

## 基于着色 Petri 网的多 Agent 系统交互协议建模

韩冰娣, 郑丽英

(兰州交通大学 信息与电气工程学院, 甘肃 兰州 730070)

**摘要:**多 Agent 系统(Multi-Agent Systems, MAS)中, 多个 Agent 通过交互和协作来完成一系列任务或实现一些目标。Agent 之间有效、有序地进行交互是 MAS 成功运行的关键。文中采用着色 Petri 网来表示一个多 Agent 系统。利用着色 Petri 网, 便于描述并发现现象和模拟平行系统, 除了直观的图形化表示, 还具有精确的形式化定义, 并且有完善的分析工具。最后对 FIPA 规范中的 FIPA Inform 和 FIPA Request 两个协议进行实例分析, 说明如何用着色 Petri 网进行建模。

**关键词:**MAS; 着色 Petri 网; 交互协议; 建模

**中图分类号:**TP18

**文献标识码:**A

**文章编号:**1673-629X(2006)11-0042-02

Interaction Protocols Modeling for Multi-Agent Systems  
Based on Colored Petri Net

HAN Bing-di, ZHENG Li-ying

(School of Information &amp; Electrical Engineering, Lanzhou Jiaotong University, Lanzhou 730070, China)

**Abstract:** In a multi-agent system (MAS), multi agents complete a series of task or realize some goals by interaction and cooperation. The key point of MAS successful running is efficient and ordered interaction among agents. In this paper, present a colored Petri net (CPN) express MAS. CPN describes concurrent phenomena and simulates a parallel system. Besides a visual graphics description, CPN has an exact definition and perfect analysis tool. Then analyse FIPA request and FIPA inform protocol of FIPA criterion and present how to model with CPN.

**Key words:** multi-agent system; colored Petri net; interaction protocol; modeling

## 0 引言

在 MAS 中, 多个 Agent 通过交互和协作来完成一系列任务或实现一些目标, Agent 之间有效、有序的进行交互是 MAS 成功运行的关键。KQML 与 FIPA ACL 是 Agent 之间消息传递的重要通信语言, 保证了交互各方都能够交换并理解消息, 但 Agent 要想完整地表达自己的意图, 必须通过一系列有组织和结构的消息交换来实现, 即信息交换需要遵循一些标准的模式, 这种模式即 Agent 交互协议。由于 MAS 的应用领域变得越来越复杂, 现有的许多 Agent 交互协议都暴露出了或多或少的缺点, 阻碍了 MAS 的应用<sup>[1-3]</sup>。总之, 现有 Agent 交互协议灵活性和健壮性的缺乏限制了 MAS 的应用, 而如何建立一种灵活的 Agent 交互协议则成了 MAS 研究领域的一个重要问题。着色 Petri 网 (Colored Petri Nets, CPN) 为描述、构建和分析分布式和并发系统提供了一种合适的数学工具<sup>[4]</sup>。CPN 进一步扩展了 Petri 网的令牌 (token) 机制, 使它具有

颜色 (Color) 和类型 (Type), 能用于描述复杂的数据对象。利用 Petri 网或 CPN 为 Agent 交互协议建模已经取得了一些成果。

## 1 着色 Petri 网 (CPN) 系统

文中采用着色 Petri 网来表示一个多 Agent 系统。着色 Petri 网是为多 Agent 系统交互协议进行建模的最好工具之一。利用着色 Petri 网, 便于描述并发现现象和模拟平行系统, 除了直观的图形化表示, 还具有精确的形式化定义, 并且有完善的分析工具。

文献[4]和[5]用3个定义分别对 CPN 系统、变迁实施的条件和实施后的结果给出定义。在定义中,  $S_{MS}$  表示集合  $S$  上的所有有限多重集所成之集合;  $[S \rightarrow R]_l$  表示从集合  $S$  到集合  $R$  的所有线性函数的集合;  $D$  为一个已知的颜色集。

定义1  $\sum = (P, T; F, C, I_-, I_+, M_0)$  为着色 Petri 网系统, 它的充要条件是:

(1)  $(P, T; F)$  为有向网, 称为  $\sum$  的基网;

(2)  $C: P \cup T \rightarrow P(D)$ ,  $P(D)$  为颜色集  $D$  之幂集合, 使得: 对  $p \in P$ ,  $C(p)$  是库所  $p$  上所有可能的托肯色 (资源类) 集合。对,  $t \in T$ ,  $C(t)$ ,  $C(t)$  是变迁  $t$  上所有可能

收稿日期: 2006-01-16

基金项目: 教育部“春晖计划”资助项目 (20455)

作者简介: 韩冰娣 (1977-), 女, 黑龙江宾县人, 硕士研究生, 研究方向为移动 Agent 技术、分布式入侵检测; 郑丽英, 教授, 研究方向为数据结构与算法、移动 Agent 技术、分布式入侵检测。

的出现色集合;

(3)  $I_-$  和  $I_+$  分别是  $P \times T$  上的负函数和正函数,使得对所有  $(p, t) \in P \times T$ ;

$I_-(p, t) \in [C(t)_{MS} \rightarrow C(p)_{MS}]_L$ , 且  $I_-(p, t) = 0$  的充要条件是  $(p, t) \in F$ ;

$I_+(p, t) \in [C(t)_{MS} \rightarrow C(p)_{MS}]_L$ , 且  $I_+(p, t) = 0$  的充要条件是  $(t, p) \in F$ ;

(4)  $M_0: P \rightarrow D_{MS}$  称为  $\sum$  的初始标识,它必须满足条件:  $\forall p \in P, M_0(p) \in C(p)_{MS}$ , 即  $M_0(p)$  是  $p$  的令牌颜色集合上的多重集。

定义 2  $M: P \rightarrow D_{MS}$  为着色网系统  $\sum = (P, T; F, C, I_-, I_+, M_0)$  的标识,它的条件是:  $\forall p \in P: M(p) \in C(p)_{MS}$ 。  $X: T \rightarrow D_{MS}$  为  $\sum$  的一步的条件是:  $\forall t \in T: X(t) \in C(t)_{MS}$ , 且  $\forall p \in P; \sum_{t \in T} I_-(p, t)(X(t)) \leq M(p)$  (1)

$\sum$  的标识就是为每个库所  $p$  指定它的托肯色集合  $C(p)$  上的一个多重集  $M'$ , 即各类资源各有几个。 $\sum$  的一步就是为每个变迁  $t$  指定它的出现色集合  $C(t)$  上的一个多重集, 即指明各出现色出现的次数。

定义 3 设  $X$  为着色网系统  $\sum$  在标识  $M$  下的一步, 则  $X$  发生后的后继标识  $M'$  有:

$$\forall p \in P, M'(p) = M(p) + \sum_{t \in T} I_+(p, t)(X(t)) - \sum_{t \in T} I_-(p, t)(X(t)) \quad (2)$$

$M$  和  $M'$  的后继关系记作  $M[X > M']$ 。

定义中的“+”和“-”均是多重集间的运算。由式(2)的定义可知“-”运算是定义的。

## 2 使用着色 Petri 网为 MAS 交互协议建模

使用着色 Petri 网为 MAS 交互协议建模, Agent 交互的状态被表现为着色 Petri 网中的库所(Place)。库所的类型决定了其所能容纳的 token 的类型。token 代表了 Agent 之间交换的数据, token 的颜色代表了 token 的数据值。变迁(Transition)和与变迁相连的弧(Arc)上规定了协议中 Agent 的交互策略(Interaction Policies)。具体的描述含义见表 1。

表 1 着色 Petri 网的符号定义描述

名称	描述
Place	Agent 交互的状态, 代表库所
Token	代表了 Agent 之间交换的数据, 代表消息
Arc	变迁和与变迁相连的弧, 代表消息传输机制
Transition	代表消息处理单元

如果所有的输入库所都有 token, 并且 token 的颜色都满足在弧上规定的约束, 变迁就可以被启动。变迁启动后, 根据对话策略消耗掉所有输入库所中的相应的 token, 在所有输出库所产生新的 token。变迁发生后, 协议的状态就发生了改变, 当无变迁再可被启动时, 协议就到达了终止状态。下面以 FIPA 规范中的定义的两个 MAS 协议

为例来说明怎样利用着色 Petri 网来为 Agent 交互协议建模。

### 2.1 FIPA Inform 协议

在 Inform 协议中, 参与会话的只有两个 Agent, 其中一个 Agent 发送消息给另外一个 Agent。接收者 Agent 收到消息后, 根据自己的心智状态等对其进行处理。如图 1 所示: 此交互过程共有 5 个状态, 分别用一个库所来表示。如果在 Start 库所中有一个或多个 token, 并且这些 token 满足 Start 弧上规定的约束(属于“message”数据类型), 变迁“Inform”就可以启动。当“Inform”变迁启动后, 一个 token 将从 Start 库所, 分别转移到“Received”和“Terminated1”库所。 “Received”库所得到一个新的 token 后, “process inform”变迁就满足了启动的条件。最后当变迁“process inform”启动后, 交互将被终止。

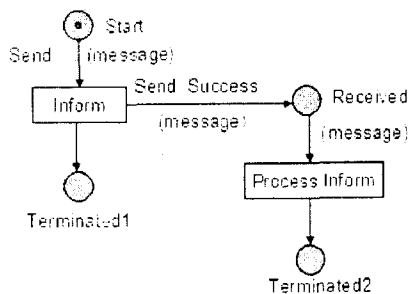


图 1 FIPA Inform 协议的 CPN 模型

### 2.2 FIPA Request 协议

FIPA Request 协议也是 FIPA 规范中的定义的一个基本 Agent 协议。它允许一个 Agent 请求另一个 Agent 完成相应的行为。这个请求可能被拒绝, 也可能被接收; 接收者 Agent 可能成功完成请求的行为, 也可能失败。如图 2 所示: 在 Start 库所中有一个或多个 token, 当 token 满足 Start 弧上规定的约束, 变迁“Prepare Request”就可以启动。当“Prepare Request”变迁启动后, 一个 token 将从 Start 库所, 分别转移到“Refuse”和“Agree”库所。“Agree”库所得到一个新的 token 后, “process inform”变迁就满足了启动的条件。最后当变迁“process inform”启动后, Request 协议就到达了终止状态。

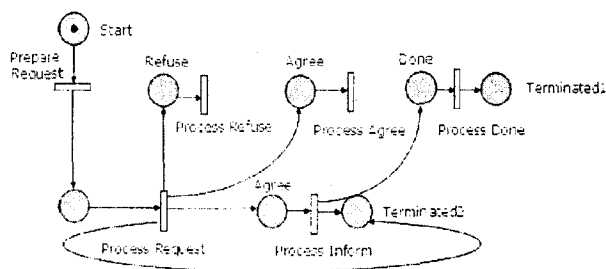


图 2 FIPA Request 协议的 CPN 模型

## 3 结束语

在多 Agent 系统中, 为了实现系统的共同目标, 每个 Agent 都不可避免地要和其他 Agent 进行交互。鉴于多

(下转第 46 页)

阵之和:  $Q = Q_1 + Q_2$ , 每次应选择当前误差值  $Q(v)$  最小的边进行折叠。

要使折叠后的误差值最小, 则须对误差代价  $Q(v) = v^T Q v$  求偏导, 令偏导值为 0, 解出最佳折叠后新顶点位置向量  $v = -a^{-1}b$ , 相应的最小值为  $Q(v) = -b^T a^{-1}b + c$ 。有时  $a$  无法求逆矩阵, 也可在两个端点中选择一个代价较小的作为新顶点。这样的好处是程序更简单、速度更快、内存占用更小, 但简化的效果上要差一点。

### 1.3 算法梗概

根据上面的讨论, 文中提出的递进网格的生成方法可用下面的算法描述:

Step1 读入初始网格  $M = M_n$ 。

Step2 对  $M$  中的所有边, 计算折叠误差, 并根据误差的大小进行排序。

Step3 取折叠误差最小的边进行折叠操作, 记录折叠信息, 更新与之相关的所有信息。

Step4 重复 Step3, 直至当前的简化网格满足要求。

Step5 输出简化网格  $M_0$  和边折叠记录序列(即递进网格)。

## 2 实验结果

文中所介绍的算法已经在 P4 2.4G, 256M 内存, NVIDIA Geforce2 MX 400 64M 显卡的联想品牌机上用 VC++ 6.0 和 DirectX8.0 实现, 系统为 Microsoft Windows XP。

图 3 是一组实例, 执行结果验证了算法的有效性。从图中可以看出, 不仅网格模型的几何信息得到了较好的保留, 其色彩信息也得到了较好的保留。

## 3 结束语

文中提出并实现了一种新的递进网格简化处理方式。新的方法采用扩展二次误差测度表示简化网格与初始网

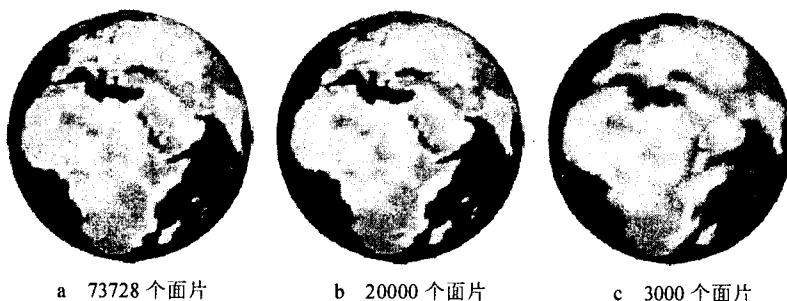


图 3 多属性递进网格模型简化实例

格的几何特征和其它属性信息的匹配程度, 利用二次误差值最小原理, 指导网格简化操作的进行。同 Hoppe 提出的基于能量优化的算法和以往大多只能完成几何特征简化的算法相比, 该算法不仅效率高, 而且可以保证简化模型同初始模型在几何特征和色彩、纹理等属性信息上尽可能相似。

在以后的研究中, 将考虑将该算法与视点和绘制算法结合起来, 形成与视点相关的递进网格, 从而支持实时图形绘制。

### 参考文献:

- [1] Hoppe H. Progressive meshes[J]. Computer Graphics, 1996, 30(1):99-108.
- [2] Taubin G, Gueziec A, Horn W, et al. Progressive forest split compression[C]//SIGGRAPH'98 Proceedings. [s. l.]: [s. n.], 1998:123-132.
- [3] 陶志良, 潘志庚, 石教英. 支持快速恢复的可逆递进网格及其生成方法[J]. 软件学报, 1999, 10(5):503-507.
- [4] 卢章平, 赵 泉. 基于“边折叠”的累进网格生成算法的研究[J]. 工程图学学报, 2004, 24(1):37-41.
- [5] 刘焕敏, 杨克俭, 王玉华. 一种面积加权的半边折叠网格简化算法及其递进网格构造[J]. 武汉理工大学学报, 2005, 29(2):76-78.
- [6] Garland M. Simplifying Surfaces with color and Texture using Quadric Error Metrics[C]//Proceedings of IEEE Visualization'98. [s. l.]: [s. n.], 1998:263-269.

(上接第 43 页)

Agent 系统的复杂性及其交互的动态性, 需要一种形式化的方法对交互协议进行建模分析和验证。文中提出一种基于着色 Petri 网的 MAS 交互进行建模研究方法, 通过 FIPA 交互协议建模过程, 可以看出 Agent 可利用这种方法来分析是否可理解接收到的协议、是否交互可被 Agent 的现有状态接收、是否与 Agent 自利的目标冲突等。

### 参考文献:

- [1] Nowostawski M, Purvis M, Cranefield S. A Layered Approach for Modelling Agent Conversations[C]//In: Proceedings of the 2nd International Workshop on Infrastructure for Agents,

MAS, and Scalable MAS. Montreal, Canada; [s. n.], 2001:163-170.

- [2] Cost R. Modelling Agent Conversations with Coloured Petri Nets[C]//Working Notes of the Workshop on Specifying and Implementing Conversation Policies. Seattle, Washington; [s. n.], 1999:59-66.
- [3] Poutakidis D, Padgham L, Winikoff M. Debugging Multi-Agent Systems Using Design Artefacts: The Case of Interaction Protocols[C]//In: Proceedings of the 1st International Joint-Conference on Autonomous Agents and Multi Agent Systems. Bologna, Italy; [s. n.], 2002:960-967.
- [4] 袁崇义. Petri 网原理[M]. 北京: 电子工业出版社, 2005.
- [5] 张维明. 信息系统建模[M]. 北京: 电子工业出版社, 2002.