基于遗传算法求解 TSP 问题的一种新方法

傅玉芳
(浙江大学 计算中心紫金港分中心, 浙江 杭州 310027)

摘　要: 针对基于遗传算法求解 TSP 的效率问题, 提出了一种基于位操作编码技术, 并给出了基于位操作的交配、变异等基本操作的实现方法, 有效地提高了计算过程中的空间利用率和计算效率。

关键词: 遗传算法; TSP 问题; 位操作
中图法分类号: TP301.6　　文献标识码: A　　文章编号: 1001-3695(2004)09-0045-02

A New Method of TSP Problem Based on Genetic Algorithms

FU Yu-fang
(Computer Center of Zijingang, Zhejiang University, Hangzhou Zhejiang 310027, China)

Abstract: The paper makes a new method of bit-operation aiming at the efficiency of solving TSP based on genetic algorithms. The paper brings out an approach of genetic operation based on bit-operation like crossover or mutation. This kind of storing can reduce storage greatly and benefit to practice an effective route of genetic operation.

Key words: Genetic Algorithms(GA); TSP Problem; Bit-operation

1　引言

遗传算法(Genetic Algorithm, GA)是模拟自然界生物进化过程的搜索最优解的方法, 被广泛用于机器学习、人工智能、人工神经网络训练、图像处理、TSP 组合优化等问题的求解。TSP (Traveling Salesman Problem, 又称货郎担问题)是一个典型的、易于描述却难以处理的 NP 完全问题, 在遗传算法研究中, 常用于评价不同的遗传操作及选择机制的性能。因此, 如何在进行大规模数据处理中提高计算机效率, 有效地解决 TSP 问题具有重要的理论价值。TSP 问题就是在一个网络图(本文假设是完全图) G 中, 寻找一条最短的遍历 n 个城市的路径, 或者说搜索整数子集 $X = \{0, 1, 2, \cdots, n-1\}$ (X 的元素表示对 n 个城市的编号) 的一个排列 $\prod(X = \{v_0, v_1, v_2, \cdots, v_{n-1}\})$, 使

$$Td = \sum_{i=0}^{n-2} d(v_i, v_{i+1}) + d(v_0, v_{n-1})$$

取最小值。式中 $d(v_i, v_{i+1})$ 表示 v_i 到城市 v_{i+1} 的距离。

我们知道, 遗传算法的处理对象不是参数本身, 而是对参数集进行了编码的个体, 即染色体。对于 TSP 问题, 处理对象是 n 个城市的编号排列。以往的编程方法是对城市编号用数组结构形式存储。如果我们利用一些低级操作进行遗传算法的二进制编码和基因操作, 如汇编语言和 C 语言就具有对数据的内码进行操作的功能。改用二进制编码对城市编号进行处理, 能显著提高处理速度和计算机的利用效率。

2　常用二进制编码及基因操作编码方法

在求解 TSP 问题的各种遗传算法中, 多采用以遍历城市的次序排列进行编码的方法。如在八个城市组成的网络完全图中, 码串 32405617 表示自城市 3 开始, 依次经城市 2, 4, 0, 5, 6, 1, 7, 最后返回城市 3 的遍历路径。满足 TSP 问题约束条件的可行解空间规模共有 8! = 40 320 种。

以上八个城市编码组成的码串 32405617 形成一个"染色体", 若用字符串数组 ts1 存储, 则有 ts1[8] = '32405617', 占 64bits; 若用整型数组 tt1 存储, 则有 tt1 = {3, 2, 4, 0, 5, 6, 1, 7}, 占 128bits。遗传算法就是对由多个"染色体"组成的群体, 按优胜劣汰的原则进行演化计算, 通过复制、交配和变异三种基本遗传操作以及其他特殊操作来实现。复制是遗传算法的基本算子, 按"适者生存"的选择原则, 直接将优良个体传到下一代新群体中繁殖。交配是仿照生物学中杂交的原理, 将两个个体的部分字符互相交配, 以实现遗传物质的互配。

例如在由 Goldberg 和 Lingle 提出的部分匹配交配(Partially Matched Crossover, PMC)法中, 先随机产生两个位串交配点, 定义这两点之间的区域为一匹配区域, 并使用位置交配操作交配两个父串的匹配区域。如两父串及匹配区域为

A = 74|561|203　　B = 36|270|541

首先交配 A 和 B 的两个匹配区域, 得到

A^1 = 74|270|203　　B^1 = 36|561|541

对于 A^1 和 B^1 两子串中匹配区域以外出现的遍历重复, 依据匹配区域内的位置映射关系, 逐一进行交换。对于 A^1 匹配区以外的 2, 7, 0 分别以 5, 6, 1 替换, 则得 A^2 = 64|270|513

同理可得 B^2 = 37|561|240

这样, 每个新串的次序部分由其父串确定, 不存在未能完全遍历所有城市的非法路径。

变异是次要操作, 起恢复丢失的遗传信息的作用。它是将个体字串中某一位字符"求异", 即将 0 变为 1, 1 变为 0。为保持群体内个体的多样化, 采用连续多次对换的变异技术, 使可行解有较大的顺序排列上的变化。如对换变异, 随机选择串中两点, 交换其码。对于串 X = 012|345|67

收稿日期: 2003-05-15; 修返日期: 2003-09-23

若对换点为 2 和 5, 则经对换后, 得 X^1 = 01534267

3 位操作编码技术

位操作是指对变量或指针所在的储存单元内码进行移位、复制及计算等操作, 在汇编语言提供了如下逻辑指令:

(1) 逻辑与指令 AND

格式: AND 目的操作数, 源操作数

如 AND AL, 06H, 若 AL 的值是 00001001B(09H), 则逻辑与后的值是 00000000B。用 AND 指令可以使操作数的某些位被屏蔽。

(2) 逻辑或指令 OR

格式: OR 目的操作数, 源操作数

如 OR AL, 06H, 若 AL 的值是 00001001B(09H), 则与 06H 逻辑或后的值是 00001111B(即十六进制 0FH, 十进制 15)。用 OR 指令可以使操作数的某些位置 1。

(3) 异或指令(XOR)

格式: XOR 目的操作数, 源操作数

如 XOR AL, 17, 若 AL 的值是 00001101B, 则与 00010001B(17) 异或后的值为 00011100B(即 28)。用 XOR 指令可以使操作数的某些位置反。

(4) 逻辑非指令(NOT)

格式: NOT 操作数

如 NOT AL, 若 AL 的值是 00011001B, 则取非后的值是 11100110B(即十进制 230)。

(5) 算术左移指令(SHL)

格式: SHL 操作数, 移位次数

如 SHL AX, CL, 若 AX 的值是 00010101B, CL 的值是 2, 则 AX 的值左移 2 位后得 01010100B。

(6) 算术右移指令(SHR)

格式: SHR 操作数, 移位次数

如 SHR AX, CL, 若 AX 的值是 00101101B, CL 的值是 2, 则 AX 的值右移 2 位后得 00001011B。

另外, 汇编语言中还有算术左移、算术右移、循环左移、循环右移等操作指令。

3.1 基于位操作技术实现基因的存储、编码和解码

前面讨论的用一个数组表示一个染色体的方法中, 一个数组元素仅能存放一个基因。在求 n 个城市(假设 $n \le 256$) 的 TSP 问题中, 一个染色体所需的二进制位数为 8n(字符数组) 或 16n(整型数组)。在汇编语言中, 可以采用位操作方法, 直接读出任何整数的机器内码, 并对它们进行运算和操作。为了有效使用内存, 本文提出用一个或若干个整型数存放一个染色体。这样在求 n 个城市的 TSP 问题中, 一个染色体只需用 $n \times \lceil \log_2 n \rceil$ 位二进制码表示。表 1 表示不同城市数与一个染色体所占的二进制位数的关系。

表 1 不同城市数的取值与一个染色体所占的二进制位数的关系

城市数 n	[129, 256]	[65, 128]	[33, 64]	[17, 32]	[9, 16]	[5, 8]	[3, 4]
一个染色体所占位数	8n	7n	6n	5n	4n	3n	2n

从表 1 中看出, 用一个或若干个整型数存放一个染色体, 当 $n \in [129, 256]$ 时, 与字符型数组存放的空间相等, 但比用整型数组存放节省 50.0%; 当 $n \in [65, 128]$ 时, 比用字符型数组存放节省 12.5%, 比用整型数组存放节省 56.25%; 当 $n \in [33, 64]$ 时, 比用字符型数组存放节省 25.0%, 比用整型数组存放节省 62.5%; 当 $n \in [17, 32]$ 时, 比用字符型数组存放节省 37.5%, 比用整型数组存放节省 68.75%; 当 $n \in [9, 16]$ 时, 比用字符型数组存放节省 50.0%, 比用整型数组存放节省 75.0%; 当 $n \in [5, 8]$ 时, 比用字符型数组存放节省 62.5%, 比用整型数组存放节省 81.25%; 当 $n \in [3, 4]$ 时, 比用字符型数组存放节省 75.0%, 比用整型数组存放节省 87.5%。由此看出, 用一个整数存放若干个基因的方法可最大限度地使用内存空间, 有利于进行海量数据的演算。

运用汇编语言提供的逻辑指令可方便地实现编、解码。假设 n = 16, $X = \{v_0, v_1, v_2, \cdots, v_{n-1}\}$ 为 n 个城市编号一个排列, $y_1, y_2, \cdots, y_w$ 为无符号整型数。由 $L = \lceil \log_2 n \rceil$, 得 L = 4; 同理由 $w = \lceil (n \times \lceil \log_2 n \rceil) / 16 \rceil$, 得 w = 4。编码程序段如下:

```
x dw n Dup(?); x 为一条遍历 n 个城市的路径, 即一个染色体
y dw w Dup(0)
…
mov bp, n
dec bp ; 初始位置指向 x 数组的最后一个元素 x(n-1) 的地址
mov di, w
dec di ; 初始位置指向 y 数组的最后一个元素 y(w-1) 的地址
mov cx, w ; 计数
push cx
loop1: mov dx, 0
mov ch, 4 ; 由 16/L 求得, 这里 L = 4, 所以有 ch = 4
mov cl, L ; 移位次数, 即一个基因所占二进制位数
rotate: shl dx, cl
mov bx, x[bp]
add dx, bx
dec bp
dec ch
jnz rotate
mov y[di], dx
dec di
pop cx
loop loop1
```

解码与编码相反, 即将 y 数组中的每一个基因分别放入 x 数组的相应元素中, 其代码略。

3.2 用位操作实现基因操作

在一个或若干个整型数中包含全部基因, 在汇编语言中, 无须更多指令就可实现基因操作, 所以能有效提高运算速度。

(1) 交配操作。在部分匹配交配(PMC) 法中, 先随机产生两个位串交配点。以下面两个码串(两个不同的染色体) 为例, 假设交配位置为 4, 12。

W1: 0111 | 0100 0101 | 0110 W2: 0011 | 0110 0010 | 0111

则其交配代码段如下:

```
mov ax, w1
and ax, 0f0ffh ; ax <— 0111 0000 0101 0110
mov bx, w1
and bx, 0f00h ; bx <— 0000 0100 0000 0000
mov cx, w2
and cx, 0f0ffh ; cx <— 0011 0000 0010 0111
mov dx, w2
and dx, 0f00h ; dx <— 0000 0110 0000 0000
add ax, dx
mov w1, ax ; w1 <— 0111 0110 0101 0110
add bx, cx
mov w2, bx ; w2 <— 0011 0100 0010 0111
```

(2) 变异操作。在对换变异法中, 随机产生串中的两点, 交换码值。对于下列码串:

w: 1011 0111 1101 0110

交换 1011 和 1101 的值, 得 w: 1101 0111 1011 0110

其对换代码段如下:

```
mov ax, w and ax, 0f000h
```

(下转第 49 页)

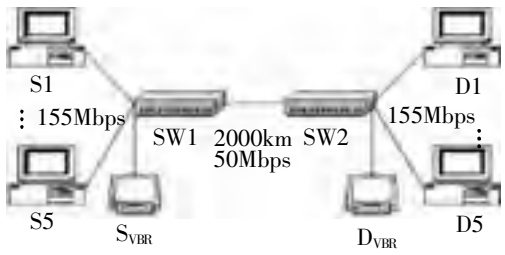

图 1　自相似特性 VBR 背景下单一瓶颈链路网络仿真模型

表 1　ABR 服务端系统的主要参数

参　数	NRM	ICR	PCR	MCR	RIF	RDF	CRM	ADTF	CDF	TRM
选用值	32	4MB	149Mbps	1.49Mbps	1	1/16	1500Cells	500ms	1/16	100ms

4.2　仿真结果

图 3 至图 5 分别是三种算法的仿真结果。在每个图中, 左边的纵轴标志 ABR 源系统 S1 的 ACR 值, 单位是 Mbps; 右边的纵轴标志交换机 SW1 的队列长度, 单位是 Cell。由图可见, EFCI 算法的 ACR 值在 MCR 和 PCR 之间大幅震荡, 同时交换机队列长度也在 0 与最大值 5 000 之间震荡, 这说明 EFCI 机制的不稳定, 同时由于队列溢出造成了信元丢失; ERICA 算法和 PD 算法的 ACR 初始值为 ICR, 30ms 左右之后开始在 6Mbps 上下抖动; ERICA 算法的交换机队列长度一直在 0 ~280 之间震荡, 而 PD 算法的交换机队列长度在到达稳态值 150 之后 (经过 50ms) 在 90 ~220 之间抖动。

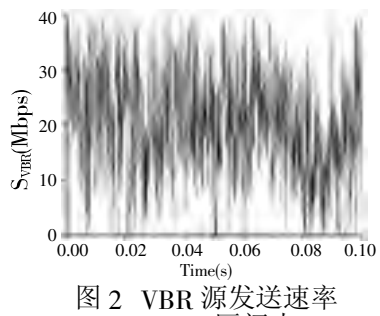

图 2　VBR 源发送速率 (0s ~0.1s 区间内)

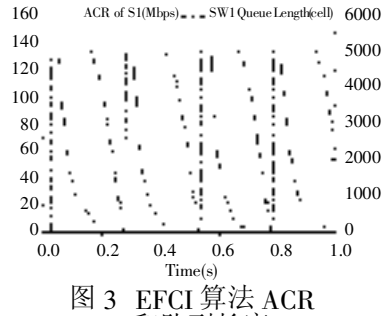

图 3　EFCI 算法 ACR 和队列长度

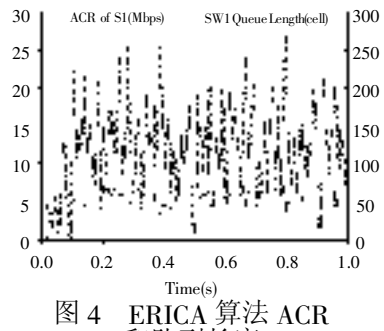

图 4　ERICA 算法 ACR 和队列长度

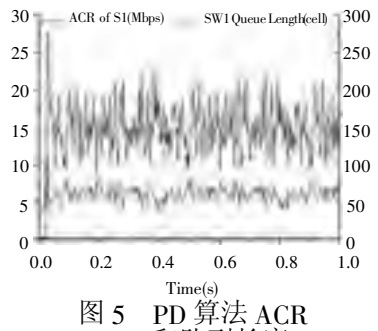

图 5　PD 算法 ACR 和队列长度

从以上分析可知: EFCI 算法虽然实现简单, 但是性能最差, 并且有信元丢失。ERICA 和 PD 算法都可以在自相似流的干扰下把剩余带宽公平地分享给各个源端系统, 而 ERICA 算法虽然不能保持队列长度的稳定, 但是依然使队列长度保持在一个有限的范围内, PD 算法由于主要以队列长度作为算法的测度, 在保证可公平分享剩余带宽的同时, 也能使交换机队列长度保持稳定。并且 PD 算法实现要比 ERICA 算法简单得多。

5　结论

本文比较了在自相似业务背景下几种 ABR 控制机制的性能。由反映网络运行性能的两个动态参数: ABR 源端允许信元速率 ACR 和瓶颈交换机输出缓冲区队列长度 q。仿真实验表明 EFCI 算法实现简单但是性能最不稳定。ERICA 和 PD 算法都可以在自相似流的干扰下把剩余带宽公平地分享给各个源端系统, 使网络链路带宽得到充分利用, 并具有极好的瞬态和稳态性能, 其中 PD 算法在保证可以公平分享剩余带宽的同时, 也能使交换机队列长度保持稳定, 而且计算的复杂度较低。

参考文献:

[1]　raffic Management Specification 4.1[EB/OL]. http://www.atmforum.com, 1999.

[2]　张孝林, 吴介一, 朱正强, 等. ATM 网络中面向 ABR 服务的一种流量控制机制[J]. 电子学报, 2002, 30: 503-507.

[3]　alyanaraman. Traffic Management for the Available Bit Rate (ABR) Service in Asynchronous Transfer Mode (ATM) Networks[D]. The Ohio State University, 1997.

[4]　eran J, Sherman R, Taqqu M S, et al. Long-range Dependence in Variable-bit-rate Video Traffic[J]. Communications, IEEE Transactions, 1995, 43: 1566-1579.

[5]　axson V, Floyd S. Wide Area Traffic: The Failure of Poisson Modeling Networking[J]. IEEE/ACM Transactions on, 1995, (3): 226-244.

[6]　axson V. Fast Approximation of Self-Similar Network Traffic[J]. Laurence Berkeley Laboratory, 1995, (4).

[7]　张少博, 张孝林, 吴介一, 等. ATM 网络中一种基于传统控制理论的流量控制机制(论文集)[C]. 第六届全国计算机应用联合学术会议, 北京: 北京邮电大学出版社, 2002. 193-197.

作者简介:

张少博(1974-), 男, 博士研究生, 主要研究方向为高性能网络; 吴介一(1941-), 男, 教授, 博士生导师, 主要从事自动控制及计算机应用的教学、科研; 刘鑫(1979-), 女, 硕士研究生, 主要研究方向为高性能网络; 张飒兵(1968-), 女, 高级工程师, 主要研究方向为集成制造环境。

(上接第 46 页) mov cl, 8

```
    rol ax, cl ; ax < —0000 0000 1011 0000
    mov bx, w

and bx, 00f0h
    rol bx, cl ; bx < —1101 0000 0000 0000
    add ax, bx
    mov bx, w
    and bx, 0f0fh ; bx < —0000 0111 0000 0110
    add ax, bx
    mov w, ax ; w < —1101 0111 1011 0110
```

4　结束语

按照生物进化论演变而来的遗传算法, 对 TSP 组合优化的难题, 有了新的思路。在基于 GA 的 TSP 问题中, 进化系统保持了可能得到的整个解空间, 体现了 "适者生存、优胜劣汰" 的选择过程。在随机扰动情况下进行个体(染色体)的交配、复制和变异, 其计算量是十分巨大的, 所以要求尽可能优化计算程序, 以提高计算效率。利用汇编语言操作指令进行基因的编、解码, 并在此基础上实现基因操作, 有效地提高了 TSP 算法的计算效率。

参考文献:

[1]　陈国良, 等. 遗传算法及应用[M]. 北京: 人民邮电出版社, 1996.

[2]　沈美明, 温冬婵. IBM PC 汇编语言程序设计[M]. 北京: 清华大学出版社, 1995.

[3]　陈超, 陈家联, 余丰. 用位操作编制遗传算法程序的一种技术[J]. 计算机应用研究, 2002, 19(11): 20-22.

作者简介:

傅玉芳(1969-), 女, 讲师, 在职硕士研究生, 研究方向为计算机应用与计算机图像处理。