第 5 卷第 4 期
1994 年 4 月

软 件 学 报
JOURNAL OF SOFTWARE

Vol. 5, No. 4
Apr. 1994

用 Boltzmann 网络解任务安排问题*

张德富　　顾卫刚

(南京大学计算机科学系,南京 210008)

摘要　本文提出了一种解任务安排问题的 Boltzmann 网络算法,已在 Transputer 多处理机系统上并行模拟实现,该算法具有优化程度高,运算速度快等特点.

关键词　任务安排,多处理机系统,模拟退火.

任务安排问题是组合优化中的 NP 难题之一. 该问题可推述为:设有 L 台完全相同的机器 $m_1, m_2, \cdots, m_l$,加工 n 个彼此无关的任务 $j_1, j_2, \cdots, j_n$,用一台机器加工它们所需时间分别为 $t_1, t_1, \cdots, t_n$,要求用 L 台机器加工 n 个任务的时间最短.

该问题的确定解不能在用多项式表示的时间内得到,因此人们探求有效的近似算法. 1983 年,Kirkpatrick[1]提出用模拟退火方法解这类组合优化问题,可使解由局部最优达到全局最优. 1989 年,B. H. L Aarts[2]等基于模拟退火算法,提出用 Boltzmann 网络解旅行商问题. 本文基于模拟退火算法,提出用 Boltzmann 网络解任务安排问题,并已在 Transputer 网络上并行模拟实现,取得了较好的结果.

1　用 Boltzmann 网络解任务安排问题

用 Boltzmann 网络解任务安排问题,是在任务安排方案和 Boltzmann 网络结构间建立对应关系,使最大一致性的结构对应于最佳任务安排方案.

定义 1.1. 设有 $L*n$ 维数组 $u=[u_{ij}]$, u_{ij}表示在第 i 台机器上是否执行第 j 个任务,其值为 1 或 0. 数组 t 由各个任务执行的时间 $t_j(1\leqslant j\leqslant n)$组成,则任务安排问题为

$$\min F=\max_{1\leqslant i\leqslant l}(\sum_{j=1}^{n}u_{ij}t_j)-\min_{1\leqslant i\leqslant l}(\sum_{j=1}^{n})u_{ij}t_j)$$

根据以上定义,每个任务由一列向量表示其执行情况,由于每个任务只能在一台机器上执行,故一列向量只能有一个元素为 1. 每台机器上执行的任务由一行向量表示,有几个任务在该机器上执行,则其相应的元素为 1,其余为 0. Boltzmann 网络采用 $l*n$ 个单元,每个单元状态值为 0 或 1. 单元集由 $u=[u_{ij}]$表示, u_{ij}在结构 K 时的状态由 $r_{k(Uij)}$表示. 矩阵 $R=[r_k$

---

* 本文 1991 年 8 月 28 日收到,1991 年 12 月 17 日定稿

本课题得到国家自然科学基金资助. 作者**张德富**,57 岁,教授,主要研究领域为计算机科学技术,并行处理与分布式计算. **顾卫刚**,女,27 岁,1990 年硕士毕业于南京大学,主要研究领域为计算机仿真,并行处理.

本文通讯联系人:张德富,南京 210008,南京大学计算机科学系

(i,j)]表示在结构 K 时的各单元状态. $s(u_{ij}u_{lm})$表示单元 u_{ij}到 u_{lm}之间的连接权值.

定义 1.2. 网络局部一致性定义为在结构 k 时网络某一单元子集中所有被激活连接之和.

则 Blotzman 网络第 i 行的局部一致性函数为:

$$C_{Ki}=\sum_{j,m=1}^{n} S(u_{ij},u_{lm}) * r_k(u_{ij}) * r_k(u_{lm})$$

算法的目标是选择适当的连接权值,使局部一致性最大的结构对应于最佳任务安排方案.

定义 1.3. 结构 K 的 2—邻构定义为结构 K 中第 i 列某两个元素状态发生变化而其余状态保持不变所得的结构.

2 算 法

本文提出的算法已在 Transputer 多处理机系统(图 1)上并行模拟实现.

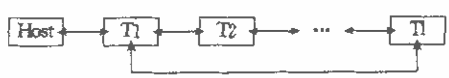

图1

设 l * n 阶矩阵 u,R 分别表示网络当前状态和下一步状态,S 为连接权值矩阵,则数据分布如表 1 所示.

表 1

T1	T2	···	Tl
u[1,j]	u[2,j]		u[l,j]
R[1.j]	R[2,j]		R[l,j]
S[1,j;1,j]	S[2,j;2,j]		S[l,j;l,j]
j=1,···,n	j=1,···,n		j=1,···,n

本文提出的算法如下:

- 接收 Host 送来的数据
- 初始化
- While m<M

  SEQ i=1 For l

  - 计算每个结点上局部一致性函数 C_0
  - 网络上每个结点的 C_0 值相互比较得出最大值 max_0 和最小值 min_0
  - 网络任意两个单元改变状态,并计算此时的局部一致性函数 C_1
  - 各个结点的 C_1 值相互比较,得出最大值 max_1 和最小值 min_1
  - $Delt_0 \leftarrow max_0 - min_0$,

    $Delt_1 \leftarrow max_1 - min_1$
  - 若 $Delt_0 > Delt_1$,则接收状态变化,$m \leftarrow 0$

否则，若 $Delt_0 = Delt_1$，则拒绝，$m \leftarrow m+1$

否则，若 $\exp(Delt_0 - Delt_1)/C > X$，X 为 $(0<X<1)$ 的任意数，则接收 $m \leftarrow 0$

否则拒绝，$m \leftarrow m+1$

· $C \leftarrow C * 0.95$ {end of while}

· 把最终结果送至 Host

算法中 m 表示系统状态到当前没有发生变化所经过的步数，若 m 超过一定值 M，则系统收敛过程结束. C 为控制参数.

图 2 表示了用该算法在 8 台机器上计算 25 个任务的情况，其结果较佳. 各项任务所需时间的平均值为 93.

J1	J2	J3	J4	J5	J6	J7	J8	J9	J10
12	34	23	32	12	43	43	48	23	34

J11	J12	J13	J14	J15	J16	J17	J18	J19	J20
45	32	27	12	36	32	24	43	23	35

J21	J22	J23	J24	J25
28	34	23	12	36

M1	J1 J9 J17 J25	Sum 95
M2	J10 J18 J23	Sum 100
M3	J3 J11 J19	Sum 91
M4	J4 J12 J20	Sum 99
M5	J2 J13 J21	Sum 89
M6	J6 J14 J22	Sum 89
M7	J5 J7 J15	Sum 91
M8	J8 J16 J24	Sum 92

图2

3 算法分析

该算法经过 M 步停止，每步计算包括两个 n 维向量内积，即 n 次乘法和 n－1 次加法，设一次乘法和一次加法为一个计算单元，则每步计算量为 O(n)，由于计算过程的不确定性，总的计算量也不确定.

算法每步的通信分为两个阶段，首先是两个相同列元素变化时其所在的两个单元需要通信，由于这两个单元的选择是任意的，网络采用环状结构，其长度为 l，故最多需 l－1 次通信，最少需一次通信，平均需 l/2 次通信；第二阶段是对 l 个局部一致性函数作比较，再把比较结果广播到各个结点，此时需 2l 次通信，这样每步的通信量为 O(l).

参考文献

1 Kirkpatrick S, Gelalt C D, Jr M P. Vechi optimization by simulated annealing science. 1983;**220(4598)**.

2 Aarts B H L, Korst J H M. Boltzmann machines for TSPs. European Journal of Operational Research. 1989;**39**.

SOLVING TASK ALLOCATION PROBLEM ON BOLTZMANN MACHINES

Zhang Defu and Gu Weigang

(*Department of Computer Science, Nanjing University, Nanjing* 210008)

Abstract This paper presents an algorithm of task allocation problem using Boltzmann machines. This algorithm is simulated parallelly in Transputer multiprocessor system. It is characterized with high optimization degree and computation speed.

Key words Task allocation, simulated annealing, multiprocessor system.