DOI:10.15938/j.jhust.1997.05.021

第2卷 第5期 哈尔滨理工大学学报 Vol. 2 No. 5
1997年10月 JOURNAL HARBIN UNIV. SCI. & TECH. Oct., 1997

# 基于模拟退火算法旅行商问题的并行实现

郭茂祖 洪家荣

(哈尔滨工业大学)

摘 要 旅行商问题属于NP难题, 不存在多项式时间的算法. 提出一个基于模拟退火算法求解旅行商问题的算法, 并在并行设计环境Multi-pascal中加以实现.

关键词 模拟退火算法; 旅行商问题; NP难题; 加速比

分类号 O157

旅行商问题(TSP)属于NP难题, 不可能在多项式时间内找到最佳解, 因此, 对有关求其近似最优解算法的研究一直是算法设计的一个重要课题. 模拟退火(SA)算法是解决组合优化问题的有效方法, 但它计算时间较长. 为此, 提出一个基于模拟退火算法求解旅行商问题的算法, 并将它在并行设计环境Multi-pascal中加以实现, 实验结果比较理想.

## 1 Multi-pascal概貌

Multi-pascal是由美国Maharishi国际大学的BruceP. lester博士等研制的一种基于Pascal的并行设计环境和并行算法评价工具, 是在标准Pascal语言中加入表达并行的成分而构成的一种并行程序设计语言. 它要求程序指定数据分解和进程间通信, 支持对共享存储和分布存储两类多机系统的编程. Multi-pascal的解释器解释用Multi-pascal语言编写的程序在多机系统上执行, 不仅给出程序要求的输出结果, 还给出该程序并行执行相对于其串行执行的加速比、算法效率等.

Multi-pascal系统汇报的加速比是采用并行算法的串行执行时间和其并行执行时间之比, 而通常在并行算法的性能衡量中提到的加速比是指最好的串行执行时间和待考察的并行执行时间之比[2].

## 2 旅行商问题与模拟退火算法

### 2.1 旅行商问题

**定义1 (TSP)** 设有 $n$ 个城市和距离矩阵 $\boldsymbol{D}=[d_{ij}]$, 其中, $d_{ij}$ 表示城市 $i$ 到城市 $j$ 的距离, $i, j=1, 2, \cdots, n$, 问题则是要找恰好遍访每个城市一次的一条回路, 且其路径长度为最短.

为简便起见, 仅讨论Euclid平面上的TSP, 其距离矩阵 $\boldsymbol{D}=[d_{ij}]$ 满足对称性 $dij=dji$ 和三角不等式 $d_{ij}+d_{jk}\geqslant d_{ik}$, 且 $d_{ii}=0$, 并假设 $d_{ij}$ 均舍入为整数, $i, j, k=1, 2, \cdots, n$.

### 2.2 模拟退火算法

在解决组合优化问题时, 与贪婪策略不同的是, 模拟退火算法摆脱了局部最优的束缚, 它的解

收稿日期: 1996-10-17

逼近全局最优.

SA 算法可用类 PASCAL 语言描述如下：

```
procedure Simulated-Annealing;
begin
  INITIALIZE (i0, t0, L0);
  k: =0;
  i: =i0;
  repeat
   for l: =1 to Lk do
   begin
     GENERATE (j from Si);
     if f (j) ≤f (i) then i: =j
     else
     if exp ((f (i) -f (j)) / tk) >random [0, 1)
     then i: =j
   end;
  k: =k+1;
  CALCULATE-LENGTH (Lk);
  CALCULATE-CONTROL (tk)
until stopcriterion
end;
```

# 3 模拟退火算法用于求解旅行商问题

## 3.1 解空间与初始解

解空间 $S$ 可表为 $\{1, \cdots, n\}$ 的所有循环排列的集合，即

$$S=\{(\pi_1, \cdots, \pi_n) \mid (\pi_1, \cdots, \pi_n) \text{ 为 } \{1, \cdots, n\} \text{ 的循环排列}\}$$

其中每一循环排列表示遍访 n 个城市的一条回路，$\pi_i=j$ 表示在第 $i$ 个次序访问城市 $j$，并约定 $\pi_{n+1}=\pi_1$. 初始解可选为 $(1, \cdots, n)$

## 3.2 目标函数

此时的目标函数即为访问所有城市的路径长度或称为代价函数，须求其最小值，选为

$$f(\pi_1, \cdots, \pi_n)=\sum_{i=1}^{n} d_{n\pi_i\pi_i+1}$$

注意到约定 $\pi_{n+1}=\pi_1$.

## 3.3 新解的产生

任选访问的序号 $u$ 和 $v$，对调 $u$ 和 $v$ 的访问次序. 此时新路径为 (设 $u<v$)

$$\pi_1 \cdots \pi_{u-1} \pi_v \pi_{u+1} \cdots \pi_{n-1} \pi_u \pi_{v+1} \cdots \pi_n$$

## 3.4 代价函数差

由于这里讨论的问题是对称的，即距离矩阵 $D=[d_{ij}]$ 满足对称性：$d_{ij}=d_{ji}$，所以伴随新解的代价函数的差可由下面的公式计算：

$$\Delta f=(d_{\pi_{u-1}\pi_v}+d_{\pi_v\pi_{u+1}} d_{\pi_{v-1}\pi_u}+d_{\pi_u\pi_{v+1}})-(d_{\pi_{u-1}\pi_u}+d_{\pi_u\pi_{u+1}}+d_{\pi_{v-1}\pi_v}+d_{\pi_v\pi_{v+1}})$$

## 4　算法描述

在该算法中，采用随机交换两城市访问次序的方法产生新解. 冷却进度表中，控制参数 $t$ 的衰减函数设为 $\alpha(t)=\alpha t$，Markov 链长为定长 $L$，当停止准则为连续的 $s$ 个 Markov 链中对路径无任何（优化或暂时恶化的）变动时，即停止算法运行.

模拟退火具有内在的串行性，每一步循环中，状态改变都是在前一步状态的基础上进行的，这种依赖性使得模拟退火算法只能进行部分并行计算. 本算法主要在以下两部分进行并行计算：

1) 在进行初始化时，将计算任意两个城市间距离的过程进行并行处理；

2) 在每一温度 $t$，Markov 链长为定长 $L$. 由于 $L$ 值一般较大，所以将每一温度下的循环进行并行处理. 但在每一循环体内，热平衡过程，即 Metropolis 过程必须串行执行，可通过语句 LOCK 与 UNLOCK 建立临界区实现.

算法描述

取马尔科夫链长度为定长 $L$，Boltzmann 常数 $k=1$，终止参数 bound＝10.

①城市坐标初始化，计算任意两个城市间的距离. // ** 该部分进行并行处理；

②按 3.1 节产生初始回路 $C$；

③1time←1，same←0；1time 为循环次数，same 为终止变量；

④计算回路 $C$ 的费用 $f \leftarrow f(C)$；

⑤如果 1time＞$L$，则转向⑦；否则随机选择两个城市并作交换，得到新回路 $C'$. 计算费用 $f' \leftarrow f(C')$，$f(C')=f(C)+\Delta f$. 其中 $\Delta f$ 按第 2.4 节计算，1time←1time＋1. // ** 该部分进行并行处理；

⑥计算回路费用差 $\Delta f = f' - f$. 若 $\Delta f=0$，则 same→ same＋1，否则 same→ 0.. 若 same＞bound，则终止并输出结果 $C'$；否则，执行 Metropolis 过程决定是否接受 $C'$；若 $C'$ 被接受，则令 $C \leftarrow C'$，$f \leftarrow f'$，并转向⑤. // ** 对判定是否接受建立临界区；

⑦按 $T=\alpha T$ 方案降温，转向③.

## 5　实验结果分析

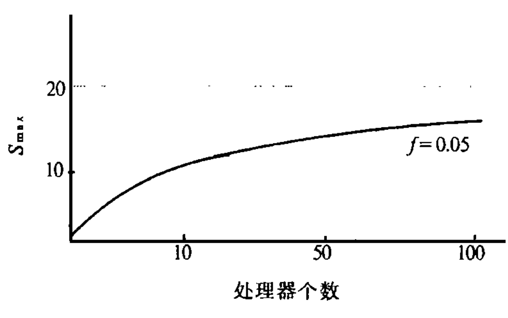
分别利用 12 和 30 个城市进行试验，从程序运行结果看，运行结果正确，但加速比较低，为 1.56. 这有以下两个方面的原因：

1) 不能破坏模拟退火的核心部分，即 Metropolis 过程的串行顺序. 只有维持其串行顺序，才能使解接近全局最优；

2) 代码的串行部分所占比例较大. 并行程序中一个严重限制加速比的因素是代码的串行部分，这可以从如下著名的 Amdah1 定理看出.

**Amdah1** 定理　假定某计算中有 $f$（$f \in [0, 1]$）部分操作必须串行执行，该程序在具有 $p$ 个物理处理器的计算机上运行，则最大加速比为

$$S_{\max}=\frac{1}{f+\frac{1-f}{p}}$$




附图　**Amdah1** 定理图示

当 $p$ 趋于无穷时，表达式值收敛于 $1/f$，这意味着并行程序中的串行码部分会严重限制最大加速比，如附图所示.

## 参 考 文 献

1 唐立山，谢 云，尤矢勇，罗祖华．非数值并行算法．北京：科学出版社，1994

2 Lester B. The Art of Parallel Programming. Englewood Cliffs, New Jersey. Prentice-Hall, Inc., 1993

3 Van Learhoven P J M, Aats E H L. Simulated Annealing: Theory and Applications. D Reidal Publishing Company, 1987

4 郭茂祖，洪家荣．基于遗传算法的示例学习系统的并行实现．计算机工程与应用．1996，(5)

5 倪 南，王晨，张德富．冒险模拟退火算法及其在任务映射上的应用．计算机研究与发展，1996，(3)

# A Parallel Algorithm for the Simulated Annealing Based Traveling Salesman Problem

*Guo Maozu, Hong Jiarong*

**Abstract** The traveling salesman problem (TSP) is shown to be NP-hard and has no algorithm with polynomial time-complexity. In this paper, a simulated annealing-based TSP algorithm is presented and then is implemented in the parallel developing environment——Multi-pascal.

**Key words** simulated annealing algorithm; traveling salesman problem; NP-hard; speedup

(审稿：陈是荣教授，周洪玉教授；编辑：王剑波)