

组合数学趣题的 Mathematica 算法

胡 晶

(同济大学生命科学和医学工程研究院, 上海 200092)

摘 要 列举的几个组合数学难题, 有的来自传统的计算机竞赛, 有的经常出现在考验智力的杂志上。要用计算机来解决这些问题, 传统上是通过 C 语言或 Pascal 语言编程来实现解题算法。通过使用 Mathematica 来实现这些算法, 并对各个算法进行了比较和分析。

关键词 Mathematica; 程序设计语言; 穷举算法

Solving Combinatorial Programming Puzzles with Mathematica

HU Jing

(Life Science and Biomedical Engineering Research Institute, Tongji University, Shanghai 200092)

【Abstract】 We analyze here some of the interesting combinatorial puzzles which are traditionally solved by C or Pascal programming languages. These algorithms can easily be implemented and analyzed in Mathematica, which indicates its powerful nature.

【Key words】 Mathematica; Programming language; Enumerative algorithm

Mathematica 下文简称为“麦卡”是一个很好的数学工具软件。它不仅可用于解决数学问题, 它独特的集成环境, 丰富的编程思想对于学习使用计算机解决其它各种问题都是很有帮助的。这里列举的几个组合数学难题, 有的来自传统的计算机竞赛, 有的经常出现在考验智力的杂志上。这些问题描述起来很简单、有趣, 但解答起来就不是很容易了。要用计算机来解决这些问题, 传统上是通过 C 语言或 Pascal 语言编程来实现解题算法。本文通过使用 Mathematica 来实现这些算法。本文假定读者已经具有使用 Mathematica 的基础知识, 因此叙述主要集中在算法实现上。关于 Mathematica 的基础知识可参考国内出版的一些基础教程, 此外该软件的在线帮助和其中 Stephen Wolfram 写的“The Mathematica Book”也是学习麦卡的很好的教材。

1 独特的商(Unique Quotients)

这是一道 1992 年国际计算机竞赛的问题, 现复述如下。

· 写一个程序, n 为输入参数, 求出所有满足表达式

$$\frac{ABCDE}{FGHIJ} = n \quad (\text{其中 } n \text{ 为正整数})$$

的 A, B, C, D, E, F, G, H, I, J。这 10 个字母分别代表 0 到 9 这 10 个不同的数字。例如当 $n=9$ 时, $A=5, B=7, C=4, D=2, E=9, F=0, G=6, H=3, I=8, J=1$ 就是问题的一个解。

这是组合数学中典型的一道字母代数的问题。而使用计算机解决组合数学的问题, 最常用的就是穷举法。也就是说将满足一定条件的可能的解列举出来, 然后从中搜索出完全满足题设条件的最终的解。

首先, 要把可能的解列举出来。麦卡中的内部函数 Permutations 很有用处, 例如对于 0, 1, 2, 3 这 4 个数字的全排列:

```
Permutations[{0, 1, 2, 3}]
```

```
{0, 1, 2, 3}, {0, 1, 3, 2}, {0, 2, 1, 3}, {0, 2, 3, 1}, {0, 3, 1, 2}, {0, 3, 2, 1}, {1, 0, 2, 3}, {1, 0, 3, 2}, {1, 2, 0, 3}, {1, 2, 3, 0}, {1, 3, 0, 2}, {1, 3, 2, 0}, {2, 0, 1, 3}, {2, 0, 3, 1}, {2, 1, 0, 3}, {2, 1, 3, 0}, {2, 3, 0, 1}, {2, 3, 1, 0}, {3, 0, 1, 2}, {3, 0, 2, 1}, {3, 1, 0, 2}, {3, 1, 2, 0}, {3, 2, 0, 1}, {3, 2, 1, 0}
```

一共 24 种情况全被列举出来了。

我们先简化前面的问题, 假设要求出 4 个字母 A, B, C, D 对应的 0 到 3 的 4 个数字, 使得

$$\frac{AB}{CD} = n \quad (\text{其中 } n \text{ 为正整数})$$

那么, 0 到 3 的全排列就是 4 个字母所有可能解的空间。现在就很容易构造麦卡的算法程序了:

```
uniqueQuotients[n_] :=
Module[{pos, quotients, n1, n2, numbers, digits = {0, 1, 2, 3}},
  numbers = Permutations[digits],
  n1 = Map[Take[#, 2] &, numbers],
  n2 = Map[Drop[#, 2] &, numbers],
  n1 = Map[FromDigits, n1],
  n2 = Map[FromDigits, n2],
  quotients = n1/n2,
  pos = Position[quotients, n], Map[{Take[n1, #], Take[n2, #]} &, pos]]
```

运行一下试试:

```
uniqueQuotients[4]
{{{12}, {3}}}
```

可见, 当 $A=1, B=2, C=0, D=3$ 时, 满足

$$\frac{AB}{CD} = 4$$

实际上我们的程序做得更多, 代码“quotients = n1/n2;”已经把所有可能的商的集合全都求出来了。

$$\left\{ \frac{1}{32}, \frac{1}{32}, \frac{2}{16}, \frac{2}{16}, \frac{1}{4}, \frac{1}{4}, \frac{1}{8}, \frac{1}{8}, \frac{5}{16}, \frac{5}{16}, \frac{3}{8}, \frac{3}{8}, \frac{3}{8}, \frac{13}{24}, \frac{20}{24}, \frac{20}{24}, \frac{7}{12}, \frac{23}{12}, \frac{5}{6}, \frac{5}{6}, \frac{10}{12}, \frac{10}{12}, \frac{31}{32}, \frac{31}{32}, \frac{16}{8}, \frac{16}{8} \right\}$$

也就是说, 只有当 $n=4, 7, 23$ 或 32 时字母 A, B, C, D 才有解。

这个算法解决简单问题很有效, 问题再复杂一些呢?

$$\frac{ABC}{DEF} = n$$

现在是 3 位数除以 3 位数, 一共是 6 个数字的组合。只

作者简介: 胡 晶(1976-), 男, 硕士生, 主研心血管系统建模、动态系统数字仿真函数型编程、数学软件开发。

收稿日期: 2001-03-14

最终解。穷举的可能的解的集合小了，算法效率就相应提高。而要使可能的解的集合变小，就要加入约束条件来筛选。这可以说是解决这样一类问题的总的思路。

而使用麦卡来实现算法, 和传统的计算机语言编程有很大的区别。例如对于组合数学的问题不必经过“编码”, 转换成传统计算机语言可实现的数据结构, 而只要通过麦卡的表达式变换和符号运算的丰富内部函数就可以方便地进行算法实现。并且还可以方便地对算法的效率及运算时间做出比较。

从以上对于3个组合数学趣题的讨论中可以看到,用麦卡实现穷举算法还是很有特色的。虽然麦卡的代码是在一个解释执行的系统中运行,执行速度可能比不上C语言等编译执行的程序快,但是它在描述和分析一个算法时很清晰明了。这将大大有助于在算法开发阶段,算法的试验和验证时节省大量的时间。

而通过清晰明了, 接近人类思维方式的计算语言和计算机沟通, 这正是高级语言系统追求的目标。麦卡的编程语言很好地体现了这个特性。

尤其值得指出的是，麦卡语言中可以使用模式(Pattern)

(上接第26页)

RGB_r取值合适的(c)图可见, 尽管期间路面背景有些小变化, 但除了车辆(或非机动车)通过的时间, 其它时刻的V值都极小。这里动态背景技术起了很大的作用。

需要指出的是,在按策略(2)和(3)强制进行背景刷新时,有可能取到假背景,造成车辆的漏识或拒识。由于(2)和(3)的刷新出现得很少,确定背景的时间又很短,因此对识别率的影响是很小的。

3 系统检测效果

下面以上海市交警总队提供的一段录象的检测结果为例,说明系统的检测效果。该录象记录了上海市延安路和中山北路交叉路口在2000年8月21日的交通情况。下面取其中11:00~12:00一段的一个车道,其检测结果如表1所示。

表1 车辆检测结果表

时段	实际通过车辆	多识车辆	总漏识车辆	因遮挡漏识车辆	总识别率(%)	除去遮挡原因的识别率(α_0)
11:00~11:10	78	1	7	4	89.7	94.6
11:10~11:20	91	0	7	7	92.3	100.0
11:20~11:30	93	0	6	5	93.5	98.9
11:30~11:40	76	1	9	5	86.8	93.0
11:40~11:50	81	1	3	1	95.1	96.3
11:50~12:00	65	3	4	3	89.2	93.5
11:00~12:00	484	6	36	25	91.3	96.3

由表1可见,本文方法把背景识别成车辆的次数是极少的,一般只在背景突变的情况下发生。而漏识率却相对较大,这里有很大一部分是由车辆的前后遮挡引起的。由于实验场景位于交叉路口,在绿灯刚亮的一小段时间,车辆相互紧接通过检测线;又由于摄像头的视角是倾斜的(如图1所示),因此造成了前面的车辆遮挡住了后面的车辆。这种遮

匹配和基于规则(Rule)的编程方法。例如第三个趣题算法实现中出现的Select和Cases语句,其中的模式匹配表达式,就很方便地把问题的假设条件变成了麦卡的代码。

麦卡的模式匹配和基于规则的编程思想使之很接近人的思维方式,从而使得它不仅成为一个很好的数学工具软件,而且也可用于实现其他计算机算法。例如用麦卡实现一个冒泡排序算法。

```
bubble[x_List] = x[[{p___, a_, b_, q___} /; b > a] > {p, b, a, q}]
运行一下:
bubble[{9, 7, 1, 6, 11, 21}]
{21, 11, 9, 7, 6, 1}
```

参考文献

- 1 Wolfram S. The Mathematica Book. Cambridge University Press, 1999
- 2 Piele D. Mathematica Pearls: Mathematica in Education and Research 1998, 8(1)
- 3 Trott M. Trott's Corner, Solving Puzzles with Mathematica The Mathematica Journal, 1997
- 4 吴文虎, 赵鹏 1993-1996 美国计算机程序设计竞赛试题与解析 北京: 清华出版社, 1999

挡现象是视频方法检测车辆流量的通病,解决方法是摄像头需要垂直安装^[4]。这里将考虑遮挡和不考虑遮挡的检测结果分别给出。从结果来看,本系统的检测效果令人满意。

4 结论

与以前的识别系统^[1,2]相比,本系统引入了动态投影的概念,动态投影图完全表现了车辆通过检测线的情况。在由动态投影图绘制出的特征值波形图中,可以清楚地看到阈值变化对检测效果产生的影响,使原来完全凭经验的阈值确定过程变得直观而可靠。当特征提取的方案确定以后,即可通过阈值变化,在训练样本的范围内得到最优的识别效果。

特征提取公式(1)是经过大量的实验得到的,它充分利用了实时信号的RGB信息,从图3中可以看到,它能够有效地反映动态投影的变化情况。不同的特征提取方法会产生不同的检测效果,从原理上来说,这些方法都是大同小异的。

动态背景技术的采用保证了检测过程中背景值的真实性,这对于本方案的检测系统是至关重要的。

系统的识别正确率能达到90%以上,若摄像头垂直安装,解决了车辆的遮挡现象,则识别率更可达到95%以上,为本系统在ITS中的实际应用提供了可能。

参考文献

- 1 Guorong X, Jianqiao J, Peiqi C, et al Real Time Traffic Flow Measuring System Based on Gap Video Image Processing, Proc. of 7th Symposium on Transportation System Theory and Application of Advanced Technology, 1994:1134-1137
- 2 姚健超, 宣国荣. 基于图象处理的实时交通流检测系统 交通与计算机, 1991, 23(1):12-17
- 3 Bersen. Dynamic Thresholding of Grey-level Images Proc. of Int. Conf. on Pattern Recognition, 1986:1251-1255
- 4 Autoscope Solo User Guide. Image Sensing System Inc. Version 3 00. 2000-05
- 5 上海市科技攻关成果鉴定: 交通图象处理和语音台成系统技术报告 上海市科委, 1993-04