生物序列模式分析中神经网络的并行训练策略*)

王 镝 吴青泉 王国仁 于 戈
(东北大学信息学院 沈阳110004)

摘 要 神经网络作为模式识别、数据挖掘等方面的有效工具,已被广泛应用到生物序列的模式分析中,而生物序列的超大规模、超长同时也给神经网络提出了挑战,即必须解决训练时间过长、效率低下的问题。本文提出了若干适合生物应用的神经网络并行训练策略,并按其神经网络粒度进行分类,同时分析和比较了各种策略的代价。

关键词 生物信息,生物序列分析,神经网络,并行训练

Parallel Training Strategies of Neural Networks for Bio-Sequence Analysis

WANG Di WU Qing-Quan WANG Guo-Ren YU Ge
(College of Information Science and Engineering, Northeast University, Shenyang 110004)

Abstract As an analysis method, neural network has been successfully used in the field of bioinformatics, such as recognition of gene and promoter in DNA sequence, classification of DNA and protein sequences. In the field the bio-sequences are very long and in very large scale, some even up to 6Gbps, and with the rapid development of sequencing technology of the genes, a huge amount of public bio-sequence data are available. The extra huge scale and extra long size of bio-sequences provide some challenges on the neural network. Thus it is very important to reduce the network training time to meet the requirements of bioinformatics and parallel neural network seems one of promising approaches to the problem. In this paper, we have summarized some parallel training strategies of neural network suitable for bioinformatic applications. We also classified them in terms of the granularity of the neural network, analyzed and compared the cost of each strategy.

Keywords Bioinformatics, Bio-sequence analysis, Neural network, Parallel training

1 引言

自从1953年人类首次揭示DNA的结构以来,尤其是随着基因测序技术的快速发展以及人类基因组项目[1,2]在1990年启动以来,人们对基因(包括DNA、RNA、蛋白质)的结构进行了非常深入的研究工作,结构基因学也因之日趋完善。与此相反,由于目前基因序列分析工具的局限性,使得功能基因学的研究大大滞后于结构基因学。功能基因学主要研究基因的遗传性质、遗传特征、基因相似性、基因病变等与人类生活密切相关的许多问题。目前功能基因学的研究工作主要是基于计算机技术来进行的,因此有时也称之为计算生物学。在生物序列中,高的序列相似性通常意味着重要的功能相似性[7],因此可以通过基因的序列来研究基因的功能。目前在世界范围内积累了大量的公开发布的各种基因的生物序列数据,如美国国家生物技术信息中心发布的核酸序列与蛋白质序列数据库GenBank[1],欧洲生物信息研究所发布的EMPL核酸序列数据库[3],欧洲生物信息研究所建立的蛋白质序列数据库[4],人类基因数据库GDB[5],果蝇基因数据库[6]等。而且很多生物序列数据是非常长的,例如在公开发布的DNA生物序列中最大长度已经超过6Gbps[8,9]。

神经网络作为模式识别、数据挖掘等方面的有效工具,已被广泛应用到生物序列的模式分析中。例如,在蛋白质二级结构的预测问题中,神经网络通过学习现有的氨基酸序列与二级结构间的关系,可以对蛋白质的二级结构做出预测[10,11]。在蛋白质分类中,神经网络也有许多应用,如对不同起源(如人类、马、鸟)的某种感冒病毒的分类[3],根据蛋白质序列特征对贝叶斯神经网络进行训练,完成训练的神经网络可对特定蛋白质进行识别[14]等等。启动子是DNA序列中能与DNA聚合酶相互作用的片段,生物信息学中常常要对启动子进行分析,在神经网络中利用启动子的保守性来识别DNA序列中的启动子[16,17],并可以用神经网络的剪枝学习方法来改良真核细胞启动子在原始DNA中位置的判定[15]。在启动子分类识别中,可用贝叶斯神经网络作为分类器,用启动子的特征训练神经网络,完成训练后可以对特定的启动子进行分类[18,19]。相比其它方法,神经网络在这些应用中,具有高效准确的特点。但在生物序列分析中,特别是DNA序列分析中,DNA序列超长,其中又包含GC含量、启动子、重复序列和基因等大量特征[22],神经网络结构复杂,训练集庞大,往往需要很漫长的时间才能完成神经网络的训练。因此,在这些应用中,神经网络如果串行训练,会存在很明显的效率问题。而从神经网络的本质上讲,人工神经网络是要模仿人类大脑大量简单神经元细胞互连而获得的强大分析计算能力。人脑的这种能力则是各神经元并行计算达到的[23]。自然地,我们可以考虑用并行策略提高生物序列分析中神经网络的训练效率。

实现神经网络的并行有两类主要方法[27],一是建立专门的VLSI硬件系统,实现并行神经网络,即神经元计算机;二是开发并行计算机系统上的神经网络软件。本文仅讨论第二类方法。并行计算机系统可以根据处理机与主存、外存的关系

*)本文受国家自然科学基金项目资助(60273079)。

分为三类[28]：①共享主存结构，所有处理机可以直接访问主存和所有外存；②共享外存结构，每个处理机拥有私有主存，但它们可以直接访问所有外存；③非共享结构，每个处理机拥有私有的主存和一个或多个外存，各处理机间的通信是通过互联的高速网络实现的[28]。前两种结构在近年虽然有了一些新发展，如基于共享外存的 KSR1并行机[27]，但它们实现复杂，需要较多的编程接口，规模可扩展性差[28]，而非共享结构的并行系统具有较好的性能加速比和规模可扩展性，因此本文讨论的并行神经网络也是基于这种结构的。在非共享结构的并行网络环境中，网络的拓扑结构有多种，如总线结构、环状结构、网状(mesh)结构等[28]。本文的讨论基于两种代表性结构：有主机(master)的总线结构(主/从模式)和无主机的非总线结构。

在神经网络的传统应用中，针对特定的需求，已经建立了一些并行神经网络，并得到成功的应用。如基于 MultiSpert 并行体系建立的用于语音识别的并行神经网络，并行化使原来需要几周完成的训练过程缩短到几天[25]，以及用于自动导航、手写体识别[23]等的并行神经网络，它们在缩短训练时间，提高效率上显示出强大的优势。在这些应用中，提出了大量的神经网络并行训练策略，而每种策略适用的应用领域不尽相同，因此，我们有必要了解和比较各类并行策略，从而选择最优的，或根据实际问题改进现有策略或提出自己的新方法，将它们应用于生物序列分析中。本文正是基于此目的提出的。

2 神经网络的并行策略

目前，为实现并行神经网络提出过许多不同的策略。这些策略可根据神经网络的分解粒度，主要分为三个层次上的神经网络并行[24,34]。(1)模式并行[24~26,34]。这种方法利用已给定的大规模的训练模式集作为并行的资源，它并不利用神经网络自身的并行能力。通过将训练模式集分片到各处理机上来实现神经网络训练时间的降低。(2)网络并行[24~26,34]。这种方法将神经网络分块，而后将各分块分配到各处理机上训练。(3)神经元并行[24]这种方法把神经元看成可并发处理的单元，随机放在各处理机上训练。这种方法是不可行的，尤其是大型网络神经元的个数远远大于网络中处理机的个数[32]。下面将应用较为广泛的反向传播算法(BP 算法)的训练规则描述前两个层次上的并行神经网络实现策略。

2.1 模式并行

训练模式是指神经网络的输入与目标输出对，训练集是由一组训练模式构成的。模式并行(Pattern Parallel)神经网络是指将训练集中的训练模式均匀地分布在多个并行节点上。如图1和图2所示，模式并行策略在每个并行节点上保存完整的神经网络结构，将所有训练模式平均分配到网络各节点上。无论基于哪种网络结构，模式并行都是每个并行节点上的每个模式训练后，将按误差梯度调整的各层间的连接权值告知主机或网络的其他节点。

文[25]引入了模式束的概念，即在各节点将模式分成一定大小的组，每组即为一束。各节点上束的大小(一束中含有的模式数)相同。累计一束中各模式的误差梯度，待一个模式束输入完毕，根据累计所得的梯度调整一次各层间的连接权。若束的大小为 m，则网络的通信量将减少 m－1/m (相对于每个模式都要通信)。同时，这种思想在非并行网络训练中为加快网络训练已经提出过[29]，当束的大小得当时，模式束的方法并不影响神经网络的收敛及准确性。束的大小可以通过实验得到，也有一些经验值[30]。

对于基于主/从方式的模式并行网络[25]，如图1，当各节点上的神经网络完成一个模式束的训练后，将权值的调整量传送给主机，主机计算求和，并将最终的调整值广播给各并行节点，当各节点完成全部模式的训练后，各节点上的神经网络是相同的。

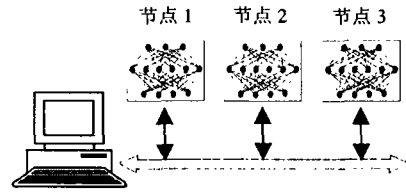

图1 总线结构的模式并行

对于基于非总线方式的模式并行神经网络[24]，如图2，当一个节点计算出一束的误差梯度后，调整连接权，并将连接权的改变量广播给其它节点，当一个节点接到所有其它节点的连接权调整信息后，累加计算神经网络的权值，调整完毕后各节点上的连接权是相同的。整个网络训练结束后各节点的神经网络亦是相同的。

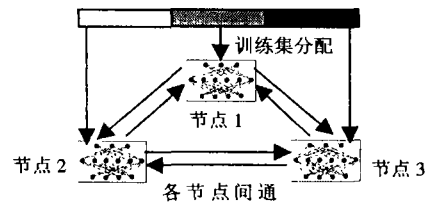

图2 非总结结构的模式并行

2.2 网络并行

网络并行将神经网络分块，每一部分保存在一台处理机上。这种方法是根据神经网络的自身特点提出的[24]，神经网络的每一层均由可并行操作的神经元组成[29]，因此可以考虑将同层神经元放在不同的处理机上[25]；另一方面，对于多层神经网络可通过管道的方法使各层并行训练[32]。当然在应用中亦可以将这两种方法相结合。在这里根据上面提到的两个方面，分别构造神经网络并行训练模型，如图3、4所示。图3中各节点保存神经网络输入层和输出层的完整结构，而将隐含层节点分别放在不同的处理机上[25]，图4中按 BP 网的三层结构[24]，将输入层、隐含层、输出层分别放在不同的处理机上。

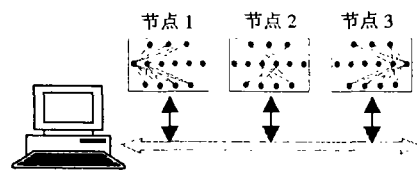

图3 总线结构的网络并行

图3是基于主/从方式的一种网络并行策略，当然在图2所示的网络结构中也可实现，与前述的模式并行策略相比，每个节点必须保存全部训练模式。在各节点上同时用相同的模式计算，得到输出后，各节点将输出传送给主机。主机读入所有节点的输出结果，求和计算出对于一个模式的全局输出，并得到误差向量，再将此误差向量广播，各节点根据得到的误差调整各自的连接权。由于计算连接权的调整量时，必须先得到误差向量，因此处理机需要等待主机广播的误差向量。这里，为

了提高各处理机的利用率，可以采用管道的方法。即在等待误差向量时，先计算下一个模式的输出，并传送给主机，在等待下一模式的误差向量时，计算本次权值的调整量，以此类推。由于权值每次的调整量非常小，这种延时更新的影响相对于训练的正确性与收敛性可以忽略不计。

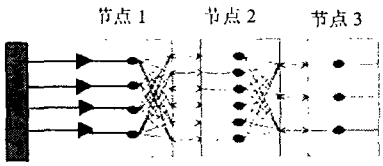

图4　非总线结构的网络并行

图4是基于非总线方式的一种网络并行策略。保存输入层神经元的处理机首先开始基于输入数据的计算，将输出传送给保存下层神经元的节点，当到达保存输出层神经元的节点后，开始计算误差并反传，即把误差向量传给当前节点的前层节点。如图所示，节点1计算隐含层的输入，节点 2计算输出层的输入，节点3计算神经网络的输出及误差梯度并从节点3经节点2到节点1反传误差向量调整连接权。在此方法中由于每个节点要等待与它通信的节点的结果，我们在这里采用管道技术，实现神经网络的并行训练[32]，如图5。同时，在2.1节模式并行中提出的模式束的方法在网络并行中也是适用的。

3　并行策略的比较

上面描述了基于粒度的神经网络并行训练策略，下面从神经网络的计算代价、通信代价及训练完成时间等方面，对各策略进行比较。为了使分析问题简单，假设神经网络有 L 层（包括输入层），每层的神经元个数相同，为 M，每个神经网络中的连接权数目为 W，则 $W=(L-1)\cdot M^2$。假设并行网络中有 P 个节点（总线结构中不包括主机），训练集中模式的个数为 N。由于神经网络的训练复杂，并不是 N 个模式，通过 N 次训练即可完成，因此，比较各策略对某个模式训练一次的代价。为了分析的方便，设定一些变量：

·计算代价：假设神经网络对一个模式训练并修改的代价为 C，其中前向传播代价为 C_f，计算并反传误差调整权值的代价为 C_b，有 $C=C_f+C_b$。两权值、误差等进行一次加、减算术运算的代价为 C_s。

·通信代价：假设神经网络的每个权值、输出等的长度相同，为 ω，且网络传送 ω 长度的代价为 g。

·同步代价：对于非总线网络，要引入同步机制，这里是这样规定的：①每个节点用本地数据进行计算；②节点间通信；③同步，各处理机的本地存储器得到数据后同步。同步一次的代价为 l。

同时，需要指出的是在下面的代价分析中，并没有把将训练模式传送到各处理机的通信代价计算到上面提到的通信代价中。

3.1　模式并行策略的比较

对于总线结构的并行模式，P 个节点上同时训练 P 个模式，各节点的计算代价和为 P·C，然后各节点将连接权的调整量传送给主机，通信代价为 W·g·P，主机的计算代价为 $W\cdot(P-1)\cdot C_s$，主机将最后的权值调整量广播的通信代价也为 W·g·P。平均来说，训练一个模式一次的代价为：计算代价 $C+(W\cdot(P-1)\cdot C_s/P)$，通信代价为 2W·g。

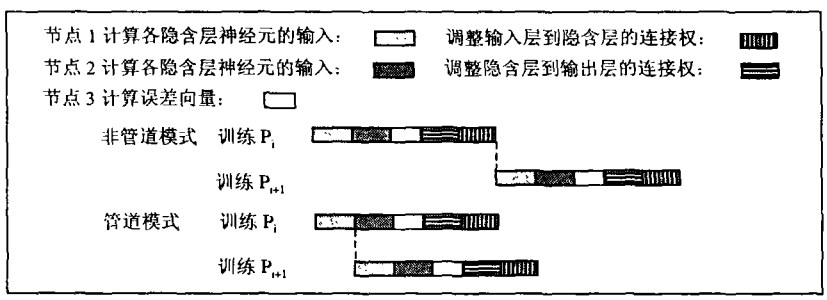

图5　有无管道的对比

对于非总线模式，与总线模式类似，首先各节点的计算代价为 C，而后通信的代价为 W·g(P−1)，各节点同步，代价为 l，各节点再进行计算得到最后的权值，代价为 $W\cdot(P-1)\cdot C_s$。因此，训练一个模式一次的代价为：计算代价 $C+(W\cdot(P-1)\cdot C_s)$，通信代价为 W·g(P−1)，同步代价为 l。

可见，由于总线结构是在主机上计算最终的权值，而非总线结构是每个节点各自计算，因此总线结构的计算代价相对较小。但需要指出的是，在完成时间上并没有明显的区别。在通信代价上总线结构与非总线结构的比为 2/(P−1)，可知，随着网络中节点的增加，非总线结构的通信代价要远大于总线结构。同时，非总线结构中还要付出同步代价，这在总线结构中是不需要的。

这只是评价模型的一部分特征，我们并不能因此得出总线结构更好的结论。一个主要的原因是总线结构有其自身的缺陷，主机是总线结构的瓶颈，这是由于其结构特点决定的。

3.2　网络并行策略的比较

网络并行是通过将神经网络分块来实现的。相对于模式并行，某个模式训练一次也就意味着还有 P−1个模式也完成了一次训练不同，网络并行中，各节点计算通信一次仅完成对一个模式的一次训练。同时，网络并行的方法很多，前面提到的仅是众多方法中最为基本的两种，根据具体问题可以提出很多不同的实现方法。这里，仅对前面提出的方法做出代价分析。

对于总线结构的网络并行，把神经网络隐含层的神经元平均分配到 P 个节点上，假设 M 可以被 P 整除（这不是必需的，这里仅是为说明问题方便）。在某个模式的一次训练过程中，每个节点首先完成 C_f/P 的计算量，而后进行通信，即各节点将神经网络的输出传送给主机，每个节点的通信代价是 M·g，总的通信代价为 M·g·P。主机的计算代价是主机将收到的各节点的神经网络输出相加，得到完整的神经网络的输出，代价为 $M\cdot(P-1)\cdot C_s$，然后主机再将计算所得的误差广播给网络的各节点，通信代价为 M·g·P，各节点根据得到的

误差计算误差梯度并调整权值，代价为 C_b/P。因此，一个模式训练一次的总代价为：计算代价($C_f/P+C_b/P$)·P+M·(P−1)·C_s，即 C+M·(P−1)·C_s，通信代价2M·g·P。

对于非总线模式，前面给出了最简单的方法之一。神经网络的每层神经元放在一个网络节点上，也可以称作层并行，基于此条件，网络中的节点数与神经网络的层数相同。显然，对于一个模式的一次训练网络中各节点的计算代价和为 C。整个网络在训练过程中，存储前层神经元的节点将计算所得的后层神经元的输入传送给保存下层神经元的节点，此过程的通信代价为(P−1)·M·g。在保存输出层神经元的节点计算完误差后，将误差梯度传送给保存前层神经元的节点，由此节点计算调整连接此两层神经元的权值，直至保存输入层神经元的节点。此过程的通信代价亦为(P−1)·M·g。因此，总代价为：计算代价 C，通信代价2(P−1)·M·g。此处可以不需要同步，但正如已经指出的这是一种最简单的实现策略，如果将同一层的神经元保存在多个网络节点上，模型就需要同步，同时通信代价也会大量增加。

这两种网络并行策略，在通信代价上相差不大，但正如前面提到的若层并行中的同层神经元放在多台处理机上，通信代价将增加，同步代价也会出现。尤其是当隐含层神经元远多于输出层神经元时，为了保证各台处理机上负载平衡，必然会出现刚刚提到的情形。在计算代价方面，总线型结构需要更多的计算代价，但这部分计算代价与 C 相比是非常小的。同时这种网络当隐含层神经元非常多时，仅是增加网络的节点数或增加每个节点上的隐含层神经元数。不会出现网络负载不平衡的问题。而且，若仅增加在每个节点上的隐含层神经元个数，则通信代价不会增加。

3.3 模式并行与网络并行的比较

(1)在前述的分析中，有假设 $W=(L-1)\cdot M^2$，因此模式并行的通信代价要大于网络并行的通信代价。

(2)总线结构的网络并行策略与模式并行策略相比，一个缺点是在每个节点上必须保存完整的训练集，这需要大量的存储空间。

(3)层次并行与其它并行策略相比会出现负载不平衡的问题。

(4)网络并行在每个节点上仅保存整个神经网络的一部分，因此适合训练规模大的神经网络。

(5)网络并行和模式并行都可以采用模式束的方法，网络并行为了提高各节点利用率可以采用管道技术，这在模式并行中不存在。

(6)在总线结构中，如果把训练集从主机传送到各节点的代价也作为通信代价的一部分，那么在这一点上网络并行的通信代价是模式并行的 P 倍。神经网络的训练集的规模一般是非常大的，尤其是那些大规模问题，因此这部分代价也是非常高的。

表1对各并行策略进行了综合比较。从并行环境看，总线结构存在着一定的瓶颈，但当网络节点增多时，总线结构的通信代价增加不是很明显，而非总线结构的通信代价显著增加(可参见前两个小节中的代价分析)，同时还需要进行同步。从两类并行策略的角度看，模式并行的代价要大于网络并行，在神经网络规模不是很大的情况下这种差距并不明显，在大规模的网络中网络并行才显示出它的优势。而在中小型神经网络中，网络并行对于加快完成训练的效果并不明显，相反，模式并行尤其是总线结构的模式并行(网络节点多时通信代价小)在中小规模的网络中优势显著。

表1 各并行策略的综合比较

并行策略 \ 性能		计算代价	通信代价	同步问题	负载问题	模式束	管道	网络结构瓶颈	适用神经网络规模	节点增加引起代价增加
模式并行	总线	较大	较大	无	小	支持	不可	大	中小	小
	非总线	大	大	有	小	支持	不可	小	中小	大
网络并行	总线	中等	较小	无	小	支持	可以	大	大规模	小
	非总线	较小	较小	可能有	大	支持	可以	小	大规模	大

结论 本文描述了生物信息领域的新发展及神经网络在此领域中的应用，综述了在并行网络环境中，基于不同网络结构的，可用于生物序列分析的神经网络并行训练策略的实现方法，并对各实现策略做了简要的分析。

近年，随着测序工作的蓬勃发展，研究者可以在全球范围内获得大量的研究数据。神经网络方法被经常地用于这些数据的分析处理中。尤其当前的一些应用中，如蛋白质二级结构预测，为了进一步提高准确率，研究人员采用了神经网络集合(Neural network ensemble)[33,34]方法，这就需要训练大量的神经网络。当面对大量的数据，需要训练的多个网络，以及大量可获得的计算机资源时，选择合适的训练策略，进行并行训练，能够很好地提高工作效率。

参 考 文 献

1 The Human Genome Project (HGP). http://www.nhgri.nih.gov/HGP/

2 Collins F, Patrinos A, Jordan E, Chakravarti A, et al. New goals for the us human genome project: 1998-2003. Science, 1998, 282(5389): 682～689

3 European Bioinformatics Institue. EMPL at EBI. http://www.ebi.ac.uk

4 European Bioinformatics Institute. SWISS-PROT Database. http://www.ebi.ac.uk/swissprot

5 The Genome Database. http://www.gdb.org

6 FlyBase: the Database of the Drosophila Genome. http://www.flybase.org

7 Gusfield D, Stoye J. Linear time algorithms for finding and representing all tandem repeats in a string: [Tech Report CSE-98-4]. Dept of Computer Science, University of California, 1998

8 National Center for Biotechnoloy Information. GenBank Overview. http://www.ncbi.nlmnih.gov/Genbank/Genbank Overview.html

9 2E. Hunt. PJama stores and suffix tree indexing for bioinformatics applications. In: Proc. of ECOOP'00, 2000

10 Qian N, Sejnowski T J. Predicting the secondary structure of globular proteins using neural network models. Mol. Biol, 1988, 202: 865～884

11 Genamics Expression, Predicting the secondary structure of you rprotein. http://genamics.com/expression/strucpr ed.html

12 Nagl S. Neural network models of protein domain evolution. Int. Journal for Philosophy of Chemistry, 2000, 6(2): 143～159

13 Campitelli L, Delledonne L, Salvini A. A neural network approach to protein sequence processing. http://www.es.isy.liu.se/norsig2000/publ/page283_id090.pdf

(下转第178页)

该函数和对象定义在该类所在的模块中，因为 FORTRAN 2000的访问机制是对应于模块而言的。

```
class c_example {
  public:
    void operate();
}
...
c_example example;
example.operate();
```

(a)C++对成员方法的调用

```
module f_example
  type f_type
       procedure,pass(are)::operate=>op
  end type f_type
  interface operate
       module procedure op
  end interface
  contains
     subroutine op(arg)
        type(f_type),intent(in)::arg
     end subroutine op
  end module f_example
  ...
  use f_example
  type(f_type)example
  !第一种调用成员方法的方式
  call operate(example)
  !第二种调用成员方法的方式
  call example.operate
```

(b)FORTRAN 2000对成员方法的调用

图2 C++与 FORTRAN 2000类操作调用的比较

结束语 FORTRAN 语言是一门历史悠久的语言，有很强的数值计算能力，它曾经在科学计算、工程问题和企事业管理中发挥巨大的作用。但是今年来随着面向对象技术的流行，越来越多的人认为现在最为流行的面向对象语言 C++有可能替代 FORTRAN。本文在此环境下向人们介绍了 FORTRAN 2000的面向对象机制，并把它与 C++语言的面向对象机制作了比较。从中我们可以看出，FORTRAN 2000已经具有了一定的面向对象特征，基本上可以从面向对象的角度来对问题进行分析并编程实现。由于 FORTRAN 语言在面向对象机制上起步较晚，FORTRAN 2000与 C++语言相比其面向对象机制还不完善。但 FORTRAN 语言凭借其在数值计算上的高效性和稳定性，它在工程应用中的地位还是无可替代的。如何完善 FORTRAN 语言的面向对象机制同时又保持其在数值计算上的优势是 FORTRAN 语言今后发展的方向。

参 考 文 献

1 Reid J. The New Feature of FORTRAN 2000. ISO/IEC JTC1/SC22/WG5 N1495

2 ISO/IEC 1539. WORKING DRAFT,J3/02-007R3,Sep. 2002

3 Liang Xianzhong, Wang Zhenyu. Ada-based Support for Abstraction, Encapsulationg and Unit Hierarchy. In: Proc. of TRI-Ada'91,San Jose. New York: ACM Press ,Oct. 1991

4 Booch G. Object-Oriented Development. IEEE Trans,1986,SE-12(2)

5 Sheidewitz E. Object-Oriented Programming in Smalltalk and Ada. ACM OOPSLA'86 Proc. ,1988

6 Liang Xianzhong. GRASIS: Graphical Ada-based Specification and Implementation Support for Object-oriented Development. CSRDA (China Ship Research and Development Academy). Thesis of MSc (in Chinese),Mar. 1988

(上接第133页)

14 Ma Q, Wang J T L, Shasha D, Wu C H. New techniques for extracting features form protein sequences. IBM Systems Journal (Special Issue on Deep Computing for the Life Sciences),2001, 40(2):426～441

15 Reese M G,Eeckman F H. Novel Neural Network Algorithms for Improved Eukaryotic Promoter Site Recognition. In: The 7th Int. Genome Sequencing and Analysis Conf. 1995

16 Neural Network Promoter Prediction: Input. http://www.fruitfly.org/seq_tools/promoter.html

17 Knudsen S. Promoter2.0: for the recognition of PolII promoter sequences. Bioinformatics, 1999,15(5): 356～361

18 Ma Q,Wang J T L,Gattiker J R. Mining biomolecular data using background knowledge and artificial neural networks. Handbook of Massive Data Sets, Kluwer Academic Publishers, 2002

19 Ma Q, Wang J T L, Shasha D, Wu C H. DNA sequence classification via an expectation maximization algorithm and neural networks: a case study. IEEE Transactions on Systems, Man, and Cybernetics, Special Issue on Knowledge Management, 2001

20 Noordewier M O,Towell G G,Shavlik J W. Training Knowledge-Based Neural Networks to Recognize Genes in DNA Sequences. Advances in Neural Information Processing Systems, 1993. 530～536

21 Wu C H. Gene classification artificial neural system. Neural and Biological Systems, May 1995. 102

22 黄小兵，陈锋，等. 1%人类基因组数据库系统. 计算机科学，2002，29(8. 增A):273～276

23 Misra M. Prarllel environment for implementing nerual networks. Neural Computing Surveys, 1997,1:48～60

24 Rogers R O,Skillicorn D B. Using the BSP cost model for optimal parallel neural network training. Technical report of department computing and information science, Queens University, Dec. 1997

25 Färber P,Asanovic K. Parallel neural network training of Multi-SPERT. http://cag-www.lcs.mit.edu/～krste/papers/MStrain.ps

26 Färber P. Quicknet on MultiSpert: fast parallel neural network training:[Technical Report TR-97-047]. International Computer Science Institute, Dec. 1997

27 Coetzee L,Botha E C. Parallel neural net traininig on the KSR1. Concurrency - Practice and Experience, 1996,8(8): 617～638

28 DeWitt D J,Gray J. Parallel database systems: the future of high performance database processing. ACM, 1992,36(6)

29 Hagan M T,Demuth H B,Beale M H. 戴葵等 译. 神经网络设计. 北京:机械工业出版社,2002

30 Rogers R O, Skillicorn D B. Batch size and training times in supervised and unsupervised networks, Dec. 1997

31 Färber P. Parallel computing on MultiSpert:[Technical Report TR-97-046]. International Computer Science Institute, Dec. 1997

32 Ahmad A S, Zulianto A,Sanjaya E. Design and implementation of parallel batch-mode neural network on parallel virtual Machine. Proceedings, Industrial Electronic Seminar, 1999

33 Opitz D, Maclin R. Popular Ensemble Mehtods: An Empirical Study. Journal of Artificial Intelligenc Research,1999,11:169～198

34 Riis S K,Krogh A. Improving Prediction of Protein Secondary Structure using Structured Neural Networks and Multiple Sequence Alignments. Journal of Computational Biology,1996,3:163～183