

一个新的时序推理系统TRS

丁占鳌 (黑龙江大学计算机科学系 哈尔滨150080)

TP301.5

摘 要

本文分析了J. F. Allen的时序推理算法的局限性,给出了一种新的有效算法,并基于ATMS系统的特点,建立了一个新的时序推理系统TRS。

一 引言

在许多领域中,包括计算机科学、哲学、心理学、语言学等等,都涉及到时序知识的表示和时序推理问题。这是当前推理系统中一个很热门的课题。时序知识表示方法很多,但笔者认为应以J. Allen^[1]的时间区间代数表示法最吸引人。它简单明了,而且通过一种约束传递算法使之容易实现。尽管Allen的算法能用来很好地进行时序推理。但是,经分析表明,当在时间网络中引进参考区间的概念来减少系统对空间的要求或者考虑非单调的时序约束传递时,要计算时序断言的结论是相当困难的。同时,如正Allen所指出的他的算法是不完全的,即不能保证总体上的相容性。针对上述问题,我们给出了一种新的基于三角形的时序约束算法,并且采用了J. de Kleer的基于假设的真值维护系统(ATMS)^[2,3,4],从而设计出一种新的时序推理系统TRS。它不仅能有效地进行各种时序推理,而且能对时序关系的析取进行推理,并且生成时间关系网络的完全解。

二 分析Allen算法的局限性

1. 时间区间代数

在Allen算法中引进了时间区间代数的概念。这里所指的时间区间表示的是区间上的一些事件。有关时间区间的时序知识构成一个时间关系网络,每个区间为这个网络中

的节点,两个节点之间的弧表示相应的时间区间之间可能的时序关系集。在两个区间之间共有十三种可能的明确的时序关系原语。

原语1° A与B同时开始,且同时结束。记号为[=],逆记号[=]。

例子 — | AAA
 | BBB

原语2° A在B开始前已结束。记号为[<],逆记号[>]。

例子 — | AAA BBB

原语3° A结束B马上开始。记号[m];逆记号[mi]。

例子 — | AAABBB

原语4° A与B交错发生。记号[o];逆记号[oi]。

例子 — | AAA
 | BBB

原语5° A在B发生期间发生。

例子 — | AAA
 | BBBBB

原语6° A与B同时开始,但A首先结束。记号[s];逆记号[si]。

例子 — | AAA
 | BB BBB

原语7° A与B同时结束,但A迟开始。

例子 — | AAA
 | BBB BB

丁占鳌 副教授,主攻方向:人工智能、机器学习、专家系统。

应注意，上述关系是定性的，我们并不需要知道区间开始或结束的确切时间，当然如果能提供的話，也许是有用的。同时，由于这种区间代数允许时序关系的不确定性，所以我们可以表示两个区间之间任何我们所希望的不确定的时序关系。例如，已知两个区间A和B没有公共发生的部份，则我们可以用一个向量 $V(A, B)$ 来表示这个事实：

$$V(A, B) = [<, m, m_i, >]$$

在这种区间代数里，定义了两种向量运算：加法与乘法。已知两个不同的向量，它们描述同一对区间之间的时序关系，则加法运算是将两个向量“相交”，得到它们共同允许的最大约束关系。用集合的语言定义 V_1 与 V_2 的和为： $V_1 + V_2 = V_1 \cap V_2$ ；而乘法运算是涉及到三个区间A, B和C的向量对之间定义的。如果在区间A和B之间有关系原语 V_1 ，在区间B和C之间有关系原语 V_2 ，那么 V_1 和 V_2 的乘积仍是一个向量记为 V_3 ，即向量 V_1 和 V_2 共同允许的A和C之间最大的约束关系(图1)。它可以通过查找传递表^[1]得到，并且记成： $V_3 = V_1 \times V_2$

当描述两个区间之间的时序关系是由两个以上的原语组成时，只要简单地利用析取，便可以将上述的乘法规则加以推广。确切地说，若 $V_1 = [x_1, x_2, \dots, x_m]$, $V_2 = [y_1, y_2, \dots, y_n]$ ，则用集合的语言可以定义 V_1 和 V_2 的乘积为

$$V_3 = \bigcup_{i,j} \{ V | V = x_i \times y_j, \text{ 其中 } x_i \in V_1, y_j \in V_2 \}$$

为方便起见，我们也把 V_3 记为 $V_3 = V_1 \times V_2$ 。

很明显，当网络中一条弧上的时序约束发生变化时，必须从含有这条弧的所有弧三角形出发来传递这种变化。因此，我们下面作的一切讨论将总是与弧三角形联系在一起。现在，我们仍以图1中的弧三角形为例。假设 V_1 是该网络中区间A与B之间的已被更新的关系向量。按照Allen的算法，必须利用下面的公式分别重新计算B和C之间的关系向量 V_2 ，以及A和C之间的关系向量 V_3 ，即

$$V_2' = V_2 + [\bar{V}_1 \times V_3] \dots\dots\dots(1)$$

$$V_3' = V_3 + [V_1 \times V_2] \dots\dots\dots(2)$$

这里， \bar{V}_1 是区间A和B之间的逆关系向量。注意到，在每做一次这样的传递后， V_2 和 V_3 是否会被更新，将直接影响着 V_1 是否会进一步被更新。这种传递的不平衡有可能造成下面我们提到的问题。

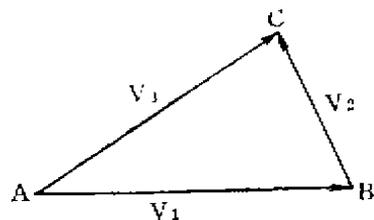


图1 向量乘法

2. Allen 算法的局限性

按照Allen的算法，每当在一条弧，例如(A, B)上作出一个新的时序约束断言时，必须保证这个断言与那条弧上原先存在的时序约束集是相容的。换句话说，当弧(A, B)上的断言被输入时，那个新的时序约束集必须与原有的时序约束集相交。如果所得的交集是原来约束集的真子集，则弧(A, B)将被放入一个数据集queue。执行算法时将不断地从queue中将弧移去。对于每条移去的弧，算法将确定A和B之间的关系是否能用来约束A与其他区间或者B与其他区间之间的关系。如果一个新的关系能够成功地约束，那么这个关系所对应的那对区间就要被放入queue中。这个过程一直进行到queue成为空集为止。容易看出，这种算法能够保证网络中每个弧三角形里那三条弧上的约束集总是相容的，每条弧上的约束集里不会含有多余的约束关系，即绝对没有可能出现一些新的约束关系。这是因为Allen的算法是一种单调的推理系统，它不允许一些弧上的新断言的加入会导致原有结论完全变为无效，只能在原有结论的基础上作进一步的推理。因此它也没有必要记录推出每条弧上的约束集的那些证实。

计科二

陈装刚印

遗憾的是,正是由于上述原因,这个算法不能很好地处理现实世界中出现的三种情形, i) 信息的不完全性 ii) 变换的环境 iii) 解决一些复杂问题时需要作的一些假设。这些情况都有可能产生非单调推理。此外,当我们在网络中引进参考区间的概念后,在约束传递上也遇到某些困难。在上述情况下,如果仍然使用Allen的算法,则可能出现约束传递过程被不恰当地终止,导致冗余约束关系的出现,无法发现互相矛盾的断言集等问题。详细的分析可参见[7]。

最后,虽然Allen的算法可以保证三个结点所形成的子网络中约束集的相容性,但却无法发现它在输入中蕴含的不相容性,即不能保证总体的相容性。这就是所谓的相容性标记问题^[8,9]。

三 基于三角形的约束传递算法

时序推理系统TRS的设计主要是根据一种称为基于三角形的约束传递算法,它是在Allen算法的基础上加以改进而得到的。

1. 网络中三角形的识别

正如前面所指出的,时序约束传递的基本单元是弧三角形。因此,在网络中识别弧三角形的重要性可以从以下三个方面来加以说明:

(1) 当一条弧上的时序约束发生变化时,必须从含有这条弧的所有弧三角形开始来传递这种变化。

(2) 假定在网络中已经引进了参考区间作为分层结构。如果不能保证区间A与C之间存在一条弧的话,那么就不能够利用区间A和B以及区间B和C之间的约束关系由算法推导出A和C之间的关系。换言之,在一个由节点组成的子网络中进行约束传递的先决条件是,必须存在一个以这三个结点为顶点的弧三角形。

(3) 当用ATMS来进行非单调推理约束传递时,必须作出每条弧上新的时序约束是怎样得来的证实。这些证实总是依赖于含有那条弧的三角形中三条弧上所对应的三个当

前时序约束集,它们被作为一个整体来看待,不可能从它们当中区分出哪一个时序约束是造成新约束的主要原因。

鉴于以上理由,我们必须研究一种方法来识别出正在考虑的网络中所有三角形。

在我们的算法中,每一个时间区间都被作为网络中的一个结点。它可以从属于一个或者多个参考区间。由于参考区间本身也是区间,所以它们也可以有自己的参考区间,从而形成一种分层结构。我们定义两个区间A和B是相互可比较的,如果它们有相同的参考区间或者它们中一个是另一个的参考区间。只有可比较的区间在网络中才存在一条连接它们的弧。当用户将一个区间,比如A,及其参考区间输入时,系统为区间A在数据结构INTERVALS表中建立一项,其形式为:

[A REFS(A) COMPARABLE-LIST(A)]

其中,REFS(A)是由A的所有参考区间所组成的数据集,而数据集COMPARABLE-LIST(A)中每一个元素是A的参考区间。根据可比较性定义,如果区间B是COMPARABLE-LIST(A)中的一个元素,那么区间B与A之间一定存在一条弧(A, B)。网络中所有的弧都将被放置在一个数据集ARCLIST里面。现在,我们来考察COMPARABLE-LIST(A)与COMPARABLE-LIST(B)的交集。容易看出,这个交集集中的每一个元素,例如区间C,一定分别与A及B以弧连接。因此,便得到一个三角形ABC(图2)。用这种方法我们最终就能够得到网络中所有的三角形,并且组成一个数据集TRIANGLE-LIST。

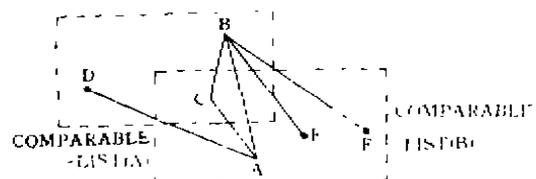


图2 一个三角形的生成

2. 时序约束的传递

我们仍然以图1中的三角形ABC为例。

在时序约束传递算法中,下面定义的关系向量 V_1 的子集 θ , V_2 的子集 φ 以及 V_3 的子集 ψ 起着关键的作用。

定义1 x 是 θ 中的一个元素,当且仅当在 V_2 中存在一个元素 y ,使得 x 和 y 的乘积与关系向量 V_3 的交集 σ 非空即:

$$\sigma = (x \times y) \cap V_3 \neq \emptyset$$

当从 V_1 中逐个取出 x ,并且轮流地与 V_2 的每个元素 y 相乘,然后将乘积与 V_3 相交得出 σ 之后,将所满足上述条件的 x , y 及 σ 分别取并集,便可得到 V_1 的子集 θ , V_2 的子集 φ 和 V_3 的子集 ψ 。

根据定义1,我们在下面给出一种基于三角形的时序约束传递算法。对于任意的原语关系 r_1 和 r_2 ,设 $T(r_1, r_2) = r_1 \times r_2$ 是传递表中相应的一项。令 ϕ 是一个空集, $R(i, j)$ 是弧 (i, j) 上的当前时序约束关系, \cap 表示集合的交集, \cup 表示集合的并集。网络中所有弧上的时序约束一开始全部缺省地被定义为 $[\lt, =, \gt, d, di, \dots, s, si]$,即由所有十三个原语关系所组成的一个关系向量。数据集 *quene* 不仅用来记录约束传递过程中约束关系发生了变化的弧,而且也用来记录变更假设时任何约束关系发生变化的弧。于是,用来更新时间网络的算法如下:

```

TO CLOSE
/*计算输入到数据库中的断言族*/
/*Find-triangles(i, j)的作用是产生由所有含有弧(i, j)的三角形所组成的LIST */
Unless quene is empty do
  Got next(i, j) from quene;
  Find-triangles(i, j)→List;
  Propagation(i, j);
end Unless;
TO Calculate-triangles (R1, R2, R3)
/*变元R1, R2和R3分别是三角形里三条弧上当前约束关系 */
 $\phi \rightarrow \theta$ ;  $\phi \rightarrow \varphi$ ;  $\phi \rightarrow \psi$ ;
for each r1 in R1 do
  for each r2 in R2 do
    R3 ∩ T(r1, r2) →  $\sigma$ ;
    if  $\sigma \neq \phi$  then
      [r1] ∪  $\theta \rightarrow \theta$ ; [r2] ∪  $\varphi \rightarrow \varphi$ ;  $\sigma \cup \psi \rightarrow \psi$ ;
    endif;
  endfor
endfor
TO Propagation(A, B)

```

```

/*当区间A和B之间的约束关系发生变化时,调用此函数将这种变化传递给所有有关的区间 */
for each triangle ABC in LIST do
  calculate-triangles(R(A, B), R(B, C), R(A, C));
  if  $\theta = \phi$  or  $\varphi = \phi$  or  $\psi = \phi$  then {signal contradiction};
  elseif  $\theta \neq R(A, B)$  then add (A, B) to quene;  $\theta \rightarrow R(A, B)$ ;
  elseif  $\varphi \neq R(B, C)$  then add (B, C) to quene;  $\varphi \rightarrow R(B, C)$ ;
  elseif  $\psi \neq R(A, C)$  then add(A, C) to quene;  $\psi \rightarrow R(A, C)$ ;
endif
endfor

```

3. 算法分析

不难证明,定义1中的 φ 就是(1)中的 V_2' , ψ 就是(2)中的 V_3' ^[1]。这意味着我们的算法与Allen的算法是兼容的。可是,新算法还具有许多其它优点。首先,由于它采用了ATMS一种基于假设的真值维护系统,所以提供了一种灵活的交互方式,用户可以根据需要将弧上的时序约束指定为前提或者假设。所谓前提是指一个普遍成立的断言,是不能变更的;而假设是一个没有逻辑基础的任意假定,以后是可以变更的。变更一个假设不一定是由于它在逻辑上导致了不相容性,也许只是想考察一下变更假设会造成哪些不同。这样一来,人们不仅可处理当前的世界,而且可以处理过去的或者未来的世界,这只要通过变更弧上的断言即可。其次,时序约束传递将以更加平衡的方式进行,无论对单调推理或是对非单调推理,在网络中考虑参考区间或者不考虑参考区间,在每一个约束传递过程中,我们都全面地考虑到一个三角形里三条弧上的时序约束的相互制约关系,从而达到一个更为精细的标志法,即每条弧上将不含有冗余的时序约束出现。最后,新算法的效率比Allen的算法要高,因为它将约束传递过程变得更紧凑了。事实上,在[12]中已表明Allen算法的时间复杂度 $O(n^3)$ 次向量运算,其中一半是乘法运算。与加法运算比较起来,执行乘法运算所消耗的时间要多得多。因此,加快传递过

程的关键是减少所需要的乘法次数。而新算法已经将以前的乘法次数减少了50%。显然效率提高是很明显的。

四 ATMS的应用

ATMS是由 J.de Kleer提出的一种基于假设的真值维护系统，它作为整个推理系统的“管家”子系统。一般来说，一个推理系统可以看成由两部份组成：问题求解系统和TMS系统（图3）。问题求解系统把它所作的推理传递给TMS，然后TMS应用这些推理的结构来组织问题求解系统的信念。问题求解系统通过询问TMS将获得可以相信的那些信息。这两个子系统之间有各种各样的交互方式，可是基本上TMS总是提供不相容性所依据的信念以及相互相容的信念集合的有关信息。

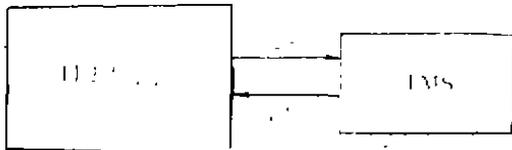


图3 一个推理系统

与 Doyle 的基于证实的真值维护系统 (JTMS) 不同^[9]，de Kleer的 ATMS 是基于假设的。在ATMS里，每个数据最终所依赖的假设集将被识别，并且与它一起被记录下来，而且，ATMS是多上下文的，我们同时可以作几个假设（它们可以是互相矛盾的），看看会发生什么情况。ATMS可以告诉问题求解系统，什么时候一个上下文会变成不相容的，一个节点是否在具体的上下文里成立等等。我们在时序推理系统TRS的设计中，利用了ATMS的这些特点，达到了以下几个目的：

(1) 当网络中约束传递出现不相容性时，TRS会通知用户引起这种现象的根本原因，或者是由于作了某些假设集，或者前提本身是矛盾的。

(2) 能够随时了解某条弧上的约束关系最终依赖于我们所作的假设集中哪几个假

设。

(3) 当我们更改弧上的约束关系，并且对新网络进行约束传递前，能够确定这种更改所带来的影响，尽可能多地利用以前那个网络的时序信息，避免重复性的工作。

(4) 为了克服 Allen 算法的不完全性，我们在得到网络的一个不完全解后，将采用宽度优先搜索法，使得每条弧上的约束关系仅为一个原语关系，这时我们仍称之为网络的完全解。如果不能成功的话，则可以检测出输入中的不相容性。不过，在进行宽度优先搜索时，由于不断地调用约束传递过程，所以目标状态通常都能较快地达到。我们知道，ATMS 是最适宜在宽度优先搜索中使用的^[10]。

五 时序推理系统TRS的实现

下面我们简单介绍一下时序推理系统 TRS 的结构。它主要由五大部份组成(图4)

人机接口 (MMI)：用户将各个区间及其参考区间，以及已知的部分区间之间的时序约束输入系统，这时系统就接受了一个问题。用户通过接口，可以随时与推理系统进行交互作用，给予指示。

网络生成模块 (NGM)：根据输入的数据生成一个时间网络，并且找出网络中所有的三角形。这时，系统形成一个可能世界的部分描述，它由用户所输入的区间以及它们

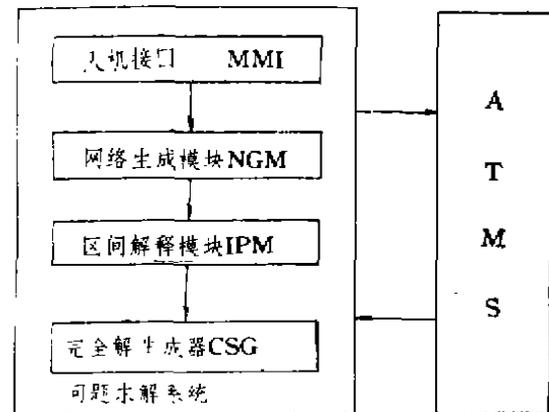


图4 时序推理系统TRS的结构

可演绎系统的研究

李磊 (中山大学岭南学院计算机系 广州510275)

J. Kouloumdjian (法国里昂INSA信息科学实验室)

koulo, J

TP311.13

摘要: 本文综述了可演绎系统目前的研究情况, 认为以DATALOG为代表的演绎数据库系统进化为实用系统是困难的。文中简介了一个基于PROLOG的可实用的可演绎系统, 还对进一步的研究内容及其原因进行了讨论。

考虑计算机的应用历史大致可分为三个阶段: 科学计算、数据处理和知识处理。每个阶段都有对应的工具系统, 它们分别是: 算法语言, 数据库系统和知识处理系统。目前, 知识处理系统还仍然处于发展阶段, 并且成为计算机科学中非常活跃的研究领域。

处理对象不同决定了处理工具特性的不同, 例如, 算法语言和关系数据库系统分别对于科技计算和数据处理是十分适宜的, 但是目前对于知识处理来说还缺乏如此适宜的工具系统。五花八门的知识处理系统和干脆用传统工具来处理知识就说明了这一点。

知识的最基本操作之一是演绎, 因此知识处理系统应具有的最基本功能之一也应当是演绎, 我们把这种具有演绎功能的系统称为可演绎系统。本文首先介绍这种系统的研究与发展, 然后介绍一个实际的可演绎系统。

一、概况

知识系统的广泛应用大大地刺激了可演

之间的时序约束所组成。

区间解释模块 (IPM): 在网络中利用基于三角形的约束传递算法生成一个不完全解, 即对那个可能世界中的每个区间进行解释, 使得网络中任意一条弧上都有一个非空的原语关系集。这时也许会报告矛盾出现的信息。

完全解生成器 (CSG): 它的主要功能

是研究系统的研究与发展, 主要方向集中于: 演绎数据库和逻辑程序设计。

关系数据库有一个逻辑解释, 还可以定义视图(View), 因此可以认为关系数据库系统是非常初等的可演绎系统。这一点强烈地引诱着数据库领域的专家来扩充这种演绎能力, 直至成为一个实用的可演绎系统。这就是目前研究的演绎数据库, 在这一领域中人们已经取得大量结果。

可以认为演绎数据库领域最基本的研究内容是对递归查询的研究。历届重要的关于数据库方面的国际会议上都有大量的论文讨论这一问题, 例如: 1990年8月16日的VLDB上发表的九篇关于演绎数据库的论文中, 有八篇是讨论递归查询的。这一状况说明了此问题的重要性, 同时也说明了这个问题还没有很好地获得解决, 虽然人们已经提出各

是根据不完全解生成网络的完全解 (可能有多个), 或者测试出整体的不相容性。

ATMS: 基于假设的真值维护系统。

· 本文所描述的时序推理系统 TRS 正在 VAX II/75 机上进行模拟试验, 效果良好, 预计能达到预定的设计目标。(参考文献 12 篇略)