计算机科学 1999 Vol. 26№. 1

人工神经网络并行处理的实现模型

The Implementation Models of the Parallism of Artificial Neural Network

竺 莹 童 频 唐 毅

（上海大学计算机科学系 上海 201800）

Abstract It is found that the key of implementing the parallelism is to implement the parallel computation of summing the build-in weighted-product of Artificial Neuron. In this paper, the characteristics of three main kinds of Artificial Neural Network have been analysed. On the basis of introducing a two-way MAT model, another model, called one-way MAT, has been presented. They are analysed and compared on such respects as structures, mechanisms of running, overheads, advantages and limit.

Keywords Artificial Neural Network, Parallel computation, Implementation model

一、人工神经网络的并行性分析

人工神经网络(Artificial Neural Network，简称ANN)具有并行处理、连续计算等特点，其并行性蕴涵在神经元模型与整个网络两个逻辑层中。

1. **神经元模型的并行性**

ANN的基本组成单元——M-P神经元模型如图1所示。其中：$(x_1, x_2, \cdots, x_n)$是神经元的n维输

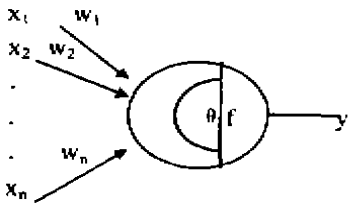

图1 M-P神经元模型

入向量，$(w_1, w_2, \cdots, w_n)$是与之对应的n个连接权。θ是神经元的阈值，输出为y，则：

$$y = f\left(\sum_{i=1}^{n} w_i x_i - \theta\right), i = 1, 2, \cdots, n$$

一般地，神经元响应函数f取Sigmoid函数：

$$f(x) = \frac{1}{1+e^{-x}}.$$

我们知道，神经网络计算的并行性就是体现在大量神经元的并行计算处理上。不仅如此，从图1中，我们可以看到，对一个神经元来说，n维输入向量的各分量与相应权值的乘积是相对独立的，因此神经元内部加权乘积也可以并行执行。这就是实现ANN并行处理的关键。

2. **网络的并行性**

下面分析三类主要ANN的结构、运行机制、并行处理方面的特点。

a. 前馈反传型ANN(Feedforward with Back Propagation ANN，简称FFBP ANN)

①结构特点：由输入层、若干中间层(隐层)、输出层构成的阶层型人工神经网络。

②运行机制特点——有监督学习。网络从输入层开始，层层计算每个神经元的响应值，根据计算出的实际输出值与理想输出值进行比较，得到误差(一般采用累计误差计算法)并调整网络连接权与学习率参数，接着开始下一模式的训练，直至误差减小到期望的范围内。

③并行处理特点：1)同一层上每个神经元在同一时刻具有相同的输入向量，可以并行地通过矩阵运算得到各自的响应值。2)若使用累积误差计算法修正连接权矩阵和学习率，则具有不同输入向量的不同模式，从输入层开始最终得到实际输出值的训练过程只要允许一个时间差的存在，也是可以并行实现。

b. Hopfield ANN

①结构特点：全互连型网络。

②运行机制特点——无监督学习。网络运行时，

每个神经元在同一时刻具有相同长度的输入向量、其状态(响应值)根据一定的工作规则得到更新,最终当网络稳定,即每个神经元的响应值不变或在允许的范围内变化时,网络停止运行。

③并行处理特点:同一时刻,每个神经元接收其余所有神经元的响应值作为其输入向量,通过矩阵运算更新其自身的响应值。网络中各个神经元更新一次响应值的处理可以并行实现。

c. 竞争型 ANN

①结构特点:由输入层、竞争层构成的两层网络。

②运行机制特点——无监督学习。通过竞争层神经元对输入模式的竞争响应,最后得到获胜神经元并将与之有关的连接权朝着更有利于它竞争的方向调整,并对邻近区域的神经元值进行抑制,再开始下一模式的训练。为了获得竞争层获胜神经元,存在一个求最大(小)值的 logN 复杂度的问题(设参与竞争的神经元个数为 N)。

③并行处理特点:1)竞争层每个神经元由矩阵运算求得响应值的过程可以并行实现。2)求获胜神经元时,也可以用成对并行比较方法处理。

二、两种 ANN 并行处理的实现模型

1. 双向 MAT 模型

附加网格树(Mesh of Appendix Trees,简称 MAT)模型是由 Malluhi 等人于 1995 年提出[3]。图 2 是 MAT 模型示意图(以 4 * 4 网络为例),其中,* 表示基本处理器,通常包括局部存储器和只具有赋值、乘法功能的专用运算器;· 表示附加处理器,通常包括局部存储器和只具有赋值、加法功能的专用运算器。基本处理器网络由 N * N(N=4)个处理器构成。每行(列)都附加一棵完全处理器树,每个树根又连着一个行(列)处理器 RAP_i(CAP_i)。由于结构是双向对称的,我们称之为双向 MAT 模型。其网络迭代过程如图 3 所示。

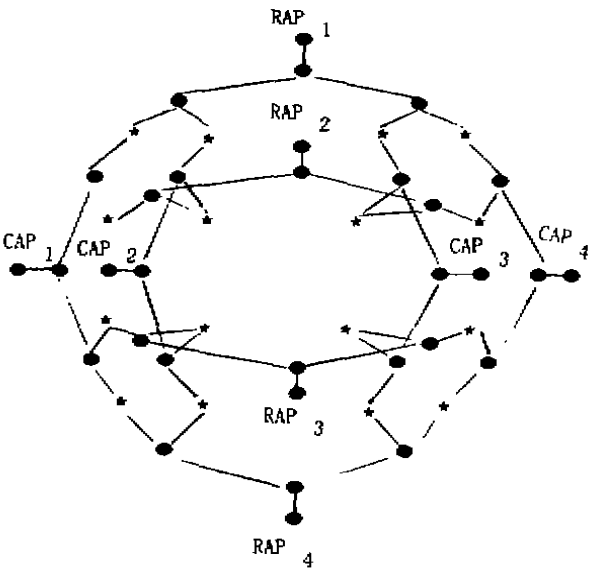

图 2 4 * 4MAT 模型

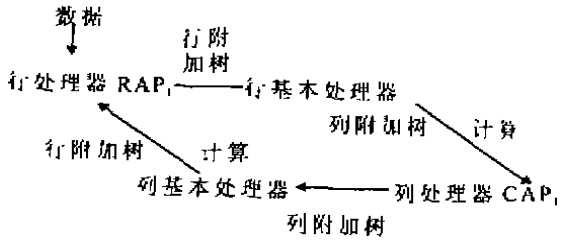

图 3 双向 MAT 反复迭代过程示意图

由于采用流水线技术的多处理器结构特别适用向量处理[4],因此将 k 个模式以流水线方式从列处理器送入双向 MAT 模型进行计算,直至运行完毕,能提高全部模式的网络计算效率,已经证明[2]:

$k = 2\log N + 2$

以竞争型 ANN 为例,设输入层神经元个数为 m;竞争层神经元个数为 N(N>m),建立 N * N 基本处理器网络。N 行表示 N 个竞争层神经元,前 m 列表示 m 个输入层神经元。N * m 的连接权矩阵存在基本处理器网络的局部存储器中。

网络运行过程如下:

(1)从 CAP_j 开始,通过列附加树将各输入分量传递到第 j 列各个基本处理器上。

(2)各个基本处理器内部加权乘积运算。

(3)通过行附加树将每一行的乘积之和加到行处理器 RAP_i 上。

(4)对所有 RAP_i 比较求得获胜神经元,并修改与之相应的连接权值。

(5)反复以上迭代。

采用双向 MAT 模型,所需处理器总数为 $3N^2$ 个,由于必须等到全部竞争层神经元计算输出值后,才能找到最小值,修改相应权值,同时由于不同模式要在调整连接权后才能参与计算,所以模式之间不能采用流水线方式,假设处理器进行一次基本运算(赋值、加法、乘法等)所耗费的时间为 1 个单位时间,则全部模式(设模式数为 p)迭代一次所需的时间为:

$t=p(2\lceil\log N\rceil+3+\lceil\log N\rceil)$

其中 $2\lceil\log N\rceil+3$ 是数据从行(列)附加器传送、计算直到列(行)附加器所需时间，$\lceil\log N\rceil$ 是 N 个竞争层神经元比较得到最大值所需的时间。

双向 MAT 模型同样也能应用于 Hopfield ANN 与 FFBP ANN[3]。

可以看出，双向 MAT 模型能充分地利用对称特性进行网络计算，但是当 N 偏大时，双向 MAT 模型需要的处理器数目比较多，何况对于竞争型 ANN，由于各个模式之间不存在并行性，而且也不能使用流水线方式，因而不太适合用双向 MAT 模型来实现计算，这就是双向 MAT 模型的最大局限。

2. 单向 MAT 模型

在上述双向 MAT 模型的基础上，我们提出了一种称为"单向 MAT"模型。它与双向 MAT 模型有些类似，其示意图如图 4 所示。其中，基本处理器还是 N * N 网络，与双向 MAT 模型不同的是，这里只存在行附加树和 RAP_i，并且每个 RAP_i 与第 I 列处理器相连，迭代过程是单向进行的。这种单向 MAT 模型适合于竞争层神经元数目 N>>输入层神经元数目 m 的竞争型 ANN。

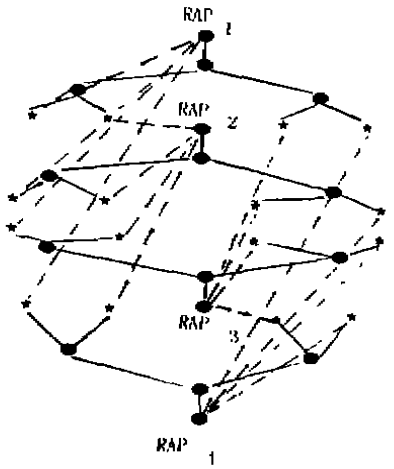

图 4　4 * 4 单向 MAT 模型

取 m * m 个处理器构成基本处理器网格，将竞争层 N 个神经元分成 m 组。每组 $\left\lceil\frac{N}{m}\right\rceil$ 个神经元。这 m 组之间是并行计算的，而每组内部的神经元是以流水线方式送入单向 MAT 模型。当竞争层神经元的各组当前神经元加权和计算结束放至 RAP_i 时，与已经计算过的同组神经元的计算值进行比较，保留最大值，直至各组神经元全部计算完毕，再比较 m 组的各局部最大值，求出 N 个竞争层神经元中的最大值，这一步需要 $\lceil\log m\rceil$ 个单位时间数，再调整相应获胜神经元的连接权值，输入第 2 个模式重复进行。

可以计算出总处理器数为 $2m^2$ 个($<<3N^2$)，其 p 个模式处理总时间为：

$t=p((\left\lceil\frac{N}{m}\right\rceil-\lceil\log m\rceil)+(\lceil\log m\rceil+3)+\lceil\log m\rceil)$

其中，$(\left\lceil\frac{N}{m}\right\rceil-\lceil\log m\rceil)+(\lceil\log m\rceil+3)$ 为每组竞争层神经元以流水线方式进行计算时耗费的时间，$\lceil\log m\rceil$ 为对 m 组局部最大值求得最终最大值的时间，可以看出当 $\left\lceil\frac{N}{m}\right\rceil-\lceil\log m\rceil\approx 0$ 时，并行处理效果最佳。

对于 Hopfield 模型来说，可以证明当受到某种约束条件即 $p<\lceil\log N\rceil$ 且 N 足够大时，用单向 MAT 模型且对 p 个模式采用流水线方式，既可减少处理器数目为 $2N^2$，又能减少处理时间，提高效率。

结论　本文提出的单向 MAT 模型与双向 MAT 模型相比较，前者需要处理器数目较少，而且特别适用于满足 $\left\lceil\frac{N}{m}\right\rceil-\lceil\log m\rceil\approx 0$ 且 N>>m 的竞争型 ANN，当多个模式可以采用流水线方式处理时，用双向 MAT 结构效果较好，但后者需要处理器数目较大。

这两种模型共有的特点是：各处理器皆按固定的拓扑结构连接。因此若用 SCSI 电路制成专用神经网络并行系统，则只能训练某一类 ANN，也就是说，它的通用性不够，今后我们将进一步在如何提高神经网络并行处理的通用模型上深入研究。

参 考 文 献

1　王伟．人工神经网络原理．北京航空航天大学出版社，1995

2　张立明．人工神经网络的模型及其应用　复旦大学出版社，1993

3　Malluhi Q M, et al. Efficient Mapping of ANNs on Hypercube Massively Parallel Machines. IEEE Trans. on COMPUTER, 1995, 44(6): 772～775

4　童颖，程代杰．多处理机及智能多机系统．重庆大学出版社，1998

5　Hopfield I J. Neural networks and physical systems with emergent collective computational abilities. In: Proc. Nat'l Academy of Science, USA 79, 1988