

## Optimisation sous incertitude dans StarCraft 2

Contact : [florian.richoux@univ-nantes.fr](mailto:florian.richoux@univ-nantes.fr)

### I Contexte

Il existe de nombreux problèmes d'**optimisation combinatoire** soumis à de l'**incertitude**. On peut penser à tous les problèmes de production (énergie, bien de consommation, ...) où l'on n'a pas une connaissance parfaite de la demande avenir. L'inconnu ici ne porte pas sur ce que l'on peut ou ne peut pas faire, mais sur l'état de l'environnement ayant une influence sur la valeur optimale d'une décision.

Il existe de nombreuses extensions du formalisme de programmation par contrainte (CSP) traitant avec l'incertitude. Cependant, aucun ne convient lorsqu'il s'agit de modéliser des problème de prise de décision où l'on connaît les choix possibles qui s'offrent à nous, mais où un agent extérieur (une tierce personne, un environnement, etc) détermine les valeurs de certaines variables spécifiques, ces valeurs nous étant inconnues au moment de prendre notre décision.

Un des formalismes qui se rapproche le plus de ce que nous recherchons est Stochastic CSP. Dans ce formalisme, certaines variables ont leur valeur affectée par une source externe et non pas par l'algorithme de résolution (le solveur). Ces variables sont des variables aléatoires. Le but pour le solveur est de trouver une affectation de valeur de chaque variable sous son contrôle de manière à ce que la probabilité que cette affectation soit une solution dépasse un seuil  $\theta$  fourni en paramètre du modèle du problème étudié. Cependant, fixer la valeur de ce paramètre  $\theta$  pour modéliser un problème donné est un processus arbitraire et subjectif.

Une première proposition de modification de ce formalisme consisterait à considérer les variables non contrôlées par le solveur comme des variables aléatoires suivant chacune sa propre distribution de probabilités. Pour chaque contrainte avec au moins une telle variable, on estime la distribution de probabilité de sa satisfaction de toutes les combinaisons possibles des valeurs de chacune des variables de la contrainte. Posons  $\epsilon$  l'espérance de cette distribution. Étant donné une affectation de valeur des variables sous contrôle du solveur, on peut estimer, en fonction des distributions des variables non contrôlées, si l'espérance de cette affectation est supérieure ou égale à  $\epsilon$ . Ainsi, le but est, comme avec Stochastic CSP, de déterminer si une affectation dépasse un certain seuil. La différence avec le Stochastic CSP est que ce seuil n'est plus une valeur  $\theta$  fixée arbitrairement mais une valeur estimable dépendant de l'affectation des variables contrôlées et des distributions de probabilité des variables non contrôlées. De plus pour calculer le score d'une fonction objectif, on peut prendre l'espérance de chaque variable non contrôlée comme son affectation de valeur (même si cette espérance sort du domaine de la variable).

Un tel formalisme méritera une étude approfondie de sa complexité, ainsi que ses connexions avec des modèles comme les processus de décision de Markov. Un des buts de

ce projet de recherche est aussi de combiner des méthodes d'apprentissage automatique et de programmation par contrainte. Les méthodes statistiques en apprentissage automatique devraient nous permettre de manipuler l'incertitude des problèmes d'optimisation modélisés dans le formalisme proposé ci-dessus.

Une fois un formalisme correctement défini et caractérisé, la seconde étape de ce projet est de proposer un algorithme de recherche locale pour résoudre nativement des problèmes d'optimisation combinatoire avec incertitude modélisés dans ce formalisme, et d'implémenter cet algorithme. Le calcul d'espérances peut parfois être intractable, mais peut être facilement approximé avec des tirages de Monte Carlo.

Afin de reconnaître des motifs récurrents dans les valeurs probabilistes en fonction de l'instance du problème et de l'environnement, nous envisageons l'application suivante : prendre comme environnement de test **StarCraft 2**, un jeu de stratégie en temps réel où l'information est incomplète et où il existe depuis peu l'**API s2api** pour pouvoir programmer un agent jouant au jeu ainsi que le **framework de deep learning pyc2** et une grande collection de parties analysables, qui pourront servir à l'apprentissage des variables aléatoires introduites ci-dessus. Des problèmes d'optimisation combinatoire liés à la prise de décisions regorgent dans ce type d'environnement, par exemple « quels choix technologiques dois-je faire pour contrer mon adversaire, sachant les morceaux d'informations dont je dispose sur ses choix actuels ? ». L'avantage de ce type d'environnement, en sus de proposer divers problèmes d'optimisation avec incertitude, est qu'il permet de créer l'ensemble des données à étudier (il suffit de jouer plein de parties et de les analyser a posteriori pour apprendre).

## II Stage de Master 2 (5/6 mois entre 01/18 et 08/18)

Ce stage consiste à modéliser la partie optimisation d'un problème via le formalisme Stochastic CSP modifié, de modéliser l'incertitude lié à l'environnement via un modèle d'apprentissage statistique, d'implémenter ces modèles et de la valider expérimentalement. Plus en détail, il faudra :

- Réfléchir aux modifications à apporter au formalisme Stochastic CSP afin de pouvoir modéliser des problèmes de prises de décision sous incertitude en se débarrassant de paramètres arbitraires, et y dégager des propriétés tel que la complexité.
- Proposer un modèle d'apprentissage statistique pour appréhender l'incertitude liée à l'environnement, au contexte du problème étudié.
- Concevoir et implémenter un algorithme de recherche locale pour résoudre nativement les problèmes d'optimisation modélisés dans le formalisme proposé.
- Modéliser des problèmes d'optimisation sous incertitude dans StarCraft, et expérimenter les performances du solveur de recherche locale implémenté.

Ce stage rentre dans le cadre du projet COPUL du RFI Atlanstic2020 de la région Pays de la Loire (vidéo à 3 minutes introduisant ce projet ici : [goo.gl/AUpPxu](http://goo.gl/AUpPxu)).

Le stage se déroulera au LS2N de l'Université de Nantes et sera gratifié au hauteur d'environ 550€ par mois (pour 22 jours par mois, 7h par jours). Il couvrira 5 à 6 mois sur la période janvier - août 2018.