

# 媒体同步技术\*

肖勃雷 李新友 王 诚 陆玉昌

(清华大学计算机系 北京 100084)

**摘要** 人们可以用多种媒体形式表达同一段内容,这样,在输出这些内容的时候就会碰到媒体同步的问题.本文就多种媒体输出同一内容但有多种表达形式时的媒体同步问题进行了一些探讨,提出了一种媒体同步输出的控制模型,并且根据此模型给出了具体工作中媒体同步控制的实现.

**关键词** 媒体同步,同步控制.

多媒体技术越来越多地用于信息表达及输出,因为人可以用不同的方式接收外界事物的信息.例如,人可以通过视觉、听觉、触觉等接受不同类型的信息,即使在用一种感觉器官下,例如视觉,同样的信息内容也可以以图形文字等不同方式反映到人眼,如果信息被以多种媒体同时运行的方式反映到人脑,也就大大地提高了信息的可接受性和易懂性.

在多种媒体形式共同输出至人的各种感觉器官的时候,我们就会碰到媒体同步(Media Synchronization)<sup>[1,2]</sup>的问题.最简单的同步,如电影放映中要求画面与相应人物的配音同步进行,不能到了下一个场面时放的仍然是前一个场面的配音.复杂的同步,如虚拟现实中的飞行模拟,飞机被击中时的爆炸声,飞行员座椅的震动,眼前景象的变动,都要求输出上的同步.<sup>[3]</sup>媒体同步问题实际上是在多种媒体的操作之间有相互时间关系要求时产生的一种媒体时序控制的问题.这里的媒体操作含义很广,例如多个用户在同一份多媒体文档上工作时,每个用户对文档的修改等操作都可视为媒体操作.这些媒体操作之间也有同步的问题.例如当一个客户在修改文档中某个对象时应该禁止其他客户对此对象的操作.这类同步我们称为事件同步(Event-Driven Synchronization).<sup>[4]</sup>此文只考虑媒体输出这种操作.通常在信息输出采用多媒体形式时,就存在媒体输出同步的问题,因而必须解决各种媒体输出时的时序同步问题.也就是说,反映同一内容的用各种不同媒体方式表达的输出在时间先后与相互配合上,应该是相关的.这种同步属于媒体同步中的连续同步(Continuous Synchronization).<sup>[5]</sup>

本文只考虑媒体输出中的这种同步控制问题.

\* 作者肖勃雷,1972年生,博士生,主要研究领域为计算机应用.李新友,1961年生,副教授,主要研究领域为计算机图形学.王诚,1940年生,副教授,主要研究领域为计算机结构及计算机应用.陆玉昌,1937年生,教授,主要研究领域为知识工程与机器学习.

本文通讯联系人:肖勃雷,北京 100084,清华大学计算机系

本文 1996-01-30 收到修改稿

# 1 问题提出

在一个预测分析系统运行后,生成一些市场预测、分析及于当前市场形势的结果,这些结果可以以不同形式反映到接收者头脑中.例如这样一条信息:“云南香烟销售量为 16 万箱”,这条信息可以用语音说出来,也可以在云南省地图上用柱图表示出来,也可以填在香烟销售情况的一张表格中表示出来.

当我们预测分析的结果用多种输出形式表达时,就得到了一份多媒体形式(语音、柱图、表格)的报告.

我们目前系统的信息表达形式有下述几种:语音、图表、卡片、可视化图形、统计图形.当这份报告输出时,各种信息必须同步地反映给接收者,当语音读到“云南香烟销售量 16 万箱”时,地图上的中心点移到云南省且在云南省地图上出现一个表示 16 万箱的柱图,屏幕上的卡片显示云南香烟销售量的表格.

对这样多媒体的输出我们要解决 2 个问题,①如何描述这份报告,②输出时如何控制其媒体同步.

# 2 媒体同步模型

## 2.1 报告描述

本节我们先解决报告描述的问题.每种媒体信息都看成一个数据流,在报告中我们以函数形式描述运行的时序关系.有 5 种信息表达形式:语音、图表、卡片、可视化图形、统计图形,分别用  $A, T, C, V, S$  表示,那么在报告中描述的便是

$$\begin{cases} A=A(t) \\ T=T(t) \\ C=C(t) \\ V=V(t) \\ S=S(t) \end{cases} \quad \text{其中 } t \text{ 表示时间}$$

这是一个以时间为自变量的参数方程, $x(t)$ 表示在  $t$  时刻媒体信息应播放到的位置.具体的报告表达中,因为计算机内用的是离散的数据表达形式,我们描述的也就是一张媒体输出的时序表,用其表示各个离散的时刻及每个媒体应播放到的位置.

## 2.2 媒体输出模型

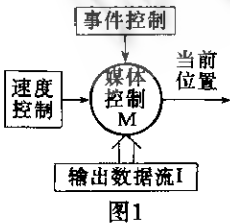


图1

对于每种媒体输出形式,可建立如图 1 模型.这是一个抽象的媒体输出模型,基本上适用于上面提到的 5 种媒体输出形式.媒体  $M$  是一个可以控制的输出单元,对它可以控制输出的速度  $v$ ,其中  $v$  是一个时间的函数,表示媒体的播放速度, $v=v(t)$ .

此单元还响应一些特殊事件,例如暂停输出、恢复输出、停止输出这些事件.输出的数据是外部提供的数据流  $I$ .

另外,此单元产生输出  $M$  表示当前媒体输出进行到的位置.一般来说, $M$  是输入数据流  $I$ ,播放速度  $v$  和时间  $t$  的函数, $M=M(I, v, t)$ .

我们用上述模型描述语言媒体输出,  $I$  为字符流,  $v$  为语音播放速度, 如果假定语音实际的播放速度与内容无关, 而只与  $v$  有关, 那么可以得到输出位置函数  $O(t)$ :

$$O(t) = \int_0^t v(x) dx$$

就是说在时刻  $t$ , 语音播放到的位置是  $O(t) = \int_0^t v(x) dx$ .

此模型也是较符合实际情况的, 例如在语音中, 我们通过控制播放速度来使预期说话的位置与实际位置一致. 而且, 一般我们对每种媒体只控制其速度, 而不是通过输出内容的增减来达到同步. 针对上述模型, 定义统一的媒体输出对象如下:

*MediaObject*

```

{
    DataBlock I;           //输入数据流 I
    SetPlayRate(int v);   //用于设置媒体播放速度
    GetPosition();        //取得当前媒体运行到的位置
    EventControl(Event e); //响应上层事件
};
  
```

上述定义中, 标出了媒体对象的输入、它应该提供的状态查询操作 *Get Position* 以及应该有的控制操作: 设置播放速度 (*Set Play Rate*) 和事件控制 (*Event Control*).

2.3 同步控制模型

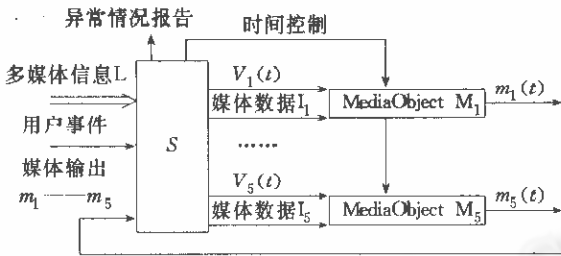


图2



图3

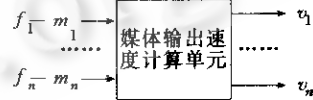


图4

图2中  $S$  为媒体同步控制单元, 它的核心工作内容是从多媒体输出信息  $L$  中分离出各媒体的预期输出函数, 经过运算产生媒体输出速度, 内部结构如图3.

计算关系如下:

$$\begin{cases}
 v_1 = v_1(f_1, f_2, \dots, f_n, m_1, m_2, \dots, m_n, t) \\
 v_2 = v_2(f_1, f_2, \dots, f_n, m_1, m_2, \dots, m_n, t) \\
 \dots \\
 v_n = v_n(f_1, f_2, \dots, f_n, m_1, m_2, \dots, m_n, t)
 \end{cases}$$

其它部分如用户事件响应单元, 媒体信息  $L$  的分离单元(产生), 上图中有所省略, 一般我们只用预期输出位置与实际输出位置的差值, 所以上图也可简化为图4所示.

2.4 模型讨论

上面的媒体同步控制模型最终目的是想使媒体实际输出函数跟预期的输出函数一致.

但由于媒体输出时的播放速度实际上是与内容有关的,因此实际输出不会与预期输出函数完全一致. 这样我们就通过控制其输出速度,使各种媒体输出跟预期的目标相近,误差控制在一定范围之内.

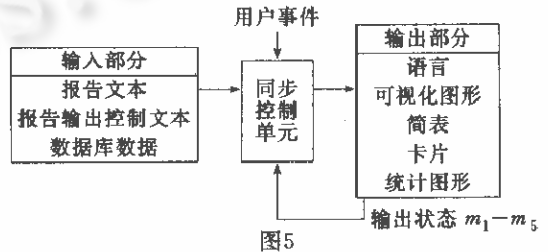
### 3 同步实施方案

#### 3.1 总体结构

市场预测分析后产生的结果是一个预测分析报告文本,以文字形式描述预测分析的内容,另外产生一个报告输出控制文本,此文本描述了各种输出媒体间的时序关系,也就是模型中的预期输出函数. 多媒体信息流中有一部分信息从数据库中取出.

上述3部分数据,即报告文本;报告输出控制文本;数据库数据,组成了同步控制的数据输入部分,合起来组成媒体输入信息流  $I$ .

媒体输出形式有5种,已在2.1节中描述过,我们定义5个媒体对象  $M_1 \sim M_5$ ,分别表示语音、图表、卡片、可视化图形、统计图形5个媒体对象. 同步控制单元总体结构如图5.



#### 3.2 同步控制单元的内部处理

本单元从输入信息中分离出预期输出函数  $F = (f_1, f_2, \dots, f_5)$ .

分别表示语音、可视化图形、简表、卡片、统计图形的输出时序关系,接收输出部分的状态  $M = (m_1, m_2, \dots, m_5)$ .

经过运算产生各个媒体控制对象需要的速度  $v_1 \sim v_5$ ,

$$V = \begin{bmatrix} v_1 \\ v_2 \\ \dots \\ v_5 \end{bmatrix} = \begin{bmatrix} v_1(F, M) \\ v_2(F, M) \\ \dots \\ v_5(F, M) \end{bmatrix}$$

在实际中,我们对模型作了一些简化,我们假定语音输出速度与内容关系不是很大,而基本上由输出速度控制,这样

$$m_1(t) = \int_0^t v_1(x) dx + i_1(t)$$

其中  $i_1(t)$  表示一个很小的误差.

令  $v_1(t) = \frac{df_1(t)}{dt}$ , 则  $f_1(t) = \int_0^t v_1(x) dx$ , 从而预期输出与实际输出的误差为  $i_1(t) = f_1(t) - m_1(t)$ .

$$\begin{cases} v_2 = \frac{df_2(t)}{dt} + c_2 \cdot i_1(t) + d_2 \cdot (f_2 - m_2) \\ \dots \\ v_5 = \frac{df_5(t)}{dt} + c_5 \cdot i_1(t) + d_5 \cdot (f_5 - m_5) \end{cases} \quad (3-1)$$

其中  $C_2 \sim C_5, d_2 \sim d_5$  为用于调节的常数因子.

## 4 同步算法

在实际工作过程中,考虑到控制误差会不断积累,因此,我们采用将输出内容分段的输出方式,上述的同步控制在一段内进行,段与段之间采用另一种被广泛采用的同步方式,即同步点控制技术,来达到积累误差消除的效果.

这种技术从各处输出媒体间插入同步点,到同步点之后,媒体输出暂停直至所有媒体都达到此同步点,再开始下一段的同步控制过程.

### 算法.

1. 初始化各个媒体控制对象
2. 初始化同步控制单元用的时钟
3. IF 报告没有结束,并且产生了时钟事件 THEN
  - 3.1 IF 某个媒体对象达到同步点 THEN
    - 等待所有媒体对象到达同步点
    - GOTO 3
  - END IF
  - 3.2 按式(3-1)计算速度向量  $V$ ,送到相应媒体输出控制单元
  - 3.3 GOTO 3
- END IF

上述算法是本系统所运用的同步控制算法的主要描述,具体实现中则有出错处理及对不同媒体控制单元特殊化的一些处理.

算法中的 3.1 就表达了对上面提出的同步点技术的运用,整个同步控制过程是一个循环,此过程按一定的时间间隔产生时钟事件.没有时钟事件发生时,同步控制单元是不做任何事情的,接收到时钟时间后,同步控制单元开始处理过程,每一个时钟时间中计算一次速度向量,从而控制各个媒体输出单元.

## 5 实验结果及结论

由于媒体同步控制的模型中,各个参数间的关系比较复杂,因此在实际工作中对模型作了简化.实际上,这些简化都是依据实际情况做出的,因此对同步控制的结果影响不是很大.一般来说,语音输出的精度,在一段几十个字的文本中,可以将输出误差控制在 1 秒中之内,对于用户来说,是可以忍受的.简表、卡片和统计图形的输出精度都可以控制的比较好,可视化图形的输出和数据的复杂程度关系较大,因此误差相对大一些.

### 参考文献

- 1 Neil Williams, Gordons Blair. Distribution multimedia application; a review. Computer communications, 1994, 17 (2).
- 2 钟玉琢. 多媒体技术. 桂林:广西科学技术出版社,1993.
- 3 Grigore burdea, Philippe Coiffet. Virtual reality technology. John Wiley & Sons, Inc., 1993.

## MEDIA SYNCHRONIZATION TECHNOLOGY

Xiao Bolei Li Xinyou Wang Cheng Lu Yuchang

*(Department of Computer Science Tsinghua University Beijing 100084)*

**Abstract** A computer system can use multiple types of media to enrich the expression of same piece of information. Thus, while presenting the information to end user, the system faces the problem of multimedia synchronization. This paper gives some exploration on the problem, and proposes a control model. The paper also gives a detailed implementation of the model as well.

**Key words** Media synchronization, synchronization controlling.