN

Introduction

Aujourd'hui les systèmes experts sont passés des laboratoires de recherche aux centres de développement et les ingénieurs doivent se préparer à l'utilisation de ces nouveaux outils. Le but de cet exercice est de vous faire comprendre comment fonctionne un système expert en réalisant un moteur "en chaînage avant" et un moteur "en chaînage arrière" sur une base de règles en logique d'ordre 0 écrits en LISP: **MICREXP** pour MICro système EXPert.

Le système MICREXP

MICREXP n'est pas réellement un système expert mais plutôt une somme de deux moteurs "chaînage avant" et "chaînage arrière" et d'une syntaxe de description des règles en logique des propositions. Les objets du système sont:

Une **proposition** qui est représentée par un atome. Une proposition peut avoir les propriétés: Vraie, Fausse, Recherchée et en_conclusion. En_conclusion a pour valeur la liste des règles dans lesquelles la proposition intervient dans la conclusion.

Une **règle** qui est représentée par une liste de la forme:

(Nom_de_la_règle SI Liste_des_premisses ALORS Liste_des_conclusions)

Une règle peut avoir la propriété : **Appliquée**, en fait, c'est l'atome Nom_de_la_règle qui a la propriété Appliquée.

La base de règles est stockée dans la *C-VAL* de l'atome **REGLES** par l'utilisateur du système, et ceci dans le fichier **REGLES**. La base de propositions est extraite de la base des règles par la fonction *Le_Lisp*: "extraire_de" qu'on applique autant de fois qu'il y a de règles dans **REGLES** avec *MAPC*. La base de propositions est stockée dans la *C-VAL* de l'atome **PROPOSITIONS** par la fonction "construire_avec".

Les principales fonctions Lisp de MICREXP

Initialiser : permet d'extraire la base des propositions de la base des règles si cela n'est pas déjà fait et d'initialiser les propriétés des règles et des propositions.

Déduire_de : constitue le moteur en chaînage arrière. Après l'initialisation demande dans quel mode de déduction il doit travailler (déterministe ou non):

Mode déterministe : si les prémisses d'une règle n'ont pas pu être prouvées Vraies alors la conclusion est mise à Fausse.

Mode non déterministe : si les prémisses d'une règle n'ont pas pu être prouvées Vraies alors la conclusion est laissée indéterminée.

Puis par "tester" regarde si la proposition sur laquelle est faite la déduction est déjà recherchée ou terminale pour inférer ou non cette proposition. Enfin présente les résultats de la déduction.

Induire_de : constitue le moteur en chaînage avant. Après l'initialisation met à Vraie les propositions de la liste à partir de laquelle il va faire son induction puis boucle tant qu'il reste des règles à appliquer. Enfin présente les résultats de l'induction.