

# 基于着色 Petri 网的项目进度计划<sup>①</sup>

## Progress of Project Based on Colored Petri Nets

任雪利 (曲靖师范学院 计算机科学与工程学院 云南 曲靖 655011)

**摘要:** 如何处理进度与成本之间的关系是提高软件项目开发成功的重要保证, 提出根据 PERT 图构造着色 Petri 网来表示项目进度计划, 依据转移条件判断变迁的发生, 从而及时调整资源与进度的相互关系, 为项目的进度计划提供有力的依据, 取得了良好的效果。

**关键词:** 着色 Petri 网 计划评审技术 资源 成本 进度

进度, 成本是软件项目管理的重要内容, 在项目进度安排时, 不仅需要考虑到时间因素, 而且应该考虑到成本以及其他因素对项目进度的影响, 在软件开发中, 成本通常以消耗的资源来计量(本文中, 成本与资源的含义相同)。进度与资源相互影响、相互制约, 人力资源的不足会导致进度的延期; 同样, 财力资源的缺乏也会影响到项目的进度安排。资源作为项目开发的重要因素是所有项目都必须考虑的, 即资源与进度的影响、制约关系在项目开发中是不可忽略的, 本文就进度(时间)与资源的相互关系进行探讨。

### 1 项目进度安排方法的比较

安排项目进度计划的方法有: Gantt 图法和计划评审技术(PERT)<sup>[1]</sup>。Gantt 图简单易用, 易于修改, 但不能显示各项活动之间的依赖关系; PERT 虽然可以明确的表示出各项活动之间的依赖关系, 但是, 该图表明了时间在项目开发过程中的重要性, 没有涉及其他因素的影响, 例如: 人力资源和财力资源, 因为在制定项目进度计划时, 不仅需要考虑到时间因素, 而且必须考虑到资源的影响。本文在 PERT 的基础上, 提出了基于着色 Petri 网的项目进度计划。

### 2 着色 Petri 网

着色 Petri 网(CPN 或 C Pnets)是一种精确的形式化定义语言, 其工具 Design/CPN 提供画图、仿真和分析等几乎所有的功能。由于 Petri 网可以简单、灵活

地描述系统的动态行为, 以及对系统的行为进行强有力的表达和分析。其形式化定义如下:

CPN 是一个 9 元组  $CPN=(\Sigma, P, T, A, N, C, G, E, I)$ <sup>[2,3]</sup>

1)  $\Sigma$  是一个类型有限非空集合, 称作颜色集, 颜色集用于标注类型、运算和函数, 例如: 弧表达式、识别、初始化表达式和颜色集。

2)  $P$  为库所(Place)的有限集合。

3)  $T$  为变迁(Translation)的有限集合。

4)  $A$  为弧(Arc)的有限集合,  $P \cap T = P \cap A = T \cap A = \phi$ , 与普通 Petri 网不同的是: 在这里网结构可以为空, 即  $P \cup T \cup A = \phi$ , 这样可以在构造一个复杂的 CPN 模型之前, 先定义颜色集合, 并对其做语法检查; CPN 允许一对有序节点之间存在几条弧, 所以把弧  $A$  定义为一个独立的集合, 而不是  $P \times T \cup T \times P$  的子集。

5)  $N$  为节点(Node)函数,  $N: A \rightarrow P \times T \cup T \times P$ , 节点函数把每条弧映射到一个二元组, 第一个为源节点, 第二个为目标节点, 且这两个节点属于不同的类型, 即(变迁, 库所)或(库所, 变迁)。

6)  $C$  为着色(Color)函数,  $C: P \rightarrow \Sigma$ ,  $C$  把每个库所  $p$  都映射到一个颜色集  $C(p)$ , 即  $p$  中的每个令牌(token)都属于颜色类型。

7)  $G$  为识别(Guard)函数,  $G: T \rightarrow \text{Exprs}$ , 且满足  $t \in T: [\text{Type}(G(t)) = \text{Bool} \wedge \text{Type}(\text{Var}(G(t))) \subseteq \Sigma]$ 。  $G$  把每个变迁  $t$  都映射到一个布尔表达式, 并且  $G(t)$  中所有变量的类型必须包含于颜色集中。

8)  $E$  为弧上的条件表达式(Arc Expression)函数,

① 收稿时间: 2008-12-22

$E:A \rightarrow \text{Exprs}$ , 且满足  $\forall a \in A: [\text{Type}(E(a)) = C(p(a))_{MS} \wedge \text{Type}(\text{Var}(E(a))) \Sigma]$ , 其中:  $p(a)$  表示节点函数  $N(a)$  中的库所,  $MS$  表示集合  $C$  上的多种集合的集合。E 把每条弧  $a$  都映射到一个具有类型  $C(p(a))_{MS}$  的表达式, 这表示  $E(a)$  的每次求值都生成一个依赖于相邻库所颜色集的多重集合。

9)  $I$  为初始化(Initialization)函数,  $I:P \rightarrow \text{Exprs}$ , 且满足  $\forall p \in P: [\text{Type}(I(p)) = C(p)_{MS} \wedge \text{Var}(I(P)) = \phi]$ ,  $I$  把每个库所映射为不包含变量的表达式, 其类型为  $C(p)_{MS}$ 。

### 3 着色Petri网在安排项目进度中的应用

PERT 网是活动之间依赖关系组成的进度安排网络图, 在活动中标识出了实现该活动计划需要的时间。在着色 Petri 网安排进度过程中, 当变迁发生时, 消耗输入库所中的令牌, 增加输出库所中的令牌; 项目资源分为两类: 人力资源和财力资源, 其中: 人力资源是一种可再生的资源, 不会随着使用而消耗, 可以无损的增加至输出库所中; 而财力资源是一种不可再生的资源, 会随着变迁的发生而消耗, 不会无损的增加至输出库所中, 只会将没有耗尽的财力资源增加至输出库所中。因此, 应区分对待这两种令牌。

使用着色 Petri 网安排进度时, 颜色集表示资源类型的集合, 库所表示软件开发过程中的状态, 变迁表示开发过程, 弧表示变迁的路径, 节点函数将弧映射到变迁路径, 着色函数表示库所中拥有的资源类型与数量, 识别函数表示实现变迁应满足的条件, 弧上的条件表达式表示变迁实现后, 资源的分配策略或变迁过程中可能出现的一些情况, 初始化与着色 Petri 网中的定义相同。在此, 为了表示出资源对项目进度安排的影响, 需要对着色 Petri 网定义中的识别函数进行扩充。

着色 Petri 网络在安排项目进度时, 识别(Guard)函数  $G$  定义为:  $G:T \rightarrow \text{Exprs}$ , 且满足  $\forall t \in T: [\text{Type}(\text{Var}(G(t))) \Sigma] \wedge [\text{Val}(\text{Type}(G(c))) \geq \text{SVal}(\text{Type}(G(c)))] \wedge [\text{Stu}(\text{Dep}(t)) = T]$ 。Val 表示目前该库所拥有的资源数量, SVal 表示变迁发生所需资源的最低要求, Dep(t)表示该变迁依赖的变迁集, Stu(Dep(t))表示变迁的状态(实现与否); 要实现变迁, 不仅要求  $G(t)$  中令牌的类型必须包含于颜色集, 而且必须满足其数量不

小于实现变迁所需此种令牌的最少量及该变迁所依赖的变迁均已实现, 如果变迁的发生不依赖于任何其他变迁, 则将  $\text{Stu}(\text{Dep}(t))$  设置为  $T$  即  $\text{Stu}(\text{Dep}(t)) = T$ 。G 将变迁  $t$  映射到一个布尔表达式, 当  $G$  为  $T$ (真)时, 该变迁可以发生, 否则, 变迁不能发生。

下面给出使用着色 Petri 网安排项目进度的一般步骤:

1) 将 PERT 图转化为 Petri 网。PERT 图中的活动(除终点外)转化为 Petri 网络中的节点(库所, 变迁), 节点间的依赖关系转化为两个节点((库所, 变迁), (库所, 变迁))之间的依赖关系, 终点作为 Petri 网的终点;

2) 标识出库所目前拥有的令牌(人力资源与财力资源)及其数量, 根据变迁发生所需的最少资源数量, 计算识别函数的值, 判断出目前变迁发生与否。库所  $S$  表示项目拥有的人力资源 and 财力资源总量。

3) 如果变迁的实现过程复杂, 可以将其分解为一个单独的着色 Petri 网络, 形成一种层次型的着色 Petri 网络。

基于着色 Petri 网安排项目进度可以表示出资源对项目进度的影响, 通过判断函数的值, 可以准确判断并发的变迁。如果存在可能拖延整个项目的变迁, 应及时调整资源的分配策略, 降低项目的进度风险, 确保项目在预定的时间限度内完成。

### 4 着色Petri网在安排项目进度的实例

以水资源调度管理信息系统开发中为例, 该项目包含 10 个活动, 其中:  $S$  表示起点,  $A$  表示分析,  $TP$  表示测试计划,  $D$  表示设计,  $TD$  表示测试数据,  $DM$  表示文档,  $TS$  测试软件,  $C$  表示编码,  $PT$  表示产品测试,  $E$  表示终点, 画出 PERT 图(图中没有标识出每个活动计划需要的时间)如图 1 所示:

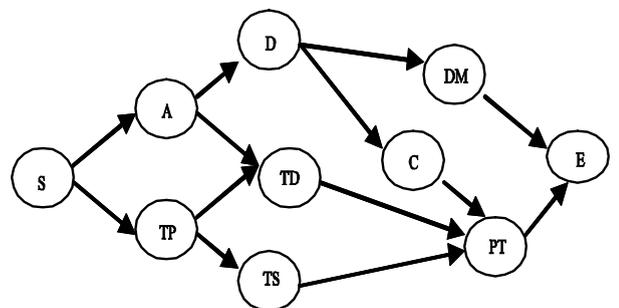


图 1 PERT 图

构造 Petri 网模型,如图 2 所示:

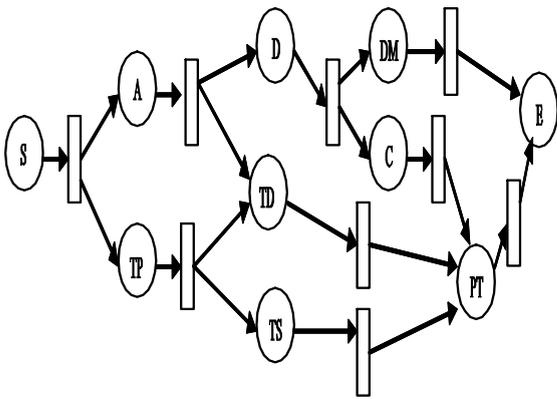


图 2 Petri 网

变迁  $t_1$  表示资源的分配策略, 如果项目可以进行,  $t_1$  总是可以发生的; 实现变迁  $t_2$ , 至少必须具备的人力资源和财力资源分别为 10 人和 2 万元, 变迁  $t_3$  至少必须具备 8 人和 1.3 万元, 变迁  $t_4$  至少必须具备 6 人和 1.2 万元... 变迁  $t_{10}$  至少必须具备 10 人和 1.5 万元, 构造着色 Petri 网模型, 如图 3 所示:

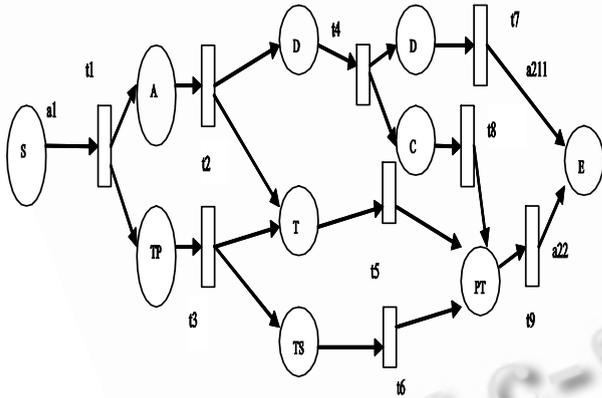


图 3 着色 Petri 网

在图 3 的着色 Petri 网中:  $\Sigma = \{R_p, R_s\}$ , 其中:  $R_p$  是人力资源,  $R_s$  是财力资源;  $P = \{S, A, TP, D, TD, TS, DM, C, PT, E\}$ ;  $T = \{t_1, t_2 \dots t_9\}$ ;  $A = \{a_1, a_2 \dots a_{22}\}$ ;  $N: \{a_1 \rightarrow (S, t_1), a_2 \rightarrow (t_1, A) \dots a_{21} \rightarrow (t_8, E), a_{22} \rightarrow (t_{10}, E)\}$ ;  $C: \{S \rightarrow \{R_p, R_s\}, A \rightarrow \{R_p, R_s\} \dots\}$ ;  $G: R_p \Sigma \wedge Val(R_p) \geq SVal(R_p) \wedge R_s \Sigma \wedge Val(R_s) \geq SVal(R_s) \wedge (Stu(Dep(t)) = T)$ ; 由图 3 可知: 库所 A 目前拥有的人力资源和财力资源分别为 15 人和 2.4 万元, 而实现变迁  $t_2$  的最少人力资源和财力资源分别是 10 人和 2 万元, 因此, 识别函数的值为真, 目前变迁  $t_2, t_3$  都是可以发生的, 由于  $t_2$  处在关键路径上, 所以应该优先保证  $t_2$  的发生, 然后再考虑  $t_3$ 。该项目的开发过程中使用着色 Petri 网来安排项目的进度取得了良好的效果。

### 5 总结

在项目进度安排中考虑资源是合理的, 也是必不可少的, 用着色 Petri 网不仅可以表示项目计划中资源的分配策略, 而且能更好的模拟变迁之间的并发, 使得项目进度安排更为合理、更加准确, 为成功的项目管理提供了依据。

### 参考文献

- 1 Pressman RS. 软件工程实践者的研究方法 (第 5 版). 北京: 机械工业出版社, 2002: 132 - 135.
- 2 Jensen K. Condensed State Spaces for Symmetrical Coloured Petri Nets. Formal Methods in System Design 9, Kluwer Academic Publishers, 1996: 7 - 40.
- 3 Desrochers A, Deal TJ, Fanti MP. Complex-valued token Petri nets. IEEE Transactions on Automation Science and Engineering, 2005, 2(4): 309 - 318.