

Approche générique pour l'acquisition de contraintes qualitatives *

Mohamed-Bachir Belaid,¹ Nassim Belmecheri,² Arnaud Gotlieb,² Nadjib Lazaar³ and Helge Spieker²

¹NILU, Norwegian Institute for Air Research, Kjeller, Norway.

²Simula Research Laboratory, Oslo, Norvège

³LIRMM, Université de Montpellier, CNRS, Montpellier, France

Abstract

Many planning, scheduling or multi-dimensional packing problems involve the design of subtle logical combinations of temporal or spatial constraints. On the one hand, the precise modelling of these constraints, which are formulated in various relation algebras, entails a number of possible logical combinations and requires expertise in constraint-based modelling. On the other hand, active constraint acquisition (CA) has been used successfully to support non-experienced users in learning conjunctive constraint networks through the generation of a sequence of queries. In this paper, we propose GEQCA, which stands for *Generic Qualitative Constraint Acquisition*, an active CA method that learns qualitative constraints via the concept of qualitative queries. GEQCA combines qualitative queries with time-bounded path consistency (PC) and background knowledge propagation to acquire the qualitative constraints of any scheduling or packing problem. We prove soundness, completeness and termination of GEQCA by exploiting the jointly exhaustive and pairwise disjoint property of qualitative calculus and we give an experimental evaluation that shows (i) the efficiency of our approach in learning temporal constraints and, (ii) the use of GEQCA on real scheduling instances.

Résumé

De nombreux problèmes de planification et d'ordonnement impliquent la création de combinaisons subtiles de contraintes temporelles ou spatiales. La modélisation précise de ces contraintes, qui sont formulées dans diverses algèbres de relations, nécessite une expertise en modélisation basée sur les contraintes et implique un grand nombre de combinaisons logiques possibles. L'acquisition active de contraintes (AC) a été utilisée avec succès pour aider les utilisateurs non expérimentés à apprendre les réseaux de contraintes conjonctives par la génération d'une séquence de requêtes. Dans cet article, nous proposons une méthode d'AC appelée GEQCA pour *Generic Qualitative Constraint Acquisition*, qui permet d'apprendre les contraintes qualitatives en utilisant des requêtes qualitatives. GEQCA combine les requêtes qualitatives avec la cohérence de chemin limitée dans le temps (PC pour *Path Consistency*) et la propagation des connaissances de base pour acquérir les contraintes qualitatives. Nous prouvons

la correction, la complétude et la terminaison de GEQCA. Nous présentons également une évaluation expérimentale qui montre l'efficacité de notre approche dans l'apprentissage des contraintes temporelles ainsi que l'utilisation de GEQCA sur des instances réelles d'ordonnement.

Introduction

Le raisonnement sur le temps et l'espace est essentiel pour résoudre de nombreux problèmes pratiques, tels que la planification automatisée [4] et l'ordonnement [2]. Dans ce contexte, le calcul qualitatif fournit un cadre algébrique qui établit des relations entre des paires d'entités à l'aide d'un langage qui est *exhaustif et disjoint par paires*. Des exemples de calculs qualitatifs incluent (sans s'y limiter) l'algèbre des points [6] ou l'algèbre des intervalles d'Allen [1] pour raisonner sur les tâches temporelles, et le calcul des connexions de régions (RCC) [5] pour raisonner sur les relations topologiques entre les régions spatiales. Dans ce contexte, les techniques de satisfaction de contraintes et de programmation par contraintes (CP) sont des cadres pratiques pour modéliser et résoudre des réseaux de contraintes qualitatives. Pour faciliter la modélisation des problèmes de programmation par contraintes, Bessière et al. ont introduit un cadre permettant d'apprendre les modèles de contraintes par un apprentissage passif à partir d'un ensemble d'exemples d'affectations étiquetées ou par un apprentissage actif avec des requêtes spécifiques permettant de classer les affectations complètes.

Cet article présente le concept de l'acquisition générique de contraintes qualitatives (GEQCA), un nouvel algorithme d'acquisition active de contraintes pour apprendre tout type de contraintes qualitatives entre entités. L'algorithme GEQCA combine des requêtes qualitatives, une cohérence de chemin limitée dans le temps, une heuristique dédiée et une propagation des connaissances de base pour acquérir des contraintes. L'algorithme est conçu pour répondre aux limitations des algorithmes d'acquisition de contraintes existants, telles que l'incapacité de traiter les disjonctions, le contrôle du nombre de requêtes et la connaissance limitée du contexte. L'objectif de GEQCA est de faciliter la modélisation et la résolution de réseaux de contraintes complexes dans des situations pratiques telles que les problèmes d'ordonnement.

* Cette présentation se base sur des résultats publiés à AAAI 2022 [3].

GEQCA : Acquisition de contraintes via des requêtes qualitatives

GEQCA est un algorithme générique conçu pour apprendre des contraintes qualitatives. Il est basé sur un nouveau concept appelé "requête qualitative", où l'utilisateur doit confirmer si une relation atomique est valide entre une paire de variables d'entité données. GEQCA prend en entrée un vocabulaire de variables d'entités, un langage de relations atomiques, des connaissances de base et un timeout comme paramètre. L'algorithme commence avec un réseau contenant uniquement des contraintes universelles entre entités, puis itère sur les paires d'entités pour réduire l'ensemble des relations possibles à un ensemble localement consistant avec ce que l'utilisateur a en tête. GEQCA utilise une procédure de propagation pour réduire automatiquement les relations incompatibles avec l'état courant de l'apprentissage sans avoir besoin de passer par l'utilisateur.

Experiments

Notre évaluation expérimentale de GEQCA porte sur l'algèbre d'Allen appliquée à des entités temporelles. Le langage Γ utilisé contient les 13 relations atomiques connues de cette algèbre.

Le tableau 1 présente l'effort fourni par l'utilisateur pour résoudre 5 instances de planification en utilisant GEQCA, avec une heuristique de sélection des paires que nous avons introduite dans ce travail. Nous avons utilisé les instances RCPSP3, disponibles publiquement, en considérant la structure du problème incluant la durée des tâches, les exigences en ressources et les capacités des sources, noté K_1 . De plus, certaines contraintes peuvent déjà être connues de l'utilisateur, telles que la contrainte globale cumulative et la contrainte de délai. Nous appelons K_2 le background knowledge incluant la contrainte cumulative et la contrainte de délai. Nous avons également utilisé une limite de temps (`cutoff`) d'une heure et noté T_{max} le temps d'attente maximum entre deux requêtes. Chaque instance de planification est caractérisée par le nombre de tâches (par exemple, `sch_30_1` fait référence à l'instance numéro 1 avec 30 TIs).

La première observation est que l'utilisation de GEQCA avec des connaissances sur la structure du problème K_1 permet de réduire considérablement l'effort de l'utilisateur (en moyenne une réduction de 38%). La deuxième observation est que l'effort de l'utilisateur est également réduit lorsqu'il utilise des connaissances qui portent sur des contraintes connues telles que la contrainte *cumulative* et les contraintes de délai. Nous observons une réduction de 41% en utilisant K_1 pour la propagation. L'utilisation de $K_1 \wedge K_2$ apporte une amélioration faible mais non significative (en moyenne une réduction de 41% au lieu de 38%). En outre, en termes de temps CPU, le temps d'attente entre deux requêtes peut atteindre le seuil d'une heure sous $\mathcal{K} = K_1 \wedge K_2$. Cela s'explique par la procédure *solve*, qui peut prendre plus d'une heure pour essayer de prouver la cohérence d'une relation avec le réseau appris.

Instance	\mathcal{K}					
	\emptyset		K_1		$K_1 \wedge K_2$	
	eF	T_{max}	eF	T_{max}	eF	T_{max}
<code>sch_30_1</code>	95%	0.79	55%	0.91	53%	1.18
<code>sch_30_2</code>	98%	0.62	52%	0.90	48%	393.08
<code>sch_60_1</code>	99%	4.80	65%	8.51	62%	13.00
<code>sch_60_2</code>	99%	7.72	64%	8.08	61%	12.55
<code>sch_60_3</code>	98%	5.94	59%	9.66	57%	3.600

TABLE 1 – Effort de l'utilisateur eF avec GEQCA agissant sur les instances RCPSP (avec `cutoff` = 3,600s, T_{max} en secondes).

Conclusion

Cet article, publié à AAAI 2022, présente un nouvel algorithme d'apprentissage actif appelé GEQCA pour apprendre des réseaux qualitatifs via des requêtes qualitative. L'algorithme utilise la propriété JEPD du calcul qualitatif et la cohérence du chemin sur les contraintes temporelles pour prouver la convergence et minimiser le nombre de requêtes nécessaires. Les résultats montrent que GEQCA est une approche appropriée et efficace pour des applications pratiques.

Remerciements

Ce travail a reçu un financement du projet T-LARGO et AutoCSP du Conseil norvégien de la recherche, accord de subvention n° 274786 et n° 324674. Ainsi que du programme de recherche et d'innovation Horizon 2020 de l'Union européenne (projet TAILOR).

Références

- [1] James F. Allen. Maintaining knowledge about temporal intervals. *Communications of the ACM*, 26(11) :832–843, November 1983.
- [2] Roman Barták, Miguel Salido, and Francesca Rossi. Constraint satisfaction techniques in planning and scheduling. *Journal of Intelligent Manufacturing*, 21 :5–15, 02 2008.
- [3] Mohamed-Bachir Belaid, Nassim Belmecheri, Arnaud Gotlieb, Nadjib Lazaar, and Helge Spieker. Geqca : Generic qualitative constraint acquisition. In *Proceedings of the AAAI Conference on Artificial Intelligence*, volume 36, pages 3690–3697, 2022.
- [4] Said Belhadji and Amar Isli. Temporal Constraint Satisfaction Techniques in Job Shop Scheduling Problem Solving. *Constraints*, 3(2) :203–211, June 1998.
- [5] David A Randell, Zhan Cui, and Anthony G Cohn. A spatial logic based on regions and connection. *KR*, 92 :165–176, 1992.
- [6] Marc B Vilain and Henry A Kautz. Constraint propagation algorithms for temporal reasoning. In *AAAI*, volume 86, pages 377–382, 1986.