

# TLP-GP, un planificateur temporel pour des problèmes temporellement expressifs

F. Maris

P. Régnier

IRIT, Université Paul Sabatier,  
118 route de Narbonne  
31062 Toulouse, cedex 9, France.  
{maris, regnier}@irit.fr

## Résumé :

*TLP-GP est un planificateur temporel complet qui permet de traiter des problèmes qui nécessitent des actions concurrentes pour leur résolution. Gérer une aéroport, une gare, cuire des céramiques, faire la cuisine... sont quelques exemples de ces applications. Nous montrerons comment encoder un problème réel pour le résoudre et manipuler ensuite le plan-solution obtenu.*

## Mots clef :

Planification, temps, problème temporellement expressif, graphe de planification, DTP.

## 1 Introduction

TLP-GP [7] [8] [6], est un planificateur temporel complet basé sur l'utilisation d'un graphe de planification simplifié [1] et d'un solveur de problèmes temporels disjonctifs (DTP). Il permet de traiter des problèmes temporels qui nécessitent des actions concurrentes pour leur résolution [3] (problèmes temporellement expressifs). Gérer une aéroport, une gare, cuire des céramiques, faire la cuisine, planter un clou... sont quelques exemples de ces applications. Du point de vue temporel, l'expressivité de son langage de représentation est supérieure à celle de PDDL2.1 [4] puisque les préconditions peuvent être requises et les effets peuvent se produire sur n'importe quel intervalle relatif à un instant caractéristique d'une action (par exemple son instant de début ou de fin). TLP-GP peut également prendre en compte, de manière simple, des événements exogènes, des buts temporellement étendus, des fenêtres d'activation... Il est également capable d'interpréter plusieurs modalités pour la sémantique des intervalles temporels associés aux effets et préconditions des actions. Ces modalités étendent significativement l'expressivité de son langage de représentation par une certaine prise en compte de l'incertitude, du choix, ou des transitions continues.

## 2 Démonstration du logiciel

Nous montrons comment encoder un problème réel dans le langage de notre planificateur, puis comment paramétrer le planificateur pour la résolution. Enfin, nous montrerons comment manipuler le plan-solution obtenu, à travers l'interface graphique.

## 2.1 Encodage d'un problème réel

Le langage de TLP-GP autorise directement une expressivité supérieure à celle de PDDL2.1. Les actions y sont représentées par des quadruplets ( $\langle$ nom-action $\rangle$ ,  $\langle$ préconditions $\rangle$ ,  $\langle$ effets $\rangle$ ,  $\langle$ durée $\rangle$ ) dans lesquels  $\langle$ préconditions $\rangle$  et  $\langle$ effets $\rangle$  sont des ensembles de propositions associées à un label temporel. Ce label représente un intervalle sur les réels, relatif à l'instant de début de l'action (start) sur lequel une précondition doit être vérifiée ou un effet produit. Pour une action,  $\langle$ durée $\rangle$  représentera sa durée. En plus de la modalité "over all" de PDDL2.1, nous avons implémenté d'autres extensions sous la forme de modalités qui permettent de définir différentes manières, pour une proposition, d'intervenir sur un intervalle temporel. La modalité "somewhere [a b] p" permet ainsi d'exprimer que l'on doit protéger tout l'intervalle pour garantir que l'effet puisse s'y produire (resp. pour garantir que la précondition puisse y être satisfaite); il existe une marge de manoeuvre, mais la valeur finale est incontrôlable : (1) p est vraie à l'instant b (incertitude sur l'instant d'apparition de l'effet, certitude sur sa production); (2)  $\neg p$  ne peut pas être établie sur l'intervalle [a, b] (protection de l'intervalle). La modalité "anywhere [a b] p" permet d'exprimer que l'effet (resp. la précondition) doit impérativement se produire (resp. être produite) dans l'intervalle, mais n'importe où : il existe une marge de manoeuvre, mais la valeur finale est contrôlable. Le planificateur peut réduire l'intervalle, sans toutefois aboutir à l'intervalle vide. On peut en plus (syntaxe : **minimal-duration d anywhere [a b] p**) imposer une durée de validité minimale à l'effet, durée pendant laquelle l'effet doit être maintenu dans l'intervalle : (1) p est vraie à un instant c de l'intervalle [a, b]. On peut, en utilisant minimal-duration d, le contraindre à rester vrai à l'intérieur de [a, b] durant d; (2)  $\neg p$  peut être vraie sur une partie de l'intervalle [a, b] mais pas partout, en particulier pas de l'instant c (inclus) jusqu'à c+d (si minimal-duration d est utilisé). TLP-GP peut également prendre en compte naturellement des événements extérieurs (ou des propositions initiales datées) ou des buts temporellement étendus. Avec ces extensions, on peut encoder le problème de réalisation de céramiques dentaires de la manière suivante :

```
(:durative-action fire-kiln
:parameters (?k - kiln)
:duration (= ?duration (firing-duration ?k))
:condition (over all (energy ))
:effect (and
  (somewhere [start (+start 2)] (ready ?k))
  (over [(+ start 2) end] (ready ?k))
  (at end (not (ready ?k))))))

(:durative-action bake-ceramic
:parameters (?p - piece ?k - kiln)
:duration (= ?duration (baking-duration ?p))
:condition (over all (ready ?k))
:effect (and
  (over [start end] (not (baked ?p)))
  (over [start end] (baking ?p))
  (at end (not (baking ?p))))
  (somewhere [(- end 5) end] (baked ?p))))

(:durative-action treat-ceramic
:parameters (?p - piece)
:duration (= ?duration (treating-duration ?p))
:condition (and (over all (baking ?p)))
:effect (and (minimal-duration 3 anywhere
  [start end] (treated ?p))))
```

## 2.2 Paramétrage du planificateur

TLP-GP est implémenté en OCaml<sup>1</sup> 3.09.2 et utilise le solveur SMT (Sat Modulo Theory) MathSat<sup>2</sup> 3.4. Avant de lancer la résolution, il faudra :

- choisir les fichiers des problèmes/domaines donnés en entrée au planificateur ;
- choisir l'algorithme et les heuristiques utilisées.

## 2.3 Manipulation du plan-solution

L'interface graphique de TLP-GP permet ensuite :

- d'afficher le plan-solution fourni par l'algorithme ;
- de modifier ce plan-solution dans la limite des contraintes qu'il impose ;
- d'obtenir des informations sur les différentes actions, préconditions, effets du plan-solution ;
- de simuler l'exécution du plan solution dans le temps pour observer les effets de l'exécution des actions ainsi que la valeur des différentes propositions ;
- d'enregistrer ce plan-solution modifié.

L'interface est codée en JAVA et peut donc être exécutée sur tous les systèmes d'exploitation courants comme Windows, Linux, MacOS. Elle permet d'exécuter TLP-GP sur un problème particulier, mais elle peut également être lancée indépendamment du planificateur et permet de manipuler des plans-solutions flottants. Voici, figure 1 ci-contre, un aperçu de cette interface.

## 3 Conclusion

TLP-GP est capable de résoudre des problèmes temporellement expressifs dans un langage dont l'expressivité est supérieure à celle de PDDL2.1. Aucun compromis ne doit être fait en terme de complétude pour obtenir cette expressivité. Sur des benchmarks<sup>3</sup>

temporellement expressifs, TLP-GP montre de meilleures performances que CRIKEY3 [2], VHPOP2.2 [9] et LPGP [5], trois planificateurs de l'état de l'art capables de résoudre ce type de problèmes. Ces résultats démontrent qu'il est possible de représenter et de résoudre en pratique des problèmes de planification temporellement expressifs proches d'applications réelles. La production d'un plan-solution flottant plutôt qu'une solution fixe permet une grande flexibilité au cours de l'exécution du plan, et l'interface graphique de TLP-GP permet de manipuler facilement ce type de solution.

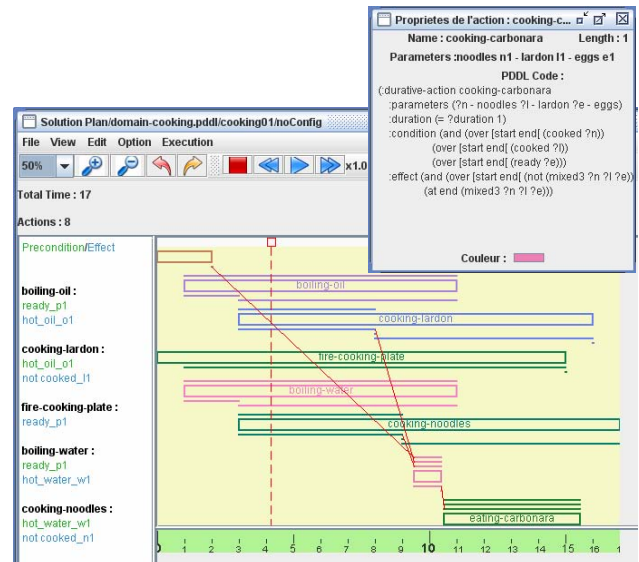


Fig. 1 – Plan-solution flottant dans l'interface de TLP-GP

## Références

- [1] A.Blum, M.Furst, "Fast Planning Through Planning Graph Analysis", IJCAI, 1995.
- [2] A. Coles, M. Fox, D. Long, A. Smith, "Planning with Problems Requiring Temporal Coordination", AAAI 2008, pp. 892-897.
- [3] W.Cushing, S.Kambhampati, Mausam, D.S.Weld, "When is temporal planning really temporal?", IJCAI, 2007.
- [4] M. Fox, D. Long, K. Halsey, "An Investigation into the Expressive Power of PDDL2.1", ECAI, 2004.
- [5] D.Long, M.Fox, "Exploiting a graphplan framework in temporal planning", ICAPS, 2003.
- [6] F.Maris, "Planification SAT et Planification Temporellement Expressive. Les systèmes TSP et TLP-GP", Thèse, 2009.
- [7] F.Maris, P.Régnier, "TLP-GP: Solving Temporally-Expressive Planning Problems", TIME, 2008.
- [8] F.Maris, P.Régnier, "TLP-GP: New Results on Temporally-Expressive Planning Benchmarks", ICTAI, 2008.
- [9] H.L.S.Younes, R.G.Simmons, "VHPOP: Versatile Heuristic Partial Order Planner", JAIR, 20, 2003.

<sup>1</sup> <http://caml.inria.fr/>

<sup>2</sup> <http://mathsat.itc.it/>

<sup>3</sup> <http://tlpgp.free.fr/>