

基于同步时间 Petri 网的细粒度多媒体同步模型研究^{*}

张广胜 吴哲辉 翟正利 贾广余

(山东科技大学信息科学与工程学院 泰安271019)

摘要 多媒体同步是分布式多媒体信息系统的关键问题。在分析和比较几种已有的典型的多媒体同步模型的基础上,本文提出了一种新的模型:同步时间 Petri 网(简记为 STPN)。STPN 不仅能完整、准确地对媒体流内和媒体流间的时间层次行为进行形式化描述,而且解决了时间流 Petri 网中存在的死托肯问题。特别地,通过引入抑弧,STPN 有效地克服了随机延迟、阻塞、丢包等因素的不良影响。尤其当丢失率较高时,STPN 更能体现其优越性和鲁棒性。

关键词 多媒体同步,抖动,扭曲,服务质量,同步时间 Petri 网

Fine-Grained Distributed Multimedia Synchronization Model Based on Synchronization Time Petri Nets

ZHANG Guang-Sheng WU Zhe-Hui ZHAI Zheng-Li JIA Guang-Yu

(College of Information Science and Engineering, Shandong University of Science and Technology, Taian 271019)

Abstract Multimedia synchronization is the key of distributed multimedia information systems. Based on the analysis and comparison of several typical multimedia synchronization models, this paper presents a new model: Synchronization Time Petri Nets. STPN not only allows the temporal behavior of both intra-media and inter-media to be described fully, accurately, and formally but also solves the problem that time stream Petri net has dead tokens. In particular, STPN introduces inhibitor arcs to handle the problem of synchronization failure because of fire interruption caused by random delay, blocking, or lost packets. Furthermore, STPN is robust for loss of objects in some extent and can effectively avoid long time disparity of media synchronization, especially, in case of high loss rate of media objects.

Keywords Multimedia synchronization, Jitter, Skew, Quality of service, Synchronization time Petri net

1 引言

随着分布式多媒体技术的发展,多媒体同步成为分布式多媒体应用的一个重要需求。计算机支持的协同工作(CSCW)、实时视频会议系统、分布式多媒体信息处理、在线点播、远程教学等多媒体应用不仅要求实时传输数据量很大的视频、声音、图像、文本等媒体信息,而且还要求各种媒体之间严格同步,即经过网络传输以后,单个媒体流内部以及多个媒体流之间的时间关系应保持不变。多媒体是各种数据流的集成,这些数据流包括时间依赖的媒体类型(如音频、视频、动画等)及非时间依赖的媒体类型(如静态文本、图像、图形等)。媒体流由一系列的信息单元组成,这种信息单元称为逻辑数据单元 LDU(Logical Data Unit)。同步可以分为两大类:媒体内同步,主要指维持一个媒体对象内部各 LDU 单元之间的时间同步关系,避免由于抖动引起的各 LDU 间时间关系的破坏而造成媒体流的不连续。例如,对于一个每秒播放25帧的视频序列而言,每帧应该播放40ms,不能维持这种时间关系,观众就会明显感到画面的停顿和不连续。确保媒体内同步是多媒体同步的首要任务。媒体间同步主要是维持一组相关媒体对象 LDU 单元之间的时间关系。它又可以分为粗粒度同步和细粒度同步。如带有背景音乐的画面之间的同步就属于粗粒度同步,而唇同步就属于细粒度同步。细粒度同步应当在

LDU 层上就能保证用户给定的服务质量 QoS(例如最大的抖动和扭曲)。

关于多媒体同步的研究分为两个问题:如何表示多媒体对象之间的时态关系—同步模型问题;如何保证时态关系的正确执行同步机制问题。对于多媒体同步模型, Petri 网是一种比较合适的表述工具。已提出的具有代表性的多媒体 Petri 网模型主要有:

(1)OCPN(Object Composition Petri Nets)^[1]和 XOCPN(Extended Object Composition Petri Nets)^[2]。前者是一种粗粒度的同步规范,将媒体类型映射到 Petri 网中的每个库所,并为每个库所分配了时间间隔和令牌数目,变迁表示同步点。OCPN 特别适合于存储多媒体应用中描述多媒体对象的时态关系。XOCPN 是在 OCPN 上的扩展,是一个细粒度的同步规范,适合于描述表现(或传输)多媒体应用中信息的时态关系。

上述两种模型都未充分考虑多媒体播放中可容许的抖动和扭曲,是一种理想状态下的同步模型,而且提供的控制信息和扩展能力都十分有限,因而,不适合于多媒体弱同步系统^[3]。

(2)TSPN(Time Stream Petri Nets)^[4]。是对分布式多媒体系统形式化建模的一个强有力的工具,它既能处理多媒体需求中的细粒度同步要求如抖动和扭曲,又能保证表现的连续性,并且提供了多种同步手段。然而,我们不容易用 TSPN

^{*}基金项目:国家自然科学基金资助课题(编号:60173053)。张广胜 硕士研究生,主要研究领域为 Petri 网理论及应用。吴哲辉 教授,博士生导师,主要研究领域为 Petri 网理论及应用,算法设计与分析。翟正利 硕士研究生,主要研究领域为 Petri 网理论及应用。

硕士研究生,主要研究领域为 Petri 网理论及应用。贾广余 硕士研究生,

来模拟受限阻塞(restricted blocking)问题^[5], TSPN 也不能有效解决由于网络严重延迟、阻塞或者丢包而导致的同步失败问题^[6]。同时, TSPN 也存在由于主弧引发, 而其他流并未引发而产生的死托肯问题。

(3) FTPN(Fuzzy-timing Petri Nets)^[7,8]。为了解决时间的不确定性, Zhou and Murata (1998) 提出 FTNMS(Fuzzy-timing Petri Nets for Multimedia Synchronization)^[7]。它是一个细粒度的时间模型, 能支持模糊剧情并且容易验证满足 OoS 条件的可能性, 但是该模型同样存在如文[6]指出的 TSPN 的同步失败问题。EFTPN(Enhanced Fuzzy-timing Petri Nets)^[8]也缺乏适应网络变化的能力。

(4) DTPN(Dynamic Timed Petri Nets)^[9]。其最大优点是能处理用户的交互操作, 并且不受同步粒度级别的限制。其缺点是没有考虑分步环境中的抖动。

本文提出一个新的多媒体同步模型: STPN(Synchronization Time Petri Nets)。该模型可以解决媒体的表现(或传输)的不确定性, 同时解决了时间流 Petri 网中存在死托肯的问题, 而且比时间流 Petri 网有更灵活的表述能力。通过引入抑弧可以有效地克服随机延迟、阻塞和丢包等因素的不良影响, 尤其当对象丢失率较高时(如面向 Internet 的实时多媒体通信等), STPN 更能体现出它的优越性和鲁棒性。

2 同步时间 Petri 网模型—STPN

定义 2.1 STPN = (P, T, F, A, M₀, SI, ST, SU, D), 其中:

P 是一个位置集合;

T 为变迁集合;

$F \subseteq (P \times T) \cup (T \times P)$ 为一个有向弧集合;

$A \subseteq P \times T$ 为抑弧集合, 抑弧用 \dashv 来表示, 是为了解决异常处理而引入的;

$M: P \rightarrow I$, 是网系统的标识, $I = \{0, 1, 2, \dots\}$, M_0 是初始标识;

SI: 是时间映射函数, $SI: T \rightarrow Q^+$ (Q^+ 为非负有理数);

ST: 是实时时间戳集合(Real-time Stamp), 每个库所都带有一个实时时间戳, $\pi_i^m \in ST$, (i 表示与库所 p_m 相对应的媒体对象如音频、视频、图像、文本等的序号) 表明在时刻 π 一个托肯到达该库所 p_m 。初始状态时 π_i^m 为 0, 当网系统运行时, 库所的实时时间戳 π_i^m 可以被实时更新为新的值;

SU: 是时间映射函数。定义如下: 令 $B = \{b = (p, t) \mid (p, t) \in F \wedge (p, t) \in A\}$, 则 $SU: B \rightarrow Q^+$, $SU(b_i) = u_i$, u_i 表示多媒体对象(或 LDU) i 的正常表现时间;

D: 是时间映射函数。定义如下: 令 $C = \{c = (t, p) \mid (t, p) \in F\}$, 则 $D: C \rightarrow Q^+$, $D(c_i) = d_i$, d_i 表示媒体对象 i 的准备时间或者是 LDU 之间插入的时间间隔。

对于 STPN, 记 $X = P \cup T$, 对 $x \in X$, $x = \{y \mid (y, x) \in F \wedge (y, x) \in A\}$ 称为 x 的前集(pre-set)。

定义 2.2 STPN 的形式语义和引发规则

(1) 当一个托肯到达库所 p_m 时, 托肯立即处于加锁状态, 同时与该库所 p_m 相对应的多媒体对象 i 开始表现, 表现 u_i 段时间后托肯解锁。

(2) 当每个前集库所中的托肯都处于解锁状态时, 变迁 t 立即引发。

(3) 当任意 $t_k \in T$ 发生时, 从它前集中的每个输入库所吸收一个托肯, t_k 变为进行态, 并在进行态持续 $SI(t_k)$ 时间后, 变迁 t_k 发生结束。引发变迁 t_k 所产生的托肯经过 d_i 段延迟时间后送入每个相应的输出库所 p_n 。

(4) 此时库所 p_n 的实时时间戳为 $\pi_i^n = \pi_i^m + u_i + SI(t_k) + d_i$ 。假如托肯不能在时刻 π_i^n 到达 p_n , 则异常发生, 抑弧的输出变迁被条件: 现在时刻(current time point) $\tau > \pi_i^n$ 引发而进入异常处理状态, 异常处理完后继续表现。

(5) 媒体对象表现完毕, 置实时时间戳为空。

为了实现多媒体同步的需要, 文[4]中, 对 TSPN 提出了三种基本引发规则: strong-or 规则, weak-and 规则, master 规则。并通过这三种规则进而组合成九种引发策略。这就使得 TSPN 有效的描述媒体内部和媒体间的同步关系, 并允许在多媒体系统内存可容许的时间抖动, 这大大的加强了它的模拟能力。我们完全可以用定义 2.2 提出的 STPN 的形式语义和引发规则来实现 TSPN 的多种变迁引发策略。这表明 STPN 只用一种引发规则就实现了 TSPN 的模拟能力, 而且 STPN 模型解决了 TSPN 中存在死托肯的问题, 且有更灵活的表述能力。

在 STPN 中, 当一个变迁 t_k 的每个前集库所包含一个没有锁定的托肯时, 变迁 t_k 持续 $SI(t_k)$ 后立即引发, 这样相当于变迁引发就是被最迟到来的托肯引发的, 即 TSPN 中的 weak-end 规则与 STPN(当 $SI(t_k) = 0$) 的引发规则相同。

下面我们首先用 STPN 来模拟 TSPN 的 strong-or 与 master 规则。需要指出的是, STPN 模型解决了 TSPN 中的死托肯问题, 且具有更灵活的表述能力。

① 模拟 master 变迁引发规则

在多个媒体流的同步要求中, 有时候其中某个媒体(称为主媒体)流的进程起到决定性的作用。譬如, 在音频流和视频流同时播放的过程, 音频流为主要媒体流。因为视频流跳过少量几帧, 对媒体的表现效果不会产生太大的影响, 所以一般的要求一旦音频流播放完毕, 视频流则停止播放。图 1(a) 为 TSPN 给出的主弧策略, 这虽然可以反映出主要媒体流的决定性作用, 但在 V_{n-1}, V_{n-2}, \dots 等库所中可能有残留托肯, 对系统的持续运行将产生影响。图 1(b) 为 STPN 来模拟图 1(a)。从图 1(b) 可以看出 STPN 在模拟图 1(a) 时有很大的灵活性。我们可以设定 $SI(t_k) = t_k$ 的值为主媒体流播放完毕等待的上限, 一旦 t_k 时间到, 其它流的播放立即停止。

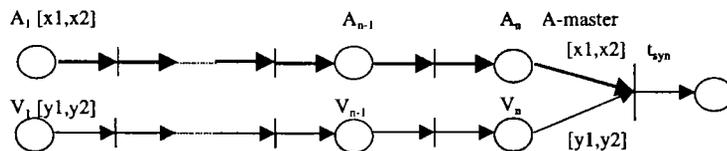


图 1(a) 带有主弧引发规则的 TSPN

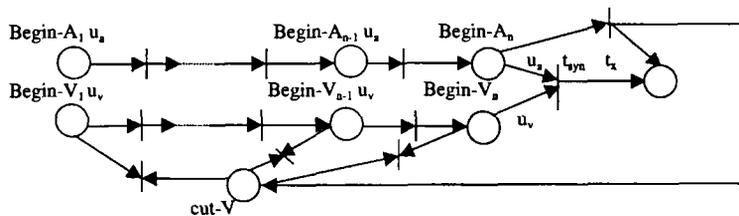


图1(b) STPN 模拟主弧引发规则

② 模拟 Strong-or 引发规则

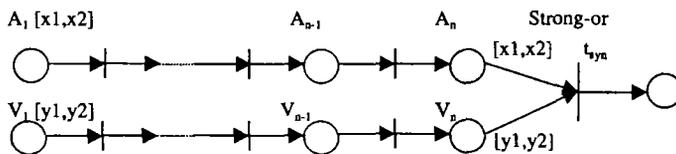


图2(a) 带有 Strong-or 引发规则的 TSPN

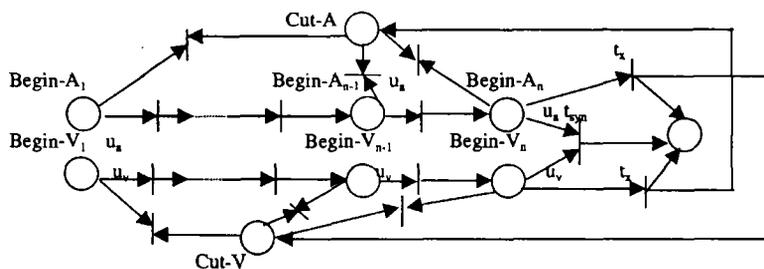


图2(b) STPN 模拟 Strong-or 引发规则

图2表明:如果 video 或 audio 有一个播放完毕,后续媒体(如文本)将开始播放。图2(a)为 TSPN 给出的 Strong-or 规则,但是同样存在残留托肯的问题。图2(b)为 STPN 模拟 Strong-or 引发规则,可以看出通过指定最先完成的媒体流等待时间的上限的值 $SI(t_x) = t_x$,一旦超过此值,另一个媒体流将被迫终止播放。可以看出,STPN 有比 TSPN 更灵活的模拟能力,且不存在残留托肯问题。

③ 表示基本的时间关系

多媒体同步时序关系是通过时间模型来抽象和描述的,根据采用的基本时间单位,当前用于描述时间的概念模型主要有两类:第一类是基于时刻(time instant)的时间模型,第二类是基于时段(time interval)的时间模型。由于每个多媒体对象都有表现持续时间,所以基于时段的模型更适合于描述相对的时序关系,并且比基于时刻的模型具有更强的描述能力。当前都采用该模型作为多媒体同步关系的时间概念模型,时段表示多媒体对象的持续时间,它们的相互位置表示媒体对象的时间相关性即同步关系。Aellen^[10]的7个基本时序关系足以描述任何时序关系的合成如图3左边部分。STPN 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如图3右边部分。虽然这里描述两个对象之间的时序关系,但可以证明这些关系和 STPN 模型足以说明由成对的相关对象组成的复杂多媒体时序关系,这是因为多个对象之间的时序关系可以用两两之间的时序关系逐级描述出来。

对图3中的 $x \text{ equal } y$,如果 x 与 y 中没有主媒体对象,则 t_x 为最先完成的那个媒体对象的最大等待时间,超出这个时间,落后的媒体流将被迫终止表现。如果 x 为主媒体对象, X 一旦表现完毕并且等待了 t_x 时间,不管 y 是否表现完毕都将

终止其表现。 $X \text{ finish } Y$ 的情况与 $x \text{ equal } y$ 类似。对 $x \text{ overlap } y$ 和 $x \text{ during } y$,通常 y 为主媒体对象,一旦 y 表现完毕并且等待了 t_x 时间,不管 x 是否表现完毕都将终止其表现。

3 同步问题讨论

3.1 多媒体流内同步

对于流内同步的情况,由于时间相隔较远的媒体单元的时间关系较弱,我们将主要注意力集中在相邻媒体 LDU 上,保证相邻媒体 LDU 之间的时间抖动在 QoS 规定的最大抖动范围内即可同步。许多方面影响抖动如客户端缓冲区的大小,网络带宽等等,假设有足够的缓冲区容量,我们发现有两个主要因素影响抖动^[6]:第一,媒体对象表现(或传输或播放)期间的偏移量 ϵ ,第二,LDU 的准备时间(或插入的时间间隔)。采用类似 TSPN 的方法,假设每个 LDU 的正常播放时间为 u_i ,实际播放过程中, ϵ 为 LDU 偏离正常表现的时间(延迟为负数)。由于我们引入了 RTS,因此,该媒体流内表现过程中的最大抖动可以计算:

设 j 为媒体对象 i 表现过程中第 m 个 LDU 与第 n 个 LDU 之间的最大抖动, m 和 n 之间插入 d_i 段准备时间, $SI(t_x) = 0$,则

$$\pi_i^n = \pi_i^m + (u_i + \epsilon) + d_i \tag{1}$$

$$j = \pi_i^n - (\pi_i^m + u_i - \epsilon) = d_i + 2\epsilon \tag{2}$$

设用户给定的 QoS 中最大抖动为 η ,那么必有 $j \leq \eta$ 。

当 $j \geq \eta$ 时,即媒体对象的第 m 个 LDU 受到较大的延时,超出规定的时间范围,则当引入抑止弧来进行如下文所述的异常处理。

3.2 多媒体流间同步

媒体间同步主要是维持一组相关的媒体对象 LDU 之间的时间关系,本文主要研究细粒度的同步问题。给定的 QoS 中的最大扭曲通常是用户可接受的媒体流间的最大的扭曲值,因此,我们必须指定适当的同步粒度。事实上,扭曲主要是由于流内的抖动而影响的^[8]。设两个媒体流(如音频视频)为 o₁,o₂从第 1 个 LDU 单元开始同步,它们的同步粒度为 n,那么,二者的最大扭曲 Sⁿ_{o₁,o₂}可如下计算:

$$\text{令 } s_{o_1} = \sum_{k=1}^{l+n} j_{o_1}^k + nu_{o_1} \quad (3)$$

$$s_{o_2} = \sum_{k=1}^{l+n} j_{o_2}^k + nu_{o_2} \quad (4)$$

$$\text{则: } S_{o_1,o_2}^n = |s_{o_1} - s_{o_2}| \quad (5)$$

其中,j^k表示媒体流 o₁的第 k 个 LDU 单元与第 k+1 个 LDU 单元之间的最大抖动,其余的类似。为了讨论方便,假设 s_{o₁} ≥ s_{o₂},那么由(2),(3),(4),(5)得:

$$S_{o_1,o_2}^n = 2n(\epsilon_{o_1} - \epsilon_{o_2}) + n(u_{o_1} - u_{o_2}) + \left(\sum_{k=1}^{l+n} d_{o_1}^k - \sum_{k=1}^{l+n} d_{o_2}^k \right) \quad (6)$$

设给定的 QoS 中最大扭曲为 γ,则必有 Sⁿ_{o₁,o₂} ≤ γ。当 n=1 时,我们得到最小的同步粒度,若 S¹_{o₁,o₂} ≤ γ,称为固有同步。如果给定的 QoS 中最大扭曲调整小了,我们也可以通过减小 n 来适应新的最大扭曲值 γ,可见该模型有很强的动态适应性,这一点对于设计多媒体软件是非常重要的。

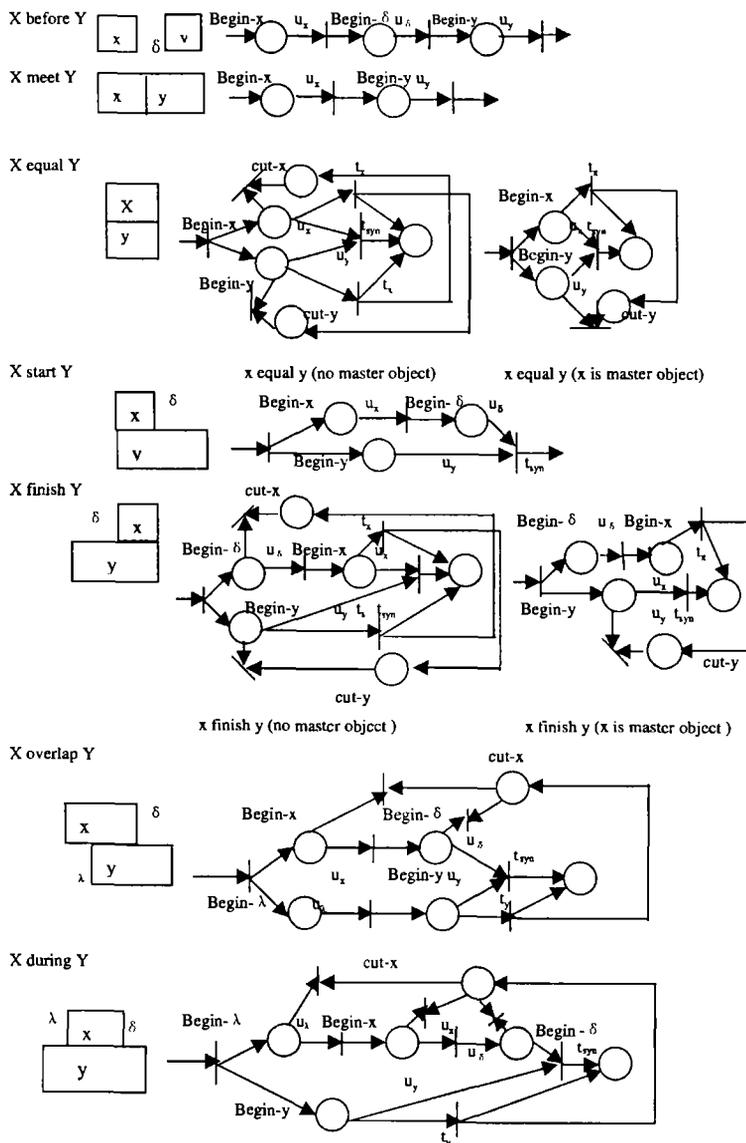


图3 STPN 表示基本的时间关系

4. 实例分析

音频、视频的唇同步问题如图4,图5。音频、视频流每秒表现10帧,QoS 中最大扭曲为80ms,最大抖动为10ms。

由于音频、视频二者有相同的正常表现时间,此时表现是的最大扭曲如下计算:

对于(6)式,如果 0 ≤ ε_{o₁} - ε_{o₂} ≤ min{d_{o₂}^{l+1}, d_{o₂}^{l+2}, ..., d_{o₂}^{l+n}} 则有:

$$S_{o_1,o_2}^n \leq n\gamma \quad (7)$$

满足同步要求;如果 0 ≤ ε_{o₁} - ε_{o₂} ≥ max{d_{o₂}^{l+1}, d_{o₂}^{l+2}, ..., d_{o₂}^{l+n}} 那么 s_{o₁} - s_{o₂} 的上限可以如下得到:令 ε_{o₂} = 0, d_{o₂}^{l+1}, d_{o₂}^{l+2}, ..., d_{o₂}^{l+n} 均为 0, 所以有:

$$s_{o_1} - s_{o_2} \leq 1.5n\gamma \quad (8)$$

对于图4,当 n=4 时由(7)得 4 * 10 < 80,同步粒度可接受。由(8)得 1.5 * 4 * 10 < 80,同步粒度可以接受。由(8)得 1.5 * 2

(下转第108页)

(否则它将置 valid 为 0 然后返回)。

参考文献

- 1 Bonacina M P. On the reconstruction of proofs in distributed theorem proving: a modified Clause-Diffusion method. *Journal of Symbolic Computation*, 1996, 21: 4~6, 507~522
- 2 Bonacina M P, Hsiang J. The Clause-Diffusion methodology for distributed deduction. *Fundamenta Informaticae*, 1995, 24: 177~207
- 3 Bonacina M P, Hsiang J. Parallelization of deduction strategies: an

- analytical study. *Journal of Automated Reasoning*, 1994, 13: 1~33
- 4 胡春华, 吴波, 杨叔子. 基于多自主体的分布式智能控制系统研究. *中国机械工程*, 1999, 9(7)
- 5 王万森. 人工智能原理及其应用. 电子工业出版社, 2000
- 6 杨炳儒. 知识工程与知识发现. 冶金工业出版社, 1998
- 7 <http://www.tcd.ie/Psychology/Ruth-Byrne/mental-models/critics.html>
- 8 <http://www.prdm.net/papers/knowledge/KM-20for-20CE.htm>
- 9 <http://www.cs.cf.ac.uk/Dave/AI2/node102.html>

(上接第 81 页)

* $10 < 50$, 同步粒度也可以接受。

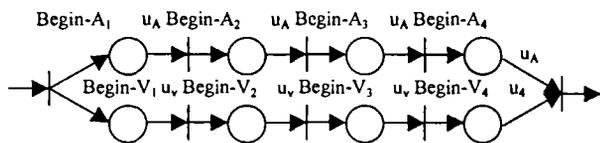


图 4 当 $n=4$ 时的唇同步

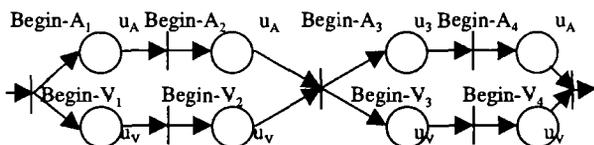


图 5 当 $n=2$ 时的唇同步

5 异常处理

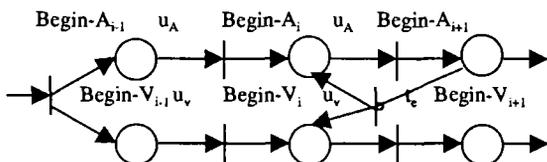


图 6 STPN 的异常处理

为了解决对象间因同步等待引起的“时间狭缝”(gap problem)^[5], “restricted blocking”提供了一种解决的办法。通常该方法被用在同步点时刻, 实际上最终难以接受的扭曲主要是流内的偏移量超出 QoS 给定的最大抖动而导致的, 因此我们可借鉴文[8]思想来解决此类问题, 但需要说明的是, 我们的模型, 对音频和视频采取了不同的处理方法, 如图 6 上边的媒体流为音频。

在实时多媒体通信中, 多媒体数据通常采用非可靠传输协议(如 UDP, RTP 等)进行传输的, 数据包的丢失是不可避免, 因此, 有必要研究同步模型对象 LDU 丢失对同步的影响, 可现有的 Petri 网同步模型均没有考虑对象丢失的情况。当 $j \geq \eta$ 时, 即媒体对象的第 m 个 LDU 受到较大的延时, 超出规定的时间范围, 通常在要求不高的时候, 原则上应该丢弃, 但本模型对音频、视频采用了不同的处理方法。由于人的听觉对音频变化比较敏感, 因而不能随意丢弃音频流的 LDU, 可以引入抑止弧来进行如图 6 所述的异常处理。对于视频媒体流来说, 丢失有限几个的媒体单元不会对用户带来明显的影响, 少量的可直接丢弃。由于 RTS 和表现时间 u_i 已知, 因而当

LDU_k (k 为自然数, 指第 k 个 LDU) 丢失时, 只要在库所 p_k 插入相同数目与 LDU_k 有相同时间属性的延迟节点, 就不会影响后续媒体的同步。可见该模型对于一定程度的对象丢失具有鲁棒性。

结论 本文针对时间流 Petri 网的不足, 引入了 STPN 模型, 有效地解决了多媒体通信、播放的同步问题。它是细粒度模型, 支持并保证服务质量, 并根据不同的 QoS 要求, 提供不同的同步响应, 具有很强的适应性。另外, 该模型对于一定程度的对象丢失具有鲁棒性, 不会影响到后续媒体的同步。

今后, 我们需要进行仿真试验, 验证 STPN 模型用于面向 Internet 的实时多媒体通信时的实际特性, 需要进一步研究对该模型的性质分析(如可达性、活性、公平性^[11]), 以及对该模型的多媒体同步机制问题等。

参考文献

- 1 Little T D C, Ghafoor A. Synchronization and Storage Models for Multimedia Objects. *IEEE Journal on Selected Area in Communications*, 1990, 8(3): 413~427
- 2 Woo M, Quzi N U, Ghafoor A. A Synchronization Framework for Communication of Pre-orchestrated Multimedia Information. *IEEE Network*, 1994, 8(1): 52~61
- 3 Tan Kun, Shi Yuan-chun, Xu Guang-you. Supporting weak synchronization over world wide Web. *Journal of Software*, 2000, 11(7): 853~862
- 4 Senac P, Diaz M. Modelling logical and temporal synchronization in hypermedia systems [J]. *IEEE Journal on Selected Area in Communications*, 1996, 14(1): 257~273
- 5 Steinmetz R. Synchronization properties in multimedia systems. *IEEE Journal on Selected Area in Communications*, 1990(April): 401~412
- 6 单志广, 杨扬, 王任, 于兴业. 扩展的时间流 Petri 网多媒体模型. *计算机研究与发展*, 2000, 37(2): 223~273
- 7 Zhou Y, Murata T. Fuzzy-timing Petri net model for distributed multimedia synchronization. In: *Proc. of the 1998 IEEE Int. Conf. on Systems, Man and Cybernetics*, Lolla, California, 244~249
- 8 Han Ying-jie, Sun Yong-qiang, Wu Zhe-hui. Fine-grained distributed multimedia synchronization model-Enhanced Fuzzy-timing Petri net. *Journal of Shanghai Jiaotong University*, 2001, E-6(1): 62~66
- 9 Prabhakaram B, Raghavan S V. Synchronization models for multimedia presentation with user participation. *Multimedia Systems*, 1994, 2(2): 53~62
- 10 Allen J F. Maintaining knowledge about temporal intervals. *Communication of the ACM*, 1983, 26(11): 832~843
- 11 吴哲辉. 有界 Petri 网的活性和公平性的分析与实现. *计算机学报*, 1989, 12(4): 267~278