维普资讯 http://www.cqvip.com

使用遗传算法改进R_时刻表算法

钱付兰,程家兴
(安徽大学 计算机科学与技术学院,安徽 合肥 230039;
安徽大学 计算智能与信号处理教育部重点实验室,安徽 合肥 230039)

摘 要:给出了对于多成分R_时间表的解决办法。利用集合的性质把它看成是几个单成分时间关系约束的并集,采用算法I将多成分分解成单成分。使用遗传算法把问题的多成分的时间关系作为染色体的基因位,问题的所有的多成分的时间关系就构成了算法的染色体,以求解关系矩阵中求到的相容子集的个数作为染色体的适应度。算法的终止条件是根据关系矩阵求得的相容子集数等于事件的个数,利用算法II使用遗传算法求解多成分时间规划问题。对实际的问题进行了讨论,使用该算法在有穷次迭代后可得到可行解。

关键词:遗传算法;R_时刻表;多成分时间关系;时间规划
中图分类号:TP301.6 **文献标识码**:A **文章编号**:1673-629X(2008)03-0074-04

Improving Algorithms of R_Time Table by Using Genetic Algorithm

QIAN Fu-lan, CHENG Jia-xing
(School of Computer Science and Technology in Anhui University, Hefei 230039, China;
Ministry of Edu. Key Lab. of Intelligent Computing & Signal Processing, Anhui Univ., Hefei 230039, China)

Abstract: Gives the solution of R_time table under the multi-ingredient time relations. It can be looked on as the union of several simple-ingredient time relation constraints using the set property. Adopting the algorithm, can resolve the multi-ingredient as several simple-ingredients. Using genetic algorithm multi-ingredient time relations of the problems act as loci. The chromosomes of the algorithm are composed by all multi-ingredient time relations in the problem. The fitness in genetic algorithm is the amount of consistent subsets resulted from resolution of the relation matrix. And if the amount of events equals amount of consistent subsets, the algorithm is finished. The multi-ingredient temporal planning using the algorithm II can be resolved. It gives a discussion on the actual problem. The feasible solution is received after the finite iteration.

Key words: genetic algorithm; R_time table; multi-ingredient time relation; temporal planning

1 R_时刻表算法

对于时间世界模型,Allen提出使用区间以及区间之间的13种不同关系来描述[1]。对13种关系约束可以表示成使用$(X_1 \times X_2)$的积空间来表示(见图1),继而进一步可以表示成$(X_1 \times X_2) \times (Y_1 \times Y_2)$中子集形式(见图2)。

单成分定义:

[定义1]:$R(I_1, I_2)$称为是单成分(或单项)的$\Leftrightarrow$若存在$A_1 \subset X_1, A_2 \subset X_2; B_1 \subset Y_1, B_2 \subset Y_2$,使得$R(I_1, I_2) = [(A_1 \times A_2) \times (B_1 \times B_2)] \cap R$。

其中R是所有13种关系在$(X_1 \times X_2) \times (Y_1 \times Y_2)$上对应的点的全体。

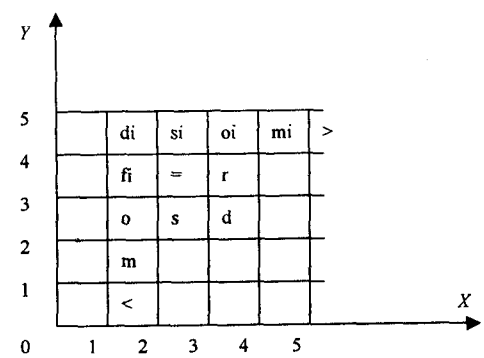

图1 时间关系平面表示图

从图2中可以看到凡是带有转折但是不连续有跳跃的点所构成的子集都不能表示成$X_1, X_2(Y_1, Y_2)$中的子集之积的形式,即都不是单成分的,或者说在图1中连线不属于图1的,它们也一定是多成分的。

收稿日期:2007-06-24
基金项目:教育部课题基金(200403057002)
作者简介:钱付兰(1978-),女,安徽蚌埠人,硕士,研究方向为人工智能、计算智能;程家兴,教授,博士生导师,研究方向为计算智能、最优化理论。

维普资讯 http://www.cqvip.com

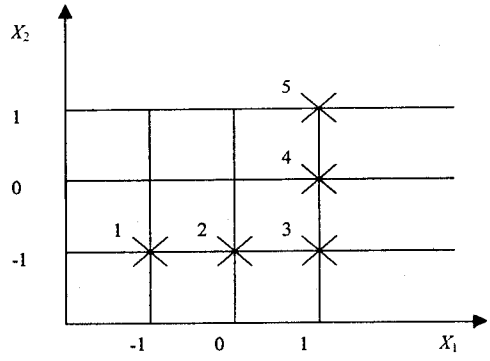

图2　3×3平面表示图

X_1对于单成分的时间规划问题,可以用关系矩阵表示[2]。

设$R(i,j)=(A(i,j)\times A(i,j))\times(B(i,j)\times B(i,j))\ \forall\ i,j=1,2,\cdots,n$,作2×2矩阵如下:

$$\boldsymbol{M}(i,j)=\begin{pmatrix}0 & -1\\ 1 & 0\end{pmatrix}$$

$$\boldsymbol{M}(i,j)=\begin{pmatrix}A_1(i,j) & A_2(i,j)\\ B_1(i,j) & B_2(i,j)\end{pmatrix},i\neq j$$

令

$$\boldsymbol{M}=\begin{pmatrix}M(1,1) & \cdots & M(1,n)\\ \vdots & & \vdots\\ M(n,1) & \cdots & M(n,n)\end{pmatrix}\qquad(1)$$

即$\boldsymbol{M}$是由n^2个2×2阶矩阵,按照公式(1)嵌套而得的$2n\times 2n$阶的矩阵。称$\boldsymbol{M}$为$\{R(i,j)\}$对应的关系矩阵,简称为关系矩阵。而对于多成分问题不能直接使用关系矩阵。

所谓R_时刻表就是一致满足时间关系约束的一个端点集的有序划分。

R_时刻表算法是基于单成分的使用关系矩阵的一种算法,具体算法和流程图见文献[3]。

2　多成分时间规划问题的化简

在对具体问题的时间关系约束进行分析时候发现,其中包含的时间关系既有单成分的又有多成分的,而对于多成分的时间关系约束[4~8],可以利用集合的性质把它看成是几个单成分时间关系约束的并集。

如$R(i,j)=\{<,>\}$可以看成是$R'(i,j)=\{<\}\cup R''(i,j)=\{>\}$,而$R'$和$R''$是单成分的时间关系约束,采用这种做法可以把任何一个多成分的时间关系约束划分为几个单成分时间关系约束的并的形式。

[例1]:$R(1,2)=\{<,>\}=\{(1\times 1),(5\times 5)\}$是多成分的,令$I=(1,2)$将$I$分离成$I'$和$I''$,其中$I'=(1',2')$,$I''=(1'',2'')$,定义$R(1',2')=(1\times 1)$,$R(1'',2'')=(5\times 5)$。

[定义2]:如果一个多成分时间关系的某个单成分分解在所有可能的单成分分解中其包含的单成分时间关系的个数为最少,则称该单成分分解是此多成分时间关系的最优解。

在求解多成分时间关系的单成分分解过程中,为了使得这种划分做到最优,即用最少的单成分集合来表示这个多成分时间关系约束,笔者分析发现对于时间关系的平面图1中任意两个时间关系,如果它们之间的对应连线不属于图1,则它们一定是多成分的,或者说在图2存在跳跃点的,也不是单成分的。

因此解决问题的步骤如下:

第一步:对于给出的多成分时间关系,在图1中找到对应的最少的连线方法,位于连线上的时间关系就可以看作是多成分时间关系的一个子集,为了使这种划分做到最优,做法如下:

说明:图中的m和<、mi和>在进行划分时具有相同的性质,为了方便进行算法描述,不妨把m和<记为r_1,mi和>记为r_2,记r_1和r_2统称为r类点。

下面给出求多成分时间关系A的最优单成分分解的算法I:

*A中无r类点的情况:

设给定的子集A为,将A分解成列的形式:$(1)\times b_1,(2)\times b_2,(3)\times b_3$。

1.若b_i只有一个不空,则A是单成分的;

2.若b_i恰有二个不空,若两者相等,则A是二成分;

3.若b_i都不空,则

3.1 三者相等,A是单成分的;

3.2 有两者相等,或其中一个b_i是另外两个b_i之并,则A是二成分的;

3.3 非上述情况,则A是三成分,即$A=\{(1)\times b_1,(2)\times b_2,(3)\times b_3\}$。

*A中有一个r类点的情况:

因为r_1和r_2具有相同的性质,所以不妨设此r类点即为r_1,取第一个成分为$(1)\times b_1$,然后对$A\cap[(2,3)\times(3,4,5)]$进行讨论,将它分解为$(2)\times b_2$和$(3)\times b_3$。若$b_2=b_3$,则$A$分解两个成分$(1)\times b_1$和$(2,3)\times b_2$;若$b_2\neq b_3$,则$A$分解为三成分,即$A=\{(1)\times b_1,(2)\times b_2,(3)\times b_3\}$。

*A中有二个r类点的情况:

将A分解成:$(1)\times b_1,a_5\times(5)$。然后对$A\cap[(2,3)\times(3,4)]$进行讨论,将它分解成$(2)\times b_2'$,$(3)\times b_3'$。若$b_2'=b_3'$,则分解成三个成分:$(1)\times b_1,a_5\times(5)$和$(2,3)\times b_2'$。不然$A$分解成四个成分,即$A=$

维普资讯 http://www.cqvip.com

$\{(1)\times b_1, a_5\times(5), (2)\times b_2', (3)\times b_3'\}$。

[命题 1]:根据算法 I 所求出的单成分分解一定是该多成分时间关系的最优单成分分解。

证明:

首先:对于第一类情况:

对于其中的单成分和二成分分解的情形证明显然。

对于三成分分解的情形,证明如下:

用反证法,若不然,则 A 可表示成两个成分:$a_1\times c_1, a_2\times c_2$。

令 $d_1 = a_1/a_2, d_2 = a_1\cap a_2, d_3 = a_2/a_1$。

若 d_i 中有一个空,不妨设 d_3 为空,则 d_1、d_2 中必有一个含两列,不妨设 d_1 含两列,则这两列对应的 b_i 相同,矛盾。

若 d_i 均不为空,于是可得到 d_i 各含一列,不妨设定 d_i 含 i 列,得第 2($d_2 = a_1\cap a_2$) 列等于第 1,3 列之并,矛盾。

故得 A 必定是三个成分。

其他情况的证明同理可得。

第二步:对于通过上述方法找出的每个划分,显然都可以把它表示成 $X\times Y$ 的形式,而根据单成分的定义知道 $X\times Y$ 还要能化为$(X_1\times X_2)\times(Y_1\times Y_2)$ 的形式,但对于 X 和 Y 由图 2 得到仍有以下 9 种子集不能表示成两个子集之积的形式:

{1,2,4},{1,2,5},{1,2,4,5},{1,4},{1,5},{1,4,5},{2,4},{2,5},{2,4,5}

但是对于这 9 种子集可以很容易地划分为两个单成分子集。

通过上面两个步骤就可以把任意一个多成分的时间关系划分为几个单成分时间关系的并的形式,而对于它的每个单成分划分,都可以和其他的单成分时间关系构成一个单成分的关系矩阵,从而可以利用求解 R_ 时刻表算法来进行求解,有 n 个划分对应的就有 n 个不同的关系矩阵,从中可以计算出满足条件的 R_ 时刻表。

3 利用遗传算法解决多成分问题的算法

在利用遗传算法[9,10]来解决这类问题时,把问题的多成分的时间关系作为染色体的基因位,基因位的值即为它的单成分时间关系子集,那么问题的所有的多成分的时间关系就构成了算法的染色体,染色体的空间即为单成分时间关系子集连接构成的空间,例如 $R(i,j) = \{<, >\}$ 的单成分划分为 $R'(i,j) = \{<\}\cup R''(i,j) = \{>\}$,则对应的基因位的取值即为{<}和{>},把染色体对应基因位的值加入关系矩阵中的对应位置就得到一个新的关系矩阵,在算法中,以求解关系矩阵中求到的相容子集的个数作为染色体的适应度,每个相容子集同时应包含求解简化矩阵时所简化的行列的信息,算法的终止条件是根据关系矩阵求得的相容子集数等于事件的个数。

利用遗传算法求解多成分时间关系下的 R_ 时刻表的完整算法 II 如下:

第一步:把多成分的时间关系化为多个单成分时间关系的并。

1. 根据算法 I 完成 A 的最优分解,把 A 分解为 $X\times Y$ 的形式。

2. 根据单成分时间关系的定义,把 1 中的 $X\times Y$ 化为$(X_1\times X_2)\times(Y_1\times Y_2)$ 的形式。

第二步:把每个多成分时间关系的多个单成分分解作为染色体对应基因位的取值范围,从所有多成分时间关系的单成分分解构成的问题空间中取出 N 个作为染色体的初始群体。

第三步:以每个染色体对应的关系矩阵计算出来的相容子集数(含简化过程中删除的行或列)作为算法的适应度函数,在给定的概率下对 N 进行遗传操作。

第四步:如果计算出来的染色体适应度满足问题的要求(等于事件的个数),则算法结束,否则重复第二、三步。

4 实例分析

对于简化的田径运动会比赛项目的实例,作

径赛[3]包括 100 米、200 米、400 米、800 米、110 米栏、400 米栏、4×100 米接力;田赛包括跳高、跳远、三级跳远、撑杆跳高、铅球。

并作如下规定:

1. 进行男子(女子)项目径赛的同时交叉进行女子(男子)项目的田赛(但不包括投掷项目)。

2. 投掷类比赛在所有径赛结束以后进行。

3. 同一项目总是男子先进行,女子后进行。

4. 赛事之间的休息时间最长不超过 15 分钟。

5. 假定有选手甲、乙、丙分别参加多项比赛,甲:男子 400 米和男子 800 米;乙:女子 100 米(初赛)和女子 100(决赛);丙:男子跳高和男子撑杆跳高。

为了简化问题,排除了在多个比赛场所同时进行比赛的情况(如在不同校区同时举行比赛的情况。)

将每个比赛项目看成不同的事件,共得 23 个事件,每个事件的描述如下:

1. 男子 100 米(初赛) 2. 男子 400 米 3. 男子 800 米 4. 男子 110 米栏 5. 男子 4×100 米接力

维普资讯 http://www.cqvip.com

6.男子跳高 7.男子跳远 8.男子撑杆跳高 9.男子铅球 10.男子标枪 11.男子铁饼 12.女子100米(初赛) 13.女子400米 14.女子800米 15.女子110米栏 16.女子1×100米接力 17.女子跳高 18.女子跳远 19.女子铅球 20.女子标枪 21.女子铁饼 22.男子100米(决赛) 23.女子100米(决赛)。

由问题描述和约束条件,首先分析出求其R_时刻表所要求的各个项目之间的时间关系。限于篇幅且不失去一般性给出项目1与其他项目之间的含多成分时间关系:

$R(1,2)=R(1,3)=R(1,4)=R(1,5)=R(1,7)=R(1,8)=R(1,10)=R(1,11)=R(1,12)=R(1,13)=R(1,14)=R(1,15)=R(1,16)=R(1,17)=R(1,18)=R(1,19)=R(1,20)=R(1,21)=R(1,22)=R(1,23)=\{<\}$

$R(1,17)=R(1,18)=\{>\}$

$R(1,6)=R(1,9)=\{s\}$

其结果如下:

交叉概率为0.8,变异概率为0.2,种群大小为23,迭代次数为1045,计算出来的染色体适应度满足问题的要求(等于事件的个数)。

参考文献:

[1] 张 铃,张 钹.问题求解理论及应用[M].北京:清华出版社,1990.

[2] 张 钹,张 铃.时间规划的关系矩阵法[J].计算机学报,1991,14(6):411-422.

[3] 钱付兰,程家兴,阚 涛,等.时间规划问题中R_时刻表及其在实际中的应用[J].微机发展,2004,14(12):139-141.

[4] 李敏强,寇纪淞,林 丹,等.遗传算法的基本理论与应用[M].北京:科学出版社,2002.

[5] 张 铃,张 钹.遗传算法机理的研究[J].软件学报,2000,11(7):945-952.

[6] 张 铃,张 钹.统计遗传算法[J].软件学报,1997(5):16-25.

[7] 张 铃,张 钹.佳点集遗传算法[J].计算机学报,2001,24(9):917-922.

[8] 赵春英,张 铃.佳点集遗传算法的应用[J].微机发展,2000,10(5):1-3.

[9] 陈国良,王煦法.遗传算法及其应用[M].北京:人民邮电出版社,1996.

[10] 张文修,梁 怡.遗传算法的数学基础[M].西安:西安交通大学出版社,2000.

(上接第73页)

URL:http://Address:8080/index.html。例如:键入本机链路本地地址URL:http://[fe80::2e0:50ff:fe00:3ce8%5]:8080,浏览器输出结果如图6所示。

6 小 结

通过以上的结果,利用基于双协议栈的代理服务器进行IPv6的访问,编写基于IPv6的客户端和服务器端程序,已实现了基于IPv6协议的数据包的发送与接收。

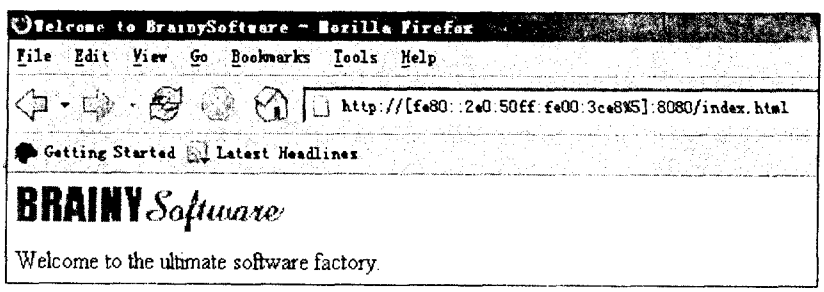

图6 浏览器输出结果

参考文献:

[1] Deering S, Hinder R. RFC2460. Internet protocol version 6 (IPv6) specification[S]. 1998.

[2] Stevens W, Alta M T. RFC 2292. Advanced Sockets API for IPv6[S]. 1998.

[3] Kitamuta H. RFC3089. A SOCKS-based IPv6/IPv4 gateway mechanism[S]. 2001.

[4] 谭汉松,刘安丰.从IPv4到IPv6网络程序的迁移[J].中南工业大学学报,2001,32(4):422-424.

[5] 赵文清,姜 波.基于Socket的Java语言网络通讯机制和程序设计[J].信息技术,2002(7):66-67.

[6] 金勇华,曲俊生.Java网络高级编程[M].北京:人民邮电出版社,2002.

[7] 李颖华,侯自强.IPv6高级套接口的研究与实现[J].计算机工程与应用,2003(21):19-22.

[8] 资武成,贺志苗.基于SOCKET的JAVA网络编程[J].娄底师专学报,2003(2):36-38.