Vo1.27 Nº11

Computer Engineering

November 2001

・软件技术与数据库・

文章编号。1000 - 1128(200(1)11 - 505] - 01

文献标识码: /

中国分类号- 11961-6

组合数学趣题的 Mathematica 算法

胡茄

《同济人学生的科学和医学主程研究院、上海 2000/02)

摘 要 列平的几个组合数字数据,有的水门技统的计算机竞赛,在的经常出现在考验智力的是出了。要用计算机来解决这些问题、传统主是通过:语言或Paseal 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:"方字算法

Solving Combinatorial Programming Puzzles with Mathematica

HU Zhe

(Little Science and Biomedical Engineering Research Institute, Tongji University , Shanghar200092)

[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语言或子asul语言编程来实现解题算法。本文通过使用 Mathematica 京实现这些算法。本文假定读者已经具有使用 Mathematica 的基础知识。同时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 Woltram 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、现复还如下。每一个程序、正为输入参数、求出所看满足表达式。

ABCDE

FGH17 == n (其中(i(为正整数)

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

这是组合数学中典型的一边字时代数的问题。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至出来、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。 麦卡中的内部所数 Pertinutations 很有用处,例如列于0, 1, 2, 3 这4个数字的 全排列:

Permutations $\{\{0, 1, 2, 3\}\}$

一共 24 种情况至被殉举出来了

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

那么。(2)到3的全排列就是4个字母所有可能解的空间 现在就很容易构造支卡的算法程序了:

uniqueQuotients[n_] :=

Module[(pos, quotients, n1, n2, numbers, digits = {0,
1, 2, 3});

numbers = Permutations[digits].

nl = Map[Take[#, 2] &, numbers].

n2 = Map[Drop[#, 2] &, numbers]

nl = Map[FromDigits, n1].

n2 = Map[FromDigits, n2];
quotients = n1/n2

pos = Position[quotien(s, u], Map[[[[ake]ij], #], [[ake[u2, #]], &, nost]

运行一下试试:

uniqueQuotients[4]

{{{12}, {3}}}

可见,当A=1,B=2,C=0,D=3时,满足

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

$$(\frac{1}{27}, \frac{1}{12}, \frac{2}{17}, \frac{2}{21}, \frac{1}{4}, \frac{1}{7}, \frac{12}{27}, \frac{15}{26}, \frac{4}{16}, \frac{2}{5}, \frac{1}{6}, \frac{1}{6}, \frac{1}{27}, \frac{13}{27}, \frac{20}{13}, \frac{20}{11}, \frac{7}{7}, \frac{7}{10}, \frac{23}{23}, \frac{17}{10}, \frac{5}{27}, \frac{51}{27}, \frac{51}{27}, \frac{51}{27}, \frac{16}{57}, \frac{16}{57}, \frac{1}{27}, \frac{1}{2$$

也就是说, 只有当 n=4, 7, 23 或 32 时字母 A, B, (,) 才有解

- 这个算法解决简单问题很有效,问题再复杂一些呢?

现在是3 位数除以3 位数,一块是4 个数字的组合。以作者简介: 胡 届(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 要把上面的程序修改一下就行了。

运行的结果是:

uniqueQuotients[2] // Timing

{0.11 Second, {{{304}}, {152}}, {{430}, {215}}}} 一共找到两个解,读者可以自己验证一下。在我的 Pentium II 上程序花了 U.11s、那是不是可以把这一算法中的 数字改为{0, 1, 2, 3, 4, 5, 5, 7, 8, 9}来解决"独 特的商"问题呢?简单计算可以知道,由于程序中是通过全 排列来列举问题所有可能的解, 1) 到 5 的全排列有 6' 个, 也就是 720 个元素, 而 ロ 到 リ 的全排列有 10' 个, 就是 3628800 个, 是前者的 5040 倍 也就是说, 光考虑 n1/n2 这个除法运算,就可能要 500 多秒 4 能完成,更何况还要存 储3兆多的全排列元素。从中搜索出符合条件的解。这些对 于现在条件下的计算机,似乎不算是很困难的事。不过只要 稍微动一下脑筋,加入一些约束条件,就可能大大缩小穷举 的可能的解集合的大小。

首先,两个5位数,其中各个位上的数字各不相同。这 样整数 ABCDE 最大可以是 98765, 而最小的 FGHU 为 01234。要使得这样两个 5 位数的商也是一个整数 n, 也就 是说 FGHD 所有可能的解集合是从 1234 到 98765年的整数, 并且这个整数中各个位上的数字不相同。例如, 当 n 是整 数り肘:

n = 9:

imp = Select[

Map[IntegerDigits[#, 10, 5] &, Range[1234,

IntegerPart[98765/n])], (Length[#] === Length(Union(#]]) &];

Length[1mp]

这样一共有 3350 个可能符合条件的 FGHIJ、和原来的 3628800 个列举项比较一下,可见约束条件大大缩小了搜索 解的空间大小、

其次, FGHIJ 在和正整数 n 的乘积就是可能的 ABCDE。 outp = Map[IntegerDigits[#, 10, 5) 6, Map(FromDigits, imp)];

最后按题设的要求, A 到 J 是 10 个互不相同的数字。加 上这个约束条件,就可以求得问题的最终解答了。

result = MapThread[Join(#1, #2) &, {outp, ump}]; Select[result, {10 = Length[Union(#]]) 6}

以上两段代码中、除了使用了麦卡的内部函数 FromDigits和IntegerDigits之外,内部函数Select 和 Union 也起了关键作用。它们把各个位上数字互不相同 的解选了出来。

完整的代码如下:

uniqueQuotients[n_] = Module[{imp, outp, result}, imp=Select[Map[IntegerDigits(#,10,5]&,

Range[1234, IntegerPart[98765/n]]],

(Length[#] = Length(Union(#])) 6).

outp = Map[IntegerDigits[#, 10, 5] &, n Map(FromDigits, imp]];

result = MapThread(Join(#1, #2] &, {outp, imp}]: Select(result, (10 = Length[Union(#])) 6]] //

Map{Transpose({{"A", "B", "C", "D", "E", "F", "G", "H", "I",

"J"), #}] &, #] & // Apply(Rule, #, {2)] & 读者可将其输入麦卡的集成环境运行一下,例如:

uniqueQuotients[9] // Timing // TableForm

0.93 Second $\begin{array}{c} B \rightarrow 7 \\ \xi \rightarrow 4 \end{array}$ $B \rightarrow \theta$ B → 5 $B \rightarrow 5$ $B \rightarrow 5$ B - 7 $C \rightarrow 2$ $C \rightarrow 2$ $C \rightarrow 7$ $\mathbb{C} \to \delta$ $0 \rightarrow 2$ $\tilde{D} \rightarrow 3$ $\tilde{\mathbf{h}} \rightarrow \mathbf{4}$ $D \rightarrow 4$ $\begin{array}{c} I \rightarrow I \\ E \rightarrow J \end{array}$ $D \rightarrow 2$ Ε → 9 $E \rightarrow 9$ $\mathbf{E} \rightarrow 2$ $E \rightarrow 4$ $\mathbf{F} \rightarrow \mathbf{0}$ $\mathbf{F} \to 0$ $\theta \rightarrow 6$ $S \rightarrow 6$ $G \rightarrow 0$ $H \to 6$ $H \rightarrow 4$ $H \rightarrow 3$ 8 → 6 $I \rightarrow 6$ $I \rightarrow 3$ $J \rightarrow 8$

1 - 7

 $J \rightarrow \delta$

 $J \rightarrow ?$ 只用了093x就将所有的6个解全都求出来了。

2 化整为零

这里再列举一道类似的字母代数题:

写一个函数 ZeroSum[n_],求出当 n 为一个输入的正整 数时,所有满足表达式

 $1 a 2 \bar{a} 3 a c... a n == 0$

的运算符句, 其中应可以代表"+","-",和""」数字。 合并)运算。例如:

n=7 时, 1-23-45+67=0 就是问题的一个解。

问题是不是很简单有趣呢?我们再用穷举法,让计算机 来帮助解决问题。不妨先假设 n=9 的情况,这样在 9 个数 字中共有 8 个心, 而每个心可以取 3 种运算符, 所以可能解 的集合为3°,也就是6561个。把它们全都列出来·

chars = {"+", "-", " "}

digitsequence = ToString /@ Range(9)

{+, -, }

{1, 2, 3, 4, 5, 6, 7, 8, 9}

使用麦卡的内部函数 Distribute 把 6561 种心的可能的解 全列举出来了:

charsequences = Distribute(Join(Table[chars, (8)], {""}], List]:

Short[charsequences, 6]

ي ل يورون ويون المراجع في المراجع

再把这些运算符和り个数字结合起来・

digitchar = Flatten[Transpose[[digitsequence, #]]] & 1a charsequences;

Short[digitchar, 6]

({1, +, 2, +, 3, +, 4, +, 5, +, 6, +, 7, +, 8, +, 9, }, 46559∞, {1, ,2, ,3, ,4, ,5, ,6, ,7, ,8, ,9, 1}

possible = StringJoin //a/ digitchar,

Short(possible, 4)

[]+3+3+45+5+6+8+9, [+3+1+44546474]+9, [42+64445447469, *c6555*, [234567849, [234567849, 234567859]

以上就是 6561 个可能的解的集合,下一步就是搜索出 完全符合条件的最终解:

solutions = Select(possible, ToExpression(#)==0 &) 1.+2-3+4-5-6+7, 1-2-3-4+5-5-7, 1-3+3+4-5+6-7, 1-2-3-4-5+6+7, 1-23-4+5+6-7, 1-23-45-6-7

可见,对于 n=9 的情况,共有 6 个最终解。

下面是这个程序的完整代码:

ZeroSum[k_Integer] :=

Module[{chars = {"+", "-", ""},

digitsequence, charsequences),

digitsequence = ToString / d Range[k],

charsequences = Distribute[Join[Table[chars, {k - 1}]], {iiii}], List], Select[StringJoin/\(\delta\) (!*Latten[Transpose {{digitsequence, #{}]} & \(\hat{ha}\)

charsequences), ToExpression[#] == 0 @]] 例如n=7, 代入程序运行--下

ZeroSum[7] // Timing

(9.11 Second,)142-344 (464), 142-144-545-7, 1-2-444-545-7, 1-2-3-4-54647, 1-23-4-54647, 1-23-4-54647

在我的Pentium II 上又花了 II II、一共求出了 6 个最终解。该算法是由 Eric Rimbey 提供的。简单计算可以看到,对于 n 个连续的正整数,穷举的可能解的集合有 3nd 个元素,因此这个集合的大小随着 n 增长得很快。就如前面提到的那样,n 变大时该算法就不那么有效了 我们是否能通过引入适当的约束条件,把可能解的空间变小呢?这就留给读者作为练习吧。

3 谁喝水,谁拥有斑马

对于一个穷举的集合,逐渐引入约束条件,直到求出最终解。这样的方法也适用于解决如下的逻辑趣题。这个逻辑问题经常出现在一些考验智力的杂志上,通过推理确定"准喝水,谁拥有斑马"。

而一共有 16 个假设条件: (1)共有 5 栋房子; (2)英国人住在红房子里; (3)西班牙人有一条狗; (4)住在绿房子里的人喝咖啡; (5)乌克兰人喝茶; (6)绿房子紧挨在象牙色房子的右边; (7)抽 Old Gold 牌香烟的人有蜗牛; (8)抽 Kools 牌香烟的人住黄色的房子; (9)住在中间房子的人喝牛奶; (10)挪威人住在第一栋房子里; (11)抽 Chesterfield 牌香烟的人住在有狐狸的人的隔壁; (12)抽 Kools 牌香烟的人住在有马的人的隔壁; (13)抽 Lucky Strike 牌香烟的人喝桔子汁; (14) 日本人抽 Parliaments 牌香烟; (15)挪威人住在蓝色房子的隔壁; (16)在每栋房子里有一个特定国籍的居民,有一个宠物,抽一种牌子的香烟,喝一种饮料。

下而使用麦卡来实现算法、解决这个有趣的问题。

先分析一下可能解的集合,对于每一个可能解是由集合 (房子编号、房子颜色、住户国籍、喝的饮料、抽的烟、养的 宠物)组成。房子的编号为 1 到 5、代表从左到右排列的 5 标 房子。而其余的 5 个选项,每个有 5 种可能的排列,因此可 能解的集合共有 51⁵=24883200000 个元素。要在这么大的一 个集合中搜索符合条件的解可不是好办法。

我们要做的就是逐步扩充可能解的集合,同时加入约束 条件,保持被搜索的解的集合尽可能小。

在麦卡中如何表示这个问题呢? 非常直接:

houseNumbers = $\{1, 2, 3, 4, 5\};$

houses = {red, green, yellow, blue, ivory};

nations = {Englishman, Spaniard, Ukraninian, Norwegian, Japanese);

drinks = {coffee, tea, water, orangeJuice, milk};
smokes = {OldGold, Kools, Chesterfield,
LuckyStrike, Parliaments};

pets = {dog, fox, horse, snails, zebra};

不妨先从房子的颜色开始穷举可能的解。对于编号 1 到 5 的房子,每栋房子的不同颜色的全排列共有 51 即 120 个可

能的全排列:

houseNumbersAndHousesAll =

Transpose[{houseNumbers,#}]&/@

Permutations[houses].

Length[houseNumbersAndHousesAll]

120

再加入约束条件,根据已知条件 6: 绿房子紧挨在象牙色房子的右边。这下把可能解的集合减少到只有 24 个元素了:

houseNumbersAndHouses=Select[houseNumbersAndHousesAll.

```
Cases[#,{_,green})[[1,1]]-Cases[#,{_,lvory}][[1,1]] == 1 &);
```

Short[houseNumbersAndHouses, 6]

现在加入国籍的全排列,按照乘法原则,可能解的集合 共有24×5! = 2880 个元素了。

houseNumbersAndHousesAndNationsAll =

Map[Flatten,Flatten[With] {p=Permutations[nation
s]}.

 $Function[x, -Transpose][x, -\#][-\& - \hbar \underline{w} - p] - \hbar \underline{a}$ bouseNumbersAndHouses], 1], (2)];

1 ength[houseNumbersAndHousesAndNationsAli] 2880

再使用假设条件 2,10,15 来约束和搜索可能解的集合,这下只剩下 12 个元素了:

houseNumbersAndHousesAndNations =
Select(houseNumbersAndHousesAndNationsAll,
(MemberQ[#, {_, red, Englishman}] &&
MemberQ[#, {1, _, Norwegian}] &&
Abs[Cases[#, {_, _, Norwegian}]][[1,1]]Cases[#, {_, blue, _}][[1, 1]] == 1) &];
Length[houseNumbersAndHousesAndNations]

同理可以逐个把饮料,抽烟,和宠物的全排列加入,同时根据已知条件约束可能的解。限于篇幅,这里就不再一一列举了。

最后可以得出,只有唯一的一个解满足全部的 16 个假设条件:

houseNumbersAndHousesAndNationsAndDrinksAndSmok esAndPets

//TableForm

i .	2	j	4	5
yellow	blue	red	19059	green
Narwegien	Ukranınıan	Englisman	Spanierd	Japanese
Tater	tea	mik	orangeJuije	coffee
Kools	Chesterfield	OldGold	LuckyStrike	Perliaments
fo	torse	scalls	dag	2ebta

从这个麦卡得出的结果可以看到,挪威人是那个喝水的,而日本人是那个拥有斑马的。

4 结束语

可见,用穷举法通过计算机来求解组合数学的问题,主要是穷举所有可能的解,然后在其中搜索出完全符合条件的

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洗。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)

RGB₇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

时间段	实际通 过车辆	多识 车辆	总漏识 车辆	因避挡漏 识车辆	总识别 李(%)	除去选挡原因 的识别率(%)
11.00~11:10	78	1	7	4	89.7	94,6
11.10~11:20	9l	0	7	7	92.3	100,0
11:20~11:30	93	0	6	5	93.5	989
11:30~11:40	76	L	9	5	84.8	93.u
11:40~11:50	81	1	3	1	95.1	96.3
11:50~12:00	65	3	4	3	R9.2	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;又由于摄象头的视角是倾斜的(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

匹配和基于規则(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, Cambride Univer-sity 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:潜华大学出版社,1999

挡现象是视频方法检测车辆流量的通病,解决方法是摄象头需要垂直安装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^{1.3}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,若摄象头垂直安装,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, Jianguo 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视系统 交通与计算机 1991.23(1):12-17
- 3 Bersen Dynamic Thresholding of Grev-level Images Proc. of Int. Conf. on Pattern Recognition, 1986:1251 – 1255
- 4 Autoscope Solo User Guide.lmage 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