

# 有限状态机在动画角色行为中的建模应用

朱晟仁, 黄瑞光

(华中科技大学 电子与信息工程系, 湖北 武汉 430074)

**摘要:**角色行为设计是动画设计的重要环节,也是动画设计的难点。文中提出使用有限状态机来指导动画的角色行为设计,并对有限状态机在动画中的角色行为设计应用作了比较全面的论述。阐述了有限状态机的数学模型,介绍了利用有限状态机建立动画角色行为模型的基本原理。以键盘打字动画中的角色行为设计为例详细介绍了有限状态机建模方法的应用。结果表明,通过使用有限状态机的方法来对动画角色行为建模,能够有效处理复杂的动画角色行为模型,并简化动画的设计。

**关键词:**有限状态机;动画;行为模型

**中图分类号:**TP391.9

**文献标识码:**A

**文章编号:**1673-629X(2007)06-0130-03

## Application of Finite State Machine in Cartoon Role Behavior Modeling

ZHU Sheng-ren, HUANG Rui-guang

(Dept. of Electronic and Information Engineering, Huazhong University of Science and Technology, Wuhan 430074, China)

**Abstract:** Roles behavior design is a very important and difficult part in cartoon design. Uses finite state machine (FSM) to guide their design and makes a comprehensive discussion on the application of FSM in the cartoon role behavior design. First introduces the mathematical model of FSM, then gives the principle of using FSM to model the cartoon roles behavior. Finally, introduces an application of FSM-modeling method by taking the keyboard type cartoon role as an example. The result shows that this method can effectively deal with complicated behavior model and simplify cartoon design.

**Key words:** finite state machine; cartoon; behavior model

### 0 引言

近年来,随着计算机硬件技术的高速发展和计算机图形研究的不断深入,计算机动画技术取得了很大的进展,目前已广泛地应用于教育、娱乐、影视制作以及计算机图形用户接口等众多领域,是深受人们喜爱的媒体表现形式之一。

动画中的角色行为设计是计算机动画设计中的重要内容。在现代的动画设计中,为了让角色的行为具有更好的效果,更能逼真地贴近生活实际。角色的行为设计越来越复杂,而如何对动画中的角色行为进行建模则是角色行为设计的核心,是动画设计人员面临的一大难题。文中提出采用有限状态机(finite-state

machine, FSM)来对动画中的角色行为进行建模。通过使用FSM来管理动画角色的各种状态和行为,然后根据用户的输入以及角色所处的环境确定角色在不同状态之间进行转移并产生对应的动作行为。

### 1 有限状态机的工作原理

有限状态机是一种逻辑系统,是由有限状态组成的一个机制。它包含一系列代表特定行为的状态和一系列代表从一个状态转换到另一个状态的规则。假设 $x(t)$ ,  $s(t)$ 和 $y(t)$ 分别表示一个FSM的输入、状态和输出,那么它的示意图如图1所示。

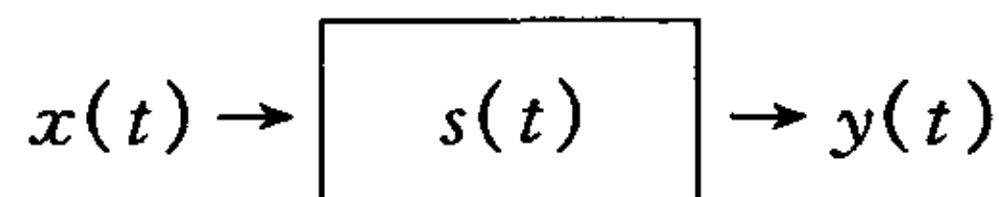


图1 有限状态机示意图

有限状态机由状态集合、输入集合、输出集合和状态转换函数组成。其数学定义<sup>[1]</sup>如下:

有限状态机是一个五元组 $[I, O, S, v, \mu]$ ,其中 $I$

收稿日期:2006-09-18

作者简介:朱晟仁(1983-),男,江西九江人,硕士研究生,研究方向为计算机图形处理、计算机动画研究和通信系统仿真;黄瑞光,教授,研究方向为现代通信系统、信号处理、计算机图形处理和人工智能。



是有穷输入集且  $I = \{i_0, i_1, i_2 \dots\}$ ,  $O$  是有穷输出集且  $O = \{o_0, o_1, o_2 \dots\}$ ,  $S$  是状态集合且  $S = \{S_0, S_1, S_2, \dots\}$ ,  $v$  是  $S \times I$  映射到  $S$  状态的转换函数,  $\mu$  是  $S$  映射到  $O$  的状态输出函数。

在给定一组状态集  $S$  和输入集  $I$  的前提下,一旦输入  $i_{n-1}$  和状态  $S_{n-1}$  确定,在映射  $v$  和  $\mu$  的作用下,则可以确定有限状态机的下一个状态  $S_n$  和输出  $O_n$ 。可以通过状态转移表和状态转换图等方式来描述 FSM 的工作原理。例如一个系统有三个状态,使用状态转移表描述如表 1 所示。

表 1 三状态 FSM 状态转移表

当前状态	输入	下一个状态	输出	输入	下一个状态	输出
$S_0$	0	$S_1$	0	1	$S_0$	1
$S_1$	0	$S_2$	1	1	$S_1$	0
$S_2$	0	$S_0$	1	1	$S_2$	0

而通过状态转换图描述如图 2 所示。

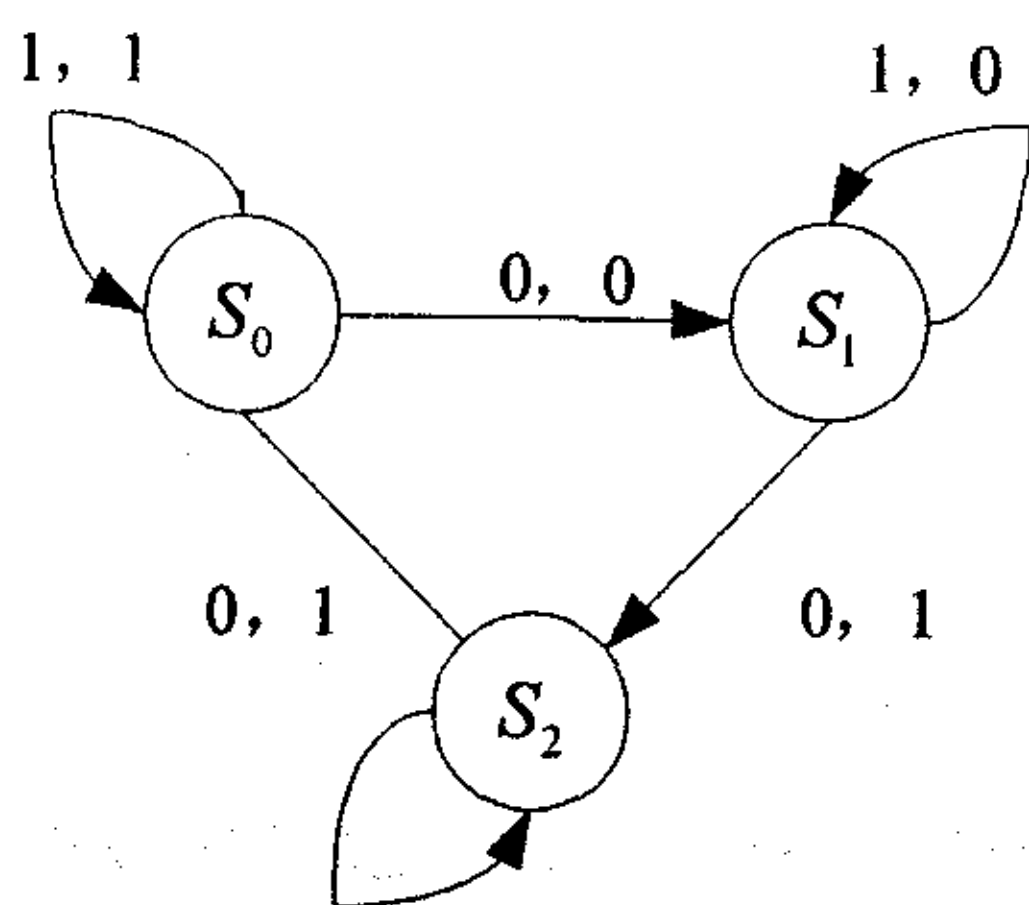


图 2 三状态 FSM 状态转换图

## 2 用有限状态机指导动画中的角色设计

通过 FSM,动画中的角色能够模拟出非常复杂的行为,其中不仅包括运动,还可以根据其他动画物件的状态或动画自身所在环境的全面状况采取不同的行为。FSM 还可进行简单的行为扩充,所有这些只要给出新状态下的行为逻辑,以及进入和离开该状态的逻辑即可。

应用 FSM 来指导动画中的角色,关键是将角色进行建模。图 3 给出了建模的一般原理图。

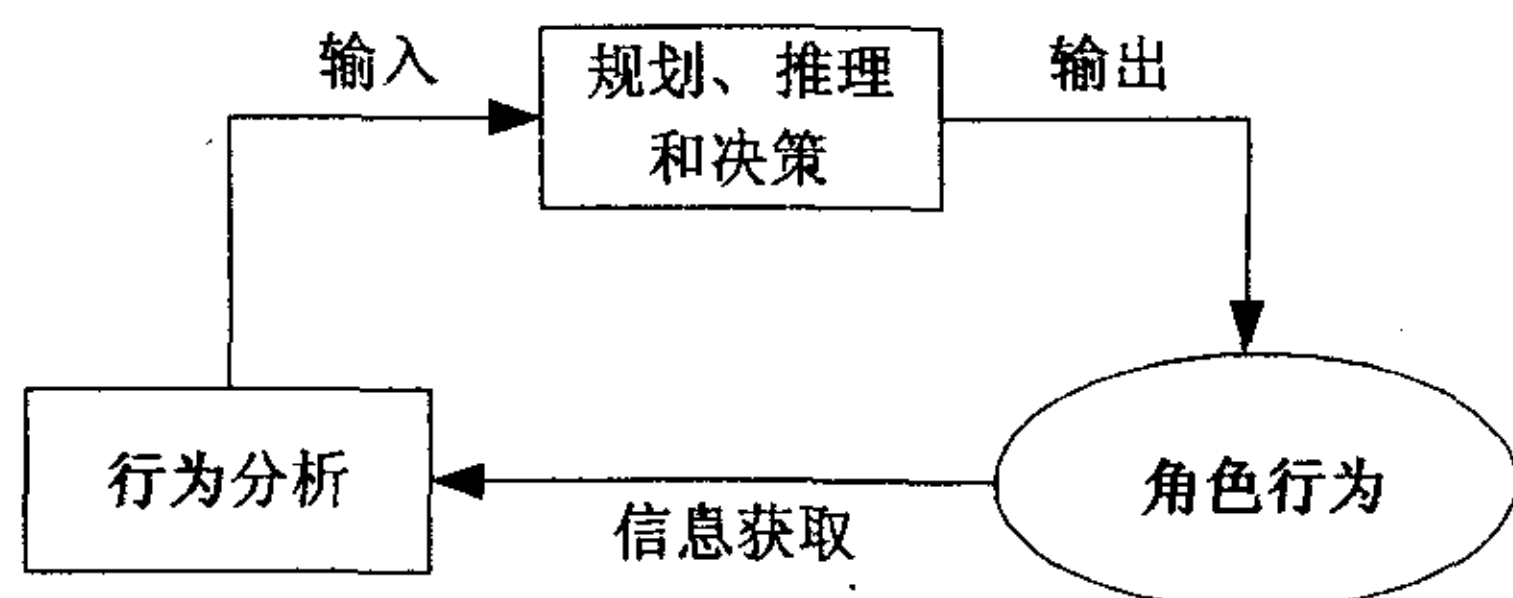


图 3 角色行为建模原理图

在上图中,信息获取包括角色的行为状态以及角色的外部环境。角色行为建模的核心问题是如何根据角色当前的行为以及角色所处系统,决策、推理下一个角色的状态行为。根据前文叙述的有限状态机基本原理,很容易得出两者之间存在着很多的共同之处。因

而,只要确定了角色的行为集合,以及角色在这些不同行为状态下面进行转换的条件,也就是状态转换规则,就可以把角色的行为状态以及转换过程通过有限状态机来进行描述,这就是对角色行为进行建模。

总之,利用 FSM 来指导动画中角色行为建模的一般过程如下:

(1)针对每一个具体的角色对象,结合其所处的环境,将其行为模型建立 FSM 模型,并明确组成该 FSM 的所有状态;

(2)对应每一状态,应明确在各种可能输入(触发事件)的情况下角色所对应的状态转移;

(3)对应每一状态,应明确在各种可能输入的情况下角色所对应的动作响应;

(4)根据动画角色中的当前状态行为进行分析,规划、推理和确定角色下一个状态以及该状态的动作行为。

## 3 FSM 在键盘打字动画中的角色行为模型应用

此处通过一个简单的键盘打字动画中对角色行为建模并实现的具体应用,阐述了怎样利用 FSM 来对动画中的角色行为进行设计。对于复杂的角色行为,其分析方式是相同的。

键盘打字动画中的角色从地上 26 个字母洞中随机出现。角色不断检测用户的输入状态,在角色出现的时间内,如果输入的字母和它头上的字母相同,它会给用户一个“礼品”以示奖赏。如果用户输入的正确率越高,则该礼品越“珍贵”。角色在画面上出现的数量以及出现的时间是根据用户输入的正确率来决定的。当输入的正确率越高,则角色在画面上出现的个数越多而且停留的时间越短。反之数量越少并且停留的时间越长。

为了实现该系统,采用 Microsoft DirectX 中的 DirectDraw 技术,使用 C++ 语言,在 Visual C++ 6.0 平台下开发实现。

### 3.1 DirectDraw 介绍

DirectDraw<sup>[2,3]</sup>是 DirectX 的组件之一。DirectX 是 Microsoft 公司推出的一种在 Windows 环境下进行游戏设计和多媒体开发的工具包。DirectDraw 是 DirectX 的组件之一,使用它开发的程序可以利用硬件厂商提供的驱动程序接口和最佳的设备性能。

DirectDraw 采用了组件对象模型(COM)标准,因此对于不同对象的版本可以有不同的接口,这使得用 DirectX 开发的程序在未来将得到完全兼容和支持的保证。此外,它还和 Windows 图形设备接口 GDI 保持



兼容。利用 DirectDraw 的这些特点,设计人员可以开发出性能优异的 Windows 多媒体程序。

### 3.2 建立角色的 FSM 模型

由前文分析可知,建立角色行为的有限状态机模型的重要步骤是明确该角色行为的状态集合,根据程序设定,角色的状态有如下的几个状态:“出现”、“消失”、“角色把礼盒打开并相应地从礼盒出现若干不同的礼品”。

根据角色的行为状态,以及它们在不同状态进行转换的条件,很容易得出该行为状态的模型图。

键盘打字动画中角色的 FSM 模型如图 4 所示。

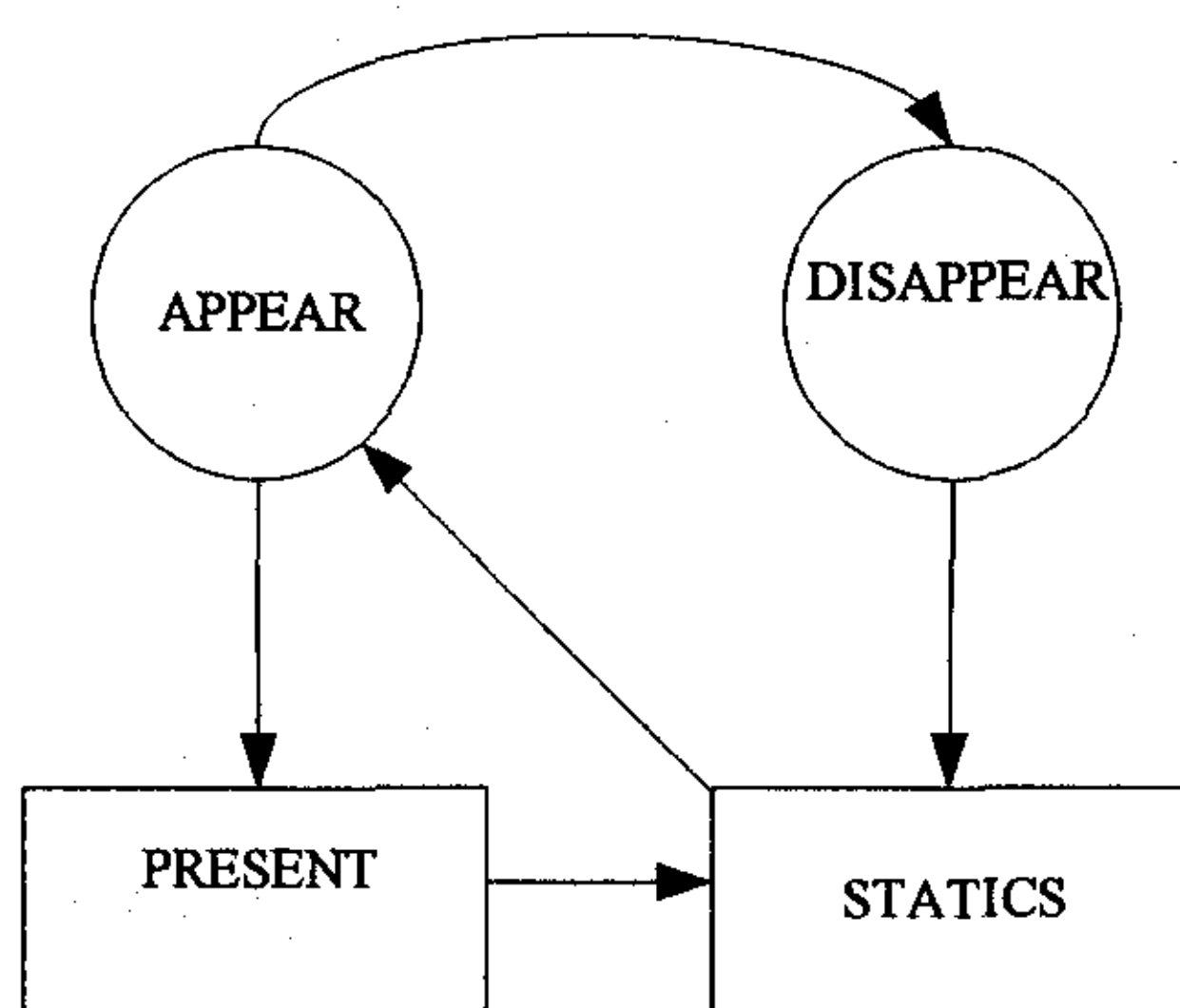


图 4 键盘打字动画中的角色 FSM 模型

这个状态机的默认状态是“DISPEAR”,也就是程序的初始状态没有任何的角色出现。在程序的主进程启动后,画面首先会随机地在 26 个洞中出现若干角色,引起角色的“APPEAR”条件成立,这导致了从默认的“DISAPPEAR”状态到“APPEAR”状态的转移。当状态机处于“APPEAR”状态且玩家输入字母正确时,就会产生一个打开礼盒的动作并从礼盒中拿出礼物给玩家“奖赏”。反之,如果在角色出现的时间范围内玩家输入不正确,则角色直接转移到“DISAPPEAR”状态。

该动画中的角色行为模型主要存在下面几类数据结构:一是通用有限状态机的实现,根据 FSM 的原理,实现了一个通用的 FSM 类以方便程序调用;二是角色自身的数据结构,包括打开礼品的动作等等。下面重点阐述一下通用有限状态机的数据结构。

根据有限状态机的特性,可以从“状态对象”、“状态对象集合”和“有限状态机”这三个方面来阐述有限状态机的数据结构<sup>[4]</sup>。“状态对象”在实现中封装成类<sup>[5]</sup>CState。它定义了一个状态的数据元素、状态转移的条件和建构状态对象的操作(如增加或者删除转移关系等)。“状态对象集合”定义为一个双向链表 SList,数据项是每个状态对象的指针和相应状态的 ID 编号。“有限状态机的表达”在 CState 类对象和 SList 变量的基础上包含了有限状态机的各个状态,它保存了当前的状态变量,默认状态信息,输入、输出、条

件、状态的序列,并且提供了访问外部程序数据的接口函数。它是 FSM 中的容器,是应用程序和 FSM 本身的接口,在实现中封装成 CFMS 类。

### 3.3 角色的状态管理器

状态管理器实现了对动画中的角色行为进行管理的体系和框架,它用来对动画角色的整个状态进行统一管理。程序中是通过创建一个 CTaskManage 类来实现的。在类中应当注册下面的内容:

①函数指针。对于状态管理器可能要激活的状态,都应当将对应的函数指针在状态管理器中进行注册。其中主要包括:用以判断某条件是否满足的谓词逻辑函数、状态转移函数;

②公共数据区。包括状态标识号、角色状态行为统计等变量。例如状态标识号使用一个整型变量来表示,默认值是 0,表示角色没有出现。此外,当角色状态发生改变时,状态管理器应当确保这些数据的及时更新;

③状态列表和动作列表。状态的迁移是根据用户的输入情况来确定,状态管理器利用这些信息确保状态的转移和动作的执行是依照信息流的逻辑顺序进行的。

一旦状态管理器中的信息确定下来之后,状态管理器就可以利用这些信息来控制角色的执行。具体的算法可以描述如下:

(1)首先进行初始化,包括设置角色的默认状态以及相应的动作行为等;

(2)将角色的每一个状态都创建一个 CState 类对象。并对每一个 CState 对象建立和该状态对象相关的转移关系;

(3)创建一个状态管理器 CTaskManage 类对象,并将角色的状态、动作分别添加到状态管理器的状态列表和动作列表里面;

(4)初始化程序的运行环境,包括创建 DIRECTDRAW<sup>[2,3]</sup>对象、DIRECTSOUND 对象等,然后启动主进程开始执行。

程序的实现流程图如图 5 所示。

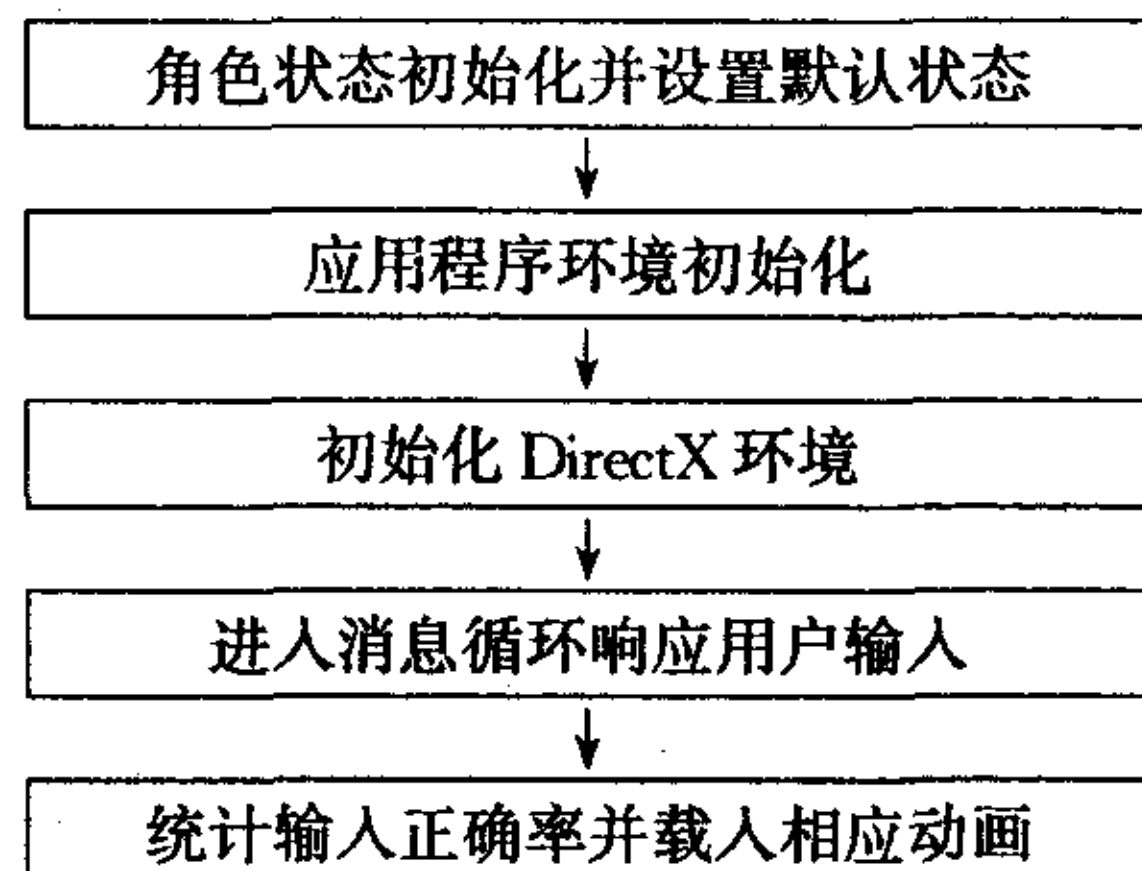


图 5 程序流程图 (下转第 182 页)



在处理过程中申请人需要向保险公司提供相关的保险申请信息,需要向认证中心提交病历资料,以及需要向银行提供账号信息,利用多方数据通信模型来保证这些敏感信息在申请过程中的身份验证和信息完整性、交易防抵赖的保护。其主要处理流程为:

(1)作为数据发送方,首先利用对称密钥生成算法,生成三个对称密钥,记为:K1, K2 和 K3,分别加密,分别用保险公司、认证中心以及银行的公钥加密各自对称密钥;

(2)申请人对病历资料、账号信息以及保险申请信息运用散列算法,生成三者的消息摘要,连接三个消息摘要,并进一步运用散列算法生成连接后的消息摘要,然后用自己的私钥加密连接的消息摘要,生成整体签名,一并发给保险公司;

(3)保险公司接收申请人发送过来的加密数据和整体签名,取出自己敏感数据——加密的保险申请信息以及加密的对称密钥,然后用自己的私钥解密加密的对称密钥,用对称密钥解密加密的保险申请数据,同时分别验证发送者的身份以及信息的完整性,保险公司根据相关规定确定是否接收申请人的申请,如果符合要求,则将申请人发送的数据转发给认证中心;

(4)认证中心接收转发的数据,与保险公司处理过程相似,解密病历资料并验证发送者的身份以及信息的完整性,认证中心根据规定进行认证,并将认证结果通知保险公司;

(5)保险公司分析认证结果,如果符合要求,保险公司将申请人发送的数据转发给银行;

(6)银行接收到申请人的数据,与(3)、(4)处理过程相似,解密账号信息并验证发送者的身份以及信息

的完整性,银行根据规定进行的相关转账处理。

利用多方数据通信安全模型技术优势使得其能够很好地应用于保护保险业务处理过程中多个通信方安全地传输信息,用于其多方通信时的身份验证和信息完整性、交易防抵赖的保护,同时对发送的数据进行整体签名、分别验证。

## 5 结束语

利用数字信封技术保证数据传输的安全性,利用数字签名技术实现整体签名,建立多方数据通信安全模型,解决了多方(超过三方)通信数据安全问题,从应用层的角度提供了多方通信时的身份认证和数据完整性、交易防抵赖的保护,同时对多个信息进行整体签名、分别验证。它所具有的技术优势使其在以电子商务为代表的网络应用中有很好的前景。

### 参考文献:

- [1] Stallng W. 密码编码学与网络安全[M]. 第 2 版. 北京:电子工业出版社,2001.
- [2] Mertz D. 密码学简介:第一部分[M/OL]. 2002-04-26. <http://www-900.cn.ibm.com/developerWorks/cn/cnedu.nsf/security-onlinecourse-bytitle/>.
- [3] Mertz D. 密码学简介:第二部分[M/OL]. 2002-05-17. <http://www-900.cn.ibm.com/developerWorks/cn/cnedu.nsf/security-onlinecourse-bytitle/>.
- [4] Mertz D. 密码学简介:第三部分[M/OL]. 2002-05-24. <http://www-900.cn.ibm.com/developerWorks/cn/cnedu.nsf/security-onlinecourse-bytitle/>.
- [5] 张先红. 数字签名原理及技术[M]. 北京:机械工业出版社,2004.

(上接第 132 页)

## 4 结束语

FSM 作为形式语言的分析工具,它提供了非常简单的计算平台,而且计算的额外开销很小。此外,它还具有结构清晰、逻辑关系明确和系统实现相对容易等优点。因而人们广泛地利用 FSM 来设计动画中的角色行为。当然,有限状态机也存在着不足之处,主要体现在实现角色的复杂行为比较困难。实际上,FSM 的结构复杂性随着状态规模的增加呈现极大程度的增长,这给开发、维护带来了很大的困难,此时可以借鉴其他的建模技术。因此,如何将 FSM 技术和其他的如神经网络和模糊逻辑等结合起来,构建功能强大、系统复杂的角色行为,将是一个重要的研究方向。

### 参考文献:

- [1] 李为建,王 文,秦 兴. 有限状态机在数控系统软件中的应用研究[J]. 组合机床与自动化加工技术,2003(4): 50-53.
- [2] 乔 林,杨志刚. Visual C++ 6.0 高级编程技术 DirectX 篇[M]. 北京:中国铁道出版社,2000.
- [3] 孙 勇,陈胜军. 基于 DirectDraw 的动画设计技术[J]. 现代电子技术,2003,26(20): 3-6.
- [4] 耿卫东,陈 为. 计算机游戏程序设计[M]. 北京:电子工业出版社,2005.
- [5] 刘正林. 面向对象程序设计[M]. 武汉:华中科技大学出版社,2001.