

Projet QuoridorIA

INFO B317 Techniques d'IA

J.-M. Jacquet
Faculté d'informatique, Université de Namur

Année académique 2019-2020

1 Introduction

Quoridor est un jeu de stratégie développé par la société Gigamic. Dans le cadre du projet, nous considérons une légère variante, appelée ci-après QuoridorIA.

QuoridorIA, tout comme Quoridor, se déroule sur une grille de 9 colonnes et de 9 lignes. Quoridor peut se jouer entre deux joueurs ou quatre joueurs. Pour QuoridorIA, nous considérons la version la plus complexe de 4 joueurs. Deux des joueurs sont supposés être des humains et deux autres des robots munis d'intelligences artificielles, potentiellement différentes pour les deux robots.

Chacun des joueurs (humain ou robot) est représenté par un pion de couleur différente. Nous conviendrons que les robots utilisent les pions de couleur jaune et rouge et que les humains utilisent les pions de couleur verte et bleue. Tout comme dans Quoridor, chaque joueur place son pion au début du jeu au centre d'un des côtés de la grille. Par convention, nous représentons le jeu en plaçant le pion rouge en bas de la figure et en numérotant les colonnes de A à I de gauche à droite et les lignes de 1 à 9 en partant du bas vers le haut. Par convention aussi, le pion bleu sera placé en haut en face du pion rouge et les pions jaune et vert respectivement sur les bords gauche et droit. La figure 1 représente le début du jeu.

L'objectif du jeu est pour chaque joueur d'amener son pion sur une des cases du côté opposé. Le déplacement des pions est toutefois freiné par des barrières placées comme indiqué ci-après.

2 Règles du jeu

2.1 Ouverture du jeu

Chaque joueur dispose de 5 barrières. Par convention, le joueur du pion bleu commence. Il est suivi après par le joueur du pion rouge, pour celui du pion vert, puis celui du pion jaune.

2.2 Déroulement d'une partie

A tour de rôle, selon l'ordre du début de la partie, chaque joueur choisit de déplacer son pion ou de poser une de ses barrières. Lorsqu'il ne possède plus de barrières, le joueur est obligé de déplacer son pion.

2.3 Pose des barrières

Chaque barrière a une longueur de deux cases. Elle se pose entre deux blocs de deux cases, soit verticalement, soit horizontalement. Une fois posée, il n'est pas possible de la déplacer.

La pose de barrières permet à un joueur de se créer son propre chemin ou de ralentir la progression du pion d'un autre joueur. Il est toutefois interdit de fermer totalement l'accès à la ligne de but d'un joueur.

2.4 Déplacement des pions

Les pions se déplacent d'une case horizontalement ou verticalement, en avant ou en arrière. Les barrières doivent être contournées. Lorsque son pion fait face à un autre pion, le joueur peut sauter au-dessus sauf s'il

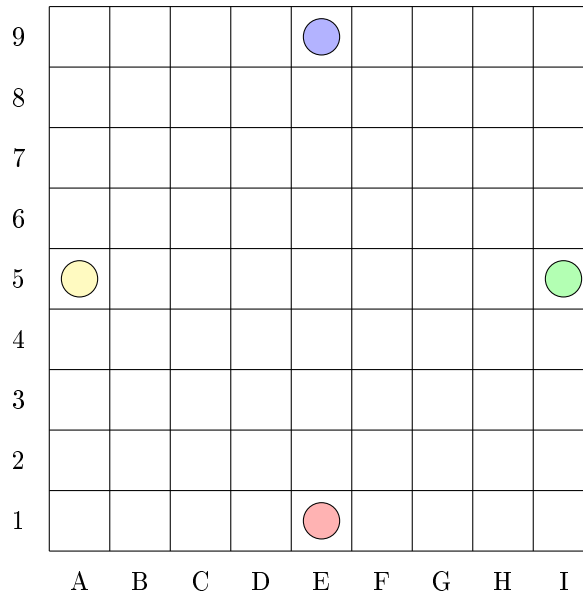


FIGURE 1 – Jeu Quoridor en début de partie. Les pions jaune et rouge sont déplacés par des robots munis d’une intelligence artificielle. Les pions bleu et vert sont manipulés par des êtres humains.

franchit une barrière ou doit sauter au-dessus d’un deuxième pion. Au cas où une barrière est située derrière le pion sauté ou au cas où un deuxième pion se trouve derrière le pion sauté, le joueur peut choisir de sauter en diagonale soit à gauche soit à droite du (premier) pion sauté.

2.5 Fin de la partie

Le premier joueur qui a atteint une des 9 cases de la ligne opposée à sa ligne de départ gagne la partie.

2.6 Vidéo

La vidéo disponible à l’adresse <https://boardgamestories.com/learn-to-play/quoridor-how-to-play/> explicite très bien le déroulement du jeu Quoridor.

3 Projet

L’objectif du projet est de développer un logiciel permettant à un utilisateur de jouer à tour de rôle à la place des deux joueurs humains et de faire jouer les robots. Ceci bien entendu en fonction de l’ordre de jeu mentionné ci-dessus.

Le projet est à articuler en différentes phases :

1. la construction d’une interface de type web représentant le jeu courant et permettant de lire des instructions de déplacement de pions ou de placement de barrières
2. la construction en Prolog d’un bot capable de répondre à des questions élémentaires sur le jeu
3. l’élaboration de modules d’intelligence artificielle pour guider les robots.

3.1 Interface de type web

Le choix de la conception de l’interface est laissé à l’imagination des étudiants. Il est conseillé de la concevoir comme une page web de manière à pouvoir bénéficier des nombreuses bibliothèques javascript/ccs mais aussi de manière à connecter facilement l’interface aux développements en Prolog via les websockets. Pour ce dernier point, nous suggérons de consulter le tutorial disponible à la page <https://gist.github.com/willprice/684291f147151db86f531fdec31b36be>



FIGURE 2 – Prototype d'interface

Un premier prototype de cette interface est représenté à la figure 2. Il fait apparaître en haut le jeu dans son état courant, au milieu une boîte de dialogue permettant de donner des directives de jeu et en bas une boîte permettant de questionner le bot.

Les directives de jeu seront données en référant les cases de la grille selon les coordonnées induites par les lettres et les chiffres. Dans ce format la case occupée par le pion rouge est E1 et celle occupée par le pion jaune est A5.

Les mouvements des pions seront donnés en indiquant la couleur du pion suivie de la case où il se déplace. Bouger le pion rouge d'une case vers le haut est ainsi dénoté par **rouge-E2**.

La pose d'une barrière est dénoté en donnant la case située au nord-ouest de la barrière ainsi que par sa direction (horizontale ou verticale). La pose d'une barrière verticale en bas de la grille juste à la gauche du pion rouge est ainsi noté **mur-D2-v** alors que la pose d'une barrière horizontale juste au-dessus du pion jaune et cadrée à gauche est dénoté par **mur-A6-h**.

3.2 Bot explicateur du jeu

L'objectif du bot est de permettre à l'utilisateur de poser des questions sur le déroulement du jeu ainsi que d'obtenir des conseils sur les coups à jouer. Dans ce cadre, il est demandé de construire un prototype capable des dialogues suivants. D'autres dialogues imaginés par l'étudiant sont, bien entendu, aussi les bienvenus.

Bonjour, je suis QBot, le bot explicateur de QuoridorIA.
En quoi puis-je vous aider ?

Vous : Qui commence le jeu ?

QBot : C'est au pion bleu de commencer. Puis aux pions rouge,
vert et jaune.

Vous : Combien de barrieres ais-je au debut du jeu ?

QBot : 5

Vous : Puis-je deplacer une barriere placee ?

QBot : Non

Vous : Puis-je sauter au-dessus d'un pion ?

QBot : Oui, s'il n'est pas suivi d'un autre pion ou d'une barriere.

Vous : Puis-je placer une barriere ou je veux ?

QBot : En principe oui mais vous ne pouvez pas enfermer un pion adverse.

Vous : Je joue avec le pion bleu. Quel coup me conseillez-vous ?

QBot : bleu-E6

Le bot est à réaliser en Prolog sur base du prototype proposé dans le fichier `qbot-elm.pl`.

Par facilité, on supposera que les questions posées par l'utilisateur se limitent à une ligne de texte (pouvant cependant remplir plusieurs lignes d'écran). Ce faisant, il est possible de transformer ces questions en une liste de mots en utilisant le prédicat `read_atomics/1` développé par M. Covington. Le squelette proposé inclut déjà ce prédicat. Il est à noter qu'il transforme tous les mots en minuscules et ne retient aucun signe de ponctuation.

En sens inverse, un prédicat `ecrire_reponse/1` a été écrit par l'équipe enseignante. Il prend comme argument une liste de listes de mots écrits en minuscule ainsi que des suites de caractères, y compris des signes de ponctuation ('.' et ','). Le prédicat écrit à l'écran ces listes de mots comme des lignes de texte, en insérant les espaces nécessaires et en transformant les premières lettres de mots en majuscule quand nécessaire.

De plus, le prédicat principal `quoridora/0` a été écrit en fin de fichier. Il fait apparaître une boucle – grâce au prédicat `repeat` (qui réussit toujours, en particulier, en cas de backtracking) – dont la terminaison est obtenue lorsque l'utilisateur introduit `Fin`.

Dans ce contexte, la tâche consiste à coder le prédicat `produire_reponse/2`, situé au début du squelette de programme. Au vu d'une liste de mots `L_mots` retournée par le prédicat `read_atomics/1`, il doit retourner une liste de listes de mots `L_lignes_reponse` à produire comme réponse par le prédicat `ecrire_reponse/1`.

Pour ce faire, il est suggéré de baser votre travail en raisonnant sur les mots clefs (plutôt qu'en procédant par une analyse grammaticale des requêtes) et en construisant, par suite, des règles similaires aux deux règles déjà incluse dans le prototype.

3.3 Intelligence artificielle

De manière à permettre aux robots de jouer par eux-mêmes, il est demandé de créer en Prolog une intelligence artificielle basée sur les algorithmes du chapitre 6 "Recherche et jeux". Pour ce faire, deux tâches principales sont à effectuer. D'une part, il convient de définir avec rigueur les états et les transitions entre les états. D'autre part, il convient de définir la profondeur d'exploration des états ainsi qu'une fonction heuristique d'évaluation des états sans successeur. Une première approche de cette fonction heuristique serait, pour un pion donné, d'évaluer la distance en ligne droite par rapport au bord à atteindre. Il est cependant certain qu'une telle fonction est quelque peu naïve puisqu'elle ne prend nullement en compte les barrières sur le chemin vers ce bord et qu'elle ne considère pas du tout les pions adverses et, en particulier, leur difficulté à se déplacer. C'est sur ce point qu'il est fait appel à la sagacité des étudiants!

4 Remise des travaux

Le projet sera réalisé par groupe de trois étudiants, selon les inscriptions effectuées sur webcampus. Le code produit accompagné d'un rapport explicitant sa réalisation sera remis sur webcampus pour le dimanche 17 mai 2020 à minuit. En outre, de manière à permettre une évaluation à distance, une vidéo sera jointe (pour la même date). Elle aura pour objectif d'expliciter le déploiement du code et de faire la démonstration des deux composants principaux du logiciel : le bot et l'intelligence artificielle.