DOI:10.19461/j.cnki.1004-8626.2014.04.014

有限状态自动机及在字符串搜索中的应用

程晓锦，徐秀花
（北京印刷学院，北京 102600）

摘 要： 有限状态自动机是计算机科学的重要基石，对有限自动机及其应用做了讨论，特别是应用有限自动机描述了简单模式匹配算法及 K. M. P. 算法，并对 K. M. P. 算法的时间复杂度进行了较详细的分析。为了应用有限状态自动机解决实际问题，对有限状态自动机的存储结构做了分析，给出了一种高效的有限状态自动机的存储表示，基于这种存储表示，应用确定有限状态自动机可以建立一种效率高于 K. M. P. 算法的模式匹配算法。使用有限状态自动机建立的算法简单、易懂，且高效，对学生理解掌握有限状态自动机有极大的帮助。

关键词： 有限状态自动机；模式匹配；KMP 算法
中图分类号： TP301.1 **文献标志码：** A
文章编号： 1004-8626(2014)04-0045-04

Finite State Machine and Its Application to the String Searching

CHENG Xiaojin, XU Xiuhua
(Beijing Institute of Graphic Communication, Beijing 102600, China)

Abstract: Finite state machine is an important foundation of computer science, the finite state machine and its application are discussed. The simple pattern matching algorithm and KMP algorithm with finite state machine, and the time complexity of the KMP algorithm are analyzed in detail. In order to apply the finite state machine to solve practical problems, the storage structure of finite state machine is analyzed. An efficient storage structure of finite state machine is constructed. Based on the structure, more efficiency than KMP algorithm of pattern matching algorithm is established using deterministic finite automaton. Algorithms constructed using finite state algorithm is simple, easy to understand, and efficient, and it is help for students to master the finite state automata.

Key words: finite state automata; pattern matching; K. M. P. algorithm

有限状态自动机(FSM)是一种用于设计计算机程序和时序逻辑电路的数学模型，是计算机科学的重要基石。在编译原理课程中，虽然对有限状态自动机有较详细的讨论[1]，但仅限于将 FSM 应用于词法分析中，且对如何将一个 FSM 应用于程序设计中讨论较少。因此，学生在学习 FSM 有关内容后不知如何将 FSM 应用到解决实际问题之中。为了使学生能够熟练应用 FSM 解决实际问题，本文对 FSM 做较深入的讨论，应用 FSM 描述字符串搜索算法，并对搜索算法的效率进行讨论。为了获得高效率搜索算法，对 FSM 及其状态函数的存储表示做了深入研究，给出一种高效的 FSM 状态函数的存储方法及高效的字符串搜索算法。

1 有限状态自动机

有限状态机是一种计算机设备，输入是字符串，输出是两个值，称为接受(Accept)与拒绝(Reject)，有限状态机亦称有限状态自动机[2]。

如果 M 是一个 FSM，则由左向右一次读取输入字符串中的一个字符送到 M 中。M 每次接受一个字符时，由当前状态及读入的新字符而选择下一个状态。可以将一个或多个状态标注为接受状态。如果 M 输入完字符后终止在接受状态，则称输入字符串是接受的，否则称之为拒绝的。这样的 FSM 形式上可以定义如下：

一个确定的有限状态自动机 M 是一个五元组，$M=(K,\sum,\delta,A,\delta)$，其中：$K$ 是状态的有限集合，$\sum$ 输入的字母表，$s\in K$ 是开始状态，$A\subseteq K$ 是一组接受状态，δ 是转换函数，将 $K\times\Sigma$ 中的元素映射到 K 中的一个元素。

应用确定有限状态自动机可以很容易地验证输入是否是可以接受的，验证方法如下：

```
state = startState;
ch = nextChar();
while(ch! = EOL)
{
   state = δ(state,ch);
```

收稿日期：2014-04-30
基金项目：北京印刷学院精品课程建设项目(22150114065)

```
    ch = nextChar();
  }
  if(state in endStates)
    return Access;
  else
    return Reject;
```

其中:startState 表示开始状态,endStates 表示终止状态集,nextChar()表示读取一个输入符号,δ 为状态转换函数,δ(state,ch)表示状态 state 接收符号 ch 后转移到下一个状态,EOL 表示输入结束。

确定有限状态自动机如此表示不易理解,因此,在使用时,一个确定有限状态自动机使用一个有向图(称之有状态转换图)表示。用圆表示一个状态,双圆表示接受状态,用一个无标记的箭头指向开始状态,弧表示状态转换,弧上的标记符号 a 表示由弧尾状态读取符号 a 后转换到弧头状态。如图1所示,表示一个确定的有限状态自动机只有当输入符号串是 abca 时是接收的,否则是拒绝的。

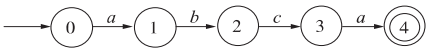

图1 接收 abca 的确定有限状态自动机

如此状态转换图具有如下特点:①有且只有一个开始状态,可以有多个终止状态;②每个弧上只有一个符号,且同一个状态引出的两个弧,如果到达两个不同的状态,则弧上的标记符号不同。

如果减少对状态转换图的约束,即允许有多个开始状态,允许由一个弧引出多个标记相同符号的弧到达不同的状态,甚至允许有标记为空符号 ε(即无任何符号)出现。这种有向图表示一个不确定的有限状态自动机。图2表示一个接受以 abca 为子串的不确定的有限状态自动机。

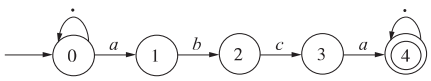

图2 接收含有子串 abca 的不确定有限状态自动机

图中的·表示任何一个符号,标记为·的弧实际上是多条弧的简单表示。不确定的有限状态自动机表示简单容易理解,图2表示的是接受以 abca 为子串的不确定有限状态自动机;图3表示的是接受由 a、b 和 c 3个符号组成的任何仅出现两个符号的字符串的不确定有限状态自动机。由一个不确定的有限状态自动机 M,可以构造一个与其等价的确定有限状态自动机 M'。确定有限状态自动机与不确定有限状态自动机皆称为有限状态自动机

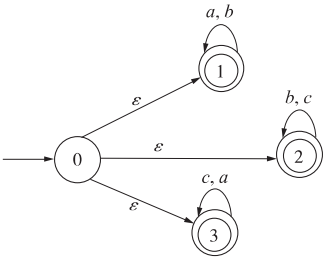

图3 缺项符号串的不确定有限状态自动机

(FSM)。

FSM 不仅可用于表示输入字符串是否能接受,亦可表示一个对象的状态转换,如图4所示,表示道路交通灯的转换。即红灯亮 40 s 后转换成黄灯,黄灯亮 5 s 后转换成绿灯,绿灯亮 40 s 后转换成黄灯,黄灯亮 5 s 后转换成红灯。

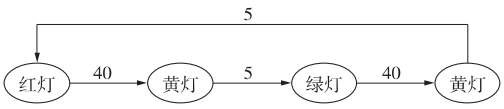

图4 表示交通灯的有限状态自动机

2 字符串搜索算法分析

字符串搜索是文件与字符串的基本操作之一[3-4],文献[5-8]对如何快速搜索字符串进行了大量研究。所谓字符串搜索指:设给出一个称为正文的字符串 T 和一个称为模式的字符串 P,检测 P 是否在 T 中出现。设 $P = a_0 a_1 \cdots\cdots a_{n-1}$,则简单的字符串搜索过程可用图5所示的不确定有限状态机描述。

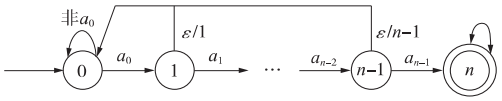

图5 简单匹配算法对应的不确定有限状态自动机

图5中标记为 a_i 的弧表示当前输入符号与 a_i 相同,转换到弧头表示的状态,并读取下一个符号;标记为 ε/i 的弧表示当前符号不与符号 a_i 相同,回退 i 个符号,然后读取串 T 中的符号,并将状态转换到开始状态。匹配失败时,无论什么情况,皆回退若干个符号并由开始状态重新匹配模式串。这样效率较低,极端条件下其读取正文串的时间复杂度是 $O(m \times n)$(m 是正文串的长度,n 是模式串的长度)。一种改进的串搜索算法是 K. M. P. 算法,即:当匹配失败时不是回退字符并回到开始状态匹配模式串,而是回退到某一个状态 i,由状态 i 开始匹配模式串 P。当模式串是 abaabcac 时,可以

使用图 6 表示的不确定有限状态自动机表示其搜索过程。

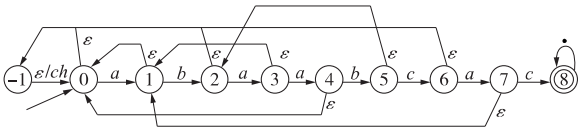

图 6　搜索 abaabcac 的 K. M. P. 算法的不确定有限状态自动机

图 6 中标记为 a_i 的弧的含义与图 5 中的相同，标记为 ε 的弧表示当前符号不与 a_i 匹配，将状态转换到弧头状态继续匹配。标记为 ε/ch 的弧表示读取下一个符号并转移到状态 0。显然，由于 K. M. P. 算法无回退操作，读取正文串 T 最多读取 $|T|$（T 之长度）次，效率明显要高于简单搜索算法。K. M. P. 算法可描述如下：

```
state = 0;
ch = nextChar();
while(state! = n && ch! = EOL)
  if(ch = = P[state]){
    state + +;
    ch = nextChar();
  }else{
    state = next[state];
    if(state = = -1){ ch = nextChar();state = 0;}
  }
if(state = = n)搜索成功; else 搜索失败;
```

由上述算法可见，每读取一个字符发生一次状态转换，每发生一次状态转换需要进行两次字符比较，有可能还要有一次终止状态比较。因此，如果减少状态转换次数，则可以减少状态比较次数。

在第 i 状态处匹配失败，需要转移到第 j = next[i] 状态（$j < i$）。如果当前输入字符不是 a_j，则将再次发生状态转移，转移到第 k = next[j] 状态。因此，在第 i 状态处匹配 a_i 失败后不立即将状态转移到第 j 状态，而是再继续往下匹配 a_j，直到匹配某个符号 a 转移到第 k 状态，读取下一个符号进行匹配。显然经这样处理后，搜索效率要高于 K. M. P. 算法。修改后的算法可以使用一个确定的有限状态自动机描述，图 7 所示确定的有限状态自动机描述了当模式串是 abaabcac 时的搜索算法。

在字符搜索中，不仅搜索一个给定的字符串，而且当需要搜索多个给定字符串或具有某种特征的模式字符串时，是无法应用 K. M. P. 算法搜索，而应用 FSM 则可以方便快捷地实施搜索。

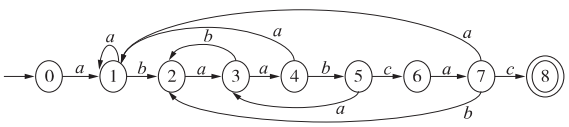

图 7　搜索 abaabcac 的确定有限状态自动机

3　有限状态自动机的存储表示

应用确定的有限状态自动机识别字符串的关键问题是状态转换函数 δ 的存储问题。一种简单的存储方式是使用一个 mxn 的二维数组存储 δ，这里 m 是状态个数，n 是输入符号的个数。如果仅搜索由英文字符及数字组成的文本串，且区分字母的大小写，则需要 62 个字符；如果搜索标准 ASCII 字符组成的文本串，则需要 94 个字符；如果在汉字文本中搜索一个字符串，则需要成千上万个字符。如此存储 δ 不仅占用大量存储空间，还要将输入符号转换成字符表中的下标，需要大量的字符比较操作，效率极低。

事实上，在搜索符号串的确定有限状态自动机中，一个状态仅接受数量不多的符号转移到某个中间状态，而大量的符号，其接受后皆转换到开始状态。因此，对每一个符号使用一个动态长度的数组存储接受的符号及接受符号后的转移状态。为了方便数组的访问，同时存储一个数组长度信息。用 C 语言描述一个状态的转移函数的结构如下：

```
struct Map
{
  char * chs;
  int * nextStates;
  int count;
};
```

用一个 Map 结构数组 maps 存储状态转移函数。计算转移状态算法如下：

```
int map(int state, char ch){
  for(i = 0;i < maps[state].count;i + +)
    if(maps[state].chs[i] = = ch) break;
  if(i < maps[state].count)
    return maps[state].nextStates[i];
  else
    return startState;
}
```

这样存储状态函数虽然比 K. M. P. 算法占用较多存储空间，但显著提高了搜索效率。

4　结　语

应用 FSM 不仅可以直观地描述求解问题的模型，还可寻找到解决问题的高效算法。应用不确定

《北京印刷学院学报》征稿简则

1. 稿件要求:来稿要求论点明确,数据可靠,文字精练,图表清晰,字数在 6000 字以内。

2. 文章书写顺序:中文题名(不超过 20 个字);作者姓名;作者单位,所在地市名,邮政编码;中文摘要;关键词(3 个以上);中图分类号;正文(文中量和单位的用法要符合国家标准);参考文献(3 个以上);英文题名;英文作者姓名(中国人用汉语拼音,一律姓前名后,姓全部用大写字母,名字的首字母大写,);英文作者单位(标准译名),城市邮编;英文摘要;英文关键词。若论文为基金项目,请在文章首页下角注名基金项目名称和编号。

3. 论文摘要书写要求:摘要应以第三人称书写,勿出现"本文研究了"字样,字数 150 ~ 300 字。摘要应该反映论文的核心内容,应包括研究的目的、方法、结果、结论四方面内容。英文摘要应与中文摘要内容相对应。

4. 图表要求:论文中的曲线图、示意图等应放在正确的位置。图片宽度一般不超过 8cm。论文中的表格要求采用三线表,必要时可加适当的辅助线。表号和表题放在表格的上方。表中的参数应标明量和单位(用符号),若单位相同可统一写在表头或表顶线上右侧。表中重复出现的文字,不能用"同前"、"同左"等表示,必须全部写出。论文插图应以矢量格式插入 word,不要用*.jpg、*.tif、*.bmp、*.png、*.gif 等的格式(此类格式的线条和文字编辑不了,照片除外)。矢量格式有:*.ai、*.eps、*.PDF、*.fig、*.dwg、*.org、*.vsd 等。可保存(或输出、或另存为)矢量格式的软件:Illustrator(*.ai、*.eps、*.PDF)、Matlab(*.fig、*.ai)、AutoCAD(*.dwg)、origin(*.org、*.ai)、Office Visio(*.vsd)、Excel 等等。

5. 参考文献的著录:文后参考文献只选最新的、最主要的列入,数量在 3 个以上,只列公开发表的文献,除在文末列出必要的参考文献外,还应在正文中相应引用处按出现的先后顺序用\[1\]、\[2\]……等标出,且著录格式应符合国家标准GB/T 7714 - 2005。现举例如下:

期刊类:\[序号\] 作者(不论中外人名,一律姓前名后,3 位以下必须全部著录,超过 3 位时可只列出前 3 名,后加";等",多个作者间以";"相隔). 文献题名\[J\]. 刊名,出版年,卷(期):起止页码.

例:\[1\] 张三,郑良,刘文. 数字水印算法\[J\]. 北京印刷学院学报,2005,13(1):15-17.

专著类:\[序号\] 作者. 书名\[M\]. 版本(第一版不必写). 出版地:出版者,出版年:引文页码(也可不著录).

例:\[2\] 李四. 机械原理\[M\]. 3 版. 北京:机械工业出版社,2001:28-38.

专著析出类:\[序号\] 作者. 析出文献题名\[文献类型标志\]//专著主要责任者. 专著题名. 版本(第一版不必写). 出版地:出版者,出版年:析出文献的页码.

例:\[3\] 程根伟. 1998 年长江洪水的成因与减灾对策\[M\]//许厚泽,赵其国. 长江流域洪涝灾害与科技对策. 北京:科学出版社,1999:32-36.

报纸类:\[序号\] 作者. 文献题名\[N\]. 报纸名. 出版年-月-日.

例:\[4\] 傅刚,赵承,李佳路. 大风沙过后的思考\[N\]. 北京青年报,2000-04-12.

电子文献类:[序号] 主要责任者. 题名. [文献类型标志]. 出版地:出版者,出版年(更新或修改日期) [引用日期]. 获取和访问路径.

例:[5] 张红. 出版业信息化迈入快车道[EB/OL]. (2001\|12\|29)[2006\|09\|19]. http://www.creader.com/news/20011229/200112290029.html.

6. 稿件的处理。来稿请注明作者联系电话。请勿一稿多投。本刊已入编《中国学术期刊(光盘版)》,并入网万方数据——数字化期刊群,同时,被《中文科技期刊数据库(全文版)收录》。若作者有其他要求请在来稿时声明。

的有限状态自动机描述字符串的搜索十分容易理解,且应用不确定有限状态自动机的确定化可以很容易地建立 K. M. P. 算法中所需要的 next 函数,并且应用这种方法建立的 next 函数的性能优于应用 K. M. P. 算法得到的 next 函数,便于合理地存储确定的有限状态自动机,获得较 K. M. P. 算法更高效的搜索算法。

参考文献:

[1] Alfred V. Aho, Ravi Sethi, Jeffrey D, Ullman. . 编译原理[M]. 李建中,姜守旭,译. 北京:机械工业出版社,2003:76-85.

[2] Elamine Rich. 自动机理论与应用[M]. 邱仲潘, 米哲伟,武桂香,等,译. 北京:北京大学出版社,2011:40-41.

[3] 陈倩. 一种基于有限自动机的快速串匹配算法[J]. 计算机技术与发展,2009,19(1):131-133.

[4] 严蔚敏. 数据结构(C 语言版)[M]. 北京:清华大学出版社,1997:80-84.

[5] Knuth D E, Pratt V R, Morris J H. Fast pattern matching in strings[J]. SIAM J. Comput., 1977,6(1):323-350.

[6] 李 钢,吴燎原,张仁斌,等. 基于有限自动机的模式匹配算法及其应用研究[J]. 系统仿真学报,2007,19(12):2772-2775.

[7] 陈芳,沈虹,张霞. 一种字符串模式匹配算法的实现[J]. 西安工业大学学报,2007,27(3):272-273.

[8] 刘沛骞,冯晶晶. 一种改进的 BM 模式匹配算法[J]. 计算机工程,2011,37(17):248-249.

(责任编辑:邱林华)