

Projet PontuXL

INFO B317 — IA et programmation symbolique

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

Année académique 2024-2025

1 Introduction

Pontu est un jeu de société dont le but est d'isoler les lutins de son adversaire en déplaçant ses propres lutins et en retirant des ponts. Ainsi qu'illustré en figure 1, le jeu se pratique classiquement sur un plateau de 5x5 cases, les cases étant reliées entre elles par des ponts. Chaque joueur dispose de lutins de couleur différente. Le joueur qui commence est tiré au sort. En cours de partie, quand c'est son tour, chaque joueur effectue deux actions :

- il déplace un de ses lutins d'une case à une autre en empruntant un pont (ce qui nécessite que le pont existe)
- puis, à son choix, il retire un pont – celui qu'il a utilisé ou un autre – ou tourne un pont d'un quart de tour – de nouveau celui qu'il a utilisé ou un autre – sur l'axe d'une case.

Deux lutins ne peuvent pas se trouver sur une même case.

Un joueur est éliminé lorsque tous ses lutins se retrouvent sans pont autour d'eux. Un joueur qui n'est pas dans ce cas mais dont aucun lutin ne peut se déplacer par manque de pont ou parce qu'un autre lutin occupe la case qu'il obtiendrait en traversant un pont ne bouge aucun lutin mais peut enlever un pont à son choix.

Le jeu se pratique à deux, trois ou quatre joueurs.

Une vidéo illustrant une partie est disponible sur le site YouTube à l'adresse : <https://www.youtube.com/watch?v=RYPHTus9CQg>.

Pontu est un jeu apprécié pour la simplicité de ses règles et pour la stratégie dont les joueurs doivent faire preuve. Il s'agit là d'autant de facteurs qui en font un excellent choix pour un travail dans le cadre du cours d'Intelligence Artificielle !

2 Particularisation au projet

Pour le projet, nous particulariserons le jeu comme suit :

- le jeu sera joué par quatre joueurs, utilisant des lutins de couleur bleu, vert, rouge et jaune ;
- les lutins de couleur bleu et rouge seront déplacés par une intelligence artificielle, les autres par un être humain ;
- contrairement à la version originelle du jeu, les joueurs jouent tour à tour dans l'ordre suivant : d'abord les verts, puis les bleus, puis les jaunes puis les rouges ;
- le plateau de jeu considéré est de taille 6 sur 6 (d'où le suffixe XL). Les cases seront numérotées selon les abscisses et ordonnées en prenant le coin inférieur gauche comme origine. Les ponts seront identifiés par les coordonnées des deux cases qu'ils joignent en donnant

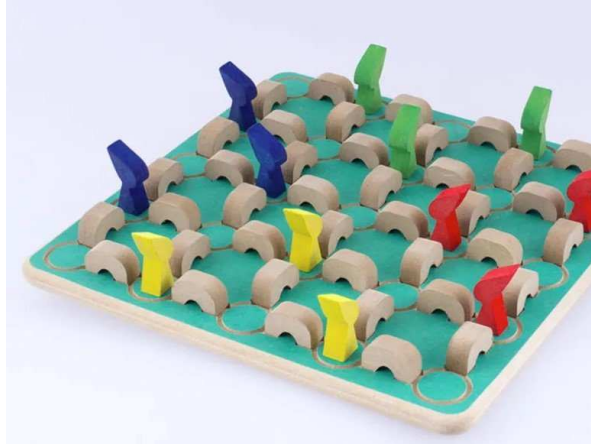


FIGURE 1 – Illustration du jeu “Pontu”

- d’abord les coordonnées inférieures selon un ordre lexicographique. Ainsi on utilisera $(2,3)–(2,4)$ et non $(2,4)–(2,3)$ et de même $(3,1)–(3,2)$ et non $(3,2)–(3,1)$;
- chaque joueur dispose de quatre lutins ;
- à tour de rôle et dans l’ordre fixé ci-dessus, chaque joueur place un lutin sur le plateau ;
- quand tous les lutins sont placés, les joueurs déplacent tour à tour un lutin selon l’ordre ci-dessus (en enlevant ou tournant un pont au passage) ;
- le dernier joueur non éliminé gagne la partie.

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.

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 déplacements des lutins et d’enlèvement ou de déplacement des ponts ;
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 librairies graphiques 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> ou encore d'utiliser la version Tau-Prolog de Prolog disponible à la page <http://tau-prolog.org/>

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 PBot, le bot explicateur du jeu PontuXL.
En quoi puis-je vous aider ?

Vous : Qui commence le jeu ?

PBot : Par convention, c'est au joueur en charge des lutins
verts de commencer la partie.

Vous : Combien de lutins compte chaque equipe ?

PBot : 4

Vous : Puis-je deplacer un lutin sur une case occupee par un
autre lutin ?

PBot : Non.

Vous : Quel pont puis-je retirer apres avoir deplace un lutin ?

PBot : Il est permis de retirer le pont emprunte ou tout autre
pont.

Vous : Je joue pour les lutins verts. Quel lutin me conseillez-vous
de deplacer et vers quelle case ?

PBot : Le lutin sur la case (3,4) vers la case (3,5). Puis je
vous conseiller d'enlever le pont (3,4) -- (3,5) ?

Le bot est à réaliser en Prolog sur base du prototype proposé dans le fichier `pbot-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 `pontuXL/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 incluses 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. Il est aussi demandé de développer deux heuristiques de déplacement des lutins et de comparer l'efficacité de ces deux heuristiques. Une d'entre elles consiste par exemple à sélectionner à chaque tour le ou un des lutins ayant le moins de pont près de lui et d'enlever le pont que le lutin empruntera. Enfin dans un souci d'optimisation, il est demandé d'appliquer les algorithmes Minimax et Alpha-Beta.

4 Remise des travaux

Le projet sera réalisé par groupe de quatre étudiants, selon les inscriptions effectuées sur Web-campus. Le code produit accompagné d'un rapport explicitant sa réalisation sera remis sur Web-campus pour le dimanche 18 mai 2025 à minuit.

Une vidéo sera rentrée pour la même date (dimanche 18 mai 2025). 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.