基于带标记的并发可达标识图的关键路径的求解方法

韩耀军

(上海外国语大学国际工商管理学院信息管理系 上海 200083)

摘 要 将AOE网转换成有色时延Petri网模型,在模型转换过程中同时计算出各位置所对应的事件的最早开始时间,给出了模拟AOE网的有色时延Petri网模型的带标记的并发可达标识图的构建算法;利用并发可达标识图中的标记序列直接得到关键路径并计算出完成所有活动所需的最短时间。实例与仿真实验结果表明,当AOE网中平均存在3个以上的并发活动时,所提方法执行效率优于传统的求解关键路径的算法,并发活动越多,所提算法效率越高。

关键词 有色时延Petri网,并发可达标识图,AOE网,关键路径

中图法分类号 TP301 **文献标识码** A **DOI** 10.11896/j.issn.1002-137X.2016.11.023

Method for Finding Critical Paths Based on Concurrent Reachable Marking Graph with Tags

HAN Yao-jun

(Department of Information Management, School of Business and Management, Shanghai International Studies University, Shanghai 200083, China)

Abstract The color timed Petri net model was gotten by transforming AOE network in this paper. The earliest event start time was calculated while constructing Petri net model. The algorithm of constructing concurrent reachable marking graph with tags for color timed Petri net modeling AOE network was given. The critical paths were gotten and the shortest time of completing all activities was calculated from tags of concurrent reachable marking graph. The example and simulation show that the execution efficiency of the algorithm is better than traditional algorithm for finding critical paths when there are more than three concurrent activities in AOE network. The more the concurrent activities are, the higher the efficiency is.

Keywords Color timed Petri net, Concurrent reachable marking graph, AOE network, Critical paths

1 引言

AOE网(Activity On Edge,边表示活动的网)是一个带权的有向无环图,其中顶点表示事件(Event),弧表示活动,权表示活动持续的时间。由于AOE网常用来估算工程的完成时间,而完成工程的最短时间是指从工程开始点到完工点的最长路径的长度(即路径上各活动持续时间之和),因此路径长度最长的路径称为关键路径[1]。所以,要想缩短工程的工期,就必须提高关键活动(即关键路径上的活动)的工效。因此,求解关键路径是AOE网中的重要任务之一(本文研究的AOE网假设只有一个顶点的入度为0,只有一个顶点的出度为0)。传统的关键路径的求解算法通过分别求出所有事件的最早和最迟开始时间以及每项活动的最早和最迟结束时间来求得关键活动。为改善求解效率,国内外学者相继提出了一些新的求解AOE网关键路径的方法[2-6]。

Petri网由于是模拟与分析具有并发、异步、动态等特点事件的理想的图形与数学工具,因此得到越来越多的应用。先后有学者利用Petri网(主要是含时间因素的Petri网)求解关键路径。文献[7]通过对肯定型工程问题构建库所(表示工序,即AOE网中的活动)含时间的时延Petri网模型,利用时延Petri网计算出各库所对应工序的最早和最晚开工时间,进而求得关键工序,并利用时延Petri网的可达标识图求解出合理的工序。由于这种方法需要计算各工序的最早和最晚开工时间,因此在执行效率方面与传统的关键路径的求解算法相比并没有显著提高。由于大部分表示工程的AOE网的活动要多于事件(除非没有并行的活动),因此用库所表示活动的Petri网模型,其库所要多于变迁。相应地计算表示各活动的库所的最早和最晚开工时间的效率要低于计算变迁的最早和最晚开工时间。文献[8]把工程规划中的有向网络转换成与文献[7]一样的时延Petri网,然后利用Petri网的可达性分析方法,通过对Petri网模型进行剪枝优化,自动获取关键路径。但该文对剪枝及关键路径的自动获取过程并未详细介绍。文献[9]首先基于一定的转换规则把AOE网转换成CPN模型,然后利用CPN Tool获得CPN的状态可达图,最后通过对获得的状态可达图进行遍历得到所有的路径,并从这些路径中找出关键路径。但该方法首先要依赖于状态分析工具,其次遍历状态可达图需要花费大量时间。因为可达标识图的主要弱点是所谓的状态空间的爆炸问题,即模型的状态空间随着实际系统的规模增大而呈指数增长。本文首先将AOE网转换成变迁含时间的有色时延Petri网,并且在转换过程中计算

到稿日期:2015-07-26 返修日期:2016-01-18 本文受上海市哲学社会科学规划一般课题(2010BTQ001),上海外国语大学校级重大科研项目(2013114ZD004),上海外国语大学国际工商管理学院高层次培育项目资助。

韩耀军(1957—),男,博士,教授,CCF会员,主要研究方向为Petri网理论及应用、网格计算、云计算、信息管理,E-mail:yjhan@shisu.edu.cn。

出各事件的最早发生时间(不必计算各事件的最晚发生时间以及活动的最早、最晚开始时间),然后构建有色时延 Petri 网的带标记的并发可达标识图,利用并发可达标识图中的标记序列,即可直接获得关键路径。由于该方法只需计算各事件的最早发生时间,因此可节省执行的时间,提高执行的效率。另外,由于并发可达标识图可以在一定种程度上减小可达标识图的规模(并发变迁越多,可达标识图的规模越小),因此与传统的用可达标识图求解关键路径的方法相比,该方法可以有效地提高算法的执行效率。

2 基本概念

定义 1[10] 三元组 $N=(P,T,F)$称为网的充分必要条件是:

(1)$P\cap T=\emptyset$;

(2)$P\cup T\neq\emptyset$;

(3)$F\subseteq(P\times T)\cup(T\times P)$("×"为笛卡尔积);

(4)$\mathrm{dom}(F)\cup\mathrm{cod}(F)=P\cup T$。

其中,$\mathrm{dom}(F)=\{x\mid\exists y:(x,y)\in F\}$为 F 的定义域,$\mathrm{cod}(F)=\{y\mid\exists x:(x,y)\in F\}$为 F 的值域。P 和 T 分别称为 N 的位置(place)集和变迁(transition)集,F 为流关系(flow relation)。

定义 2[10] 四元组 $PN=(P,T;F,M_0)$称为 Petri 网,当且仅当

(1)$N=(P,T;F)$是一个网;

(2) $M:P\to Z$ 为标识函数,其中 M_0 是初始标识,$Z=\{0,1,2,\cdots\}$为非负整数集。

定义 3[10] 设 $PN=(P,T;F,M_0)$是一个 Petri 网,对于 $x\in P\cup T$:

$\cdot x=\{y\mid(y,x)\in F\}$称为 x 的输入集或前集;

$x\cdot=\{y\mid(x,y)\in F\}$称为 x 的输出集或后集。

定义 4[10] 设 $PN=(P,T;F,M_0)$是一个 Petri 网。

(1)变迁 $t\in T$在标识 M 下称为使能的,当且仅当 $\forall p\in\cdot t:M(p)\geqslant 1$,记作 $M[t>$;

(2)若 t 在 M 下使能,则 t 可以引发,且引发后产生一个后继标识 M',记作 $M[t>M'$,其中:

$$M'(p)=\begin{cases}M(p)+1, & \text{当 } p\in t\cdot-\cdot t\\ M(p)-1, & \text{当 } p\in\cdot t-t\cdot\\ M(p), & \text{其它}\end{cases}$$

定义 5[11] 九元组 $CPN=(\Sigma,P,T,A,N,C,G,E,I)$ 称为一个有色 Petri 网(Coloured Petri Nets),当且仅当

(1)Σ 是非空标记颜色的集合;

(2)P 是有限位置的集合;

(3)T 是有限变迁的集合;

(4)A 是有限弧集,且 $P\cap T=P\cap A=T\cap A=\emptyset$;

(5)$N:A\to(P\times T)\cup(T\times P)$是一个节点函数;

(6)$C:P\to\Sigma$ 是颜色函数;

(7)G 是一个保护(guard)函数,映射 T 到布尔型表达式,使得 $\forall t\in T:Type(G(t))=B\wedge Type(Var(G(t)))\subseteq\Sigma$;

(8)E 是一个弧表达式函数,映射 A 到表达式,使得 $\forall a\in A:Type(E(a))=C(p)_{MS}\wedge Type(Var(E(a)))\subseteq\Sigma$,其中 p 指在 $N(a)$中的位置;

(9)I 是一个初始化函数,映射 P 到闭表达式,使得 $\forall p\in P:Type(I(p))=C(p)_{MS}$。

定义 6[12] 五元组 $TDPN=(P,T;F,M_0,D)$是一个时延 Petri 网(Timed Petri Nets),当且仅当

(1)$(P,T;F,M_0)$是一个 Petri 网;

(2)D 是一个由 T 到有理数的映射函数,$\forall t\in T,D(t)$是 t 引发的持续时间。

3 AOE 网到有色时延 Petri 网模型的转换

定义 7 假设一个 AOE 网含有 m 个顶点$(v_1,v_2,\cdots,v_m)$和 n 条边(即 n 个活动$a_1,a_2,\cdots,a_n$),则该 AOE 网所对应的有色时延 Petri 网模型为七元组 $CTdPN=(P,T;F,M_0,C,E,D)$。转换规则如下:

(1)位置 P 与 AOE 网中的顶点相对应,即 AOE 网中的一个顶点转换为 $CTdPN$ 的一个位置;

(2)变迁 T 与 AOE 网中边上的活动相对应,即 AOE 网中边上的一个活动转换为 $CTdPN$ 的一个变迁;

(3)增加一个变迁 t_e 作为汇点 v_m 所对应位置的输出变迁及一个位置 p_e 为变迁 t_e 的输出位置;

(4)F 为连接位置与变迁而产生的弧的集合;

(5)C 是颜色的集合,由各活动的名称组成,即 $C=\{a_1,a_2,\cdots,a_n\}$;

E 是弧的函数:

$$E(f)=\begin{cases}r\cdot a_i, & \text{当 } f\in(p_j,t_i),\text{其中 } r \text{ 为 } p_j \text{ 的输入变迁个数}\\ \sum_{k=1}^{s}a_{jk}, & \text{当 } f\in(t_i,p_j),\text{其中 } s \text{ 为 } p_j \text{ 的输出变迁个数}\\ 1, & \text{当 } f\in\{(t_i,p_j)\mid p_j \text{ 为汇点所对应位置}\}\vee\{(t_e,p_e)\}\\ n_i, & \text{当 } f\in(p_i,t_e),n_i \text{ 为 } p_i \text{ 的输入变迁个数}\end{cases}$$

(6)D 是一个由 T 到有理数的映射函数,$\forall t\in T,D(t)$是 t 引发的持续时间,除 $D(t_e)$为 0 外,其余变迁引发的持续时间为对应的 AOE 网中活动的持续时间;

(7)M_0 是初始标识,除 $M_0(p_1)$(其中 p_1 为对应 AOE 网中源点的位置)为 p_1 的所有输出变迁所对应的活动名称外,其余位置的初始标识均为 0。

对于 $CTdPN$,变迁 $t\in T$ 在标识 M 下称为使能的,当且仅当 $\forall p\in\cdot t:M(p)\geqslant E(p,t)$,记作 $M[t>$。若 t 在 M 下使能,则 t 可以引发,且引发后产生一个后继标识 M',记作 $M[t>M'$,其中:

$$M'(p)=\begin{cases}M(p)-E(p,t), & \text{若 } p\in\cdot t-t\cdot\\ M(p)+E(p,t), & \text{若 } p\in t\cdot-\cdot t\\ M(p), & \text{其他}\end{cases}\tag{1}$$

在上述 $CTdPN$ 的构造过程中,同时计算出各位置所对应事件的最早发生时间。假设用 $VE(p_i)$表示各事件的最早发生时间,则有:

$$VE(p_i)=\begin{cases}0, & \text{若}\cdot p_i=\emptyset\\ \max\{VE(p_j)+D(t_k)\mid p_j\in\cdot(\cdot p_i),t_k\in\cdot p_i\}, & \text{其他}\end{cases}\tag{2}$$

从上述 $CTdPN$ 的构造过程易知,$CTdPN$ 的基网 $N=(P,T;F)$是一个 S-图,即 $\forall t\in T:|\cdot t|=|t\cdot|=1$。由于 AOE

网为一个有向无环图，而一个 S-图与一个有向图相对应[11]，*CTdPN* 的构造又严格按照与 AOE 网相对应的方式进行(增加的变迁 t_e 只有一个输入位置和一个输出位置)，因此，可以得出 *CTdPN* 的基网是一个 S-图。另外，由于 *CTdPN* 的构造严格按照与 AOE 网相对应的方式进行，因此对比文献[1]所提供的 AOE 网中各事件的最早发生时间的计算公式，本文给出的计算各事件的最早发生时间的式(2)的正确性是显然的。

下面通过一个例子来说明 *CTdPN* 的构造过程以及各事件的最早发生时间的计算。

例如：图 1 是一个有 11 项活动、9 个事件的 AOE 网[1]。

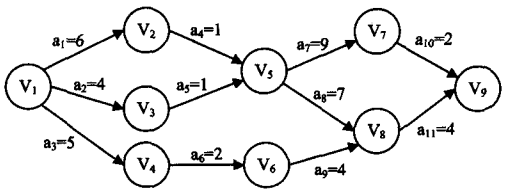

图 1　一个 AOE 网的例子

根据定义 7，图 1 所示的 AOE 网所对应的 $CTdPN=(P,T;F,M_0,C,E,D)$ 如下。

$P=\{p_1,p_2,p_3,p_4,p_5,p_6,p_7,p_8,p_9,p_e\}$；

$T=\{t_1,t_2,t_3,t_4,t_5,t_6,t_7,t_8,t_9,t_{10},t_{11},t_e\}$；

F 如图 2 所示。

$C=\{a_1,a_2,a_3,a_4,a_5,a_6,a_7,a_8,a_9,a_{10},a_{11}\}$；

$E(p_1,t_i)=a_i$，$E(t_i,p_{i+1})=a_{i+3}$，其中 $i=1,2,3$；

$E(p_i,t_{i+2})=a_{i+2}$，其中 $i=2,3,4$；

$E(t_i,p_5)=a_7+a_8$，其中 $i=4,5$；$E(t_6,p_6)=a_9$；

$E(p_5,t_{i+2})=2a_{i+2}$，其中 $i=5,6$；$E(p_6,t_9)=a_8$；

$E(t_7,p_7)=a_{10}$，$E(t_8,p_8)=E(t_9,p_8)=a_{11}$；

$E(p_7,t_{10})=a_{10}$，$E(p_8,t_{11})=2a_{11}$；

$E(t_{10},p_9)=E(t_{11},p_9)=1$；

$E(p_9,t_e)=2$，$E(t_e,p_e)=1$。

$D(t_i)$：活动 a_i 的持续时间(见图 2)($i=1,\cdots,11$)，$D(t_e)=0$；

$M(p_1)=\{a_1,a_2,a_3\}$，$M(p_j)=0$，其中 $j=1,\cdots,9$。

模型的图形表示如图 2 所示。

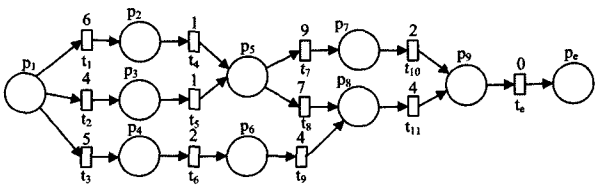

图 2　图 1 所对应的 *CTdPN*

根据式(2)计算出各位置所对应的事件的最早发生时间如表 1 所列。

表 1　图 2 各位置所对应的事件的最早发生时间

p_i	p_1	p_2	p_3	p_4	p_5	p_6	p_7	p_8	p_9
$VE(p_i)$	0	6	4	5	7	7	16	14	18

4　基于带标记的并发可达标识图的关键路径求解

定义 8　设 $CTdPN=(P,T;F,M_0,C,E,D)$ 是一个有色时延 Petri 网，$R(M_0)$ 是可达标识集。有色时延 Petri 网的并发可达标识图(Concurrent Reachable Marking Graph，CRMG)定义为一个顶点带有标记的有向图：$CRMG(CTdPN)=(V,E,F,TAG)$，其中

(1)$V=\{R(M_0)\}$。

(2)$E=\{(M_i,M_j)\mid M_i,M_j\in R(M_0),\exists t\in T:M_i[t>M_j$ 或 $\exists$ 在 M_i 可以引发的并发变迁序列 $t_{i1},t_{i2},\cdots,t_{ik}\in T:M_i[t_{i1}\|t_{i2}\|\cdots\|t_{ik}>M_j\}$，$t_1\|t_2\|\cdots\|t_k$ 表示 $t_{i1},t_{i2},\cdots,t_{ik}$ 并发引发。

(3)F：边 E 的标记。如果 $M_i[t>M_j$，则边 (M_i,M_j) 标记为“t”；如果 $M_i[t_{i1}\|t_{i2}\|\cdots\|t_{ik}>M_j\}$，则边 (M_i,M_j) 标记为“$t_1\|t_2\|\cdots\|t_k$”。

(4)TAG：标识 M 的标记，其值为有 $n+2$ 个分量的变迁序列，n 为 *CTdPN* 中变迁个数，具体取值见算法 1，其中 $M\in R(M_0)$。

4.1　带标记的并发可达标识图的构造算法

算法 1　构造有色时延 Petri 网的带标记的并发可达标识图的算法

输入：有色时延 Petri 网 CTdPN=(P,T;F,M_0,C,E,D)

输出：带标记的并发可达标识图 CRMG(CTdPN)=(V,E,TAG)

1. 设 V={M_0}，E={φ}，TAG(M_0)[i]={(φ)}，i=0,1,…,n+1
2. M_0 标记为“新”
3. 如果 V 中无“新”结点，则算法终止；否则转步骤 4
4. 从 V 中选择一个“新”标识 M，并且将 M 标记为“旧”
5. 如果在 M 中不存在使能的变迁，则将 M 标记为“终端”结点并转步骤 3
6. 对每一个在 M 使能的变迁 t∈T，构建并发变迁集 CT={(t_{j1},t_{j2},…,t_{jk})}和非并发变迁集 NT={t_{j1},t_{j2},…,t_{jk2}}
7. 对每一个在 M 下使能的非并发变迁 t_i∈NT
   7.1 t_i 引发后产生一个 M 的后继标识 M′，M′的计算见式(1)
   7.2 如果 M′∉V，则 V=V+{M′}并标记 M′为“新”
   7.3 E=E+{(M,M′)}并标记(M,M′)为“t_i”
   7.4 当 $t_i\neq te$
      7.4.1 若$|\cdot(\cdot t_i)|\leqslant 1$，则 TAG(M′)[i]= TAG(M[j])+t_i，其中 $t_j=|\cdot(\cdot t_i)$；若$|\cdot(\cdot t_i)|=0$，则 j=0
      7.4.2 若$|\cdot(\cdot t_i)|=k>1$
         (a)若 $VE(p_{sl})+D(t_{jl})$(l=1,2,…,k)全部相等，则：$TAG(M')[i]=\bigcup_{l=1}^{k}(TAG(M)[j_1]+t_i)$，其中 t_{jl}为$\cdot(\cdot t_i)$，p_{sl}为$\cdot t_{jl}$
         (b)若 $VE(p_{sl})+D(t_{jl})$(l=1,2,…,k，其中 t_{jl}为$\cdot(\cdot t_i)$，p_{sl}为$\cdot t_{jl}$)不全相等，则：TAG(M′)[i]= TAG(M)[r]+t_i，其中 $VE(p_{sr})+D(t_{jr})$为最大值
   7.5 当 $t_i=t_e$
      7.5.1 若$|\cdot(\cdot t_i)|\leqslant 1$，则 TAG(M′)[i]= TAG(M)[j]，其中 $t_j=|\cdot(\cdot t_i)$
      7.5.2 若$|\cdot(\cdot t_i)|=k>1$
         (a)若 $VE(p_{sl})+D(t_{jl})$(l=1,2,…,k，其中 t_{jl}为$\cdot(\cdot t_i)$，p_{sl}为$\cdot t_{jl}$)全部相等，则：$TAG(M')[i]=\bigcup_{l=1}^{k}(TAG(M)[j_1])$，其中 $t_j=|\cdot(\cdot t_i)$，$p_{jl}=\cdot t_j$
         (b)若 $VE(p_{sl})+D(t_{jl})$(l=1,2,…,k，其中 t_{jl}为$\cdot(\cdot t_i)$，p_{sl}为$\cdot t_{jl}$)不全相等，则：TAG(M′)[i]=TAG(M)[r]，其中 $VE(p_{sr})+D(t_{jr})$为最大值
8. 对在 M 下使能的并发变迁 ct={t_{i1},t_{i2}…,t_{ik}}∈CT
   8.1 并发变迁 ct={t_{i1},t_{i2}…,t_{ik}}引发后产生一个 M 的后继标识 M′，对 M′中所有的变迁，均按式(1)并行计算 M′
   8.2 如果 M′∉V，则 V=V+{M′}并标记 M′为“新”

8.3 E=E+{(M,M′)}并标记(M,M′)为“$t_{i1}\|t_{i2}\|\cdots\|t_{ik}$”

$M'=M-(M_{in}(t_{i1})+M_{in}(t_{i2})+\cdots+M_{in}(t_{ik}))+(M_{out}(t_{i1})+M_{out}(t_{i2})+\cdots+M_{out}(t_{ik}))$

8.4 对任一 $t_i\in ct$，并行执行步骤 8.4.1 和 8.4.2 中的任务

8.4.1 若$|\cdot(\cdot t_i)|\leqslant 1$，则 $TAG(M')[i]=TAG(M)[j]+t_i$，其中 $t_j=\cdot(\cdot t_i)$；若$|\cdot(\cdot t_i)|=0$，则 j=0

8.4.2 若$|\cdot(\cdot t_i)|=k>1$

(a)若 $VE(p_{sl})+D(t_{jl})(l=1,2,\cdots,k$，其中 t_{jl}为$\cdot(\cdot t_i)$，p_{sl}为$\cdot t_{jl})$全部相等，则：$TAG(M')[i]=\bigcup_{l=1}^{k}(TAG(M)[j_l]+t_i)$，其中 $t_j=|\cdot(\cdot t_i)$，$p_{sl}=\cdot t_j$

(b)若 $VE(p_{jl})+D(t_{jl})(l=1,2,\cdots,k)$，其中 t_{jl}为$\cdot(\cdot t_i)$，p_{sl}为$\cdot t_{jl})$不全部相等，则：$TAG(M')[i]=TAG(M)[r]+t_i$，其中 $VE(p_{sr})+D(t_{jr})$为最大值

9. 去掉 M 的“新”标记，然后转步骤 3

命题 1 设 $CRMG=(V,E,TAG)$ 是有色时延 Petri 网 *CTdPN* 的并发可达标识图，则 *CRMG* 中 $TAG(M)[n+1]$中的变迁序列所对应的活动序列即为 *CTdPN* 所描述的 AOE 网的关键路径，序列的个数即为关键路径的条数，完成所有活动所需的最短时间为：$VE(p_n)$(即 *CTdPN* 中最后一个位置所对应事件的最早发生时间)或 $\Sigma D(t_i)$，其中 t_i 为 $TAG(M)[n+1]$中的任一变迁序列中的变迁。其中 $M\in R(M_0)$为终端。

根据定义 8、AOE 网到 *CTdPN* 的转换规则以及算法 1 中标记 *TAG* 的计算方法，命题 1 容易得证。

4.2 算法的时间复杂度分析

假设 AOE 网含有 m 个顶点和 n 条边，则根据定义 7 知，Petri 网 *CTdPN* 有 $m+1$ 个位置和 $n+1$ 个变迁。算法 1 中的步骤 1 和 2 的时间复杂度为 $O(n)$；对于步骤 3－8，由于 AOE 网为无环有向图，因此构造标识图的过程最多在 $O(n+m)$时间内完成(即没有并发变迁的情况)。当有并发变迁时，其时间要小于 $O(n+m)$。再根据命题 1，一旦并发可达标识图构造完成，则关键路径可以从标记 *TAG* 中直接得到。因此，该算法的总的时间复杂度不大于 $O(n+m)$。

根据定义 7 知，AOE 网到 *CTdPN* 模型转换的时间复杂度为 $O(n+m)$(因 AOE 网的 m 个顶点对应 *CTdPN* 的 m 个位置，n 条边对应 *CTdPN* 的 n 个变迁)，而各位置所对应事件的最早发生时间的计算是在模型转换的同时计算出来的。因此，本文求解关键路径所用的时间复杂度为 $O(n+m)$。

与已有算法相比，当 AOE 网中无并发活动时，本算法的时间复杂度与已有算法相当。但当 AOE 网中存在并发活动时，该算法要优于传统的求解关键路径的算法，更优于用可达标识图计算关键路径的算法(因为可达标识图未考虑并发变迁)。

4.3 算法性能的实例验证

仍然以图 1 所给的 AOE 网为例继续讨论。根据算法 1，图 2 所示的 Petri 网的带标记的并发可达标识图的构造过程如下。

(1)$V=\{M_0(a_1+a_2+a_3,0,0,0,0,0,0,0,0,0)\}$，$E=\{\varphi\}$，$TAG(M_0)=\{(\varphi),(\varphi),(\varphi),(\varphi),(\varphi),(\varphi),(\varphi),(\varphi),(\varphi),(\varphi),(\varphi),(\varphi)\}$。

(2)标记 M_0 为“新”。

1)从 V 中选择一个“新”标识 M_0，并且将 M_0 标记为“旧”。在 M_0 下，变迁 t_1,t_2,t_3 都是使能的，t_1,t_2,t_3 为并发变迁，故并发变迁集 $CT=\{(t_1,t_2,t_3)\}$，非并发变迁集 $NT=\{\varphi\}$。

并发变迁 $ct=(t_1,t_2,t_3)\in CT$ 引发后产生 M_0 的后继标识 M_1，$M_1=(0,a_4,a_5,a_6,0,0,0,0,0,0)$，标记 M_1 为“新”。

$V=\{M_0,M_1\}$，$E=\{(M_0,M_1)\}$，(M_0,M_1)的标记为 $t_1\|t_2\|t_3$。

对于 $t_1\in ct$，由于$|\cdot(\cdot t_1)|=0<1$，$TAG(M_1)[1]=TAG(M_0)[0]+t_1=(t_1)$；同理，对于 $t_2,t_3\in ct$，有 $TAG(M_1)[2]=TAG(M_0)[0]+t_2=(t_2)$；$TAG(M_1)[3]=TAG(M_0)[0]+t_3=(t_3)$。

2)选择“新”标识 M_1，并且将 M_1 标记为“旧”。在 M_1 下，变迁 t_4,t_5,t_6 都是使能的，t_4,t_5,t_6 为并发变迁，故并发变迁集 $CT=\{(t_4,t_5,t_6)\}$，非并发变迁集 $NT=\{\varphi\}$。

并发变迁 $ct=(t_4,t_5,t_6)\in CT$ 引发后产生 M_1 的后继标识 M_2，$M_2=(0,0,0,0,2a_7+2a_8,a_9,0,0,0,0)$，标记 M_2 为“新”。

$V=\{M_0,M_1,M_2\}$，$E=\{(M_0,M_1),(M_1,M_2)\}$，$(M_1,M_2)$的标记为 $t_4\|t_5\|t_6$。

对于 $t_4\in ct$，由于$|\cdot(\cdot t_4)|=1$，$TAG(M_2)[4]=TAG(M_1)[1]+t_4=(t_1\ t_4)$；同理，对于 $t_5,t_6\in ct$，有 $TAG(M_2)[5]=TAG(M_1)[2]+t_5=(t_2\ t_5)$；$TAG(M_2)[6]=TAG(M_0)[3]+t_6=(t_3\ t_6)$。

3)选择“新”标识 M_2，并且将 M_2 标记为“旧”。在 M_2 下，变迁 t_7,t_7,t_9 都是使能的，t_7,t_7,t_9 为并发变迁。故并发变迁集 $CT=\{(t_7,t_7,t_9)\}$，非并发变迁集 $NT=\{\varphi\}$。

并发变迁 $ct=(t_7,t_7,t_9)\in CT$ 引发后产生 M_2 的后继标识 M_3，$M_3=(0,0,0,0,0,0,a_{10},2a_{11},0,0)$，标记 M_3 为“新”。

$V=\{M_0,M_1,M_2,M_3\}$，$E=\{(M_0,M_1),(M_1,M_2),(M_2,M_3)\}$，$(M_2,M_3)$的标记为 $t_7\|t_8\|t_9$。

对于 $t_7\in ct$，由于$|\cdot(\cdot t_7)|=2>1$，又由于 $VE(p_2)+D(t_4)=7>VE(p_3)+D(t_5)=5$，因此 $TAG(M_3)[7]=TAG(M_2)[4]+t_7=(t_1\ t_4\ t_7)$；同理，对于 $t_8\in ct$，由于$|\cdot(\cdot t_8)|=2>1$，又由于 $VE(p_2)+D(t_4)=7>VE(p_3)+D(t_5)=5$，因此有 $TAG(M_3)[8]=TAG(M_2)[4]+t_8=(t_1\ t_4\ t_8)$；对于 $t_9\in ct$，由于$|\cdot(\cdot t_9)|=1$，有 $TAG(M_3)[9]=TAG(M_2)[6]+t_9=(t_3\ t_6\ t_9)$。

4)选择“新”标识 M_3，并且将 M_3 标记为“旧”。在 M_3 下，变迁 t_{10},t_{11} 都是使能的，t_{10},t_{11} 为并发变迁。故并发变迁集 $CT=\{(t_{10},t_{11})\}$，非并发变迁集 $NT=\{\varphi\}$。

并发变迁 $ct=(t_{10},t_{11})\in CT$ 引发后产生 M_3 的后继标识 M_4，$M_4=(0,0,0,0,0,0,0,0,2,0)$，标记 M_4 为“新”。

$V=\{M_0,M_1,M_2,M_3,M_4\}$，$E=\{(M_0,M_1),(M_1,M_2),(M_2,M_3),(M_3,M_4)\}$，$(M_3,M_4)$的标记为 $t_{10}\|t_{11}$。

对于 $t_{10}\in ct$，由于$|\cdot(\cdot t_{10})|=1$，有 $TAG(M_4)[10]=TAG(M_3)[7]+t_{10}=(t_1,t_4,t_7,t_{10})$；对于 $t_{11}\in ct$，由于$|\cdot(\cdot t_{11})|=2>1$，又由于 $VE(p_5)+D(t_8)=14>VE(p_6)+D(t_9)=11$，因此 $TAG(M_4)[11]=TAG(M_3)[8]+t_{11}=(t_1\ t_4\ t_8\ t_{11})$。

5)选择“新”标识 M_4，并且将 M_4 标记为“旧”。在 M_4 下，只有变迁 t_e 是使能的，所以并发变迁集 $CT=\{\varphi\}$，非并发变迁集 $NT=\{t_e\}$。

变迁 t_e 引发后产生 M_4 的后继标识 M_5，标记 M_5 为“新”。

$M_5=(0,0,0,0,0,0,0,0,0,1)$，$V=\{M_0,M_1,M_2,M_3,M_4,M_5\}$，$E=\{(M_0,M_1),(M_1,M_2),(M_2,M_3),(M_3,M_4),(M_4,M_5)\}$，$(M_4,M_5)$的标记为 t_e。

由于$|\cdot(\cdot t_e)|=2>1$，又由于 $VE(p_7)+D(t_{10})=VE(p_8)+D(t_{11})=18$，因此 $TAG(M_5)[12]=(TAG(M_4)[10],TAG(M_4)[11])=((t_1\ t_4\ t_7\ t_{10}),(t_1\ t_4\ t_8\ t_{11}))$。

6)选择“新”标识 M_5，并且将 M_5 标记为“旧”。由于在 M_5 下没有使能的变迁，因此标记 M_5 为“端点”。

(3)由于 V 中无“新”结点，因此 *CRMG* 的构造过程结束。

从上述 *CRMG* 的构造结果可知，附加在终端 M_5 的标记 $TAG(M_5)[12]$有两个变迁序列$(t_1\ t_4\ t_7\ t_{10})$ 和$(t_1\ t_4\ t_8\ t_{11})$。根据命题 1，图 1 所示的 AOE 网的关键路径有两条，分别为：$a_1\rightarrow a_4\rightarrow a_7\rightarrow a_{10}$ 和 $a_1\rightarrow a_4\rightarrow a_8\rightarrow a_{11}$。

完成所有活动所需最短时间为 $VE(p_9)=18$ 或$\Sigma(D(t_1)+D(t_4)+D(t_7)+D(t_{10}))=(6+1+9+2)=18$ 或$\Sigma(D(t_1)+D(t_4)+D(t_8)+D(t_{11}))=(6+1+7+4)=18$。

图 2 所示的 Petri 网的 *CRMG* 如图 3 所示。

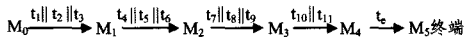

图 3　图 2 所对应的 *CRMG*

如果构造常规的可达标识图，则图 2 所示的 Petri 网所对应的可达标识图 *RMG* 的顶点多达 80 多个(因图形过于复杂，故略)。从图形规模看，其要比本文所给出的并发可达标识图大得多。其次，对于一个工程来说，可达标识图中从初始结点 M_0 到终端结点 M_e 的很多有向路径都是不可能出现的或不合理的状态。而处理这些状态需要花费大量时间。因此，处理可达标识图花费的时间要远大于处理本文所给出的并发可达标识图所花费的时间。由此可以看出本文所提方法的有效性。

5　仿真实验

本节通过仿真实验来验证本文所提方法的有效性。仿真实验软件环境为操作系统 Windows 7、编程语言 C++，硬件环境为处理器 Intel Core P8600、CPU 2.4GHz、内存 3GB。

5.1　仿真数据的产生

为了验证所提方法的有效性，首先通过编程随机产生顶点数分别为 100，150，200，250，300，350，400，450 的 AOE 网。为了防止模拟的工程出现不合理的情况，对边的产生规则有以下几点要求：

(1)第一个顶点(即源点)入度为 0；最后一个顶点(即汇点)的出度为 0。

(2)除第一个顶点外，其它所有顶点入度均大于 0；除最后一个顶点外，其它所有顶点出度均大于 0。

(3)为了防止产生的 AOE 网出现环，规定任何一个活动(即有向边)的前驱顶点编号必须小于其后继顶点的编号。

(4)为了防止产生过多的边(即活动)以及出现一些不太合常理的情况(如第一个顶点直接与最后一个顶点相连)，规定直接相连的两个顶点序号之差不超过 20。

(5)各边的权数(即完成各活动所需时间)在 1－30 范围内随机产生。

5.2　仿真结果

本仿真实验选取文献[1]中算法(称为传统算法)和文献[7]中算法(称为常规算法)与本文算法进行比较(因文献[8]的算法未给出关键路径的自动获取过程，而文献[9]利用了 CPN Tool 工具软件，不具可比性，因此未直接与这两种方法比较，但根据它们的思路，均可归为常规算法)。由于本文算法的最大特点是对可同时发生的变迁进行并行处理，因此从并发变迁的不同数量加以比较。表 2－表 4 和图 4－图 6 是平均并发变迁个数分别为 2，3.5 和 17，最大并发变迁的个数分别 3，7，42 时 3 种算法性能的比较结果(单位：μs)。

表 2　平均并发变迁个数为 2，最大并发变迁个数为 3 的比较结果

顶点数	边数	传统算法	常规算法	本文算法
100	136	6	5	1
150	200	8	8	2
200	265	11	11	2
250	348	14	15	6
300	413	20	23	21
350	483	24	23	23
400	543	27	32	40
450	625	36	37	53

表 3　平均并发变迁个数为 3.5，最大并发变迁个数为 7 的比较结果

顶点数	边数	传统算法	常规算法	本文算法
100	211	6	7	1
150	310	8	11	2
200	428	11	14	5
250	517	14	18	7
300	617	18	21	10
350	746	23	27	11
400	846	28	32	21
450	938	29	37	23

表 4　平均并发变迁个数为 17，最大并发变迁个数为 42 的比较结果

顶点数	边数	传统算法	常规算法	本文算法
100	920	13	51	4
150	1419	17	78	6
200	1849	22	100	18
250	2409	30	148	19
300	2929	38	174	26
350	3405	47	208	37
400	3856	53	231	49
450	4435	61	272	50

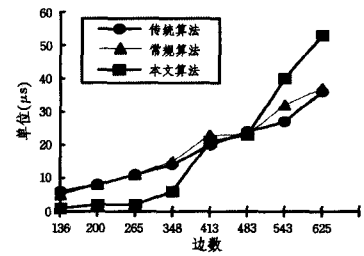

图 4　平均并发变迁个数为 2，最大并发变迁个数为 3 的比较结果

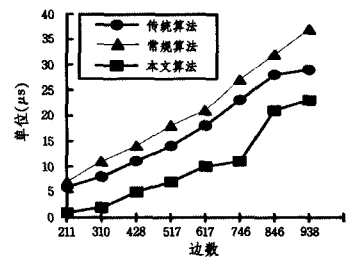

图 5　平均并发变迁个数为 3.5，最大并发变迁个数为 7 的比较结果

(下转第 141 页)

link for sensor networks[J]. Chinese Journal of Sensors and Actuators, 2007, 20(8): 1846-1851

[15] Guo Rui, Liu Chun-yu, Zhang Hua, et al. Full Diversity LDPC Codes Design and Energy Efficiency Analysis for Clustering Wireless Sensor Networks[J]. Journal of Electronics & Information Technology, 2015, 37(7): 1580-1585(in Chinese)
郭锐, 刘春于, 张华, 等. 分簇无线传感器网络中根校验全分集LDPC码设计与能效分析[J]. 电子与信息学报, 2015, 37(7): 1580-1585

[16] Yang Y, Zhong C, Sun Y, et al. Network coding based reliable disjoint and braided multipath routing for sensor networks[J]. Journal of Network and Computer Applications, 2010, 33(4): 422-432

[17] Li S Y R, Yeung R W, Cai N. Linear network coding[J]. IEEE Transactions on Information Theory, 2003, 49(2): 371-381

[18] Yang Yu-wang, et al. Reliable Braided Multipath Routing with Network Coding for Underwater Sensor Networks[J]. China Ocean Engineering, 2010, 24(3): 565-574

(上接第 125 页)

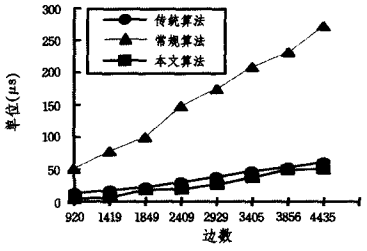

图 6 平均并发变迁个数为 17，最大并发变迁个数为 42 的比较结果

5.3 结果分析

从表 2 和图 4 可以看出，对于具有较少并发变迁的情况，当总的变迁个数较少(小于 400)时，本文算法性能优于其它两种算法，但当总的变迁个数较多(大于 400)时，本文算法性能低于其它两种算法。这是由于本文算法要对每一个变迁是否为并发变迁做出判断，需要花费一定的时间，而当并发变迁较少时，无法发挥本算法并行执行的特点。

但从表 3、表 4 和图 4、图 5 可以看出，对于具有较多并发变迁的情况(平均并发变迁个数大于 3)，本文算法性能优于其它两种算法，尤其是明显优于常规算法。因为常规算法要处理所有的可达状态，当变迁个数较多时，即便并发变迁较多，由于它没有考虑并发变迁的并发执行，因此需要花费大量时间。而对于本文算法来说，尽管总的变迁个数较多，但由于并发变迁较多，处理这些并发变迁时并行执行，因此可以节省大量时间。对于传统算法，由于它在计算各事件(对应 Petri 网中位置)的最早和最晚时间时所花费的时间要大大超过计算各活动(对应 Petri 网中变迁)的最早和最迟结束时间时所花费的时间，因此当活动的个数不变时，仅增加事件的个数其花费的时间增长，速度较慢。

结束语 本文利用有色时延 Petri 网描述了表示工程的 AOE 网，给出了构造有色时延 Petri 网的并发可达标识图的算法。利用有色时延 Petri 网和并发可达标识图给出了计算关键路径的方法。实例及仿真实验表明，当 AOE 网中的平均并发活动(即 Petri 网中的平均并发变迁)个数大于 3 时，本文算法明显优于传统计算关键路径的算法和利用可达状态求解关键路径的算法。而且 AOE 网中的并发活动(即 Petri 网中的并发变迁)越多，本文算法的优势越明显。

参 考 文 献

[1] 严蔚敏, 吴伟民. 数据结构(C 语言版)[M]. 北京: 清华大学出版社, 2008: 183-185

[2] East W. Critical Path Method(CPM) Tutor for Construction Planning and Scheduling [M]. McGraw-Hill, 2015

[3] He Li-hua, Zhang Lian-ying. An improved fuzzy network critical path method[J]. Systems Engineering — Theory & Practice, 2014, 34(1): 190-196(in Chinese)
何立华, 张连营. 改进的模糊网络关键路径法[J]. 系统工程理论与实践, 2014, 34(1): 190-196

[4] Cao Han, Liu Da-xin, Fu Rui. Workflow critical path algorithm based on activities[J]. Journal of Harbin Engineering University, 2006, 27(4): 551-555(in Chinese)
曹瀚, 刘大昕, 富锐. 基于活动的工作流关键路径算法[J]. 哈尔滨工程大学学报, 2006, 27(4): 551-555

[5] Hu Mei-qun, Xia Yin-shui, Wang Lun-yao. Critical Path Algorithm Based on Branch-and-bound[J]. Journal of Ningbo University(NSEE), 2011, 24(2): 37-41(in Chinese)
胡美群, 夏银水, 王伦耀. 基于分支限界的关键路径求解算法[J]. 宁波大学学报(理工版), 2011, 24(2): 37-41

[6] Liu Xiao-jing. The improvement and application of the key path algorithm to AOE-net[J]. Computer Systems & Applications, 2006(9): 47-53(in Chinese)
刘小晶. AOE 网的关键路径求解算法改进及其应用[J]. 计算机系统应用, 2006(9): 47-53

[7] Wu Zhe-hui, Wang Mei-qin. A kind of Peri nets involving time factors and their applications in ngineering[J]. Acta Mathematicae Applicatae Sinica, 1987, 10(3): 289-299(in Chinese)
吴哲辉, 王美琴. 一类含时间因素的 Petri 网及其在工程上的应用[J]. 应用数学学报, 1987, 10(3): 289-299

[8] Ye Shuang, Ye Jian-hong, Liu Chuan-cai. Algorithm for finding the Critical Paths Based on Petri Net[J]. Computer Science, 2012, 39(6): 201-203(in Chinese)
叶双, 叶剑虹, 刘传才. 基于 Petri 网的关键路径求解算法[J]. 计算机科学, 2012, 39(6): 201-203

[9] Zheng Wen-Yan. A New Method for Finding the Critical Paths Based on Colored Petri Nets[J]. Computer Systems & Applications, 2013, 22(8): 9-13(in Chinese)
郑文艳. 基于 CPN 的求解关键路径的新方法[J]. 计算机系统应用, 2013, 22(8): 9-13

[10] 吴哲辉. Petri 网导论[M]. 北京: 机械工业出版社, 2006

[11] Jensen K. Coloured Petri nets: basic concepts, analysis methods, and practical use[M]. Berlin: Springer Verlag, 1992

[12] Zuberek W M. Timed Petri nets: definitions, properties and applications[J]. Microelectronics and Reliability, 1991, 31(4): 627-644