

DUELLISTE

Date : 15 septembre 1995

Auteur : C.JOUSSELIN

Sujet d'intelligence artificielle

DUELLISTE
CORRIGE

3^{ème} Année SIT Centrale 95/96

DUELLISTE

"Jeu: Dépense d'activité physique ou mentale qui n'a pas de but immédiatement utile, ni même de but défini, et dont la seule raison d'être, pour la conscience de celui qui s'y livre, est le plaisir même qu'il y trouve."

Vocabulaire technique et critique de la philosophie.

Introduction

Le but de cet exercice est de vous faire comprendre comment fonctionne un mécanisme de résolution de problème de duels.

Le sujet consiste à réaliser en PROLOG (Xilog de Bull) un programme général de résolution de problèmes de duels décrit par un interface standardisé et les jeux "Allumettes" et "NIM" réalisés avec cet interface.

Spécifications du programme de résolution

Spécifications générales

Un duel met en présence deux joueurs en face d'une situation complètement connue de chacun d'eux. Tour à tour, chacun porte un coups à son adversaire, en respectant un ensemble de règles. Au moins une des règles doit indiquer le ou les états du jeu qui ne permettent plus de porter un coup parce que le joueur a gagné ou que l'autre a perdu.

Un duel se termine soit parce qu'un des joueurs a gagné ou a perdu, ou bien, lorsqu'un joueur n'a plus la possibilité de jouer, il a alors perdu. Les cas de "non fin de jeu", suite à des ensembles de coups conduisant à une périodicité d'états (match nul) ne sont pas traités.

Le mécanisme de résolution est basé sur la remarque suivante: l'état du jeu est potentiellement gagnant pour le joueur s'il existe parmi les états qui peuvent lui succéder, après avoir joué, un état perdant pour l'adversaire.

A chaque état transformé par une action (un coup) correspond un nouvel état et une valeur binaire dite récompense qui vaut:

- 1: s'il existe au moins une action qui conduit l'adversaire à perdre. L'état est dit potentiellement gagnant.
- 0: s'il n'existe aucune action pouvant conduire l'adversaire, s'il joue intelligemment, à perdre. L'état est dit potentiellement perdant.

le couple [Récompense, Action] est appelé solution.

Un état terminal gagnant est décrit dans le jeu par $termine(Etat,[1,nil])$ et un état terminal perdant par $termine(Etat,[0,nil])$.

Au cours de la recherche d'un coups gagnant, le mécanisme de résolution va tester un certain nombre d'action face à un état, l'apprentissage consiste à mémoriser dans les faits "base" les informations Etat et [Récompense,Action], pour ne plus à les tester si l'état se représente tel quel.

Pour simplifier le programme:

la recherche est exhaustive et ne repose pas sur une heuristique avec fonction de coût, c'est pourquoi, la valeur de récompense est binaire. De ce fait ce programme ne pourra pas résoudre (temps de réponse trop long ou mémorisation d'états intermédiaires trop important) des duels complexe comme le Go, les Dames ou les échecs.

le joueur commencera toujours la partie.

Visualisation

L'écran est constituée de 5 fenêtres:

La fenêtre: "Menu" indique les choix possibles: menu Duel ou menu Jeu.

La fenêtre: "Surface de jeu" permet d'afficher l'état du jeu par le prédicat "Affiche". La fenêtre a pour dimension: 76 colonnes x 15 lignes.

La fenêtre "Joueur" permet de saisir le coup du joueur et d'afficher le résultat de la partie quand c'est au joueur de jouer. Quand le coup du joueur n'est pas possible, le programme propose un conseil (travail pour le compte du joueur comme expert).

La fenêtre "Expert" permet de voir les choix effectués par le programme et d'afficher le résultat de la partie quand c'est à l'expert de jouer.

La fenêtre sans nom permet de saisir le nom du jeu et la situation initiale du jeu. Elle sert aussi de zone d'information pour le programme de résolution.

Apprentissage

L'apprentissage consiste mémoriser, dans les faits "base(Etat,Solution)", les recherches de solution effectuées lors des coups précédents.

Sauvegarde sur fichier

La sauvegarde de l'apprentissage, dans le fichier de jeu, est automatique lors de la sortie du menu "Jeu".

Impression

L'impression sur papier du se fait par appuis sur la touche "hard_copy d'écran".

Menu général

Le menu général "DUEL" sert à sélectionner les différentes commandes de ce programme de résolution:

Touche F1: Affiche le texte d'aide:

"DUELLISTE: Mini système expert adapté aux duels. Appuyer sur une touche."
sur la surface de jeu.

Touche F2: Appel le jeu: Interroge pour connaître le nom du fichier de jeu puis lance le menu "jeu".

Touche F10: Sortie du programme DUEL

Le menu "Jeu" sert à initialiser et lancer une partie, les différentes commandes sont:

Touche F1: Affiche le texte R décrivant la règle du jeu contenu dans règle(R).

Touche F2: Initialise le jeu: Interroge pour connaître la situation initiale puis affiche cette situation sur la "Surface de jeu".

Touche F3: Lance le jeu: Fait jouer alternativement le joueur et l'expert jusqu'à la fin de la partie.

Touche F10: Sortie du menu "Jeu".

Spécification de l'interface standardisé de jeux

L'interface "standardisé" d'un jeu qui peut être résolu par le résolveur de duel est très simple mais très puissant. Il est constitué par l'ensemble des six prédicats:

règles(R) : Règles du jeu (en 1 exemplaire)

où R est une liste de chaînes de caractères décrivant les règles du jeu spécifié pour servir d'aide en ligne entre deux parties.

init(S,E) : Prédicat d'initialisation (en 1 exemplaire)

où S est la situation initiale saisie par l'utilisateur du jeu (lecture faite par le prédicat read),
où E est l'état spécifié par le jeu et qui sera mémorisé dans le fait "état".

termine(E,[N,A]) : Axiomes du jeu (peut être en plusieurs exemplaires)

ces prédicats sont la plupart du temps utilisés pour ne décrire que les situations terminales du jeu
où E est l'état, N vaut 0 si la situation est perdante et 1 si elle est gagnante et A est l'action associée (vaut nil si la situation est terminale).

possibilités(E,L) : Prédicat décrivant la liste des possibilités en fonction de l'état

où L la liste des actions possibles en fonction de l'état courant E.

modifie(E1,A,E2) : Prédicat de modification d'état

où E2 est l'état du jeu après l'action A sur l'état E1.

affiche(E) : Prédicat d'affichage du jeu dans la fenêtre "Surface de jeu"

où E est l'état du jeu

Conception du programme de résolution

La conception du programme est faite de telle sorte qu'il n'y ait pas de différence entre le joueur et l'expert sauf pour le prédicat "act" qui détermine l'action à effectuer.

Le prédicat "coups" détermine l'ordre de départ du jeu: d'abord le joueur puis l'expert.

Le prédicat "jouer" liste les possibilités suivant l'état du jeu, vérifie à l'aide du prédicat "encore" si le jeu n'est pas dans un état terminal, demande une action à celui qui doit jouer, affiche la nouvelle situation et actualise l'état du jeu.

Le prédicat "act" demande au joueur d'entrer son coup et à l'expert de lancer le prédicat de résolution "expert".

Le prédicat "a" lit et teste si la lecture est une action possible en fonction de l'état du jeu, propose un conseil dans le cas contraire et appelle le programme de résolution si un conseil est demandé.

Le prédicat "expert" commence par tester si la situation est terminale ou déjà apprise sinon appelle le prédicat "expert1" qui correspond au mécanisme de résolution de duel. Ce dernier liste toutes les actions possibles du jeu en l'état et les teste une à une à l'aide du prédicat "t_act".

Le prédicat "t_act" simule la modification de l'état du jeu en appliquant l'action sous test et appelle récursivement l'expert en cherchant une situation perdante pour l'adversaire.

Spécification du jeu d'allumettes

Le jeu d'allumettes se joue avec des allumettes (pour l'exercice nous nous limiterons à $N < 20$). Tour à tour les joueurs jouent en ôtant du tas d'allumettes 1, 2 ou 3 allumettes; tout autre nombre est interdit. Le gagnant est celui qui prend la dernière allumettes.

Spécifications du jeu de NIM

Le jeu de NIM se joue avec des allumettes (nombre $N > 3$, pour l'exercice nous nous limiterons à $N < 20$). Tour à tour les joueurs jouent en ôtant du tas d'allumettes 1, 2 ou 3 allumettes pour le premier coups et ensuite on ne peut prendre le même nombre d'allumettes que le joueur précédent. A chaque coup on n'a donc que 2 possibilités, sauf pour celui qui commence. Le gagnant est celui qui prend la dernière allumette.

Sources prolog du programme DUEL

Sources prolog des jeux d'allumettes et NIM

Exemples de duels avec le jeu d'allumettes

Exemples de duels avec le jeu de NIM