XOCPN:支持用户交互的多媒体同步模型*

梁永全　　史忠植

(山东矿业学院计算机系,泰安　271019)　(中国科学院计算技术研究所,北京　100080)

摘　要　多媒体同步是多媒体应用的一个重要需求。本文对 OCPN 模型进行了扩展,提出了一种基于时间 Petri 网的多媒体同步模型 XOCPN,以对具有时间约束关系的多媒体合成进行规范描述,并具有支持灵活用户交互的能力。文中给出了处理异步用户交互的算法,并用示例说明。

关键词　时间 Petri 网,多媒体同步,异步用户交互

1　引　言

高级多媒体系统的特点是在计算机控制下各种静态和连续媒体的生成、存储、通信、和表现的集成。同步一词指的是时间,多媒体系统中的同步则指的是多媒体对象之间的时间关系,它是多媒体计算机及多媒体通信领域中的主要问题之一[1,2]。研究人员已从各方面对多媒体同步进行了研究,主要问题包括怎样规范多媒体对象之间的时间关系以及怎样实现同步,在对同步的规范描述方面已经发表了很多技术成果,已有的多媒体同步模型分 3 类:时间线模型、层次模型和参考点模型[3]。

Petri 网是一种系统描述和分析的工具,以其对并发和顺序活动的建模能力而著称。自从 C. A. Petri 博士最早于 1962 年提出 Petri 网概念以来,Petri 网理论得到了迅速的发展,并被广泛地应用于计算机科学技术和其它科学领域。Petri 网定义为一个有向二分图 $N=\{T, P, A\}$,其中 T,P 和 A 分别表示变迁、位置和有向弧集合。对于简单的 Petri 网而言,变迁的引发是一个瞬时事件。为了在 Petri 网中表示非零的时间跨度,就需要对源模型进行扩充。方案之一就是给网中的每个位置分配一个非负的延续时间,同时保持变迁的瞬时引发性质,我们一般称这类 Petri 网为时间 Petri 网[4,5]。

Little 和 Ghafoor 于 1989 年提出了一种多媒体对象的同步和存储模型,即 OCPN 模型,其全称为对象合成 Petri 网[4,6]。OCPN 是一种扩充的时间 Petri 网,它把多媒体对象的各个组成部分表示成位置,而把各部分之间的相互关系表示成变迁。在文献[4]中,分析了 OCPN 网的可达性、活性、有界性和保守性,设计了 OCPN 网的层次型存储模型,并给出了相应的对象检索和表现算法。这一切都使得 OCPN 模型在描述多媒体同步要求方面非常有效。

然而,OCPN 模型中有几个问题没有解决,其中问题之一就是没有考虑多媒体表现时的异步用户交互问题。当允许用户自由地和表现交互时,就出现了一些新的同步要求。为此,我们对 OCPN 模型进行了扩充,以处理异步用户交互问题。

2　扩充的 OCPN 模型

为了处理用户交互,我们对 OCPN 模型进行扩充,对于网中的每个位置引入一个表示资源消耗程度的度量,并把扩充后的网称为 XOCPN 网。

定义 1　XOCPN 网是一个 7 元组:

$XOCPN=(\boldsymbol{T},\boldsymbol{P},\boldsymbol{A},D,E,R,\boldsymbol{M})$,其中

$\boldsymbol{T}=\{t_1,t_2,\cdots,t_n\}$,是变迁(transition)集合,

$\boldsymbol{P}=\{p_1,p_2,\cdots,p_m\}$,是位置(place)集合,

$\boldsymbol{A}:\{\boldsymbol{T}\times\boldsymbol{P}\}\cup\{\boldsymbol{P}\times\boldsymbol{T}\}$是个有向弧(arc)集合,

$\boldsymbol{M}:\boldsymbol{P}\rightarrow I, I=0,1,2\cdots$,是标记(marking)集合,

* 国家"863"高技术计划资助课题(No. 863-317-9604)
收稿日期:1997-10-21;收到修改稿日期:1998-03-10

$D:P\to R^+$是从位置到持续时间(duration)的映射,

$R:P\to\{r_1,r_2,\cdots,r_j\}$是从位置集合到资源(resource)集的映射,

$E:P\to R^+$,是从位置到非负实数集上的映射,表示目前资源消耗程度(extent),这里要求$P\cap T=\phi$,$P\cup T\neq\phi$。

定义 2　令$t\in T$,$p\in P$,那么我们称$I_p(t)$为t的输入位置集合,$O_p(t)$为t的输出位置集合,$I_t(p)$为p的输入变迁,$O_t(p)$为p的输出变迁,其中$I_p(t)=\{p_i|(p_i,t)\in A,i=1,2,\cdots,m\}$,$(I_t(p),p)\in A$,

$O_p(t)=\{p_i|(t,p_i)\in A,i=1,2,\cdots,m\}$ $(p,O_t(p))\in A$。

定义 3　对于$x,y\in P\cup T$,如果存在$e_1,\cdots,e_l$,使得$(x,e_1)\in A$,$(e_i,e_{i+1})\in A$,$i=1,\cdots,l-1$,$(e_l,y)\in A$,那么我们称x到y是可达的,称$\{x,e_1,\cdots,e_l,y\}$为从x到y的可达路径,记作$x\to y$。

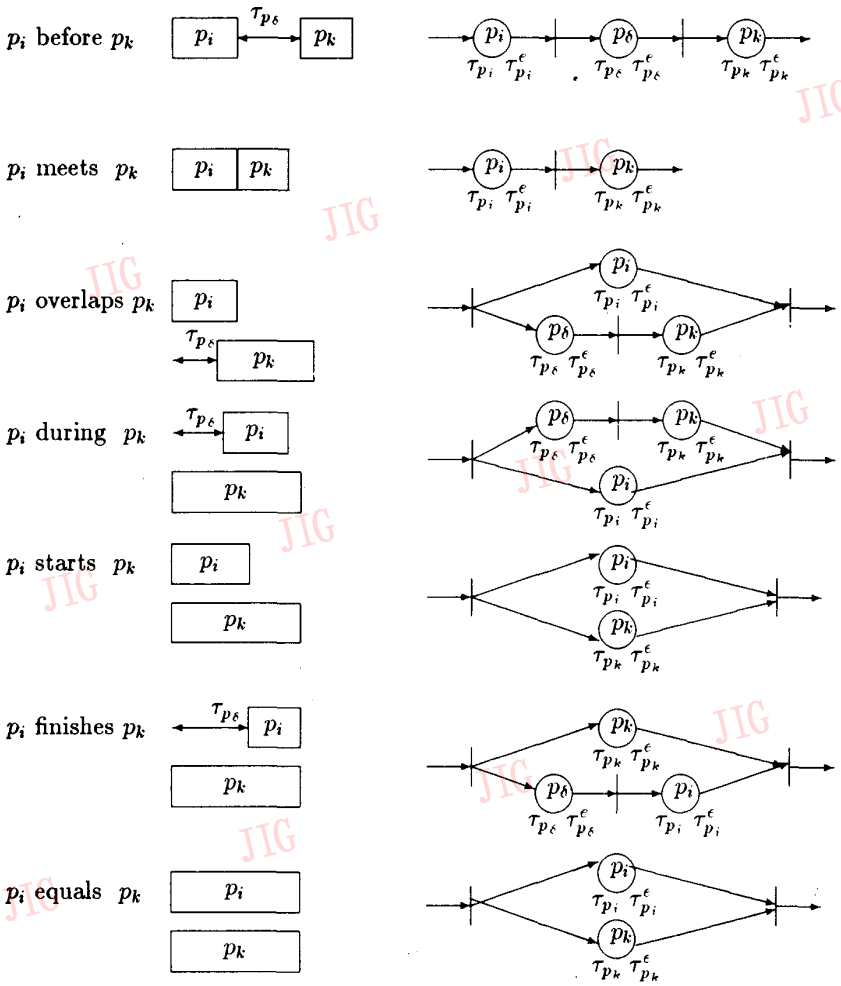
对于网中的多媒体对象,我们定义一个状态层次以描述依赖于资源消耗程度的对象状态。4 种可能的状态是空闲(Idle)、进行中(In-Process)、完成(Finished)和终止(Complete),它们定义如下:

(1)**空闲**:当$\tau^e_{p_i}=0$时,对象处于空闲状态。处于空闲状态的对象还没有开始表现,一般是因为表现还处在较前的对象。

(2)**进行中**:当$0<\tau^e_{p_i}<\tau_{p_i}$时,对象处于进行中状态。对象开始被播放,在对象$p_j$处于进行中状态期间,令牌锁定在$p_j$中。

(3)**完成**:当$\tau^e_{p_i}\geqslant\tau_{p_i}$时,对象处于完成状态。这时资源消耗已经完成,但令牌还没有取走。如果$\tau^e_{p_i}>\tau_{p_i}$,则意味着资源正等待和其它资源同步。

(4)**终止**:当$\tau_{p_i}=$null 时,对象处于终止状态。对象终止它的播放,位置上的令牌被取走,准备使能(enable)其输出变迁集,表现移出该对象。



(a)时序关系　　(b)XOCPN

图 1　时序关系和对应的 XOCPN

与 XOCPN 网定义相关联的是一个说明其语义的引发规则集。XOCPN 网的引发规则可总结为：

(1)变迁 t 立刻引发，当它的每个输入位置 p_i 都进入 **完成**状态时。

(2)变迁 t 引发时，从它的每个输入位置 $p_i(p_i \in I_p(t))$取走令牌，置 $\tau^e_{p_i}$=null(即变成**终止**状态)，同时将令牌加入到它的每个输出位置 $p_j(p_j \in O_p(t))$上。

(3)变迁 t 引发后，输出位置 p_j 进入**进行中**状态，其中令牌一直处于加锁状态，直至 p_j 以及和 p_j 具有相同的输出变迁的所有位置 p_k 都进入**完成**状态，令牌变成无锁定。

多媒体同步时序关系可通过基于时段(time interval)的时间模型来抽象和描述。两个时段间相互共有 13 种基本时序关系(见上页图 1)，可分为串行和并行两大类，它们在图 1(a)中表示出来，在这里仅看到 7 种，其余 6 种是不包括 equal 在内的其余 6 种关系的逆关系。对于逆关系可以通过 6 种关系的时间表交换而表示出来。已经证明，7 种关系足以描述基于任何时序关系的合成[7,8]。

XOCPN 能够捕捉用于说明不同对象计时和显示需求的任何时序关系，7 种时序关系在 XOCPN 中的对应表示如图 1(b)所示。

对于 XOCPN，我们可以采用类似于 OCPN 的存储机制及表现算法，在此不予详述。通过操作消耗参数 τ^e_p，可以引发变迁，向位置中存放令牌或从位置中取走令牌，资源的消耗可以被启动、停止和加速等，因此 XOCPN 形成了在表现期间支持动态用户交互的基础。XOPCN 的这种支持灵活用户交互的能力，是它超过 OCPN 的主要优点。

3 用户交互

在有用户交互的情况下，不可能总是单一固定的表现。用户可以跳到表现中的不同点，或者调整某媒体段表现速度，这就有可能影响某些对象表现持续时段，而对于同时被播放的其它对象表现持续时段没有影响。同步模型必须支持对包含用户交互一致和连贯表现的规范描述，并且必须有一种强制机制来保证当发生用户交互时维持全局状态的一致。

XOCPN 同步机制提供了修改网中每个位置的资源消耗程度的参数的能力。在不改变 Petri 网结构语义的前提下，可以改变多媒体对象的有效表现期限。由于网中变迁是瞬时的，用户交互只能在 Petri 网的执行处于位置上时才能发生。每当遇到一个特殊用户交互请求时，就暂停 Petri 网的执行并记录表现的状态(其中包括各对象的消耗参数 τ^e_p)。当恢复正常表现时，根据特殊用户交互的类型，更新对象的剩余表现持续时间。这种修改资源消耗程度的能力是支持异步用户交互的关键。

这里，我们主要讨论当用户发出暂停、恢复、快进和倒放等请求时的同步要求。

(1)暂停/恢复操作

这是最简单的特殊操作。每当发出暂停请求时，我们可以记录下表现的状态(如要检索的最后对象、当前正被表现的对象等)，停止进一步的检索。接着，当发出恢复请求时，我们记录下表现暂停的时间，并从暂停点重新启动表现。

(2)倒放操作

当用户发出倒放操作请求以观察当前已经处于**完成**或**终止**状态的对象时，就把该对象的状态变成**进行**中状态。为了保持一致性，同步机制必须传播状态变化直至达到一个全局一致的状态。为此记录下表现的当前状态，临时暂停当前表现。假设在发出倒放请求时，Petri 网的执行当前位置为 p_c，而且倒放操作仅涉及正在表现中的对象。首先，确定 p_c 以及和它具有并行关系的新的 τ^e_p，然后令 t_{p_c} 为 p_c 的输入变迁，对于 t_{p_c} 的所有输入位置，置消耗 参数为表现持续时间。对于所有和 t_{p_c} 具有并行关系(overlaps，equals，during，starts，finishes)的位置的输入变迁的输入位置的消耗参数也作同样的处理。一旦重启，各对象用新的 τ^e_p 来推进对象的消耗(表现)。

如果倒放操作涉及远位于 p_c 之后的对象，那么可以反向遍历 XOCPN 网，直至到达目标对象。在目标对象，比如说 P_T，把目标对象当做 Petri 网执行的当前位置，重复上面的步骤。然而对于 P_T 和 P_c 之间的每一个中间变迁，要重置其输入位置的 τ^e_b 为 0，表明这些位置上表示的对象仍然可以消耗。

(3)快进操作

快进操作和倒放操作类似，只是方向相反，在此不再详细讨论。

最后给出关于倒放的算法。算法中使用了下述定义：

p_c=在发出特殊交互请求时，Petri 网执行的当前位置；

p_T=表示特殊交互的目标对象的位置；

$P_p(P_i)$=和 P_i 中的对象具有并行关系的对象的位置集合；

$t_{p_i}\rightarrow t_{p_j}$=从变迁 t_{p_i} 到 t_{p_j} 的可达路径,其中 $t_{p_i}=O_t(p_i)$, $t_{p_j}=I_t(p_j)$, p_i 和 p_j 是位置。可达路径可以从 Petri 网可达树中计算;

$P(t_{p_i}\rightarrow t_{p_j})$=位于可达路径 $t_{p_i}\rightarrow t_{p_j}$ 上的位置集合;

$T(t_{p_i}\rightarrow t_{p_j})$=位于可达路径 $t_{p_i}\rightarrow t_{p_j}$ 上的变迁集合;

T_{p_i}=位置 p_i 所代表对象的表现时段上的点,表示对象的表现应该重新开始的位置。

Algorithm Fast_Reuerse

Stop Petri net execution; record state of presentation

$p_i=p_c$

if $(p_i\neq p_T)$ or $(p_T\notin P_p(P_i))$

{ while $(p_T\notin I_p(I_t(p_i)))$ and $I_p(I_t(p_i))\neq\phi)$

{$\forall P_j\in I_p(I_t(p_i))$

$\tau^e_{p_j}=0$

$p_i=p_k$ such that $p_k\in I_p(I_t(p_i))$ and $p_k\in P(t_{p_T}\rightarrow t_{p_c})$}}

if $I_p(I_t(p_i))=\phi$

{report error: beginning of presentation reached

exit }

/* target object reached */

$\forall p_m\in P_p(P_T)$ and $\forall p_x\in I_p(I_t(p_m))$ $\tau^e_{p_x}=\tau_{p_x}$

$\forall p_m\in P_p(P_T)$ $\tau^e_{p_m}=T_{pm}$

Restart Petri net execution with new settings

/* end of fast-reverse algorithm */

4 例　子

图 2 是一个题为"欧洲建筑"的多媒体表现,它描述欧洲建筑风格从哥德式到复兴时期的演变过程[9]。

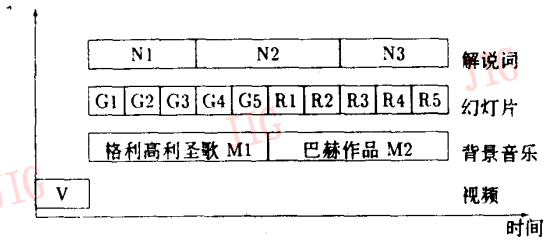


图 2　"欧洲建筑"的时间线表示

图 3 是上述例子具有初始标记的 XOCPN 表示。各对象表现持续时间没有在图中出,它们分别是 $\tau_V=120s$, $\tau_{G1}=\tau_{G2}=\tau_{G3}=\tau_{G4}=\tau_{G5}=\tau_{R1}=\tau_{R2}=\tau_{R3}=\tau_{R4}=\tau_{R5}=60s$, $\tau_{R1}=180s$, $\tau_{R2}=240s$, $\tau_{R3}=180s$, $\tau_{M1}=300s$, $\tau_{M2}=300s$,由于表现还没有开始,各对象的消耗参数 τ^e 都为 0,即所有对象都处于**空闲**状态。当开始表现时,首先引发变迁 t_1,播放视频 V,播

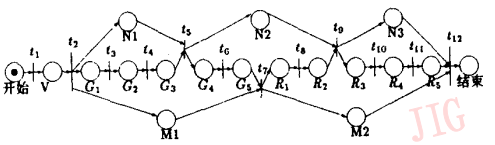


图 3　"欧洲建筑"例子的 XOCPN

放完毕后再引发 t_2,开始 N_1, G_1, M_1 的播放。图 4 中表示的是 XOCPN 执行中的一个中间状态,此时 N_1, G_1, G_2, G_3, G_4, G_5, M_1 都处于**终止**状态, R_1, M_2, N_2 处于执行中状态,且 $\tau^e_{R_1}=20s$, $\tau^e_{M_2}=20s$, $\tau^e_{R_2}=140s$,其它对象都处于**空闲**状态。

此时发生用户交互,用户请求倒放操作,希望回到幻灯片 G_4 的开始。下面我们看一看同步机制是怎样处理这一用户交互的。首先记录表现的当前状态,暂停 XOCPN 网的执行。在从当前状态 R_1 到 G_4 的可达路径上共有 3 个对象,分别为 R_1, G_5, G_4。和 R_1 具有并行关系的对象有 N_2 和 M_2,调整消耗性参数 $\tau^e_{R_1}$ 为 0s, $\tau^e_{M_2}$ 为 0s, $\tau^e_{N_2}$ 为 140−20=120s,这样 R_1 和 M_2 都变成**空闲**状态,这 2 个对象有一个共同的输入变迁 t_7, t_7 又有 2 个输入位置 M_1 和 G_5,置 $\tau^e_{M_1}$ 为 300s,置 $\tau^e_{G_5}$ 为 60s;沿着可达路径遍历,处理 G_5,和 G_5 具有并行关系的对象有 N_2 和 M_2,调整消耗性参数 $\tau^e_{G_5}$ 为 0s, $\tau^e_{M_1}$ 为 300−60=240s, $\tau^e_{N_2}$ 为 120−60=60s,这样 G_5 变成**空闲**状态, G_5 的输入变迁为 t_6, t_6 有输入位置 G_4,置 $\tau^e_{G_4}$ 为 60s;处理 G_4,和 G_4 具有并行关系的对象有 N_2 和 M_1,调整消耗性参数 $\tau^e_{G_4}$ 为 0s, $\tau^e_{M_1}$ 为 240−60=180s, $\tau^e_{N_2}$ 为 60−60=0s,这样 G_4 和 N_2 变成**空闲**状态。调整结束,调整后的状态如图 5 所示,恢复执行。

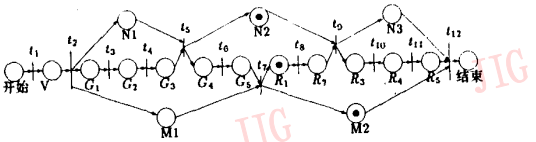


图 4　XOCPN 网执行的一个中间状态

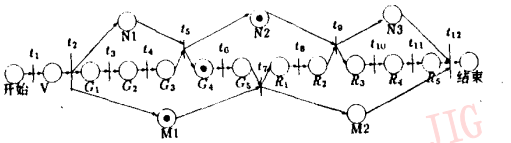


图 5　处理完用户交互后的 XOCPN 网状态

5 结束语

多媒体同步是指多媒体对象间时间关系的同步,是分布式多媒体应用的一个重要需求。多媒体与传统数据类型主要区别是引入了时间相关的连续媒体(如音频和视频)。本文提出了一个基于时间Petri网的多媒体同步模型,该模型可以很好地描述多媒体对象之间的各种时序关系,并描述了该模型对异步用户交互的支持,给出了异步用户交互算法。多媒体同步对计算机系统提出了一系列新的研究课题。

参考文献

1 Steinmetz R,Nahrstedt K. Multimedia-Computing ,communications and Applications. New York: Prentice Hall International, 1995,853.

2 Donald A Adjeroh , Lee . Synchronization and UserInteraction in Distributed Multimedia Presentation Systems. in Multimedia Database Systems, 1996:252~277.

3 Gerold Blakowski,Ralf Steinmtez. A media synchronization survey: Reference model, specification, and case studies. IEEE Journal on selected areas in communic ation,1996,14(1).

4 Little T D C , Ghafoor A. Synchronization and Storage Models-for Multimedia Objects. IEEE Journal on Selected Areas on Communications,1990,8(3):413~427.

5 Peterson J L 著. Petri 网理论与系统模拟. 吴哲辉译. 徐州:中国矿业大学出版社,1989.

6 Little T D C, Ghafoor A. Multimedia Synchronization Protocolsfor Broadland Integrated Services. IEEE Journal on Selected Areas inCommunications, 1991, 9:1368~1382.

7 Allen J F. Maintaining knowledge about temporal intervals. Communicationsof the ACM, 1983, 26(11):832~843.

8 Coolahan J E. Timing requirement for time-driven systems usingaugmented Petri nets. IEEE Transactions on Information Theory, 1983, SE-9:603~616.

9 Schnepf J,et al. Doing FLIPS: Flexible Interactive Presentation Synchronization . IEEE Transactions JSAC ,1996,1:114~125.

梁永全 讲师,博士生,研究方向为人工智能,现在的主要研究兴趣是分布式人工智能、多媒体数据库和多媒体信息检索。曾参与国家"863"课题和国家自然科学基金课题的科研工作。

史忠植 研究员,博士生导师,曾发表200多篇论文,主持多项国家自然科学基金项目和"863"项目。现在的主要研究兴趣是人工智能、机器学习、神经计算、多媒体数据库和智能决策系统。

XOCPN:A Multimedia Synchronization Model Supporting User Interaction

Liang Yongquan Shi Zhongzhi

(*Department of Computer,Institute of Shandong mining industry,Taian* 271019) (*ICT. Chinese Academy of Sciences,Beijing* 100080)

Abstract Multimedia synchronization is a significant requirement for distributed multimedia applications. In this paper,we augment the OCPN model,and present a new multimedia synchronization model XOCPN based on timed Petri nets . Using this model we can give the formal specification for multimedia object compositions with time-constrained relations and deal with flexible user interaction. Algorithms for asynchronous user interactions are also presented, and explained by an example.

Keywords Timed Petri nets,Multimedia synchronization,Asynchronization user interaction