

И. В. Трофимов

Скрытые знания в задачах планирования

Аннотация. Некоторые задачи планирования кажутся довольно простыми с точки зрения человека, но оказываются вычислительно сложными для универсальных планировщиков. В ряде случаев это обусловлено неполнотой постановки задачи планирования. Скрытие части знаний о предметной области может приводить к значительному увеличению пространства поиска, делая задачу неразрешимой. В этой статье на примере мира кубиков рассматривается, какого рода полезные знания обычно не декларируются в задачах планирования и как эти знания могут повлиять на производительность планировщика.

Введение

Люди, не занимающиеся автоматическим планированием, часто недоумевают, почему столь простые для человека задачи, как задачи из мира кубиков, представляют какую-то сложность для универсальных автоматических планировщиков? Одним из факторов, определяющих разницу в производительности между человеком и машиной при решении такого рода задач, является то, что человек и машина уже на этапе постановки задачи ставятся в неравные условия. Автоматический планировщик может использовать только те знания, которые в явном виде содержит формальная постановка задачи. В отличие от машины, человек может обладать дополнительными знаниями (накопленными в процессе опыта решения задач в данной предметной области или схожих предметных областях), которые в постановке задачи явно не представлены.

Классическая постановка задачи из мира кубиков (с четырьмя действиями) содержит знания, достаточные лишь для решения задачи комбинаторно. Комбинаторные задачи, например обход шахматной доски конем, сложны не только для машины, но и для человека. Если бы при решении задач в мире кубиков мы опирались только

Работа выполнена в рамках проекта № 223 «Исследование методов синтеза целенаправленного поведения в динамических интеллектуальных системах на основе прецедентной информации» по программе Президиума РАН «Фундаментальные проблемы информатики и информационных технологий».

на знания, непосредственно представленные в формулировке задачи, то эти задачи показались бы нам трудными. Однако мы легко справляемся с такими задачами, так как обладаем дополнительными знаниями по этой проблеме, декларация которых отсутствует в постановке задачи.

Предоставление планировщику большего объема знаний приводит к росту его производительности. На практике это подтверждают различного рода эвристики. Эвристики используют знания о предметной области, не представленные в постановке задачи. Но эвристики — это специализированные программные решения и не могут быть использованы в универсальных планировщиках. Для обеспечения гибкости универсальных систем планирования необходимо:

- либо представлять такие знания декларативно в описании домена,
- либо выявлять их непосредственно перед решением задачи (но не всякие знания могут быть выявлены таким образом),
- либо выявлять их методами машинного обучения, предъявив пакет задач и их решений, характерных для данной предметной области,
- либо выявлять их в процессе решения задач для постепенного увеличения производительности по мере накопления опыта.

В этой статье не предлагается методов решения проблемы «скрытых полезных знаний». Мы рассмотрим лишь, какого рода знания скрыты, например, в задачах мира кубиков и продемонстрируем, как оперирование такими знаниями может сказаться на производительности планировщиков.

Скрытые знания в домене «мир кубиков»

Неизбежность появления какого-либо действия в плане. В некоторых доменах достаточно рассмотреть описание задачи, чтобы утверждать, что план будет содержать определенное действие. Знание о неизбежности появления таких действий может значительно сократить пространство поиска для планировщиков, определяющих план путем наложения ограничений. Примерами таких планировщиков могут послужить SNLP [1], UCPOP [2] и RePOP [3].

Рассмотрим примеры знаний такого рода из мира кубиков. Предположим, мы хотим решить простую задачу, изображенную на рисунке 1. Очевидно, что всякое решение этой задачи будет содержать

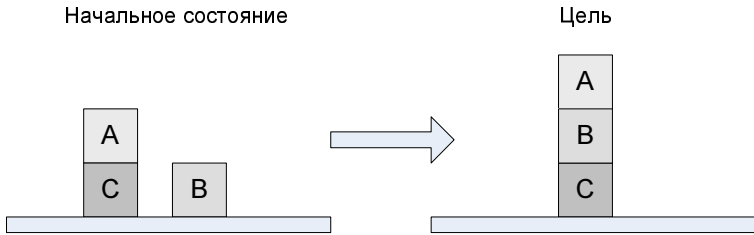


Рис. 1. Мир кубиков. Проблема A.

действие `unstack(A,C)`. Это знание не декларируется в постановке задачи. Его также трудно вывести из постановки задачи. Для этого потребуется убедиться в том, что все способы достижения подцели «кубик B на кубике C» из данного начального состояния содержат действие `unstack(A,C)`.

Знания такого рода легко закодировать в виде эвристики для данного домена, но тогда планировщик перестанет быть универсальным. Возможным выходом здесь может быть декларирование таких знаний в постановке задачи. Наиболее простым вариантом такой декларации является перечисление действий, которые неизбежно появятся в плане. Однако это может потребовать глубокого анализа задачи планирования, что сложно сделать для больших задач. Более гибкий подход — декларация условий (оперирующих описаниями начального состояния и цели), при выполнении которых действие становится необходимым. Ниже перечислены несколько примеров таких правил для домена с кубиками.

- Для любых $?x$, $?y$ и $?z$ ($?x \neq ?y$, $?x \neq ?z$, $?y \neq ?z$), если начальное состояние содержит литерал (`on ?x ?z`), а цель содержит конъюнкт (`on ?y ?z`), то план содержит `unstack(?x,?z)`.
- Для любых $?x$ и $?y$ ($?x \neq ?y$), если цель содержит конъюнкт (`on ?x ?y`), а начальное состояние не включает в себя литерал (`on ?x ?y`), то план содержит действие `stack(?x,?y)`.
- Для любого $?x$, если цель содержит конъюнкт (`on-table ?x`), а начальное состояние такой литерал не содержит, то план содержит действие `putdown(?x)`.

Фактически, эти правила отражают необходимые условия достижения цели из заданного начального условия. Аналогично, может быть представлено знание о необходимом отсутствии шага в плане.

Значимый контекст рассуждений. Часто в описании задачи присутствует большое количество сущностей, про которые с уверенностью можно сказать, что они не имеют отношения к решаемой задаче. Такое положение дел обусловлено тем, что в рамках одной предметной области часто решается множество различных задач планирования, связанных с различными объектами. Нерелевантные объекты попадают в пространство поиска и могут значительно расширить его.

Пусть, например, мы хотим решить следующую задачу (рисунок 2). Чтобы достичь подцель «кубик В на кубике С» необходимо, чтобы

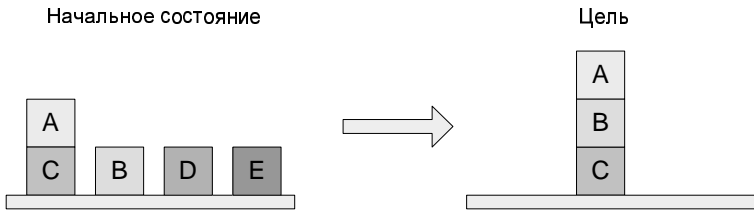


Рис. 2. Мир кубиков. Проблема Б.

кубик В оказался в манипуляторе. Это может быть выполнено двумя способами — действиями `pickup` и `unstack`. Если при поиске решения будет выбран `unstack`, то снимаемый кубик нам известен (это кубик В), а вот кубик, с которого снимают, определяется перебором. При переборе у нас нет основания исключать из рассмотрения кубики D и E, потому как снятие целевого кубика потенциально может выполняться и с них. Однако эти кубики не являются необходимыми объектами манипуляции для получения решения и составляют нерелевантный контекст этой задачи.

Проблема значимого контекста рассуждений детально рассматривалась в [4, 5]. Автоматически определить, что имеет отношение к решению данной задачи, а что нет, довольно сложно. Поэтому, возможно, имеет смысл явно описывать условия, при которых объекты относятся к релевантному (или нерелевантному) контексту. Для некоторых доменов это можно сделать. Например, для мира кубиков такое условие будет звучать следующим образом. Значимый контекст составляют объекты, фигурирующие в цели, а также все объекты,

находящиеся «непосредственно под» этими объектами и «непосредственно или опосредованно над» этими объектами.

Порядок достижения подцелей. Существуют задачи (и домены), в которых цель представлена конъюнкцией литералов (подцелей) и порядок достижения некоторых подцелей предопределен. К числу таких доменов принадлежит и домен с кубиками — все подцели формы $(on\ ?x\ ?y)$ должны быть достигнуты раньше, чем подцель $(holding\ ?z)$. Cheng и Irani [6, 7] показали, что между описаниями схем действий и порядком достижения подцелей может существовать связь. Они предложили полиномиальный алгоритм для обнаружения частичного порядка на множестве конъюнктивных подцелей, обусловленного схемами действий. Однако их алгоритм обнаруживает упорядочивающие ограничения, обусловленные только схемами действий (то есть доменом), но не учитывает ограничений, которые могут быть продиктованы задачей (то есть зависящих от начального состояния и конкретной цели).

В качестве альтернативы алгоритму Ченга-Ирани можно предложить более гибкий декларативный подход.

Недостижимые цели. Не все задачи, которые мы предлагаем решить планировщику, имеют решения. В ряде задач достаточно взглянуть на описание цели, чтобы сделать вывод о том, что она неразрешима. Например, в мире кубиков цель $(and\ (holding\ A)\ (holding\ B))$ является недостижимой, так как имеется лишь один манипулятор. Декларация и использование условий, описывающих недостижимые цели, избавит нас от ненужного планирования (причем с полным перебором пространства поиска).

Некорректные означивания. Описание домена с кубиками допускает (с формальной точки зрения) конструирование некорректно означенных действий. В планировании не регламентируется способ означивания схем действий. Обычно это осуществляется перебором всех возможных вариантов. Таким образом, мы вполне можем получить, например, означивание $stack(A,A)$ — «установить кубик A на кубик A». Это действие некорректно с точки зрения физики нашего мира. Поэтому следует вводить ограничения на множество допустимых кортежей для означивания схемы действия. Например, в случае схемы действия $stack$ можно написать $\forall ?x\ ?y. stack(?x, ?y) \rightarrow (?x \neq ?y)$.

Влияние скрытых знаний на производительность планировщиков

Для демонстрации значимости ранее упоминавшихся скрытых знаний была поставлена серия экспериментов. Использование скрытых знаний реализовывалось эвристическими функциями. Все эксперименты проводились на домене с кубиками.

Цель первого эксперимента — определить степень влияния эвристики, выявляющей значимый контекст. Для проведения эксперимента была реализована означенная версия алгоритма планирования SNLP [1] с перечисленными ниже модификациями.

- Все недетерминированные выборы заменены на систематичные.
- Предусловие может считаться угрозой (по аналогии с планировщиком RePOP [3]).
- Были использованы дизъюнктивные условия безопасности (по аналогии с планировщиком RePOP [3]).
- При добавлении в план всякого шага Step сразу добавляются ограничения ($start < Step$) и ($Step < finish$).

Данный планировщик осуществляет обход пространства поиска в глубину. Барьер глубины поиска был установлен равным 20.

В ходе эксперимента измерялось время работы планировщика над решением задачи с двумя нерелевантными кубиками, изображенной на рисунке 2. Измерялось также время работы над решением этой же задачи, но с большим числом нерелевантных кубиков, лежащих на столе. Результаты приведены в таблице 1.

Количество нерелевантных кубиков	2	3	4	5
Время работы без эвристики, сек.	12	27,6	58,6	115,4
Время работы с эвристикой, сек.	1,7	1,7	1,7	1,7

ТАБЛИЦА 1. Влияние нерелевантного контекста

Из таблицы видно, что в этом домене влияние нерелевантного контекста велико. В общем случае, степень влияния зависит от самого домена и устройства планировщика.

Для проведения следующих экспериментов планировщик был модифицирован. Вместо поиска в глубину осуществлялся эвристический поиск, основанный на принципе «чем меньше открытых условий в частичном плане, тем перспективнее его исследование». Замена планировщика обусловлена тем, что такие воздействия, как внеочередное добавление действий или упорядочивающих ограничений, могут сильно повлиять на выбор ветви поиска. Поэтому оценка будет более точной, если использовать обход в ширину или эвристический подход. Был выбран второй вариант, так как обход в ширину требует слишком много ресурсов. Кроме того, глубина поиска была ограничена 8 действиями.

Второй эксперимент проводился для демонстрации эвристики, добавляющей все неизбежные *unstack*. Эффективность использования эвристики была проверена на проблемах А и В — рисунки 1 и 3, соответственно. Результаты эксперимента отражены в таблице 2.

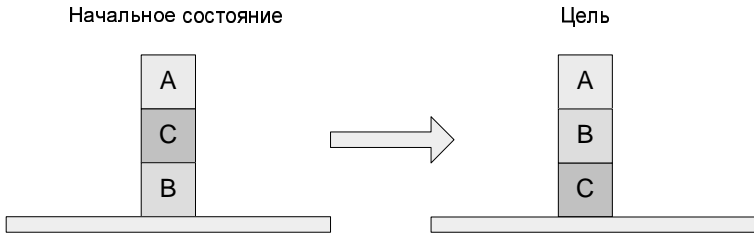


Рис. 3. Мир кубиков. Проблема В.

Задача	А	В
Время работы без эвристики, сек.	289	>300
Время работы с эвристикой, сек.	103	10

ТАБЛИЦА 2. Учет неизбежных действий

Результаты показывают, что эвристика может быть очень продуктивна в данном домене.

Третий эксперимент показывает влияние раннего упорядочивания подцелей, составляющих изначальную конъюнктивную цель. Для эксперимента были взяты задачи, изображенные на рисунках 1, 2 и 3. Результаты сведены в таблицу 3.

Задача	А	Б	В
Время работы без эвристики, сек.	289	>300	>300
Время работы с эвристикой, сек.	0,9	2,7	10

ТАБЛИЦА 3. Упорядочивание подцелей

Все три эксперимента подтверждают, что применение таких эвристик сокращает пространство поиска. Однако трудно установить аналитически, на сколько сокращается поисковое пространство. В частности, это зависит от решаемой задачи.

Заключение

Сложность решения некоторых задач планирования обусловлена неполным декларированием знаний, необходимых для эффективного поиска решения. Явная декларация полезных знаний, таких как значимый контекст рассуждений, необходимые действия, порядок достижения целей, а также написание общих процедур для использования этой информации могут существенно увеличивать производительность планировщика.

Список литературы

- [1] McAllester D., Rosenblitt D. *Systematic Nonlinear Planning* // Proceedings of the Ninth National Conference on Artificial Intelligence (AAAI-91). — Т. 2. — Anaheim, California, USA: AAAI Press/MIT Press, 1991. — ISBN 0-262-51059-6, с. 634–639. ↑
- [2] Penberthy J. S., Weld D. S. *UCPOP: A Sound, Complete, Partial Order Planner for ADL* // Principles of Knowledge Representation and Reasoning: Proc. of the Third International Conference (KR'92) ред. Nebel B. and Rich C. and Swartout W. — San Mateo, CA: Kaufmann, 1992, с. 103-114. ↑
- [3] XuanLong Nguyen, Subbarao Kambhampati *Reviving Partial Order Planning* // Proc. of the 17th IJCAI ред. Nebel B. — Seattle, Washington, USA: Morgan Kaufmann, 2001. — ISBN 1-55860-777-3, с. 459-466. ↑
- [4] Трофимов И. В. *Планирование в средах с большим количеством объектов.* // Международная научно-техническая конференция «Интеллектуальные системы» AIS'05: Труды конференции. — Т. 1. — М.: Физматлит, 2005, с. 276–282. ↑

- [5] Трофимов И. В. *Значимый контекст рассуждений в задаче планирования* // Первая международная конференция «Системный анализ и информационные технологии» САИТ-2005: Труды конференции. — Т. 1. — М.: КомКнига, 2005, с. 227–229. ↑
- [6] Irani K. B., Cheng J. *Subgoal Ordering and Goal Augmentation for Heuristic Problem Solving* // Proc. of the 10th IJCAI. — Milan, Italy, 1987, с. 1018-1024. ↑
- [7] Cheng J., Irani K. B. *Ordering Problem Subgoals* // Proc. of the 11th IJCAI. — Detroit, MI, 1989, с. 931-936. ↑

ИССЛЕДОВАТЕЛЬСКИЙ ЦЕНТР ИСКУССТВЕННОГО ИНТЕЛЛЕКТА ИПС РАН

I. V. Trofimov. *Latent Knowledge in Planning Problems.* (in russian.)

ABSTRACT. A number of planning problems seem rather simple from human viewpoint, but are computationally expensive for a universal automatic planner. Sometimes such difficulties arise due to incomplete problem definition. Lack of domain knowledge can lead to significant growth of search space, thus making the problem intractable. In this article we use the blocks world as an example. We consider the kinds of useful knowledge usually omitted from planning task definition as well as the influence of such knowledge on planner performance.