

文章编号: 1007-2691(2002)01-0049-03

有限自动机在自动控制软件设计中的应用

黄志强, 苏颖

(华北电力大学 计算机科学与工程系, 河北 保定 071003)

Application of FA in Design of Automatic Control Software

HUANG Zhi-qiang, SU Ying

(Department of Computer Science and Engineering, North China Electric Power University, Baoding 071003, China)

摘要: 简述了有限自动机的基本概念, 提出了应用有限自动机进行软件设计的处理思想, 并给出了实现方法。有限自动机在软件设计中以及应用过程中充分考虑了软件设计的规律性, 提高了软件设计的工作效率。

关键词: 有限自动机; 自动控制; 程序设计

中图分类号: TP31.51 **文献标识码:** A

Abstract: The concept of FA is introduced briefly and the concept of software design with the aid of FA is proposed. Based on the implementation procedures given in the paper, software design efficiency can be improved.

Key words: FA; automatic control; programming

引言

随着计算机控制技术的广泛应用, 软件设计越来越复杂, 要求越来越高, 其实时性、并发性、交互性及友好性等都难以用顺序的静态的程序来描述和实现。有限自动机在软件设计中的应用, 为设计者提供了一种新的解决问题的思想和方法。

有限自动机是一种具有离散输入输出系统的数学模型。它具有任意有限数量的内部格局或状态, 用此来记忆过去输入的有关信息, 根据当前的输入可确定下一步的状态和行为。一个有限自动机等价于一个状态转换图。这样得到的状态转换图可以应用有限自动机的有关定理和算法进行等价变换、化简, 然后用程序实现。由于状态转换图与程序有一定的对应关系, 所以使得程序设计比较规范化, 从

而高效地完成了自动控制软件设计工作。

(1) 用状态转换图可以方便地描述计算机在自动控制中的处理过程, 尤其是对于离散动态特征的过程。

(2) 有限自动机的有关定理和算法已为设计者提供了坚实的基础和有力的工具。

(3) 状态转换图与程序之间有对应关系。

有限自动机首先被用来作为神经细胞网络的模型。随着这种理论的深入研究和发展的, 逐渐被计算机专业人员重视, 目前主要应用于计算机操作系统、编译程序及文稿编辑设计等方面^[1], 而在自动控制等应用软件设计方面尚未引起足够认识。

1 有限自动机的基本概念

有限自动机是50年代由 Hoffman, Moore 等人提出的, 有限自动机被形式化地定义为一个5元组 (S, Σ, f, S_0, z) ^[2]。其中:

- (1) S 是一个有限的状态字母表;
- (2) Σ 是一个有限的输入字母表;
- (3) f 是一个从 $S^* \Sigma^*$ 到 S 的幂集 2^S 的映射, 即 $f: S^* \Sigma^* \rightarrow 2^S$, f 是一个多值函数;
- (4) S_0 是一个非空的初始状态集;
- (5) Z 是一个结束状态集。

以上定义的有限自动机又称为非确定的有限自动机, 若对它加以限制, 就得到了确定的有限自动机的定义, 即将 (3) 改为 f 是一个从 $S^* \Sigma$ 到 S 的映

收稿日期: 2001-01-16.

作者简介: 黄志强 (1969-), 男, 华北电力大学计算机科学与工程系副教授.

射, f 是一个单值函数; 将 (4) 改为 S_0 是唯一的初始状态。

一般地说, 非确定的有限自动机有较强的描述能力, 它在理论上具有重要作用, 而确定的有限自动机更容易用程序实现。可以证明, 一个非确定的有限自动机等价于一个确定的有限自动机, 并且可以相互转换。

在计算机自动控制应用中, 可以找到许多有限状态系统的例子, 而有限自动机是描述这些系统的有效工具。一个有限自动机可以用一个状态转换图描述。从初态出发到达一个终态, 就可以接收一个符号序列。这个序列是由从初态到达终态的路径上的字符连接而成的。一个状态转换图 (即有限自动机) 可以接收一个特定的串集合, 它是 Σ^* 的子集。

2 用有限自动机进行软件设计的方法

应用有限自动机进行软件设计可按照以下 5 个步骤:

- (1) 将系统划分为几个相对独立的模块。
- (2) 构造每个模块的状态转换图, 即形成非确定的有限自动机。
- (3) 将非确定的有限自动机转换为确定的有限自动机。
- (4) 化简确定的有限自动机。
- (5) 用程序设计语言实现确定的有限自动机。

2.1 软件系统模块划分

一个软件系统从整体上可分成几个模块, 每个模块完成一个相对独立的功能, 成为一个子系统, 如键盘——屏幕管理、打印机管理、通讯管理、控制和数据处理。这些模块分别与不同的外部设备或信息源相联系, 具有相应的特点。将它们分开处理, 可以使系统结构清晰, 还有利于对每一个模块构造一个有限自动机。当然各模块之间的转换关系也可以看作是一个有限自动机。

2.2 构造每个模块的状态转换图

对于每个模块构造相应的状态转换图。这种状态转换图含有一个初态, 允许有多个终态, 其构造过程如下:

首先建立一个初态结点, 在此结点标记上所做的初始化工作, 例如设备工作方式、数据区、计数器等初始化工作; 然后对每个状态结点做以下可能的扩展, 若在该状态下接收一个数据 (包括输入数值、命令或状态信号等), 则从该状态结点引出

一个箭弧, 在该箭弧上标记所接收的数据。在箭弧指向的状态结点 (可能是原有结点, 也可以生成一个新的结点) 上标记接收该数据所做的处理工作。这样反复进行直到没有新的状态结点生成为止。在以上过程中, 应尽量不生成标记有相同处理工作的结点, 以减少状态个数, 使状态转换图简单。这里需要权衡所生成的状态转换图结点数的多少和结点上标记所做工作多少的矛盾, 以便使逻辑上合理、自然, 整体上简单。

2.3 将非确定的有限自动机转换为确定的有限自动机

一般来说生成的状态转换图对应一个非确定的有限自动机, 这里非确定的有限自动机是指在某一状态下存在大于等于 2 条标记相同的箭弧, 而这样的箭弧指向不同的状态结点, 这难于用程序直接实现。所以, 需要将这样的非确定的有限自动机 F 转化为等价的确定的有限自动机 F' , 等价是指 F 和 F' 所接收数据 (符号串) 一样, 功能相同。因 F 不存在有任何一个状态结点, 它引出大于等于 2 条标记相同的箭弧, 指向不同的状态结点, 这一步工作称为“确定化”, 可以用“子集构造”算法实现^[3]。笔者已经在计算机上实现了该算法。它的输入是一个非确定的有限自动机的描述, 输出是与输入等价的确定的有限自动机的描述。

2.4 有限自动机化简

这里说的化简是指对一个确定的有限自动机进行转换, 使得经转换后所得到的确定的有限自动机状态结点最少, 但它仍然与原有限自动机等价。一般来说, 状态结点越少, 实现它的程序越简练。这一步工作可用“逐次分割”算法实现。笔者已经在计算机上实现了该算法, 它可以与 2.3 中的程序联接, 得到一个化简的确定的有限自动机。

2.5 用程序设计语言实现确定的有限自动机

对于经上一步得到的有限自动机, 可编写一个程序实现它所描述的功能, 即对每个状态结点所标记的处理工作编写一个程序段。各状态之间的关系相应于程序设计中的分支和循环这 2 个最主要和复杂的程序结构^[4]。

以上过程中, 对于不熟悉有限自动机的人可以直接写出一个对应确定化的有限自动机的状态转换图, 然后进行程序语言实现。如果处理得好, 会使设计出的程序更直观, 实现起来更自然。这样一种处理方法使得整个软件设计过程思路清晰, 具有一定的规律性, 设计出的程序也符合结构化程序的原

则,拓宽了有限自动机理论的应用范围^[5]。

3 自动控制程序设计举例

笔者在变电站模拟盘微机监控装置项目中使用有限自动机进行程序设计,下面将该实例的一部分简化后介绍如下。

在变电站某条电力线路进行合闸操作前,应事先按照操作规程在模拟盘进行手工试操作,由微机采样判别其每一步操作是否正确并且做出语音提示,最终给出结论,其模拟盘操作过程用状态转换图即有限自动机描述如图1所示。

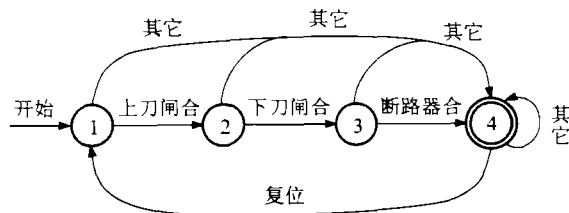


图1 模拟盘操作过程的状态转换图

Fig.1 State transformation in operations on simulation pannel

用程序设计语言实现以上有限自动机,对每个状态结点所做的工作编写一段程序实现,各段程序之间的转移由一个“判别状态”框实现,构成程序的分支^[6]。其框图如图2所示:

由于以上过程比较简单,直接画出了一个确定

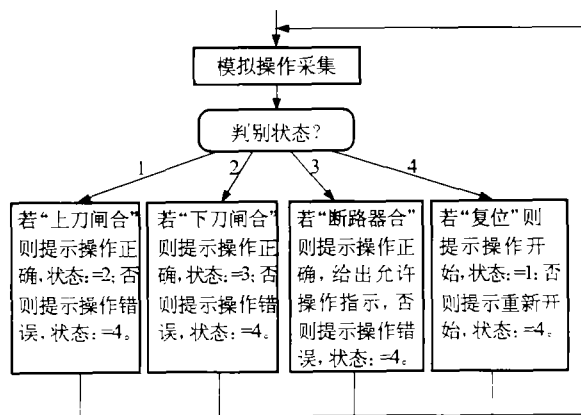


图2 模拟盘操作过程的程序框图

Fig.2 Programming frame

化的有限自动机的状态转换图,然后用程序实现,达到了预期目的。

在本项目程序设计中有合、分闸及安全联锁保护等有10余种操作,按此方法实现的操作过程描述直观,程序结构清晰,取得了很好的效果。

4 结 论

关于有限自动机在自动控制软件设计中的应用方面,笔者作了初步的探讨,在应用实践中取得了一定效果。在此,对进一步的工作提出以下设想:

应用有限自动机进行软件设计的基本步骤、程序所具有的结构已经确定,而且有限自动机的确定化,化简均有成熟的算法。因此可以在计算机上建立一个生成系统,辅助实现自动控制应用软件。这样可以大大减少程序设计的工作量,步骤考虑如下:

(1) 对软件系统进行分析,将系统分成若干功能相对独立的模块。对每一模块进行状态(含相应功能)的划分,确定各状态间的转移关系。

(2) 以交互方式,将上述各状态及相互之间的转移关系输入生成系统。

(3) 生成系统构造相应的有限自动机,并进行“确定化”和“化简”。

(4) 生成系统按有限自动机的每个状态的功能,从一个“例程库”中,选择一些例程,组织成实现相应功能的程序,如果“例程库”不够用时,也可提请用户给出相应的程序段,生成系统最后的输出是用户所要求的程序。

参 考 文 献

- [1] 邹海明. 形式语言、自动机和语法分析 [M]. 武汉: 华中工学院出版社, 1991.173-230.
- [2] 霍普克罗夫特 E. 形式语言及其自动机的关系 [M]. 北京: 科学出版社, 1987.17-160.
- [3] 高仲仪. 编译原理及编译程序构造 [M]. 北京: 北京航空航天大学出版社, 1990.90-101.
- [4] 沈昌祥. 实时系统软件设计初步 [M]. 北京: 人民邮电出版社, 1992.152-191.
- [5] 郝忠孝. 微型机实时软件设计 [M]. 北京: 机械工业出版社, 1989.39-49.
- [6] Horowitz E. 程序设计语言基础 [M]. 北京: 北京大学出版社, 1994.218-223.

(责任编辑: 马坤英)