

利用有向环的性质求解可达关系

陈秋茹¹ 文中华^{2,3} 袁润¹ 戴良伟¹

(湘潭大学信息工程学院 湘潭 411105)¹ (湖南工程学院计算机与通信学院 湘潭 411104)²

(湘潭大学智能计算与信息处理教育部重点实验室 湘潭 411105)³

摘要 不确定规划研究的最终目标是求出规划解,但是由于缺少引导信息,直接求规划解会导致大量的无用状态和动作被搜索。获得状态间的可达关系可以避免冗余计算。目前求可达关系的方法效率较低,因此设计了一种求可达关系的新方法。将不确定状态转移系统抽象成一个图,在这个图中,查找状态之间的可达信息是否形成一个有向环。若存在一个有向环,说明环内每两个状态之间都有可达关系。将其中一个状态作为父节点,并且将这个环内所有状态的可达关系记录在父节点中,通过访问父节点的可达信息更新环内状态的可达信息,减少了许多无用的状态和动作被搜索。实验结果表明,所设计的算法不仅能得到更全面的可达关系,而且效率也高于已有的算法。

关键词 不确定规划,可达关系,有向图

中图分类号 TP18 **文献标识码** A **DOI** 10.11896/j.issn.1002-137X.2016.4.041

Solving Reachability Relationship by Property of Directed Cycle

CHEN Qiu-ru¹ WEN Zhong-hua^{2,3} YUAN Run¹ DAI Liang-wei¹

(College of Information Engineering, Xiangtan University, Xiangtan 411105, China)¹

(Department of Computer & Communication, Hunan Institute of Engineering, Xiangtan 411104, China)²

(Key Laboratory of Intelligent Computing & Information Processing(Xiangtan University), Ministry of Education, Xiangtan 411105, China)³

Abstract The goal of non-deterministic planning is getting the planning solution. Due to the lack of the helpful information, searching the problem space directly leads to a large number of useless work, which can be reduced dramatically by capturing the reachability relationship between states. But current methods perform poorly with respect to efficiency, thus we designed a new algorithm. We built a graph from the non-deterministic system, and checked whether there are some paths between states leading to some cycles. We concluded that every two states in the cycle are mutually reachable, if such a cycle does exist. We could treat vertex as the parent, and tagged it with the reachability relationships. By using this relationships and updating the reachability information of the states in the cycle, we could prevent many useless states to be searched. The experimental results show that the designed algorithm not only gains more complete reachability relationships, but also outperforms the current algorithms in efficiency.

Keywords Non-deterministic planning, Reachability relationship, Directed graph

1 引言

在日常生活中,诸多领域都存在人为的或客观的不确定性^[1,2],而这些领域都存在着大量的优化问题。因此不确定环境下的优化理论具有理论价值和广泛的应用前景。我们称不确定环境下的优化理论为不确定规划^[3,4]。不确定规划是智能规划的一个重要分支,也是一个热点问题。在该领域内,有很多专家和学者取得了重要的成果^[5-8]。

在不确定状态转移系统中,不确定动作执行后可以到达多个状态,由于结果状态不明确造成大量重复访问无用的状态,降低了求解效率。如果能够快速又正确地找到引导信息,就能减少很多重复的动作。因此找到不确定状态转移系统中

状态之间的可达关系是很有意义的。针对这一问题,文献[9-15]提出了解决状态之间可达关系的方法。文献[9]通过建立不确定状态转移系统的超图的连接矩阵,并利用其得到状态之间的可达关系的一些重要性质,从而筛选状态动作序偶得到可达关系。文献[10]用矩阵模拟不确定状态转移系统,每个状态记录了其他状态到达该状态的可达信息,通过状态之间的可达信息的传递,求得不确定系统的状态可达关系。文献[12]采用不确定状态转移系统的邻接矩阵与第*i*步的可达矩阵相乘得到第*i+1*步的可达矩阵,从而统计不确定动作信念状态的可达关系。文献[14]的主要思想是用矩阵乘来模拟每一步的状态转移,对不确定动作带来的扩展和确定关系带来的聚合进行了统计和处理,使用矩阵的乘法运算计算状态

到稿日期:2015-02-05 返修日期:2015-05-16 本文受国家自然科学基金(61272295,61105039,61202398),湘潭大学智能计算与信息处理教育部重点实验室,湖南省重点学科建设项目(0812)资助。

陈秋茹(1991-),女,硕士生,主要研究方向为智能规划、图论,E-mail:150480431@qq.com;文中华(1966-),男,博士,教授,博士生导师,主要研究方向为智能规划、图论及算法;袁润(1992-),女,硕士生,主要研究方向为智能规划;戴良伟(1989-),男,硕士生,主要研究方向为智能规划。

转移,最终求得状态可达性关系。

但是当不确定状态系统中的状态数目较多时,以往的算法效率较低,具有一定的局限性。本文对如何快速得到状态可达关系展开研究,并提出了一种新方法。在不确定状态转移系统中寻找状态间的可达信息是否形成有向环。有向环具有回路的性质,环内状态两两相互可达。选择环内其中一个状态作为父节点,其收集同一个环内其他状态的可达信息,并共享可达信息给同一个环内的其他状态。每个状态能根据父节点的可达信息得到与其他状态的可达关系,减少许多搜索动作。该方法不但可以准确地求解状态之间的可达信息,而且还减少了冗余的操作,从而提高了效率。

2 相关定义

定义 1^[9] 一个规划领域是一个不确定的状态转移系统 $\Sigma=(S,A,\gamma)$,其中 S 是有限状态集, A 是有限动作集, $\gamma:S \times A \rightarrow 2^S$ 是状态转移函数。

γ 用来刻画不确定性:在状态 s 下执行动作 a 所得到的状态集合就是 $\gamma(s,a)$ 。若 $\gamma(s,a)$ 非空,则称动作 a 在状态 s 下是可执行的。

定义 2^[12] 在一个不确定状态转移系统中,从状态 s_i 开始,执行 $\gamma(s_i, a_x)$,之后可能到达状态集合 $S_L = \{s | s \in \gamma(s_i, a_x)\}$ 中的一个状态,当 $s_j \in S_L$,且 $|S_L|=1$,则称 s_i 到 s_j 确定可达;当 $s_j \in S_L$,且 $|S_L|>1$,则称 s_i 到 s_j 为不确定可达; $s_j \notin S_L$ 时,则称 s_i 到 s_j 为不可达。

定义 3 设 $\Sigma=(S,A,\gamma)$ 是一个不确定状态转移系统,两个状态之间的可达关系用函数 $g(s_i, s_j)$ 来表示,函数 $g(s_i, s_j)$ 的定义如下:

- $g(s_i, s_j)=0$, s_i 到 s_j 为不可达;
- $g(s_i, s_j)=1$, s_i 到 s_j 为确定可达;
- $g(s_i, s_j)=2$, s_i 到 s_j 为不确定可达;
- $g(s_i, s_i)=1$, 状态自己到自己为确定可达。

定义 4 设 $\Sigma=(S,A,\gamma)$ 是一个不确定状态转移系统, $g(s_i, s_j)=T$, 状态 s_u 为状态 s_i 的父节点。

- $\forall T \in \{0, 1, 2\}$, $g(s_u, s_j)=1$, 则 $g(s_i, s_j)=1$;
- $\forall T \in \{0, 2\}$, $g(s_u, s_j)=2$, 则 $g(s_i, s_j)=2$;
- $\forall T \in \{1\}$, $g(s_u, s_j)=2$, 则 $g(s_i, s_j)=1$;
- $\forall T \in \{0, 1, 2\}$, $g(s_u, s_j)=0$, 则 $g(s_i, s_j)=T$ 。

定义 5 在一个不确定状态转移系统中,用 $reachque_{ik}$ 表示状态 s_i ($1 \leq i \leq n$) 到状态 s_k ($1 \leq k \leq n$) 的可达信息,用 $reached_{ij}$ 表示状态 s_i ($1 \leq i \leq n$) 被 s_j ($1 \leq j \leq n$) 到达的可达信息。由 $reachque_{ik} = 1 \ \& \ reached_{ij} = 1$ 可得 $reached_{kj} = 1$, $reachque_{jk} = 1$ 。

3 求状态可达关系的新方法

把不确定状态转移系统抽象成一个图。本文所提的求可达关系的方法的主要思想是先收集确定动作的信息,显然,确定动作的起始状态到终止状态是确定可达的,根据定义 5 更新状态间的可达关系,只需进行简单处理。接着处理不确定动作的信息时分两种情况。第一种情况,若状态 u 是不确定动作 a 的起始状态,且不确定动作 a 的所有终止状态都能确定可达状态 v ,则状态 u 是确定可达状态 v 。第二种情况,在这个图中,寻找状态间的信息传递所形成的环,即每两个状态都有不确定或者确定可达关系。同时,根据信息的传递性,环

内的状态可以共享可达信息,能减少很多冗余的搜索。若存在信息传递所形成的环,则将最早出现在这个环中的状态 i 作为其它状态的父节点,并将其它状态的可达信息传递给父节点,更新状态 i 的 $reachque_i$ 和 $reached_i$ ($1 \leq j \leq n$)。此时,父节点拥有整个环的可达信息,环内的状态可以通过访问父节点的可达信息更新与其他状态的可达关系,达到共享信息的作用。若两个环内的某些状态的信息传递形成第三个环,可将两个环合并,选择两个环的其中一个父节点作为新环的父节点。算法流程如图 1 所示。

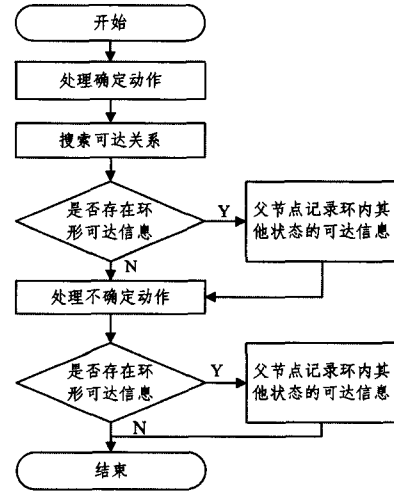


图 1 算法流程

4 状态可达性算法及分析

设 $\Sigma=(S,A,\gamma)$ 是一个不确定状态转移系统, S 中有 n 个状态,用矩阵 $reachable$ 表示不确定转移系统 $\Sigma=(S,A,\gamma)$ 的可达关系。处理确定动作的函数如下:

1. Function Action()
2. for $i=1$ to m
3. if ($num==1$)
4. $graph[u][++graph[u][0]]=v$
5. else $++mulm$
6. $mulAction[mulm][++mulAction[mulm][0]]=u$
7. for $j=1$ to num
8. $mulAction[mulm][++mulAction[mulm][0]]=v$
9. $addedge(v, mulm)$;
10. next j
11. next i
12. for $j=1$ to n
13. if ($visit[j]==0$)
14. $dfs(j, -1)$
15. next j

第 2—11 行,对确定状态转移系统进行初始化。如果该动作是确定的,将该动作保存于 $graph$ 数组;如果该动作是不确定的,将该动作存在 $mulAction$ 数组。 $graph[i][0]$ 和 $mulAction[i][0]$ 代表第 i 个动作到达的状态数; $mulAction[i][1]$ 代表的是初始状态的编号。 $mulAction[i][2 \dots num+1]$ 代表的是所有终止状态的编号。将 $mulm$ 加入 $saHead[v]$ 链表,也就是说能够通过 $saHead[v]$ 找到 $mulm$,这样记录是为了保证后面处理不确定动作时能够进行逆向查找,这里的 v 都是一个动作的终止状态。第 12—15 行,对每一个未访问过的状态进行一次搜索。通过信息的传递性,更新状态之间

的确定可达信息。

处理不确定动作的函数如下：

```

1. Function ActionUnCertain(reachable, u)
2. for i=1 to n
3.   for j=2 to mulaction[act][0]
4.     v=mulAction[act][j]
5.     if(状态 u 所有动作的终止状态 v 确定可达状态 i)
6.       更新状态 u 和状态 i 的可达关系
7.       reached[i][++reached[i][0]]=u
8.     next j
9.   next i
10. for m=1 to reached[u][0]
11.  更新可达状态 u 的状态 m 与状态 i 的可达关系
12.  for j=saHead[p] to saEdge[j].next
13.    actionQue.push(saEdge[j].act)
14.    visit[saEdge[j].act]=true
15.  next j
16. next m
17. 将剩下未处理的动作全都加入图中
18. for i=1 to n
19.  if(visit[i]==0)
20.    dfs(i, -1);
21. next i;

```

第 3—4 行, 如果状态 u 或者状态 u 的父节点确定可达状态 i , 则跳过。第 5—8 行, 当状态 u 所有动作的终止状态都确定可达状态 i , 则说明状态 u 也是确定可达状态 i 。第 9—15 行, 如果状态 u 确定可达状态 i , 则所有可达到状态 u 的状态 m 也确定可达状态 i 。更新数组 $reachque$, 则更新状态 u 确定可达状态 i , 更新状态 u 的父节点即状态 $id[u]$ 也确定可达状态 i , 则更新状态 i 被状态 u 和状态 $id[u]$ 可达。如果不确定动作 $saEdge[j]$ 没有加入队列, 则将它加入队列。第 16 行, 这里将所有保存在 $mulAction$ 中的不确定可达关系全部放入 $graph$ 中。之前 $graph$ 数组存储的只有确定可达关系。第 17—21 行, 执行一次 dfs 就可以得到所有状态之间的可达关系。

搜索状态间可达信息形成的有向环的函数如下：

```

1. Function dfs(u, parent)
2. for i=1 to graph[u][0]
3.  if (visit[v]==0)
4.    dfs(v, u);
5.  if(ancOrd[v] > frtOrd[u] || (frtOrd[v] < frtOrd[u] && visit[v]==2))
6.    for j=1 to reachque[v][0]
7.      r=reachque[v][j];
8.      reachable[u][reachque[v][j]]=reachFlag;
9.      更新 reachqueu 和 reachedr 的信息
10.     next j;
11. next i;
12. if(frtOrd[u]==ancOrd[u])
13.  while(stk[oldTop] != u)
14.    --oldTop;
15.  for i=oldtop+1 to top
16.    r=reachque[stk[i]][j];
17.    reachable[u][r]=reachFlag;
18.    更新 reachqueu 和 reachedr 的信息
19.  next i;

```

```

20. for j=oldtop+1 to top
21.  merge(stk[j], u);
22.  next j;

```

$reachque[i][2 \cdots num+1]$ 为状态 i 可达的状态, $reached[i][2 \cdots num+1]$ 为可达到状态 i 的状态。在这里维护两个数组, 一个是 $frtOrd[1..n]$, 一个是 $ancOrd[1..n]$, 其中 $frtOrd[i]$ 表示状态 i 开始访问时间, $ancOrd[i]$ 为与状态 i 邻接的状态未删除状态 j 的 $ancOrd[j]$ 和 $ancOrd[i]$ 的最小值 ($ancOrd[i]$ 初始化为 $frtOrd[i]$)。在深度优先遍历时维护一个堆栈, 每次访问一个新状态 i , 就压入堆栈。同时, 当前状态是这个有向环中最早被压入堆栈的, 那么在当前状态以后压入堆栈的并且仍在堆栈中的状态都属于这个有向环。在一次深搜的回溯过程中, 如果发现 $ancOrd[i] = frtOrd[i]$, 说明状态的可达信息形成一个有向环, 那么当前顶点就是这个有向环的父节点, 将 stk 数组存储的所有状态的可达关系都传给状态 u 。那么, 只要确定每个环中父节点, 然后根据这些父节点共享的可达关系更新状态的可达关系即可。

5 算法示例及实验分析

设 $\Sigma = (S, A, \gamma)$ 是一个不确定状态转移系统, 如图 2 所示。该不确定状态转移系统中, 有 7 个状态, 11 个动作。其中, T_1 为状态 s_2 的不确定的动作, T_2 为状态 s_1 的不确定的动作, T_3 为状态 s_3 的不确定的动作, T_4 为状态 s_5 的不确定的动作。

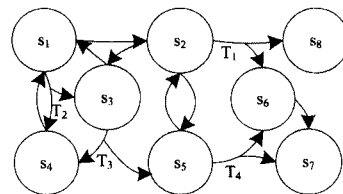


图 2 不确定状态转移系统

首先采用提出的算法求该不确定状态系统的可达关系。根据上述的算法分析可知, 状态到状态自身是确定可达的, 即 $reachable[i][i]=1$ 。算法分两部分求可达关系: 首先, 根据算法将确定动作抽象形成一个图, 可得 $reachable[s_1][s_2]=1$, 即状态 s_1 确定可达状态 s_2 。同理 $reachable[s_4][s_1]=1$, $reachable[s_2][s_5]=1$, $reachable[s_5][s_2]=1$, $reachable[s_6][s_7]=1$, $reachable[s_2][s_3]=1$, $reachable[s_3][s_1]=1$ 。

由于状态 s_2 、状态 s_3 和状态 s_1 形成环, 因此状态 s_1 、状态 s_2 和状态 s_3 所有可到达的状态, 环内的其他状态也可以到达。合并状态 s_2 、状态 s_1 、状态 s_3 的可达信息赋给状态 s_1 , 且将状态 s_1 标记为状态 s_2 、状态 s_1 、状态 s_3 的父节点。即更新可达信息 $reachable[s_1][s_5]=1$ 。要查询状态 s_2 、状态 s_3 的可达信息则访问状态 s_1 即可。由于 $reachable[s_4][s_1]=1$, $reachable[s_5][s_2]=1$, 分别更新状态 s_4 、状态 s_5 的可达信息, 即 $reachable[s_4][s_5]=1$, $reachable[s_4][s_3]=1$, $reachable[s_4][s_2]=1$, $reachable[s_5][s_1]=1$, $reachable[s_5][s_3]=1$ 。

将不确定动作加入图中。如图 1 所示, 该不确定状态转移系统有 4 个不确定动作 T_1, T_2, T_3, T_4 。对于 T_4 有 $reachable[s_5][s_7]=2$, $reachable[s_5][s_5]=2$ 。由于 $reachable[s_6][s_7]=1$, 且状态 s_5 的所有确定与不确定可达状态都确定可达状态 s_7 , 因此状态 s_5 确定可达状态 s_7 。更新状态 s_5 与状态 s_7 的可达信息, 即 $reachable[s_5][s_7]=1$ 。同时, 由于 $reachable[s_2][s_5]=1$, 根据传递性有 $reachable[s_2][s_7]=1$, 因此更新

状态 s_2 和状态 s_7 之间的可达关系,即 $reachable[s_2][s_7]=1$ 。同理,由于 $reachable[s_4][s_5]=1, reachable[s_1][s_5]=1, reachable[s_3][s_5]=1$,根据传递性有 $reachable[s_5][s_7]=1$,因此分别更新状态 s_1 、状态 s_4 、状态 s_3 与状态 s_7 之间的可达关系,即 $reachable[s_4][s_7]=1, reachable[s_1][s_7]=1, reachable[s_3][s_7]=1$ 。对于 T_3 有 $reachable[s_3][s_4]=2, reachable[s_3][s_5]=2$,但因为 $reachable[s_3][s_5]=1$,所以不需要更新此关系。

至此,我们已经处理完不确定动作的确定可达的情况,接下来处理不确定可达的情况。将不确定动作加入数组 $graph$ 。根据算法得知,最后执行一次 $solveReach$ 函数。对于 T_1 有 $reachable[s_2][s_6]=2, reachable[s_2][s_8]=2$,可得 $reachable[s_4][s_6]=2, reachable[s_4][s_8]=2, reachable[s_1][s_6]=2, reachable[s_1][s_8]=2, reachable[s_3][s_6]=2, reachable[s_3][s_8]=2, reachable[s_5][s_8]=2$ 。对于 T_2 有 $reachable[s_1][s_4]=2, reachable[s_1][s_3]=1$,可得 $reachable[s_2][s_4]=2, reachable[s_5][s_4]=2$ 。

所以,该不确定状态转移系统的可达关系如下所示:

$$reachable = \begin{bmatrix} 1 & 1 & 1 & 2 & 1 & 2 & 1 & 2 \\ 1 & 1 & 1 & 2 & 1 & 2 & 1 & 2 \\ 1 & 1 & 1 & 2 & 1 & 2 & 1 & 2 \\ 1 & 1 & 1 & 1 & 1 & 2 & 1 & 2 \\ 1 & 1 & 1 & 2 & 1 & 2 & 1 & 2 \\ 0 & 0 & 0 & 0 & 0 & 1 & 1 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 1 \end{bmatrix}$$

以下为矩阵相乘法、信息传递法与本文提出的求状态可达关系的算法的实验结果的比较。实验环境均为 Windows7 + Core(TM) i3-3220 3.3GHz + 4.0GB 内存 + VC6。3 个算法使用的数据相同,输入输出运行时间没有包括在内,运行时间比较如表 1 所列。

表 1 运行时间比较

状态数	矩阵相乘法(s)	信息传递法(s)	环形共享信息法(s)
10	0.015	0.004	0.000
20	0.045	0.009	0.000
30	0.071	0.013	0.000
50	0.887	0.050	0.000
70	4.308	0.054	0.001
90	15.442	0.073	0.001
110	30.264	0.070	0.001
130	42.551	0.101	0.001
150	256.473	0.169	0.002
170	575.141	0.197	0.003
190	956.552	0.207	0.004
200	1077.393	0.226	0.004

由表 1 可知,当不确定状态转移中的状态数目较少时,矩阵相乘法、信息传递法、环形共享信息法的运行时间都很短,此时不能显示出环形共享信息法的优势。当不确定状态系统中的状态数目较多时,环形共享信息法所消耗的时间较短,此时环形共享信息法具有很大的优势,它在求解可达关系时具有更高的效率。随着状态数的增长,由于时间复杂度不同,矩阵相乘法和信息传递法的运行时间增长速度较快,而环形共享信息法的运行时间则增长平缓。

结束语 本文设计了一种快速求解不确定系统中状态之间可达关系的新方法,其结合了信息的传递性和有向环的特

性,减少许多无用状态和动作被搜索,能准确又快速地找到可达关系。实例证明对于稠密不确定状态转移系统,本文设计的算法效率更高。在求状态可达关系问题上还有很多工作有待研究,如:1)设计更快速求解可达关系的算法;2)将方法在不确定规划器及不确定规划基准领域上进行实验。

参考文献

- [1] Jameson A. Numerical uncertainty management in user and student modeling: An overview of systems and issues[J]. User Modeling and User-Adapted Interaction, 1995, 5(3): 193-251
- [2] Vinnicombe G. Frequency domain uncertainty and the graph topology[J]. IEEE Transactions on Automatic Control, 1993, 38(9): 1371-1382
- [3] Peng Jin, Liu Bao-ding. Uncertain Programming: Current Status and Future Prospects[J]. Operations Research and Management Science, 2002, 11(2): 1-10 (in Chinese)
彭锦,刘宝碇. 不确定规划的研究现状及其发展前景[J]. 运筹与管理, 2002, 11(2): 1-10
- [4] Liu Bao-ding, Zhao Rui-qing. Uncertain Programming and Further Research Problems[J]. Journal of Institute of Command and Technology, 1999, 10(6): 102-105 (in Chinese)
刘宝碇,赵瑞清. 不确定规划及进一步的研究问题[J]. 指挥技术学院学报, 1999, 10(6): 102-105
- [5] Mattmüller R, Nebel B. Informed Progression Search for Fully Observable Nondeterministic Planning[D]. University at sbibliothek Freiburg, 2013
- [6] Bertoli P, Cimatti A, Roveri M, et al. Planning in nondeterministic domains under partial observability via symbolic model checking[C]//IJCAI, 2001. 2001, 473-478
- [7] Da Silva F A G, Ciarlini A E M, Siqueira S W M. Nondeterministic planning for generating interactive plots[M]. Springer Berlin, Heidelberg, 2010: 133-143
- [8] Fu J, Ng V, Bastani F B, et al. Simple and fast strong cyclic planning for fully-observable nondeterministic planning problems[J]. IJCAI Proceedings-International Joint Conference on Artificial Intelligence, 2011, 22(3): 1949
- [9] Wen Zhong-hua, Huang Wei, Liu Ren-ren, et al. Research on State Reachability in Planning Based on Model Checking[J]. Chinese Journal of Computers, 2012, 35(8): 1634-1643 (in Chinese)
文中华,黄巍,刘任任,等. 模型检测规划中的状态之间的可达关系研究[J]. 计算机学报, 2012, 35(8): 1634-1643
- [10] Lao Jia-qi, Wen Zhong-hua, Wu Xiao-hui, et al. Method of Information Delivery for State Accessibility in Non-Determinate System[J]. Computer Science, 2014, 41(10): 266-269 (in Chinese)
劳佳琪,文中华,伍小辉,等. 信息传递法求不确定系统中的状态可达关系[J]. 计算机科学, 2014, 41(10): 266-269
- [11] Yuan Ye, Wang Guo-ren. Answering threshold-based reachability queries over probabilistic graphs[J]. Chinese Journal of Computers, 2010, 33(12): 2219-2228
- [12] Huang Li-fang, Wen Zhong-hua, Hu Yu-long, et al. Method of getting circular reachability relation in non-determinate panning[J]. Application Research of Computers, 2013, 30(9): 2689-2693 (in Chinese)
黄丽芳,文中华,胡雨隆,等. 不确定规划中状态循环可达关系的求解方法[J]. 计算机应用研究, 2013, 30(9): 2689-2693

$$(1) \overline{\overline{PR_{\sum_{i=1}^s R_i} (PR_{\sum_{i=1}^s R_i} (X))}} = \overline{\overline{PR_{\sum_{i=1}^s R_i} (X)}}$$

$$(2) \overline{\overline{PR_{\sum_{i=1}^s R_i} (PR_{\sum_{i=1}^s R_i} (X))}} = \overline{\overline{PR_{\sum_{i=1}^s R_i} (X)}}$$

证明: (1) 对 $\forall x \in U$, 由定义 3 有

$$\begin{aligned} & \overline{\overline{PR_{\sum_{i=1}^s R_i} (PR_{\sum_{i=1}^s R_i} (X))}}(x) \\ &= \bigwedge_{i=1}^s \{ \bigwedge \{ PR_{\sum_{i=1}^s R_i} (X)(y) | y \in [x]_{R_i} \} \} \\ &= \bigwedge \{ \overline{\overline{PR_{\sum_{i=1}^s R_i} (X)(y) | y \in [x]_{R_{\max}}}} \} \quad (*) \end{aligned}$$

由引理 2 可得

$$\begin{aligned} \overline{\overline{PR_{\sum_{i=1}^s R_i} (X)(y)}} &= \bigwedge_{i=1}^s \{ \bigwedge \{ X(z) | z \in [y]_{R_i} \} \} \\ &= \bigwedge \{ X(z) | z \in [y]_{R_{\max}} \} \\ &= \bigwedge \{ X(y) | y \in [x]_{R_{\max}} \} \quad (**) \end{aligned}$$

结合 (*) 和 (**) 两式, 可得

$$\begin{aligned} & \overline{\overline{PR_{\sum_{i=1}^s R_i} (PR_{\sum_{i=1}^s R_i} (X))}}(x) \\ &= \bigwedge \{ X(y) | y \in [x]_{R_{\max}} \} \\ &= \bigwedge_{i=1}^s \{ \bigwedge \{ X(y) | y \in [x]_{R_i} \} \} \\ &= \overline{\overline{PR_{\sum_{i=1}^s R_i} (X)(x)}} \end{aligned}$$

(2) 仿(1)的证明可得。

定理 10 设 (U, \mathbb{R}) 是一个近似空间, R_1, R_2, \dots, R_s 是具有最大元的等价关系, 则 $(F(U), \cap, \cup, \sim, \overline{\overline{PR_{\sum_{i=1}^s R_i}}, \overline{\overline{PR_{\sum_{i=1}^s R_i}}})}$

构成模糊粗糙集代数。

证明: 仿定理 7 的证明可得。

同理, 还有以下定理。

定理 11 设 (U, \mathbb{R}) 是一个近似空间, R_1, R_2, \dots, R_s 是具有最大元的等价关系, 则 $(F(U), \cap, \cup, \sim, \overline{\overline{PR_{\sum_{i=1}^s R_i}}, \overline{\overline{PR_{\sum_{i=1}^s R_i}}})}$

$\dots \overline{\overline{PR_{\sum_{i=1}^s R_i}}, \overline{\overline{PR_{\sum_{i=1}^s R_i}}}, \overline{\overline{PR_{\sum_{i=1}^s R_i}}}, \dots \overline{\overline{PR_{\sum_{i=1}^s R_i}}})}$ 构成模糊粗糙集代数。

结束语 经典的 Pawlak 粗糙集代数系统已经讨论得比较清楚, 理论比较完善。徐伟华等在文献[18]中讨论了多粒化的模糊粗糙集的有关理论和性质, 并得到了许多有意义的结果。而本文则在徐伟华等人工作的基础上进一步讨论了多粒化的模糊粗糙集代数系统。为了得到更为满意的性质, 我们引入具有最小(大)元的等价关系, 讨论了基于具有最小(大)元等价关系的多粒化模糊粗糙集代数系统的性质, 得到了诸多漂亮的结论, 丰富和完善了相关的理论。

参考文献

[1] Boixader D, Jacas J, Recasens J. Upper and lower approximations of fuzzy sets [J]. International Journal of General Sys-

tems, 2000, 29; 555-568

[2] Dubois D, Prade H. Rough fuzzy sets and fuzzy rough sets[J]. International Journal of General Systems, 1990, 17; 191-209

[3] Inuiguchi M. Generalizations of rough sets; from crisp to fuzzy cases[C]//Tsumoto S, Slowinski R, Komorowski J, et al., eds. RSCTC 2004. 2004, 3066; 26-37

[4] Mi J S, Leung Y, Wu W Z. An uncertainty measure in partition-based fuzzy rough sets[J]. International Journal of General Systems, 2005, 34; 77-90

[5] Mi J S, Zhang W X. An axiomatic characterization of a fuzzy generalization of rough sets [J]. Information Sciences, 2004, 160; 235-249

[6] Pei D W, Xu Z B. Rough set models on two universes[J]. International Journal of General Systems, 2004, 33; 569-581

[7] Slowinski R, Vanderpooten D. A generalized definition of rough approximations based on similarity[J]. IEEE Transactions on Knowledge and Data Engineering, 2000, 12; 331-336

[8] Comer S. An algebraic approach to the approximation of information[J]. Fundamenta Informaticae, 1991, 14; 492-502

[9] Lin T Y, Liu Q. Rough approximate operators; axiomatic rough set theory [C] // Ziarko W., ed. Rough sets, Fuzzy sets and Knowledge Discovery. Springer, Berlin, 1994; 256-260

[10] Pomykala J A. Approximation operations in approximation space [J]. Bulletin of the Polish Academy of Sciences, Mathematics, 1987, 35; 653-662

[11] Yao Y Y. Two views of the theory of rough sets in finite universes[J]. International Journal of Approximate Reasoning, 1996, 15; 291-317

[12] Yao Y Y. Constructive and algebraic methods of the theory of rough sets[J]. Information Sciences, 1998, 109; 21-47

[13] Wu W Z, Mi J S, Zhang W X. Generalized fuzzy rough sets[J]. Information Sciences, 2003, 151; 263-282

[14] Qian Y H, Liang J Y, Yao Y Y, et al. MGRS: A multi-granulation rough set[J]. Information Sciences, 2010, 180; 949-970

[15] Qian Y H, Liang J Y, Wei W. Pessimistic rough decision [C] // 2nd International Workshop on Rough Sets Theory. Zhoushan, 2010; 19-21

[16] Qian Y H, Liang J Y, Yao Y Y, et al. Incomplete multigranulation rough set[J]. IEEE Transactions on Systems, Man and Cybernetics, Part A, 2010, 20; 420-430

[17] She Y H, He X L. On the structure of the multigranulation rough set model[J]. Knowledge-based Systems, 2012, 36; 81-92

[18] Xu W H, Wang Q R, Zhang X T. Multi-granulation fuzzy rough sets in a fuzzy tolerance approximation space[J]. International Journal of Fuzzy Systems, 2011, 13(4); 246-259

[19] Zhang W X, Wu W Z, Liang J Y, et al. The theory and method of rough set[M]. Beijing: Science Press, 2011 (in Chinese)

张文修, 吴伟志, 梁吉业, 等. 粗糙集理论与方法[M]. 北京: 科学出版社, 2001

胡雨隆, 文中华, 常青, 等. 不确定规划中状态非循环可达关系的求解方法[J]. 计算机仿真, 2012, 29(5); 114-117

[15] Zhang Shuo, Gao Hong, Li Jian-zhong, et al. Efficient Query Processing on Uncertain Graph Databases[J]. Chinese Journal of Computers, 2009, 32(10); 2066-2079 (in Chinese)

张硕, 高宏, 李建中, 等. 不确定图数据库中高效查询处理[J]. 计算机学报, 2009, 32(10); 2066-2079

(上接第 205 页)

[13] Yuan Ye, Wang Guo-ren. Answering probabilistic reachability queries over uncertain graphs[J]. Chinese Journal of Computers, 2010, 33(8); 1378-1386

[14] Hu Yu-long, Wen Zhong-hua, Chang Qing, et al. Method of Get State Accessibility of Non-Cyclic in Non-Determinate Plan[J]. Computer Simulation, 2012, 29(5); 114-117 (in Chinese)