

基于 Petri 网的关键路径求解算法

叶 双^{1,2} 叶剑虹² 刘传才¹

(南京理工大学计算机科学与技术学院 南京 210094)¹ (华侨大学计算机科学与技术学院 泉州 362021)²

摘 要 首先把工程规划中有向网络转换成带时间状态的 Petri 网,利用 Petri 网的分析方法对其不合理之处加以剖析,修正其中可能蕴含的错误,接着运行该网络并进行剪枝优化,自动获取关键路径。该方法比传统算法的执行效率要高,更易于实现。

关键词 带时间状态 Petri 网,关键路径,分析,算法

中图分类号 TP301.6 **文献标识码** A

Algorithm for Finding the Critical Paths Based on Petri Net

YE Shuang^{1,2} YE Jian-hong² LIU Chuan-cai¹

(School of Computer Science & Technology, Nanjing University of Science and Technology, Nanjing 210094, China)¹

(School of Computer Science & Technology, Huaqiao University, Quanzhou 362021, China)²

Abstract Using extended time Petri net, the directional network of project planning was converted into its Petri net model. Any potential limitations were analyzed by the methods of Petri net and corrected firstly. The critical path of net was automatically got after operating network for pruning optimization. This algorithm has better effectiveness than the existing methods, and is easy to realize.

Keywords Extended time Petri net, Critical path, Analysis, Algorithm

1 引言

在工程项目计划管理中,为了进行人力、物力的调试和分配、缩短工期,往往需要了解网络的总工期,即对关键路径加以求解。该问题已有多种算法,如传统的基于 AOE-网(Activity On Edge)算法^[1],它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。在此基础上,国内外学者相继提出一些新方法:文献[2]利用图的广度优先搜索与动态规划相结合来求解关键路径,而该算法采用图的邻接表结构,不需要进行拓扑排序,较之传统算法具有更高的效率和更强的健壮性,但算法求解过程中会出现关键路径局部路段丢失,没有把所有可能的关键路径完整输出;文献[3]在深度优先搜索的基础上,给出从源到汇的所有路径,经过分析比较找出最长的路径作为关键路径,求解过程中需多次递归回溯,算法的执行时间多于传统的 AOE-网算法;文献[4]的算法采用十字链表的存储结构,执行过程中需至少 3 次广度优先搜索;文献[5,6]利用模糊推理完成对整个计划按期完工的可能性估计,但由于引入了模糊数的概念,因此需要从每条路径的关键度去分析关键路径,在实际应用中,大型网络逐一计算每个路径的关键度比较困

难。此外,上述算法都基于一个假设,即实际项目的 AOE 网中的每个活动都至少可发生一次。在实际工程中,由于设计人员的经验缺失、对每个活动的具体情况把控不足,会导致所规划的工程网中存在诸如死锁(siphon)、陷阱(trap)、不活(no liveness)等^[7-9]不良要素。

本文给出求解关键路径的一个新方法。即先将网络图转换成带时间状态的 Petri 网(Extended time Petri Net, EtPN),接着对网做性质分析^[10],去掉其中不合理结构,随即运行该网络并将结果进一步剪枝优化,最终所得即为网中关键路径。本算法不仅能求出所有关键路径,而且时间复杂度方面略优于传统算法。

2 基本概念

假定读者已具有 Petri 网基本知识,在此仅给出本文所需概念。定义 1、2,定理 1 引自相关文献,其余定义由作者给出。

定义 1^[7-10] 一个基本 Petri 网系统是一个六元组 $\Sigma = (P, T; F, K, W, M)$, 满足:

- (1) $P = \{p_1, p_2, \dots, p_m\}$ 是一个有限库所集(place set);
- (2) $T = \{t_1, t_2, \dots, t_n\}$ 是一个有限变迁集(transition set);

到稿日期:2011-07-15 返修日期:2012-01-04 本文受福建省科技厅科技计划重大项目(2010N5008),厦门市科技局科技计划创新项目(3502Z20103027),华侨大学科研启动基金项目(09BS514),华侨大学基本科研业务费专项基金(JB-ZR1130)资助。

叶 双(1977—),女,博士生,讲师,主要研究方向为目标识别、认知计算机视觉及流媒体网络;叶剑虹(1976—),男,博士,讲师,CCF 会员,主要研究方向为大规模分布式并行计算、Petri 网理论及应用;刘传才(1963—),男,博士,教授,博士生导师,CCF 高级会员,主要研究方向为基于现代几何(包括微分几何、代数几何、随机几何)方法的目标识别、认知计算机视觉、水下图像的处理及基于视觉的导航和控制, E-mail: chuancailiu@mail.njust.edu.cn.

(3) $P \cap T = \phi, P \cup T \neq \phi$;

(4) $F \subseteq (P \times T) \cup (T \times P)$, F 是 N 上的流关系(flow relation), 其元素叫弧;

(5) $dom(F) \cup cod(F) = P \cup T, dom(F) = \{x | \exists y: (x, y) \in F\}, cod(F) = \{y | \exists x: (x, y) \in F\}$;

(6) M 是标识函数(marking): $P \rightarrow N, N = \{0, 1, 2, \dots\}$, K 为容量函数(capacity function): $P \rightarrow Z^+, Z^+ = \{1, 2, \dots\}$; $W: F \rightarrow N$ 称为网 N 上的权函数。

一般网系统都可借助原型网(elementary net system)加以定义, 即 $\forall p \in P: K(p) = \infty, \forall (x, y) \in F: W(x, y) = 1$ 。以下除非特别说明, 讨论均基于原型网系统。

定义 2^[11] 网 $N = (P, T; F)$ 称为一个标识图(T -图), 当且仅当 $\forall p \in P: |\cdot p| = |p \cdot| = 1$ 。

定理 1^[12] 若 N 是一个标识图, M_0 是其初始标识, 则 (N, M_0) 活的充要条件是对 N 的任意有向回路 $C, M_0(C) \geq 1$ 。

具有活性的系统在工作流网中也称为具有合理性^[10]。

定义 3 带时间状态 Petri 网(EtPN)是一个二元组 $EtPN = (\Sigma, DI)$, 其中 Σ 是一个标识网系统, DI 是定义在库所集合 T 上的时间函数, 即 $DI: P \rightarrow R^+, R^+$ 表示非负实数集。对于 $p \in P, DI(p) = a$ 表示托肯将在 p 中至少滞留 a 个单位时间。

定义 4 设 $EtPN = (\Sigma, DI)$ 是一个带时间状态的 Petri 网, M_0 是 Σ 的初始标识, $M \in R(M_0)$, 满足 $M[t > M']$, 若 $p \in P \wedge \cdot p = t$, 则 p 中所含 token 在 M' 下的标签定义为 $TokenFlag(M'(\tau_p) + DI(p))$, 其中 $M'(\tau_p)$ 是 t 发生后传递而来的 $\cdot t$ 中托肯标签。

3 基于 Petri 网的关键路径求解

3.1 算法实现

算法 1 基于 EtPN 的关键路径求解算法(Algorithm of Critical Path based on EtPN, CP-EtPN)

输入: 一个有向 AOE 网 $G = (V, E)$, V 中顶点代表事件, E 中弧代表活动, 弧上的权值 a 表示活动持续的时间, 其中 v_i 表示工程的起点, v_o 表示工程的终结;

输出: 网中的关键路径。

步骤 1 将 G 转换成带时间状态 Petri 网: 变迁 t_{ij} 表示顶点 v_j , 用库所 p_k 表示弧线 e_k , 弧上的权值作为库所 p_k 上的时间函数, 即 $DI(p_k) = a$ 。构成一个 EtPN 网, 转向步骤 2。

步骤 2 在此 EtPN 网中添加一个库所 p_0 , 满足 $\cdot p_0 = t_{vi}, p_0 \cdot = t_{vo}$, 标记为 $EtPN'$ 。此 $EtPN'$ 系统的初始标识为 $M_0(p_0) = 1$, 其余库所中初始托肯个数为 0。根据定理 1 判断该变迁外延子网中任意有向回路 C 中是否均有 $M_0(C) \geq 1$ 。摒弃 $M_0(C) = 0$ 的回路中不与其他回路产生交集的部分。最终所得网标记为 $EtPN''$ 。转向步骤 3。

定理 2 设 $EtPN = (\Sigma, DI)$ 是一个带时间状态的 Petri

网, 删除 $EtPN$ 网中不合理的部分并不影响整个网的关键路径。

证明: 所谓不合理部分, 即影响 $EtPN$ 网系统活的路径, 则至少存在一个回路 C , 满足 $M_0(C) = 0$ 。不妨令 C' 是一条通过辅助库所 p_0 贯穿工程起点 v_i 和工程终点 v_o 的有向回路, $M_0(C') \geq 1$ 。①若 $C \cap C' = \phi$, 删除 C 并不影响 C' 中变迁的发生; ②若 $C \cap C' \neq \phi$, 为了满足 $M_0(C) = 0$, 一定存在一段这样的路径: $\beta = (y') \cdot \rightarrow \dots \rightarrow \cdot y$, β 中的元素均属于 $C - C'$ 。由于 $EtPN$ 网是标识图, 故 y' 和 y 只能是变迁, 且均属于 $C \cap C'$ 。 β 中影响 $C \cap C'$ 中变迁发生的元素只有 $(y') \cdot$ 和 $\cdot y$, 删除它们, 无非减少了变迁 y 的一个输入条件和 y' 的一个输出库所。除此之外, β 中其他元素对 $C \cap C'$ 中变迁的发生不产生影响。故删除 β 中所有元素, 不改变 C' 中变迁发生条件。而最终关键路径一定遴选于若干条类似 C' 中的通路, 故定理结论成立。

步骤 3 运行 $EtPN''$ 网系统, M_0 是初始标识, $M \in R(M_0)$ 。如果有 $M[t > M']$, 则变迁 t 可发生的条件是 $\forall p' \in \cdot t: M(p') > 0$, 且 $\cdot t$ 库所中含最大托肯标签的计数达到; t 的发生还将向 $t \cdot$ 的库所 p 传递标签数为 $TokenFlag(M(\tau_p)) = \max(TokenFlag(M(\tau_{p'})))$, p 中所含 token 在 M' 下的标签值为 $TokenFlag(M'(\tau_p) + DI(p))$ 。我们摒弃含较小托肯标签的库所与 t 的连接弧。 $EtPN''$ 网中所有变迁发生后, 删除辅助库所 p_0 , 所得网记为 $EtPN'''$ 。转向步骤 4。

步骤 4 网中从 t_{vi} 到 t_{vo} 的一条有向路径即为 AOE 网的关键路径。转向步骤 5。

步骤 5 算法结束。

3.2 时间复杂度分析

不妨假令需求解的 AOE 网 G 中含 V 个顶点、 E 条边, 其中 $|V| = m, |E| = n$ 。CP-EtPN 算法的第一步近似认为可在单位时间内完成, 第 2 步和第 3 步需遍历所有的变迁和库所, 最终总的时间复杂度为 $O(m+n)$ 。和传统的基于 AOE-网的算法相比, 其时间常数因子略小一些, 因为 CP-EtPN 算法并不需要为每个活动求出最早开始时间和最迟开始时间, 也无需求取所有事件的最早发生时间和最迟发生时间, 只需对所有变迁发生做一次遍历, 即可求出关键路径。

3.3 算法实例

图 1(a) 引自文献[13]中旅行社业务进程实例的一部分。客户预定了某个行程 v_1 ; 旅行社的保险业务 v_2 、客户资料登记 v_3 和用户行程安排 v_4 这 3 个任务同时进行; v_2 和 v_3 完成后资料需递送旅行社主管审核 v_5 , 审核结束将进入财务环节, 等待用户支付费用 v_7 , 审核结果还将与行程安排一并送交用户, 等待反馈 v_8 ; v_4 和 v_6 分别是旅行社做详细行程安排及将其提交给用户; v_9 是客户旅程开始。图 1(b) 是依据步骤 1 所得的 $EtPN$ 网, 库所中的值代表该库所的时间函数。

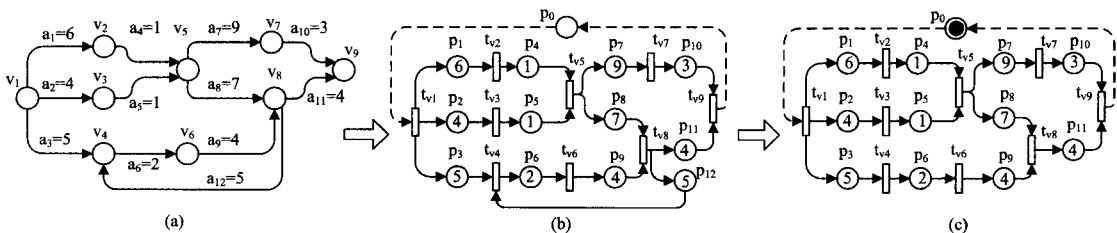


图 1 AOE 到 EtPN 的转换实例

图 1(b)中存在回路 $C=t_{v4} \rightarrow p_6 \rightarrow t_{v6} \rightarrow p_9 \rightarrow t_{v8} \rightarrow p_{12} \rightarrow t_{v4}$, $M_0(C)=0$, 根据步骤 2, 删除 C 中不与其他回路相交的部分 p_{12} 。产生 $t_{v8} \rightarrow p_{12} \rightarrow t_{v4}$ 这段不合理路径的初衷是设计者为了对 t_{v8} 的发生做出一个选择, 但这混淆了逻辑上的“任务完成”和语义的“案例决策”。 t_{v8} 的发生暗喻 p_{11} 、 p_{12} 中所含托肯的含义有所不同, 这会导致整个工作流网不具有畅通性^[14-16], 属于网中不合理部分。图 1(c)是改进后的 EtPN^{''}。运行该 EtPN^{''} 系统, 初始标识为 $M_0 = (1, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0)$, 变迁序列 $\sigma = t_{v1} t_{v2} t_{v3}$ 发生, 有 $M_0[\sigma > M]$, 标识 $M = (0, 0, 0, 1, 1, 1, 0, 0, 0, 0, 0, 0, 0, 0)$, $TokenFlag(M(p_4)) = 7$, $TokenFlag(M(p_5)) = 5$ 。若 t_{v5} 要发生, 除了保证 $M(p_4) > 0$ 和 $M(p_5) > 0$ 外, 还需等待 p_4 中所含托肯标签的计数达到。随着 t_{v5} 的发生, 系统摒弃含较小托肯标签的 p_5 与 t_{v5} 的连接弧, 并向其后续库所 p_7 、 p_8 传递标签数为 7 的托肯。其他变迁的发生类似。图 2(a)是最终所得 EtPN^{'''} 的网系统。

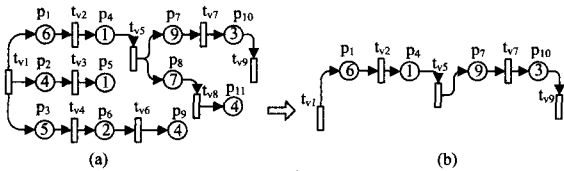


图 2 EtPN^{'''} 的关键路径求解

图 2(b)中, $t_{v1} \rightarrow p_1 \rightarrow t_{v2} \rightarrow p_4 \rightarrow t_{v5} \rightarrow p_7 \rightarrow t_{v7} \rightarrow p_{10} \rightarrow t_{v9}$ 是 EtPN^{'''} 中唯一含起点和终点变迁的有向路径, 它所对应的 AOE 网中关键活动有 v_1, v_2, v_5, v_7, v_9 , 完成整个工程最短时间为 19。显然, 要提高该旅行社业务进程的处理速度, 需特别关注保险业务的及时完工、缩短资料的审核时间及对用户款项的及时催缴。

4 实验仿真

4.1 算法实现

表 1 给出了 5 个从 YAWL Web^[1] 下载的实际工程规划案例, 每个案例均给出完整的事件集及它们之间的关系, 并依据经验值列出活动持续时间。案例 4、5 中含有不活、死锁等性质缺陷, 案例 1、2、3 满足合理性设计规范。

表 1 实际工程案例

名称	事件	活动
案例 1: 物流公司发货进程	9	10
案例 2: 保险公司业务进程	12	15
案例 3: 市政厅受理申诉进程	12	19
案例 4: 旅行社业务进程	15	18
案例 5: 有线电视公司业务进程	53	64

以图 1 展示的案例 4 的局部为例, 对算法的有效性进一步加以说明。用传统的 AOE-网算法求解图 1 给定的有向网 $G=(V, E)$, V 是事件集, E 是活动集的关键路径。需要对每个活动求出最早开始时间和最迟开始时间, 同时需进一步求

取所有事件的允许最早发生时间和最迟发生时间。整个算法对图中所有事件及活动需至少遍历 2 次。假定遍历一个结点需一个单位时间, 采用传统 AOE-网算法所需时间为 $2 * (V + E)$ 。图 1 中有 9 个事件、11 个活动, 故算法至少需要 40 个单位时间, 其中不包括 $v_8 \rightarrow v_4$ 错误路径导致进程在此无效循环上所徒耗的时间, 错误路径常会导致传统 AOE-网算法对关键路径的求解失败²⁾。采用 CP-EtPN 算法, 需要将所有的活动和事件映射成库所与变迁, 并连接成 Petri 网, 所需单位时间为 $V + E$ 。此后, 需要对 Petri 网中变迁的发生做一次遍历。不合理路径上的库所及求解过程中不会有托肯流经的库所将在遍历中被略过, 如图 1 中的库所 p_5, p_9, p_{11}, p_{12} 均在求解中被略过。CP-EtPN 算法最终所需时间为 $V + E + V + (8/12)E = 37.25$ 个单位时间。完整的旅行社业务进程中, 所涉及的活动及事件个数更多, 被剪枝及略过的库所个数也随之增加, 算法的效率提升更明显。对于整个案例 4 关键路径的求解, CP-EtPN 算法比传统 AOE-网算法减少了约 15% 的单位时间。

图 3 给出了每个案例通过 CP-EtPN 算法求得关键路径所需的时间, 分别用实线加以连接, 虚线是用传统 AOE-网算法求解的结果。显然, CP-EtPN 算法在关键路径的求解耗时略少于传统算法, 对含不合理结构的工程图, 如案例 4、5 效果更佳。

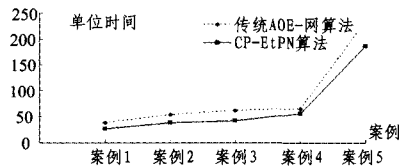


图 3 CP-EtPN 算法与传统 AOE-网算法求解时间比较

结束语 本文给出求解关键路径的新算法 CP-EtPN, 它具有以下特点: a) 对于规划图中一些设计人员疏忽所导致的错误, 可以借助 EtPN 网对其加以分析并修订, 从而为后续关键路径的求解打下良好基础; b) 由于采用了剪枝原理, 减少了遍历相应结点的次数, 同时减少了遍历路径时间的重复, 在时间复杂度上比传统算法略有提升。本算法已经应用于“基于混合模式流媒体调度技术研究”项目中, 取得了良好效果。

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, 2004: 50-52
- [2] 刘芳, 王玲. 基于动态规划思想求解关键路径的算法[J]. 计算机应用, 2006, 26(6): 1140-1142
- [3] 孟繁植. 求关键路径的一个算法[J]. 计算机工程, 1995, 21(4): 6-9
- [4] 徐风生. 一种新的关键路径求解算法[J]. 计算机应用与软件, 2005, 22(6): 97-99
- [5] Nasution S H. Fuzzy critical path method[J]. IEEE Trans on Systems, Man and Cybernetics, 1994, 24(1): 48-57

(下转第 221 页)

¹⁾ <http://www.yawlfoundation.org/>

²⁾ 在 Microsoft Visual C++ 6.0 编译环境下作者实现了传统 AOE-网算法对关键路径的求解。150 次运行程序对图 1 进行求解, 平均求解成功率仅为 34.7%, 原因在于存在 $v_8 \rightarrow v_4$ 错误路径导致算法常陷入死循环。读者可从 <http://download.csdn.net/source/3288918> 下载相关源代码自行测试分析。

近对一个连续时间系统直接分析得出的结果。

p_{ri} :是资源 ri 的耗用量在某个单位时间内落在 $[r_i^{\min}, r_i^{\max}]$ 内的概率。

P_{ri} :表示在 MI 的剩余运行时间内(假设为 l 个单位时间),资源 ri 的耗用量总是在范围 $[r_i^{\min}, r_i^{\max}]$ 内的概率。显然,如果对应于每一单位时间的 p_{ri} 都相同的话,则 $P_{ri} = (p_{ri})^l$ 。

利用一个概率模型来实现 MI 对迁移的判定,该方法基本思想如下:对于主机状态涉及到的每种资源,用耗用量的概率分布对其变化趋势建模。在每个时间步 T_i ,算法可获得关于近期资源耗用的最新统计数据,并在此基础上对概率函数进行修正。同时,MI 预测其剩余运行时间(用单位时间的数量来衡量),并将该数值传给主机。主机可计算对于每种资源的 p_{ri} 和 P_{ri} 。此外, P_i^S 和 P_i^E 也同时被计算。 P_i^S 是 MI 做出迁移决定的参考。假如它比预先规定的基准低,MI 可认为在其剩余时间内主机状态的变化不会对其产生影响。否则,MI 应该迁移到另外一台更合适的主机来完成剩余的任务。本文采用预测任务完成时间作为目标主机动态状况的度量。

5 实验

本节将通过实验验证本文提出的迁移策略的有效性。实验主要通过对主机性能的评价,说明迁移策略与随机选择方法的优劣。

作为初步研究,我们设计让 MI 计算两个 100 阶矩阵的乘积,并将所有中间数据写入一个磁盘文件。虽然这只是一中较简单的情况,但从实验得出的结论同样适用于复杂的情况。

首先在每台主机上随机启动一些程序,用以区分不同的工作及性能;依次在选定的 6 台工作机 $H_1, \dots, H_6$ 上进行评价,记录 $S(H_i)$ 和 $D(H_i)$ 。实验时,式(1)中 a 与 b 的取值均为 1。

表 1 各目标机可用度评测结果

实验次数	$S(H_i)$	$D(H_i)$	$V(H_i)$
H_1	41.8	26.1	67.9
H_2	40.7	23.3	64
H_3	44.6	24.9	73.9
H_4	45.7	28.2	73.9
H_5	51.4	26.4	77.8
H_6	48.6	28.3	73.9

从表 1 可以看出,不同的主机其静态性能指标和动态性

能指标具有较大的差异。与采用随机方法相比,在实际迁移时选择 $V(H_i)$ 较小的目的主机能有效地提高系统的效率。

结束语 迁移实例的迁移策略在迁移 workflow 系统的整体性能表现中起非常关键的作用。本文针对迁移实例迁移时遇到的服务定位、迁移目的地选择等问题提出了新的解决方案:在服务定位方案中,提出了一个模型让 MI 对网络上的服务进行定位。该模型是分层结构的注册服务器集合,它们负责管理服务并为 MI 导航;在目的主机可用度评测方案中,提出了一个包含静态性能和动态性能的评价方法。该方法的有效性通过实验得到了验证^[11,12]。

下一步研究将对本文提出的主机性能评测方法增加任务与服务之间的匹配程度来进行度量;同时赋予 MI 通过学习来智能选择特征参数进行评测的功能。

参考文献

- [1] Workflow Management Coalition, Workflow Management Coalition Terminology & Glossary WPMC-TC-1011[R], 1996
- [2] 曾广周,党妍. 基于移动计算范型的迁移 workflow 研究[J]. 计算机学报, 2003, 26(10): 1343-1349
- [3] 刘大有,杨博,杨鲲. 基于旅行图的移动 Agent 迁移策略[J]. 计算机研究与发展, 2003, 40(6): 838-845
- [4] 陈洪娜,祖旭,周峰. 工作流技术研究发展状况、研究内容及趋势[J]. 重庆工学院学报: 自然科学版, 2006(02): 65-69
- [5] Wang J, Rosca D. Dynamic Workflow Modeling and Verification[J]. Lecture Notes in Computer Science, 2006, 4(1): 303-311
- [6] 高丛. Mobile Agent 的迁移策略[D]. 青岛: 中国海洋大学, 2004
- [7] 范玉顺,吴澄. 工作流管理技术研究与产品现状及发展趋势[J]. 计算机集成制造系统-CIMS, 2000, 6(1): 1-7
- [8] 罗海滨,范玉顺,吴澄. 工作流技术综述[J]. 软件学报, 2000, 11(07): 899-907
- [9] 孙瑞志,史美林. 支持工作流动态变化的过程元模型[J]. 软件学报, 2003, 14(1): 62-67
- [10] 丁柯,金蓓弘,冯玉琳. 事务工作流的建模和分析[J]. 计算机学报, 2003, 26(10): 1304-1311
- [11] 吴修国,曾广周,韩芳溪,等. 迁移 workflow 中的目标规划研究[J]. 计算机科学, 2008, 35(01): 147-150
- [12] 唐卓,刘国华,李肯立. 多域环境下工作流访问控制时序策略组合研究[J]. 计算机科学, 2011, 38(01): 125-129
- [12] 蒋昌俊. 离散事件的动态系统的 PN 机理论 [M]. 北京: 科学出版社, 2000: 129-170
- [13] 叶剑虹. Petri 网若干关键技术的研究与应用[D]. 成都: 电子科技大学计算机科学与工程学院, 2009
- [14] van der Aalst W M P, Basten T. Inheritance of workflows; an approach to tackling problems related to change [J]. Theoretical Computer Science, 2002, 270: 125-203
- [15] Lautenbach K. Reproducibility of the Empty Marking [C]// the 23rd International Conference on Applications and Theory of Petri Nets. LNCS 2360, Adelaide, Australia; Springer, 2002: 237-253
- [16] 叶剑虹,宋文,孙世新. 空标识可再生网的运算和性质分析 [J]. 计算机研究与发展, 2009, 46(8): 1378-1385
- [6] Yao Jin-Shing, Lin F T. Fuzzy critical path method based on signed distance ranking of fuzzy numbers [J]. IEEE Trans on Systems, Man and Cybernetics, 2000, 30(1): 76-82
- [7] Reisig W. Petri Nets; An Introduction [M]. Berlin, Heidelberg: Springer-Verlag, 1985: 17-135
- [8] 袁崇义. Petri 网原理与应用 [M]. 北京: 电子工业出版社, 2005: 114-188
- [9] 吴哲辉. Petri 网导论 [M]. 北京: 机械工业出版社, 2006: 27-100
- [10] van der Aalst W M P, van Hee K. Workflow Management Models, methods and systems [M]. Cambridge: MIT Press, 2002: 22-115
- [11] Murata T. Petri Nets; Properties, Analysis and Applications [J]. IEEE Trans on Software Eng, 1987, 77(4): 541-580

(上接第 203 页)