

## 利用派生谓词和偏好处理 OSP 问题的目标效益依赖\*

蒋志华<sup>1+</sup>, 饶东宁<sup>2</sup>, 姜云飞<sup>3</sup>, 翁健<sup>1</sup>

<sup>1</sup>(暨南大学 信息科学技术学院 计算机科学系, 广东 广州 510632)

<sup>2</sup>(广东工业大学 计算机学院, 广东 广州 510006)

<sup>3</sup>(中山大学 信息科学技术学院 软件研究所, 广东 广州 510275)

### Handling Goal Utility Dependencies in OSP Problems with Derived Predicates and Preferences

JIANG Zhi-Hua<sup>1+</sup>, RAO Dong-Ning<sup>2</sup>, JIANG Yun-Fei<sup>3</sup>, WENG Jian<sup>1</sup>

<sup>1</sup>(Department of Computer Science, Ji'nan University, Guangzhou 510632, China)

<sup>2</sup>(Faculty of Computer, Guangdong University of Technology, Guangzhou 510006, China)

<sup>3</sup>(Software Research Institute, School of Information Science and Technology, Sun Yat-Sen University, Guangzhou 510275, China)

+ Corresponding author: E-mail: tjiazhzh@jnu.edu.cn

**Jiang ZH, Rao DN, Jiang YF, Weng J. Handling goal utility dependencies in OSP problems with derived predicates and preferences. *Journal of Software*, 2012, 23(3): 439-450. <http://www.jos.org.cn/1000-9825/3985.htm>**

**Abstract:** In the field of over-subscribed planning (OSP), goal utility dependencies are more useful than a single goal utility used to improve the plan quality, if goals are not independent. However, existing description models do not follow the grammatical specification of standard planning domain description language (PDDL), so they cannot be used in other OSP planning systems yet. To solve this, this paper presents a new way of describing goal utility dependencies with derived predicate rules and goal preferences, both of which are essential elements of PDDL. The goal of the process is to transform GAI (general additive independence) models into these two elements, where a derived predicate rule is used to describe the explicitly triggering conditions of a goal sub-set. A preference is used to depict explicitly its utility or value and both are indispensable. This compilation mechanism can not only maintain the characteristic of ease-of-use and straightness of GAI models in describing utility dependencies, but can also expend the ability of handling utility dependencies for general OSP planning systems. Also, this paper proves the semantic conservation in the compilation process. Experimental results in some OSP benchmark domains show that the algorithm is feasible and useful for improving the plan quality. It is the first time to describe goal utility dependency with PDDL elements in order to overcome the limitations of existing models.

**Key words:** artificial intelligence (AI); automated planning; over-subscribed planning (OSP); goal utility dependency; derived predicate rules; preference

\* 基金项目: 国家自然科学基金(61100134, 61003179, 60903178); 广东省自然科学基金(S2011040001427)

收稿时间: 2010-11-09; 修改时间: 2011-01-06; 定稿时间: 2011-01-20

CNKI 网络优先出版: 2011-01-28 14:57, <http://www.cnki.net/kcms/detail/11.2560.TP.20110128.1457.000.html>

**摘要:** 在过度规划问题(over-subscribed planning,简称 OSP)研究中,如果目标之间不是相互独立的,那么目标间的效益依赖比单个目标效益更能提高规划解的质量.但是已有的描述模型不符合标准规划描述语言(planning domain description language,简称 PDDL)的语法规范,不能在一般的 OSP 规划系统上进行推广.提出了用派生谓词规则和目标偏好描述效益依赖的方法,二者均为 PDDL 语言的基本要素.实质上,将已有的 GAI 模型转换为派生谓词规则和目标偏好,其中派生谓词规则显式描述目标子集的存在条件,偏好机制用来表示目标子集的效益,二者缺一不可.该转换算法既可以保持在描述依赖关系时 GAI 模型的易用性和直观性上,又可以扩展一般的 OSP 规划系统处理目标效益依赖的能力.从理论上可以证明该算法在转换过程中的语义不变性,在基准领域的实验结果表明其可行性和对规划解质量的改善能力.提出符合 PDDL 语言规范的目标效益依赖关系的描述形式,克服了已有模型不通用的缺点.

**关键词:** 人工智能;智能规划;过度规划问题;目标效益依赖;派生谓词规则;偏好

**中图法分类号:** TP18 **文献标识码:** A

自动规划(automated planning)<sup>[1]</sup>是人工智能(artificial intelligence,简称 AI)领域的重要分支.在经典规划问题之上,不要求实现整个目标集的一类特殊规划问题称为过度规划(over-subscribed planning,简称 OSP)问题<sup>[2]</sup>,它是这几年来来的研究热点之一.在实际问题中,很可能没有足够的资源来实现所有目标,这时,规划器的任务是找到一个能够获得最大净效益的规划.

规划问题的目标之间往往存在着许多内在联系.例如在图 1 所示的穿鞋袜任务中,只穿一双袜子获得的效益值为 20,只穿一双鞋子获得的效益值为 50,但是同时穿一双鞋子和一双袜子会大大增加舒适感,因而给予极高的效益值 300.可见在实际问题中,目标之间往往不是相互独立的,目标集的效益值不能简单地表示成单个目标效益的和.为了显式地表示这些隐含的关系,Do 等人<sup>[3,4]</sup>提出使用 GAI(goal additive independence)模型来描述目标效益依赖(goal utility dependencies)关系,即目标集的效益值等于它所包含的多个目标子集的效益函数之和.Russell 等人<sup>[5]</sup>提出使用 UCP-net 模型来表示目标之间的效益依赖,即树结点的两个分支分别对应实现和不实现某目标的效益值.虽然这些描述方式,特别是 GAI 模型具有直观和易用等优点,但都不是标准规划领域描述语言(planning domain description language,简称 PDDL)的特性,因而不能在一般的 OSP 规划系统上进行推广和使用.



Fig.1 Example for goal utility dependencies

图 1 目标效益依赖举例

借助于额外的模型来描述目标集的效益,主要是因为规划问题的状态中没有直接表示目标集合的命题.在 PDDL 中,状态是命题的集合,表示当前状态的命题集合与目标集的交集,即在当前状态下成立的目标子集.如果显式地表示这些目标子集,则状态会变得非常庞大,增加搜索的难度.因此,状态只是命题的集合,其中某些命题表示了规划问题的目标.而在处理 OSP 问题的目标效益依赖时,规划系统需要明显地知道目标子集在当前状态中是否成立,因而需要借助于额外的 GAI 模型.遗憾的是,GAI 模型不属于 PDDL 语言的语法规范.其实, PDDL 语言提供了描述动作的非直接效果(indirect effects)的派生谓词规则(derived predicate rules).目标子集可以用一个派生谓词来表示,其存在是目标在状态中成立的衍生效果.本文提出如下的转换机制:将 GAI 模型中出现的目标子集用派生谓词来描述,目标之间的依赖关系变换为派生谓词规则,目标集的效益转换为规划问题的偏好机

制.派生谓词和偏好均为 PDDL 语言的要素,转换之后的规划问题可以由支持这些语言要素的 OSP 规划系统求解,从而使得更多的规划系统能够处理带有目标效益依赖的 OSP 问题.

本文第 1 节介绍本文的研究背景,包括 OSP 问题以及派生谓词和偏好等.第 2 节给出描述目标效益依赖的 GAI 模型.第 3 节给出将 GAI 模型转换为派生谓词规则和目标偏好的具体方法.第 4 节是实验设计和分析部分.最后是结语和未来工作展望.

## 1 研究背景

智能规划领域研究计算机系统在执行任务之前如何自动制定行动步骤,其研究成果近年来得到了整个人工智能领域的认可.简单地说,一个规划问题主要由初始状态描述  $I$ 、目标状态描述  $G$  和动作集  $A$  组成<sup>[2]</sup>.规划的过程就是寻找从  $I$  到达  $G$  的一个动作序列.

### 1.1 OSP问题

自经典规划之后,不确定规划(non-deterministic planning)、时态规划(temporal planning)、过度规划(over-subscribed planning)以及在线规划(on-line planning)等非经典规划成为该领域的研究热点.过度规划放宽了经典规划中严格目标的假设,自 2004 年被提出<sup>[1]</sup>后,人工智能研究领域的顶级会议 IJCAI<sup>[3,6]</sup>和 AAAI<sup>[7]</sup>以及每年的 ICAPS 上<sup>[4,8,11,12]</sup>都有这方面的研究成果面世.

OSP 问题可以定义为<sup>[1]</sup>:给定一个规划问题  $\Pi = \langle \Sigma, I, G \rangle$ ,其中,  $\Sigma = \langle S, A, X \rangle$  是领域描述( $S$  是命题空间,  $A$  是动作集合,  $X$  是转换关系),每个动作  $a \in A$  有一个代价  $c_a \geq 0$ ,每个目标  $g \in G$  有一个效益  $u_g \geq 0$ ,找一个从初始状态  $I$  到目标子集  $S_G$  的有限动作序列  $\Delta = \langle a_1, a_2, \dots, a_n \rangle$ ,使得净效益值  $\sum_{g \in S_G} u_g - \sum_{a \in \Delta} c_a$  最大,其中,  $S_G$  是  $G$  的子集.用效益值表示对目标的偏好,效益值越大,表示对目标的偏好程度越高.可见,OSP 问题是有限资源下的最优化问题.

OSP 问题的主要处理方法有 3 种:预选目标子集、启发式求解和编译后求解.第 1 种方法在高层选择目标子集而在低层进行规划<sup>[1,8,11]</sup>,其最大的优点是在选择目标子集后,OSP 问题转换成了经典规划问题,从而减少不断评估目标集子带来的开销;主要缺点是不保证最优解.第 2 种方法使用前向状态空间搜索算法直接处理软目标,使用这类方法的典型规划器包括 SapaPS<sup>[6]</sup>和 YochanPS<sup>[12]</sup>.优点是避免了目标之间的互斥检测,适合求解在有限代价内的最优化问题;缺点是不断评估目标集带来一定的开销以及不能保证得到最优解.第 3 种方法将 OSP 问题转换为其他的问题来进行求解,例如整数规划问题(integer programming,简称 IP)<sup>[3,4]</sup>或约束可满足问题等.这类方法的优点是可以利用其他研究领域的先进技术,缺点是编译机制难以考虑目标之间的关联且不适合处理大规模的问题.

### 1.2 派生谓词和偏好

在 PDDL 中,派生谓词用来描述动作的非直接效果,即衍生效果.这与目标集的衍生效益有非常相似的地方.衍生效果不被动作模型所改变,它们的真值由其他谓词的真值通过领域规则进行推导.派生谓词自 2003 年正式提出<sup>[9]</sup>,已有不少规划系统支持,比较有代表性的包括 LPG-td<sup>[13]</sup>,Marvin<sup>[14]</sup>,SGPlan6<sup>[15]</sup>等等.处理派生谓词的方法主要有完全删除法、推导事实法、条件效果法以及激活集法 4 种方法.前三者属于编译方法,把派生谓词规则转换为新的动作或者转换为条件效果,融合到已有的动作模型中<sup>[14,15]</sup>.而激活集法不改变已有的动作模型,在规则图上计算能够替代派生谓词实例的基谓词实例集合(即激活集)<sup>[13]</sup>.

目前,规划社区对于派生谓词的研究依然比较重视.Thiebaut 等人<sup>[16]</sup>已从理论上证明,在问题规模和规划长度不超过多项式级的限定下,派生谓词规则是规划语言不可替代的基本要素.在最新的规划描述语言 PDDL3.0<sup>[20]</sup>中,派生谓词仍是 PDDL 语言中唯一的直接推理机制.国内的规划社区对派生谓词的研究主要体现在通过规则集的等价变换来加快求解速度<sup>[17,18]</sup>以及从规划例中学习派生谓词规则<sup>[19]</sup>等方面.

偏好机制(preferences)<sup>[21-23]</sup>是 PDDL3.0 语言新增的两大特性之一.在经典规划中,每个目标都是硬目标(hard goal),即要求规划系统一定要实现.而在过度规划中,每个目标都是软目标(soft goal),即不要求一定实现.不同目标有不同的权重(即效益值),目标的权重通过偏好机制来表示的.每个目标对应一个偏好,实现该目标所获

得的效益值实质上就是违反其对应的偏好所受的惩罚值.偏好机制用来衡量不同规划解的相对质量.在实现相同硬目标的基础上,所实现的软目标的总权重值越大,则规划解的质量越高.偏好机制除了用在软目标上,还大量用在软约束(即不一定要实现的约束)上.

## 2 问题表示

在 2007 年,Do 提出了用 GAI 模型来表示目标效益的依赖关系<sup>[3,4]</sup>.随后,在 2010 年,Russell 为了使用基于 SAT 技术的规划系统,将 GAI 模型转换为易于编码的 UCP-net 树结构<sup>[5]</sup>.但是,无论是 GAI 模型还是 UCP-net 都不属于 PDDL 的语法规则,只能经过特殊的编码后用于基于整数规划(integer programming,简称 IP)或者 SAT 的个别规划系统.本文试图将它们转换为 PDDL 的标准要素,在此之前,本节先详细介绍用 GAI 模型来描述目标效益依赖关系的具体方法.

先给出 GAI 模型的一般定义.设  $V=\{v_1, \dots, v_n\}$ , 其中,每个变量  $v$  具有值域  $d_v$ .对于  $X \subseteq V$ , 函数  $f(X): \prod_{v \in X} d_v \rightarrow \mathcal{R}$  将矢量  $\langle v_1, \dots, v_n \rangle$  映射到实数值.

**定义 1.** 设  $X_1, \dots, X_k$  是  $X$  的子集,如果存在  $k$  个实函数  $f_i(i=1, \dots, k)$  使得  $f(X) = \sum_{i=1}^k f_i(X_i)$ , 则称函数  $f(X)$  在  $X_1, \dots, X_k$  上存在可加的分解(additive decomposition).

根据 GAI 模型,一个目标集的效益值等于它所包含的多个目标子集的效益函数之和.

**定义 2.** 设规划问题  $\langle \Sigma, I, G \rangle$  的目标集  $G = \{g_1, \dots, g_n\}$ . 对于任意  $G' \subseteq G$ , 如果有效益函数  $u(G')$  是可加分解的,即存在  $G'_1, \dots, G'_k \in G'$  使得  $u(G') = \sum_{i=1}^k f^{G'_i}(G'_i)$ , 则称  $G$  的效益依赖关系可用 GAI 模型描述.

对于不同的规划问题,需要在领域或问题描述阶段精确地定义这些  $f^{G_i}$  函数.在领域描述阶段定义目标效益依赖关系比较困难,也不切实际.首先,规划领域是一类相同性质的规划问题的集合,不同规划问题的目标之间难以建立统一的函数关系;其次,建立领域相关的效益依赖关系,需根据语义相关性对目标划分类型,这实际上已超出了 PDDL 语言的表达能力;最后,在求解具体问题时,领域描述中的目标效益依赖关系要根据具体目标实例化,最终还是形成问题相关的目标效益依赖关系.因此,如无特殊说明,本文以下内容的目标效益依赖关系均为问题相关的,即效益函数中的变量均为具体目标,而不是目标变量.

例 1:在图 1 所示的穿鞋袜问题中,设目标  $so$  表示穿一双袜子,目标  $sh$  表示穿一双鞋子,表示目标效益依赖的 GAI 模型为  $\{f_1, f_2, f_3\}$ , 其中:

- $f_1(\{so\})=20$
- $f_2(\{sh\})=50$
- $f_3(\{so, sh\})=4*f_1(\{so\})+3*f_2(\{sh\})$

则

- $u(\{so\})=20$
- $u(\{sh\})=50$
- $u(\{so, sh\})=f_1(\{so\})+f_2(\{sh\})+f_3(\{so, sh\})=20+50+230=300$

从例 1 可以发现,GAI 模型描述目标效益依赖关系时非常地直观和易于理解.在 GAI 模型中,当函数形如  $f(\{g\})$  时,有  $u(\{g\})=f(\{g\})$ .而当函数形如  $f(\{g_1, \dots, g_m\})$  时,有  $u(\{g_1, \dots, g_m\}) \neq f(\{g_1, \dots, g_m\})$ .可见  $f(\{g_1, \dots, g_m\})$  表示目标子集  $\{g_1, \dots, g_m\}$  的额外效益, $u(\{g_1, \dots, g_m\})$  表示目标子集  $\{g_1, \dots, g_m\}$  的总体效益.二者还是有本质区别的.

**定义 3.** 在考虑目标效益依赖关系的前提下,OSP 问题的净效益值为  $u(S_G) - \sum_{a \in \Delta} c_a$ , 其中,  $S_G$  是规划问题  $\Pi = \langle \Sigma, I, G \rangle$  的目标集  $G$  的子集,  $\Delta$  是从初始状态  $I$  到目标子集  $S_G$  的有限动作序列.

进一步地,根据  $u(G)$  和  $\sum_{g_i \in G} u(g_i)$  的大小关系,可以将目标的效益依赖关系进行分类.

定义 4. 设目标集  $G=\{g_1, \dots, g_m\}$ , 若  $u(G) > \sum_{g_i \in G} u(g_i)$ , 则称  $G$  的效益依赖是正依赖(positive dependencies)关系. 相应地, 若  $u(G) < \sum_{g_i \in G} u(g_i)$ , 则称  $G$  的效益依赖是负依赖(negative dependencies)关系. 此外, 若  $u(G) = \sum_{g_i \in G} u(g_i)$ , 则称  $G$  的目标是相互独立(independent)的.

正依赖关系表示某些目标同时实现能带来更多的额外价值, 负依赖关系表示某些目标同时实现反而减少了整体效益. 例如在 Satellite 领域中, 能够完成给一颗行星拍照并且采集标本的任务会获得非常高的报酬, 而同时给处于极不方便位置的行星群拍照会得到非常低的效益. 这些不同种类的效益依赖信息可以提供给使用控制策略搜索的规划系统.

### 3 转换算法

对于求解一个规划问题来说, 规划系统就像一个黑盒子. 如图 2 所示, 通过读入规划领域描述文件和问题描述文件, 规划系统进行求解并最后给出动作序列形式的规划解. 领域描述文件刻画了领域的通用属性, 主要是动作模型和派生谓词规则(二者也称为规划领域的结构化定义); 问题描述文件表示了具体问题的特点, 主要是初始状态和目标状态. 无论是领域描述还是问题描述都应遵循 PDDL 语言的语法规则, 如果不是 PDDL 语言的描述形式, 则无法进行正常的解析, 需要在规划算法内部增加特殊的编码机制. 例如, 目前能够处理目标效益依赖的少数几个规划系统 MSATPLAN<sup>[5]</sup>和 IPUD<sup>[4]</sup>等, 将 GAI 模型编码成整数规划表示式或者 SAT 公式. 而对于其他性能良好的 OSP 规划系统, 如采用状态空间启发式搜索的  $hsp_p^*$ 、采用 BDD 技术的 Mips-XXL 或 Gamer 等, 则不能直接使用这样的编码方式. 因此, 本文提出将 GAI 模型转换为 PDDL 的语言要素: 派生谓词规则和目标偏好, 使得规划系统不需要扩展内部的规划算法就可以处理带有目标效益依赖规则的 OSP 问题. 换句话说, 派生谓词规则加上目标偏好构成了一种表示目标效益依赖关系的合乎 PDDL 语法的描述形式.

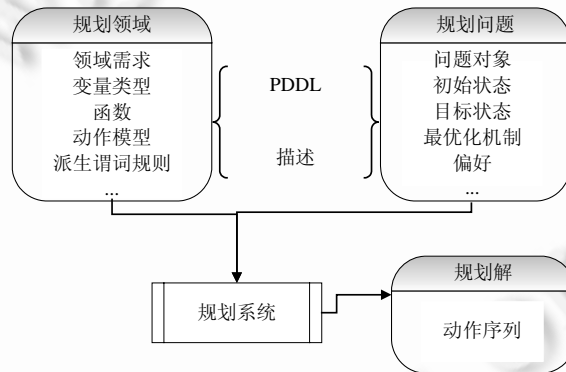


Fig.2 Planning systems

图 2 规划系统

#### 3.1 转换为派生谓词规则和目标偏好

规划问题的描述基于一阶谓词逻辑, 其中谓词分成两类: 基本谓词(basic predicates)和派生谓词(derived predicates)<sup>[9]</sup>. 二者的差别是, 基本谓词可以出现在领域动作的效果中, 而派生谓词不受动作的直接影响, 它们在当前状态下的真值是由封闭世界假设中某些谓词通过领域规则所推导出来. 派生谓词不能作为动作的直接效果, 但是可以出现在动作的前提和目标状态中. 有了派生谓词, 动作模型只需要描述动作的最直接效果, 而各种衍生效果和因果联系可以通过派生谓词规则来表示, 整个问题描述领域变得非常的简洁和清晰.

遵照 PDDL3.0<sup>[20]</sup>的语法规则, 派生谓词规则的 BNF 描述如下:

- $\langle \text{derived-def} \rangle ::= (\text{:derived} \langle \text{atomic formula} \rangle \langle \text{GD} \rangle)$

- $\langle \text{GD} \rangle ::= \langle \text{atomic formula} \rangle$
- $\langle \text{GD} \rangle ::= (\text{and } \langle \text{GD} \rangle^*)$
- $\langle \text{atomic formula} \rangle ::= (\langle \text{predicate} \rangle \langle \text{typed list (variable)} \rangle)$
- $\langle \text{predicate} \rangle ::= \langle \text{name} \rangle$
- $\langle \text{variable} \rangle ::= ? \langle \text{name} \rangle$
- $\langle \text{typed list } (x) \rangle ::= x^*$
- $\langle \text{typed list } (x) \rangle ::= \text{typing } x^+ \text{-} \langle \text{type} \rangle \langle \text{typed list } (x) \rangle$
- $\langle \text{type} \rangle ::= \langle \text{primitive-type} \rangle$
- $\langle \text{primitive-type} \rangle ::= \langle \text{name} \rangle$

对上述 BNF 描述的解释如下:derived 是派生谓词规则定义的标志,“ $\rightarrow$ ”表示逻辑蕴含关系; $\langle \text{atomic formula} \rangle$  是规则所定义的派生谓词,由单个谓词构成; $\langle \text{GD} \rangle$  称为规则触发条件,是由基本谓词或其他派生谓词组成的逻辑公式;谓词由谓词名( $\langle \text{predicate} \rangle$ )加上变量列表组成.如果变量带类型( $\langle \text{typed list } (x) \rangle$ ),则采用变量名  $x$ -类型名( $\langle \text{type} \rangle$ )的形式来声明.谓词名和类型名都是标识符,变量名是带问号的标识符.整个规则采用前缀表达式的形式.该规则的含义为:一旦 $\langle \text{GD} \rangle$ 在某状态下为真,则 $\langle \text{atomic formula} \rangle$ 在该状态下也为真.如果有多条规则同时定义一个 $\langle \text{atomic formula} \rangle$ ,只有当所有这些规则的触发条件均不成立, $\langle \text{atomic formula} \rangle$ 的真值才为假.如前所述,要在状态中显式地描述目标子集,就必须引入这些表示目标子集的派生谓词实例.此外,当规则中的变量被实例化后,这样的规则称为基规则(grounded rule).

遵照 PDDL3.0<sup>[20]</sup>的语法规范,偏好的 BNF 描述如下:

- (preference [name]  $\langle \text{GD} \rangle$ )

其中,preference 是偏好定义的标志,[name]是偏好的名字, $\langle \text{GD} \rangle$ 是由原子谓词组成的逻辑公式.一个 $\langle \text{GD} \rangle$ 对应一个[name],偏好的名字不能相同.此外,用内在谓词(is-violated  $\langle \text{name} \rangle$ )来判断是否违反以 $\langle \text{name} \rangle$ 命名的偏好.如果 is-violated 的真值为 true,则表明规划没有满足偏好 $\langle \text{name} \rangle$ ;反之,is-violated 的真值为 false,表明规划满足偏好 $\langle \text{name} \rangle$ .用目标的权重值来作为违反偏好的惩罚值.目标的权重越大,没有实现该目标所受的惩罚也越大.

下面给出将描述目标效益依赖关系的 GAI 模型转换为派生谓词规则和目标偏好的算法(如图 3 所示).

#### 算法. GAItDP&Pref.

设描述目标效益依赖关系的 GAI 模型为  $GAI\ models = \{f_1, \dots, f_n\}$ .DPRs 表示派生谓词规则集,GOAL 表示规划问题描述的目标段,METRIC 表示规划问题描述的最优化函数.变量  $k$  是 GOAL 中偏好的编号,变量 SUM 表示 METRIC 中的常数因子.算法输入 GAI models,输出 DPRs,GOAL 和 METRIC.

- 1)  $k=0, SUM=0$
- 2) 对于 GAI models 中的每个函数  $f_i(G_i)$
- 3) 如果  $G_i = \{g_1, \dots, g_k\}, k > 1$
- 4) 则生成表示目标子集  $\{g_1, \dots, g_k\}$  的派生谓词实例  $DP_i$
- 5) 生成派生谓词规则  $NewRule = (\text{derived } (\text{and } (g_1)(g_2) \dots (g_k)) DP_i)$ , 并加到 DPRs 中
- 6) 令  $P = DP_i, U = f(P) = f_i(G_i)$
- 7) 否则,如果  $G_i = \{g\}$ , 即单个目标
- 8) 令  $P = g, U = f(P) = f_i(G_i)$
- 9) 令  $k = k + 1$
- 10) 生成偏好  $NewPre = (\text{preference } P(k) P)$ , 并加入 GOAL 中
- 11) 生成优化因子  $NewIsViolated = (*(\text{is-violated } P(k)) U)$ , 并加入 METRIC 中
- 12) 令  $SUM = SUM + U$
- 13) 返回 DPRs, GOAL, METRIC

Fig.3 Algorithm for compiling GAI models into derived predicate rules and goal preferences

图 3 将 GAI 模型编译成派生谓词规则和目标偏好的算法

上述算法将描述目标效益依赖关系的 GAI 模型转化为符合 PDDL3.0 的 BNF 定义的派生谓词规则和目标偏好.首先,对于原规划问题的每个目标,其效益值需在 GAI 模型通过效益函数来指定.即 GAI 模型中的函数个

数不小于单个目标的总个数;其次,如果效益函数涉及多个目标,则算法的步骤 3)、步骤 4)对且仅对其中出现的目标子集产生对应的派生谓词实例(其正确性在第 3.2 节论证).算法的步骤 5)生成推导目标子集成立的派生谓词规则.最后,对 GAI 模型中的每个函数,不管是涉及单个目标还是多个目标,算法的步骤 9)~步骤 12)均生成对应的目标(子集)偏好和最优化函数的因子.GOAL 中的偏好按照自然数顺序编号.METRIC 中的常数因子 SUM 是 GAI 模型中所有出现的目标(子集)的效益和.由于 GAI 模型中的目标均为命题,因此生成的派生谓词规则均为基规则,增加到实例化后的领域描述文件中.GOAL 和 METRIC 增加到问题描述文件.此外还需说明的是,算法生成的派生谓词规则不包含递归并且唯一(即每个目标子集的成立条件是唯一的),因此属于形式非常简单的领域规则,且其中的命题要么表示目标,要么表示目标子集.

算法 GAItoDP&Pref 的时间复杂性分析如下:设需转换的 GAI 模型包含  $n$  个函数,其中有  $m(m \leq n)$  个函数是描述单个目标的效益.对于只描述单个目标效益的函数,算法只需生成对应的目标偏好;而对于涉及多个目标的  $(n-m)$  个的函数,算法除了生成目标偏好,还要生成对应目标子集的派生谓词实例及规则.不过,生成规则、偏好以及优化因子所花的时间都可以看成是常数时间.因此,整个算法所花的时间为  $O(m+(n-m))=O(n)$ .

例 2:将例 1 中的 GAI 模型  $\{f_1, f_2, f_3\}$  转换为派生谓词规则和目标偏好,如下所示:

```
(:derived
  (so_sh)
  (and (so)
        (sh))
)
...
(:goal
  (and
    (preference  $p_1$  (so))
    (preference  $p_2$  (sh))
    (preference  $p_3$  (so_sh)))
)
(:metric maximize
  (-300
    (+(total-cost)
      (*(is-violated  $p_1$ ) 20)
      (*(is-violated  $p_2$ ) 50)
      (*(is-violated  $p_3$ ) 230))))
```

例 2 中,  $p_1/p_2/p_3$  是目标偏好的名字,  $so\_sh$  是对应目标子集的派生谓词实例.  $metric$  表示该规划问题是一个数值规划问题.最优化函数采用前缀表达式的形式,其中, 300 是所有偏好的惩罚值总和,  $total\_cost$  是动作解的总代价.将最优化函数变为中缀表达式,调整一下可以发现,描述的是 OSP 问题中的净效益值,如下所示:

$$(300 - (\text{is-violated } p_1) \times 20 - (\text{is-violated } p_2) \times 50 - (\text{is-violated } p_3) \times 230) - \text{total-cost} = \text{总效益} - \text{总代价} = \text{净效益值}.$$

例 3(接例 2):设有 3 个规划  $Plan_1/Plan_2/Plan_3$  分别满足图 1 所示的 3 组目标,规划解的代价也如图 1 所示.表 1 列出它们所满足的偏好和 is-violated 谓词的真值( $\checkmark$ 表示满足, $\times$ 表示不满足).

**Table 1** Three solution plans which satisfy different preferences respectively  
表 1 分别满足不同偏好的 3 个规划

|          | $P_1$        | $P_2$        | $P_3$        | is-violated $p_1$ | is-violated $p_2$ | is-violated $p_3$ |
|----------|--------------|--------------|--------------|-------------------|-------------------|-------------------|
| $Plan_1$ | $\checkmark$ | $\times$     | $\times$     | 0                 | 1                 | 1                 |
| $Plan_2$ | $\times$     | $\checkmark$ | $\times$     | 1                 | 0                 | 1                 |
| $Plan_3$ | $\checkmark$ | $\checkmark$ | $\checkmark$ | 0                 | 0                 | 0                 |

根据最优化函数,3个规划的净效益值分别为:

- $Plan_1: (300-0-1 \times 50-1 \times 230)-10=10$
- $Plan_2: (300-1 \times 20-0-1 \times 230)-100=-50$
- $Plan_3: (300-0-0-0)-110=190$

可见, $Plan_3$ 的净效益值最大.这表明使用目标效益和偏好可以找到更符合期望(即净效益值更大)的解.

### 3.2 算法的正确性

如前所述,算法 GAItoDP&Pref 只对 GAI 模型中出现的目标子集产生对应的派生谓词实例,而不是产生所有目标子集(即目标集的幂集)的派生谓词实例.这样做的好处在于:一方面,突出能带来额外收益的目标子集,这些目标子集往往对于净效益值最大化起关键作用;另一方面,使得表示状态的命题集合尽可能小,从而减少状态空间的复杂性.但是对于规划算法而言,在评估一个状态的启发值时,往往以其包含的最大目标子集的效益值作为关键因素,因而上述算法需保证转换过程的语义不变性.即对于任意一个目标子集,其经过 GAI 模型计算的效益值与经过目标(子集)偏好计算的效益值是一样的.

**定理 1.** 算法 GAItoDP&Pref 能够保持语义不变性.即对于任意一个目标子集,其经过 GAI 模型计算的效益值与经过算法转换后的目标(子集)偏好计算的效益值是一样的.

证明:设规划问题  $\Pi$  的目标集为  $G=\{g_1, \dots, g_m\}$ ,任意非空目标子集  $G'=\{g_1, \dots, g_{m'}\} \in 2^G$ ,其经过 GAI 模型(GAI models= $\{f_1, \dots, f_n\}$ )计算的效益值为  $U_{GAI\ models}^{G'}$ ,经过算法 GAItoDP&Pref 转换后的目标子集偏好计算的效益值为  $U_{GAI\ to\ DP\ \&\ Pref}^{G'}$ .

首先,根据 GAI 模型的可加性,有  $U_{GAI\ models}^{G'} = \sum_{G_i \subseteq G', i=1}^{i=k} f^{G_i}(G_i)$ ,其中, $G_i$ 必须是 GAI 模型中作为函数变量的目标子集,设共有  $k$  个.由于单个目标的效益必须由 GAI 模型中的函数指定,因此有  $m' \leq k \leq n$ .不失一般性,设  $G'$ 所包含的  $G_1, \dots, G_k$  分别对应模型中的函数  $f_1, \dots, f_k$ ,且  $G_1, \dots, G_{m'}$  为单目标集合(即  $|G_i|=1, 1 \leq i \leq m'$ )以及  $G_{m'+1}, \dots, G_k$  为多目标集合(即  $|G_i|>1, m' < j \leq k$ ),则有  $U_{GAI\ models}^{G'} = \sum_{i=1}^{m'} f_i(G_i) + \sum_{j=m'+1}^k f_j(G_j)$ .

设 GAI models= $\{f_1, \dots, f_n\}$  经过算法 GAItoDP&Pref 转换后产生偏好  $p_1, \dots, p_n$ ,则有

$$U_{GAI\ to\ DP\ \&\ Pref}^{G'} = SUM - \sum_{i=1}^n is\_violated(p_i) f(P_i),$$

其中,  $SUM = \sum_{i=1}^n f(P_i)$  (算法的第 12 步). $P_i$  是  $G_i$  对应的命题,当  $1 \leq i \leq m'$  时,  $P_i = g_i$  (算法的第 8 步);当  $m' < i \leq n$  时,  $P_i = DP_i$  (算法的第 6 步),  $DP_i$  为目标子集  $G_i$  对应的派生谓词实例.由于  $G'$  包含目标子集  $G_1, \dots, G_k$ ,有

$$is\_violated(p_i) = \begin{cases} 0, & 1 \leq i \leq k \\ 1, & k < i \leq n \end{cases}$$

所以,  $U_{GAI\ to\ DP\ \&\ Pref}^{G'} = \sum_{i=1}^n f(P_i) - \sum_{i=k+1}^n f(P_i) = \sum_{i=1}^k f(P_i) = \sum_{i=1}^{m'} f_i(G_i) + \sum_{j=m'+1}^k f_j(G_j)$  (算法的第 6 步和第 8 步),其中,  $|G_i|=1, 1 \leq i \leq m', |G_j|>1, m' < j \leq k$ .最后可证  $U_{GAI\ models}^{G'} = U_{GAI\ to\ DP\ \&\ Pref}^{G'}$ .  $\square$

## 4 实验设计与分析

本实验的目的是为了验证算法 GAItoDP&Pref 的可行性.原则上,将描述目标效益依赖关系的 GAI 模型转化为符合 PDDL3.0 规范的派生谓词规则和目标偏好之后,可以用支持这些语言要素的任意规划系统来进行求解.但是,从无到有地实现这样的一个规划系统是相当困难的,并且其性能也难以与规划社区中知名的规划系统相媲美.因此,选择开放源代码的规划系统进行扩展.Gamer<sup>[24]</sup>和 Mips-XXL<sup>[25]</sup>是少数几个开源的 OSP 规划系统之一,它们由同一规划团队开发,均采用代价优先的分支限界法来求解最优化问题,并且使用二叉决策图(binary



decision diagram,简称 BDD)来表示偏好.每个偏好对应一个布尔变量,整个规划问题的偏好约束对应由这些布尔变量所构成的合取公式.不同的地方在于,Mips-XXL 需要利用外存来存放状态集合以及延迟重复状态的检测,而 Gamer 没有包括外存规划技术.本实验使用一个 C 程序来实现算法 GAItoDP&Pref,对输入的描述文件进行处理后,再分别调用这两个规划系统来进行求解.

国际规划大赛(International Planning Competition,简称 IPC)上没有包含目标效益依赖的基准测试问题,使用 GAI 模型的规划系统(如 iPUD/SPUDS/MSATPLAN 等)也没有公布所使用的测试用例\*\*,因此需对 IPC 的测试问题进行改造.选取 IPC-6\*\*\*上 net benefit optimization track 的 transport-numeric 和 elevators-strips 两个领域来进行实验,原始问题已包含动作代价和单个目标效益,但是不包含目标效益依赖.transport 领域描述了用车辆运送包裹到指定城市的问题,城市之间有距离也有油耗量.如果包裹的目的地之间距离较近,那么运送这些包裹所构成的目标子集应给予更多的效益.elevators 领域描述了用电梯运送乘客到指定楼层的问题,电梯种类分为慢梯和快梯.然而不管搭乘何种电梯,满足同方向乘客的服务请求应给予较多的效益.根据这些原则,手工产生测试问题的描述目标效益依赖关系的 GAI 模型.实现算法 GAItoDP&Pref 的程序先解析原始输入文件和手工生成的 GAI 模型,然后将转换之后的派生谓词规则和目标子集的偏好分别加入到领域文件和问题文件中,最后调用 OSP 规划系统进行求解.其中,目标子集的效益值在适当的上限和下限范围内随机选择.另外要指出的是,Gamer 和 Mips-XXL 不直接支持派生谓词规则,但是支持条件效果(conditional effects).Davidson<sup>[24]</sup>已提出将派生谓词规则编译成条件效果的通用方法.由于本文中表示目标子集依存关系的派生谓词规则不包含递归且都是唯一的,因此转换过程非常简单(只需扫描一遍动作模型,在相应位置上插入条件增加效果和条件删除效果即可),这里不再赘述.

实验环境为:CPU (Pentium Processor 1.33G)+RAM (1GB)+external disk(20G)+FreeBSD 6.2+gcc 4.3.0.规定每个规划问题的限定运行时间不超过 5 分钟.在此时限内,规划系统一直尝试找更好解,直到终止条件到达(时间到或者找到最优解),最后返回当前找到的最好解.

图 4 给出了具体的实验结果(横轴表示问题编号,纵轴表示规划解质量(即净效益值)).其中,Gamer 和 Mips-XXL 用来求解 IPC-6 的原始测试问题,新版本 Gamer+GAItoDP&Pref 和 Mips-XXL+GAItoDP&Pref 用来求解增加了效益依赖的测试问题.

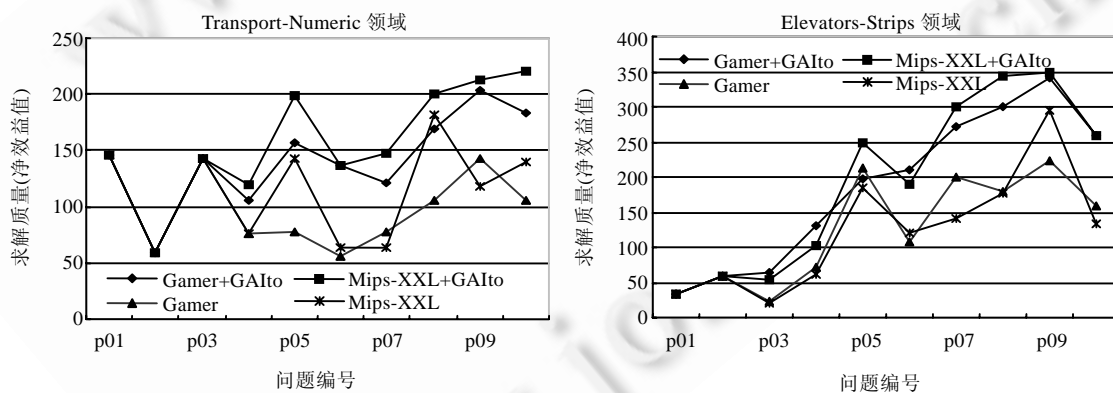


Fig.4 Experiment results

图 4 实验结果

从图 4 可以看到,在大多数实验的规划问题中,增加了算法 GAItoDP&Pref 的 OSP 规划系统比原规划系统找到了更优解.在 Transport-numeric 领域,对于规划问题 p01~p03,Gamer 和 Mips-XXL 均实现所有目标并且找到

\*\* 实际上,iPUD,SPUDS 和 MSATPLAN 也没有公布源代码,甚至可执行文件,因而实验结果无法与它们进行直接比较.

\*\*\* <http://ipc.informatik.uni-freiburg.de/Domains>

最优解,因此增加效益依赖并不能改进规划解质量.而对于该领域的其他测试问题,Gamer 和 Mips-XXL 在时限范围内只找到次优解.从图 4 的左图可以看到,Gamer+GAItoDP&Pref 和 Mips-XXL+GAItoDP&Pref 所得到的规划解的净效益值更大,因此提高了规划解的质量.类似的情况也出现在了 Elevators-strips 领域的实验.从图 4 的右图可以看到,Gamer 和 Mips-XXL 只能找到规划问题 p01~p02 的最优解,对于其余的规划问题,所找到的次优解质量均不如 Gamer+GAItoDP&Pref 和 Mips-XXL+GAItoDP&Pref 找到的解.需要说明的是,由于很多测试问题都是在限定时间内找到次优解(找到最优解可能需要更长或很长的时间),因此对求解时间进行比较没有太大的意义.

## 5 结语与展望

本文提出了一种将已有的描述目标效益依赖的 GAI 模型转换为 PDDL 要素的方法.该方法用派生谓词规则描述单个目标与目标子集的关系,用偏好来表示目标子集的额外效益,二者缺一不可.转换算法保持了 GAI 模型在描述依赖关系时易用和直观的特点,又扩展了一些 OSP 规划系统处理目标效益依赖的能力.本文还证明了转换过程的语义不变性,即对于任意一个目标子集,其经过 GAI 模型计算的效益值与经过算法转换后的目标偏好计算的效益值是一样的.在两个 OSP 问题的基准领域的实验结果表明,在不能实现所有目标的情况下,能够处理目标效益依赖关系的规划系统能够找到更好质量的规划解,从而验证了转换算法的可行性和有效性.

本文所提出的表示方法克服了已有描述模型不通用的缺点,但是还需要进一步的完善.未来工作主要包括以下两个方面:

一方面,将目标效益依赖扩展到更复杂的动作模型,例如持续动作(动作有执行时间)或者不确定动作(动作有多种可能结果)等.在扩展到持续动作时,首先需要把动作的执行时间作为代价的一部分加入到净效益值的计算中,其次是需要处理派生谓词规则如何与时态动作关联.直接编译非常困难,因为动作效果有不同的时刻点,可能需要结合规划系统 LPG 在动作图上分别处理时态推理关系(时态动作图)和派生谓词规则(规则动作图)的方法<sup>[13]</sup>.在扩展到不确定动作时,首先需要在不同组的效果中分别编译派生谓词规则,其次需要在信念空间计算规划解的净效益值.

另一方面,用户所提供的效益依赖关系不一定是正确的,在这样错误知识的引导下,规划系统可能会得到质量更差的规划解.因此,如何从规划例中学习恰当的效益依赖模型是一个很值得探讨的问题.初步设想分成两个阶段:学习针对问题的效益依赖模型,训练例采用针对同一问题的规划解-净效益值等序对形式,对不同的目标子集所带来的额外效益建立对应的整型变量,根据规划解之间的净效益差建立整数规划方程并进行求解;学习针对领域的效益依赖模型,比针对问题的更有意义但是难度更大.领域中所包含的目标需划分为类别,训练例采用不同问题的规划解-净效益值的形式,整型变量对应不同类别的目标组合所带来的额外效益.第 2 个阶段是第 1 个阶段的方法和技术从问题向领域的扩展,但是要解决的问题较多,比如目标能否分类以及目标类别的效益值是否能采用具体的量值.

## References:

- [1] Smith DE. Choosing objectives in oversubscription planning. In: Zilberstein S, Koehler J, Koenig S, eds. Proc. of the 14th Int'l Conf. on Automated Planning and Scheduling. Menlo Park: AAAI, 2004. 393-401.
- [2] Ghallab M, Nau D, Traverso P. Automated Planning: Theory and Practice. San Francisco: Morgan Kaufmann Publishers, 2004. 1-663.
- [3] Do MB, Benton J, Van den Briel MHL, Kambhampati S. Planning with goal utility dependencies. In: Veloso MM, ed. Proc. of the 20th Int'l Joint Conf. on Artificial Intelligence. Menlo Park: AAAI, 2007. 1872-1878.
- [4] Benton J. Solving goal utility dependencies and simple preferences in partial satisfaction planning. In: Proc. of the Int'l Conf. on Automated Planning and Scheduling—Doctoral Consortium. 2006. 1-3. <http://www.plg.inf.uc3m.es/icaps06/doctoral.htm>
- [5] Russell R, Holden S. Handling goal utility dependencies in a satisfiability framework. In: Brafman RI, Geffner H, Hoffmann J, Kautz HA, eds. Proc. of the 20th Int'l Conf. on Automated Planning and Scheduling. Menlo Park: AAAI, 2010. 145-152.

- [6] Benton J, Do MB, Kambhampati S. Over-Subscription planning with numeric goals. In: Kaelbling LP, Saffiotti A, eds. Proc. of the 19th Int'l Joint Conf. on Artificial Intelligence. Menlo Park: AAAI, 2005. 1207–1213.
- [7] Li L, Onder N. Generating plans in concurrent, probabilistic, over-subscribed domains. In: Fox D, Gomes CP, eds. Proc. of the 23rd National Conf. on Artificial Intelligence. Menlo Park: AAAI, 2008. 1857–1858.
- [8] Nigenda RS, Kambhampati S. Planning graph heuristics for selecting objectives in over-subscription planning problems. In: Biundo S, Myers KL, Rajan K, eds. Proc. of the 15th Int'l Conf. on Automated Planning and Scheduling. Menlo Park: AAAI, 2005. 192–201.
- [9] Edelkamp S, Hoffmann J. PDDL2.2: The language for the classical part of the 4th Int'l planning competition. Technical Report, 195, Freiburg, 2004. 1–21.
- [10] Thielscher M. Ramification and causality. *Artificial Intelligence*, 1997,89(1-2):317–364. [doi: 10.1016/S0004-3702(96)00033-1]
- [11] Meuleau N, Brafmany R, Benazeray E. Stochastic over-subscription planning using hierarchies of MDPs. In: Long D, Smith SF, Borrajo D, McCluskey L, eds. Proc. of the 16th Int'l Conf. on Automated Planning and Scheduling. Menlo Park: AAAI, 2006. 121–130.
- [12] Benton J, Kambhampati S, Do MB, YochanPS: PDDL3 simple preferences as partial satisfaction planning. In: Proc. of the 16th Int'l Conf. on Automated Planning and Scheduling—IPC5 Booklet. 2006. 1–3. <http://zeus.ing.unibs.it/ipc-5/>
- [13] Gerevini A, Saetti A, Serina I, Toninelli P. Fast planning in domains with derived predicates: An approach based on rule-action graphs and local search. In: Veloso MM, Kambhampati S, eds. Proc. of the 20th National Conf. on Artificial Intelligence. Menlo Park: AAAI, 2005. 1157–1162.
- [14] Coles AI, Smith AJ. Marvin: Macroactions from reduced versions of the instance. In: Booklet of the 4th Int'l Planning Competition. 2004. 24–26. <http://www.tzi.de/~edelkamp/ipc-4/publication.html>
- [15] Hsu CW, Wah BW. The SGPlan planning system in IPC-6. In: Booklet of the Deterministic Part of the 6th Int'l Planning Competition. 2008. 1–3. <http://ipc.informatik.uni-freiburg.de/Planners>
- [16] Thiébaux S, Hoffmann J, Nebel B. In defense of PDDL axioms. *Artificial Intelligence*, 2005,168(1-2):38–69. [doi: 10.1016/j.artint.2005.05.004]
- [17] Jiang ZH, Jiang YF. An improved method for calculating activation sets of action derived preconditions. *Chinese Journal of Computers*, 2007,29(12):2061–2073 (in Chinese with English abstract).
- [18] Jiang ZH, Jiang YF. Planning with derived predicates based on their state-independent activation sets. *Computer Science*, 2007, 34(3):176–180 (in Chinese with English abstract).
- [19] Rao DN, Jiang ZH, Jiang YF, Liu Q. Learning first-order rules for derived predicates from plan examples. *Chinese Journal of Computers*, 2009,33(2):251–266 (in Chinese with English abstract). [doi: 10.3724/SP.J.1016.2009.00251]
- [20] Gerevini A, Long D. Bnf description of pddl3.0. 2005. 1–7. <http://ipc5.ing.unibs.it>
- [21] Gerevini A, Long D. Plan constraints and preferences in PDDL3. Technical Report, 2005-08-07/, Brescia, 2005. 1–12.
- [22] Edelkamp S, Kissmann P. Optimal symbolic planning with action costs and preferences. In: Boutilier C, ed. Proc. of the 21st Int'l Joint Conf. on Artificial Intelligence. Menlo Park: AAAI, 2009. 1690–1695.
- [23] Sohrabi S, Baier JA, McIlraith SA. HTN planning with preferences. In: Boutilier C, ed. Proc. of the 21st Int'l Joint Conf. on Artificial Intelligence. Menlo Park: AAAI, 2009. 1790–1797.
- [24] Davidson M, Garagnani M. Pre-Processing planning domains containing language axioms. In: Grant T, Witteveen C, eds. Proc. of the 21st Workshop of the UK Planning and Scheduling SIG. 2002. 23–34.
- [25] Edelkamp S, Jabbar S. Cost-Optimal external planning. In: Proc. of the 21st National Conf. on Artificial Intelligence. Menlo Park: AAAI, 2006. 821–826.
- [26] Haslum P, Bonet B, Geffner H. New admissible heuristics for domain-independent planning. In: Veloso MM, Kambhampati S, eds. Proc. of the 20th National Conf. on Artificial Intelligence. Menlo Park: AAAI, 2005. 1163–1168.
- [27] Edelkamp S, Jabbar S, Nazih M. Large-Scale optimal PDDL3 planning with MIPS-XXL. In: Long D, Smith SF, Borrajo D, McCluskey L, eds. Proc. of the 16th Int'l Conf. on Automated Planning and Scheduling. Menlo Park: AAAI, 2006. 81–92.
- [28] Bacchus F, Grove A. Graphical models for preference and utility. In: Besnard P, Hanks S, eds. Proc. of the 11th Annual Conf. on Uncertainty in Artificial Intelligence. San Francisco: Morgan Kaufmann Publishers, 1995. 3–10.

## 附中文参考文献:

- [17] 蒋志华,姜云飞.一种计算动作派生前提的激活集的改进方法.计算机学报,2007,30(12):2061-2073.  
[18] 蒋志华,姜云飞.基于与状态无关的激活集的包含派生谓词的规划问题求解.计算机科学,2007,34(3):176-180.  
[19] 饶东宁,蒋志华,姜云飞,刘强.从规划解中学习一阶派生谓词规则.计算机学报,2010,33(2):251-266.



蒋志华(1978-),女,广西桂林人,博士,讲师,CCF会员,主要研究领域为智能规划.



姜云飞(1945-),男,教授,博士生导师,主要研究领域为定理机器证明,智能诊断,智能规划.



饶东宁(1977-),男,博士,讲师,主要研究领域为智能规划,图论.



翁健(1976-),男,博士,教授,主要研究领域为人工智能,网络信息安全.

www.jos.org.cn

www.jos.org.cn