

骑士巡游问题的解

肖金声

(中山大学计算中心, 广州 510275)

摘要 骑士巡游是个 np 问题, 本文再次改进其算法, 并提出了两个猜想.

关键词 回溯, np 问题, 出路数

分类号 TP301.6

国际象棋中的骑士(马)按“日”字走动. 在 $n \times n$ 棋盘上, 让骑士从任一初始点开始走动. 要求给出一个方案, 使骑士走 $n^2 - 1$ 步而遍历棋盘上的所有方格(使每个方格恰好经过一次). 这就是所谓的骑士巡游问题.

解此问题的经典算法均采用回溯^[1]. 骑士每前进一步, 须在若干个可能中进行试探, 一次不成便改试另一方向, 直到找出期望的解(存在时), 或是试探完所有可能而无解. 这种试探的不确定性正好说明这一问题是 np 难解的.

假定骑士的当前位置在 (x, y) . 如果由此出发, 下一步在棋盘内共有 k 个尚未到过的格子可供选择, 就说该位置的当前出路数为 k . [2] 中提出的一种改进策略是, 下一步移向 k 个可能的格子中出路数最小者(不唯一时任取). 改进算法见 [2] 或 [3].

采用这种改进后的回溯算法, 在有解时有两个好处: 使大多数有解初始点能无回溯地求得解; 在其余的有解初始点上, 其回溯量是可以容忍的(见图 1). 与此同时, 改进算法还留下两个问题: 某些有解点上的回溯次数仍然较大; 在无解点上把本已不能容忍的运行时间又增加了至少一个数量数. 例如, $n = 7$ 的一个无解情况在 286 机上运行了两天两夜未终止.

0		252		0		0
	0		0		1135	
0		476734		227365		0
	0		0		0	
96244		0		0		0
	0		0		1135	
97564		0		0		0

图 1 $n = 7$ 时各有解点的回溯次数

Fig. 1 The number of times of backtracking for solvable points

收稿日期: 1993-6-25

1 对有解点的猜想

如果我们能预先知道何时解或无解,在有解点才采用改进的回溯算法,则骑士巡游问题就不再是 np 难解的了.为此,笔者进行了大量的试验,得出了关于有解点的猜想.

猜想 1 对于骑士巡游问题

(i) 当 $n < 5$ 时,任何初始位置均无解;

(ii) 当 $n \geq 5$,且 n 为偶数时,任一初始点都有解;

(iii) 当 $n \geq 5$,且 n 为奇数时,若初始点纵横坐标(以四角之一为原点)之和为偶数便有解,为奇数则无解.

例如,图 1 中那些未填数字的格子都是无解的初始点.这一猜想无论做多少试验都是正确的.

2 对算法的再改进

在改进算法中,若出路数最少的点不唯一时是任取一点.考虑到未到过的空格比较集中的范围要多走走的道理,我们可以得出一个新原则:若出路数最少的点不唯一,应选附近空格较多的一点.所谓附近,可以作这样的解释:以该点为中心的五行五列与棋盘之交.

把这一新原则加到改进算法中,便可得到再改进的回溯算法.试验表明,再次改进的算法使无回溯的有解点进一步增多,而其它有解点上的回溯次数则明显下降.例如,当 $n = 8, 9, 12$ 时,所有有解点均无回溯.作为对比,图 2 给出了 $n = 7$ 时,用再改进算法得到的结果.

0		0		0		0
	0		0		62	
0		1498		0		0
	0		0		0	
0		1423		0		0
	0		0		0	
0		0		0		0

图 2 再改进算法在 $n = 7$ 时的结果

Fig. 2 The result of the second improved algorithm when $n = 7$

需要说明的是,虽然再改进算法在总体效果上比改进算法好,但并非在每个有解点上都好.这一点从图 1 和图 2 的对比中可看出.有趣的是两者的有回溯点很少相同.这引导我们得出:

猜想 2 对于充分大的 n ,在有解点上交替使用改进算法和再改进算法,实际可无回溯获解.

据此,有解时的算法复杂度应为 $O(n^2)$. 最后,作为一个实例,给出 $n = 19$,初始点在 $(4,12)$ 的无回溯解,如图 3 所示.

```

81 186  37 176  79  74  39  76  43  72  41 154  45  70  3  68  61  48  5
36 177  80 185  38 175  78  73  40 155  44  71  2 153  62  47  4  51  60
187  82 203 178 189 184  75 174  77  42 167 156 151  46  69 126  67  6  49
204  35 188 201 206 179 190 183 168 173 158  1 166 125 152  63  50  59  52
83 202 205 224 231 200 209 180 191 182 165 172 157 150 161 124 127  66  7
34 225 282 207 222 229 232 199 210 169 192 159 164 171 128 149  64  53  58
283  84 223 230 337 208 221 256 181 198 211 170 193 160 147 162 123  8  65
226  33 336 281 228 259 332 233 220 255 236 197 212 163 194 129 148  57  54
85 284 227 338 331 280 257 260 329 234 219 246 237 196 213 146  55 122  9
32 341 286 335 258 353 330 333 254 261 308 235 218 245 130 195 120 145  56
285  86 345 354 339 334 279 350 321 328 253 262 247 238 143 214 141  10 121
342  31 340 287 346 357 352 327 278 309 264 307 252 217 244 131 144 119 140
87 288 361 344 355 326 349 322 351 320 271 250 263 248 239 142 215 132  11
30 343 290 315 358 347 356 319 310 277 306 265 272 251 216 243 134 139 118
289  88  93 360 325 318 311 348 323 274 299 270 249 242 137 240 111  12 133
92  29 316 291 314 359 324 295 312 305 276 273 266 269 110 135 138 117 104
23  26  89  94 317 294 313 304 275 298 267 300 109 136 241 116 103 112  13
28  91  24  21 292 303  96  19 296 301  98  17 268 107 100  15 114 105 102
25  22  27  90  95  20 293 302  97  18 297 108  99  16 115 106 101  14 113

```

图3 $n = 19$, 开始点为 $(4,12)$ 的解

Fig. 3 The solution when $n = 19$ and the start point is $(4,12)$

参 考 文 献

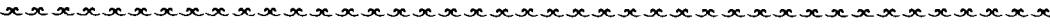
- 1 Niklaus Wirth (中译本). 算法 + 数据结构 = 程序. 北京: 科学出版社, 1984
- 2 Robert J. Rader (中译本). 高级软件设计技术. 广州: 华南理工大学出版社, 1987
- 3 肖金声, 李师贤. Pascal 程序设计教程. 北京: 科学出版社, 1990

The Solution of Knight — tour Problem

Xiao Jinsheng*

Abstract Knight — tour problem is a np problem. This paper improves that algorithm again, and advances two guesses.

Keywords backtrack, np problem, the number of outlet



· 书 讯 ·

《计算机会计系统》一书出版

本校管理学院韦沛文、陈婉玲两位副教授编著的《计算机会计系统》一书已于 1994 年 4 月由中山大学出版社出版。全书 32 万字。该书内容包括计算机会计系统概述、系统开发过程和帐务处理、工资核算、产成品及销售核算、固定资产核算、材料核算、成本核算、会计报表编制、财务分析等 8 个模块及计算机网络介绍。该书特点是各模块均按新会计制度设计, 分析详尽, 方法实用可行、容易理解和实现。帐务处理模块还列出了大量的实用程序; 各章均附有思考练习题。本书可作大专院校财经、会计类学生学习电脑会计或会计信息系统课程教材, 也可供企业财务会计人员参考。有全套软件可供上机教学或实用。

(陈 华)

* Computer Centre, Zhongshan University, Guangzhou 510275