

Compilation de connaissances pour la décision en ligne : application à la conduite de systèmes autonomes

(version courte en français)

Résumé : La conduite de systèmes autonomes nécessite de prendre des décisions en fonction des observations et des objectifs courants : cela implique des tâches à effectuer en ligne, avec les moyens de calcul embarqués. Cependant, il s'agit généralement de tâches combinatoires, gourmandes en temps de calcul et en espace mémoire. Réaliser ces tâches intégralement en ligne dégrade la réactivité du système ; les réaliser intégralement hors ligne, en anticipant toutes les situations possibles, nuit à son embarquabilité. Les techniques de compilation de connaissances sont susceptibles d'apporter un compromis, en déportant au maximum l'effort de calcul avant la mise en situation du système. Ces techniques consistent à traduire un problème dans un certain langage, fournissant une forme compilée de ce problème, dont la résolution est facile et la taille aussi compacte que possible. L'étape de traduction peut être très longue, mais elle n'est effectuée qu'une seule fois, hors ligne. Il existe de nombreux langages-cibles de compilation, notamment le langage des diagrammes de décision binaires (BDDs), qui ont été utilisés avec succès dans divers domaines de l'intelligence artificielle, tels le model-checking, la configuration ou la planification.

L'objectif de la thèse était d'étudier l'application de la compilation de connaissances à la conduite de systèmes autonomes. Nous nous sommes intéressés à des problèmes réels de planification, qui impliquent souvent des variables continues ou à grand domaine énuméré (temps ou mémoire par exemple). Nous avons orienté notre travail vers la recherche et l'étude de langages-cibles de compilation assez expressifs pour permettre de représenter de tels problèmes.

Dans la première partie de la thèse, nous présentons divers aspects de la compilation de connaissances ainsi qu'un état de l'art de l'utilisation de la compilation dans le domaine de la planification. Dans une seconde partie, nous étendons le cadre des BDDs aux variables réelles et énumérées, définissant le langage-cible des « interval automata » (IAs). Nous établissons la carte de compilation des IAs et de certaines restrictions des IAs, c'est-à-dire leurs propriétés de compacité et leur efficacité vis-à-vis d'opérations élémentaires. Nous décrivons des méthodes de compilation en IAs pour des problèmes exprimés sous forme de réseaux de contraintes continues. Dans une troisième partie, nous définissons le langage-cible des « set-labeled diagrams » (SDs), une autre généralisation des BDDs, permettant de représenter des IAs discrétisés. Nous établissons la carte de compilation des SDs et de certaines restrictions des SDs, et décrivons une méthode de compilation de réseaux de contraintes discrets en SDs. Nous montrons expérimentalement que l'utilisation de IAs et de SDs pour la conduite de systèmes autonomes est prometteuse.

Mots-clefs : compilation de connaissances, planification,
BDD, automates, model-checking, CSP