



UE PROJET

M1 ANDROIDE

Processus décisionnels de Markov et plus court chemin stochastique

Réalisé par:

Alexandre DUPONT-BOUILLARD

Adrien BROUCHET

Supervisé par:

Emmanuel HYON

Pierre FOUILHOUX

Janvier-Juin 2019

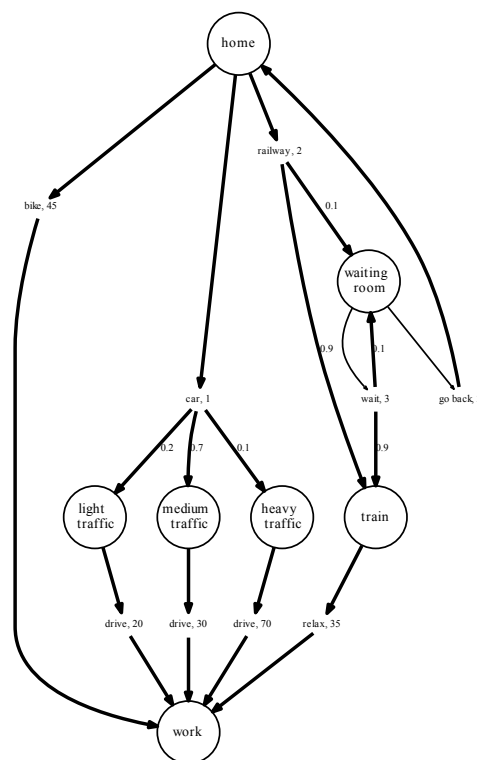
Sommaire

1	Présentation générale	2
1.1	Contextualisation du problème SSP-E	2
1.2	Objectif du projet	3
2	Tâches à réaliser	4
2.1	Appropriation du problème	4
2.2	Mise en place d’algorithmes	4
2.3	Génération d’instances	4
2.4	Tests	4
2.5	Pour aller plus loin...	4
3	Planning prévisionnel	5

Présentation générale

1.1 Contextualisation du problème SSP-E

Imaginons la situation suivante: M. Dupont se rend au travail tous les jours, et a le choix entre prendre le vélo, la voiture, et le train. Le temps de trajet varie d'un moyen de transport à l'autre, mais peut également varier pour le même moyen de transport : en voiture, par exemple, suivant le trafic, M. Dupont mettra plus ou moins longtemps pour arriver à son travail.



Ce genre de situation se formalise avec la notion de Processus Décisionnel Markovien (MDP pour *Markovian Decision Process* en anglais) : il s'agit d'un modèle stochastique à temps discrétisé dans lequel un agent (ici, M. Dupont) se déplace d'état en état en effectuant à chaque étape une action dont le résultat est généralement aléatoire ; à chaque action est associée un coût. Dans l'exemple précédent, les états sont représentés par les noeuds cerclés et les actions par les noeuds non cerclés (avec leur coût associé).

Dans le cadre du projet, on s'intéresse à l'étude de la question du plus court chemin stochastique en moyenne (en anglais : *Shortest Stochastic Path - Expectancy*, d'où le nom du problème SSP-E). Pour l'exemple de M. Dupont, cela revient à trouver les instructions à donner pour que M. Dupont mette un temps minimal en moyenne pour se rendre sur son lieu de travail, et ce quel que soit l'endroit d'où il part.

En termes plus formels, cela revient à trouver une fonction qui minimise toujours en moyenne le coût total pour arriver à l'objectif, peu importe le point de départ ou la longueur du trajet. On appelle une telle fonction une politique optimale (lorsqu'elle existe), et pour l'obtenir on va utiliser le critère de performance total, c'est-à-dire l'espérance du coût total du trajet.

1.2 Objectif du projet

Au cours de ce projet, nous allons tenter de mettre en place des algorithmes permettant d'aboutir à des politiques optimales pour le critère considéré, et de comparer l'efficacité de ces différents algorithmes sur des jeux de données que nous allons construire.

Tâches à réaliser

2.1 Appropriation du problème

Dans un premier temps, nous allons étudier et comprendre le modèle et la théorie associée à travers la constitution et l'analyse d'une bibliographie de référence.

2.2 Mise en place d'algorithmes

Une fois le problème bien compris, nous allons implémenter différents algorithmes de programmation dynamique (value iteration et policy iteration), ainsi que plusieurs algorithmes de programmation linéaire (PL "classique", algorithme primal-dual) [1], [2].

Nous utiliserons le solveur Gurobi pour les algorithmes de programmation linéaire, ainsi que l'utilitaire SWIG afin d'interfacer les algorithmes de programmation dynamique déjà existants et codés en C++.

2.3 Génération d'instances

Afin de pouvoir tester les différents algorithmes mis en place et de pouvoir comparer leur efficacité, nous allons créer un certain nombre d'instances avec des caractéristiques variées (taille...).

2.4 Tests

Nous pourrions alors enfin tester les différents algorithmes implémentés et déterminer lesquels sont les plus efficaces suivant le type d'instance considérée.

2.5 Pour aller plus loin...

Si nous en avons le temps, nous nous intéresserons à une autre manière de résoudre le problème SSP-E: l'apprentissage par renforcement, dont nous comparerons l'efficacité avec les autres méthodes implémentées précédemment.

Planning prévisionnel

Ainsi que mentionné précédemment, le travail bibliographique va se poursuivre tout au long du projet. Nous nous fixons les dates suivantes pour le projet:

- Fin février: cahier des charges final
- Fin mars: algorithmes implémentés
- Fin avril: instances créées et tests effectués
- Fin mai: rendu du rapport

Bibliography

- [1] Gautier Stauffer, Matthieu Guilloit. *The Stochastic Shortest Path Problem: A polyhedral combinatorics perspective*. European Journal of Operational Research, 2018.
- [2] Frédéric Garcia. *Processus Décisionnels de Markov en Intelligence Artificielle*. Groupe PDMIA, INRIA Lille, 2008.