

基于异构系统的生物序列比对并行处理研究进展

朱香元¹ 李仁发² 李肯立^{2,3} 胡忠望¹

(肇庆学院计算机学院 肇庆 526061)¹ (湖南大学信息科学与工程学院 长沙 410082)²

(国家超级计算长沙中心 长沙 410082)³

摘要 序列比对工作属于生物信息学的基础性研究领域。由于它具有应用广泛、计算复杂以及海量数据等特点,加之现在高性能计算的兴起,使得近年来序列比对并行处理技术快速发展。首先介绍了序列比对领域高性能计算的新进展,接着从体系结构特征入手对其研究进行分类,并对每类方法的实现细节和性能进行分析比较,从中不难看出访存控制、同步、数据交互以及算法可扩展性等问题均为目前基于异构系统的序列比对并行处理研究的关键点。最后,对该领域的未来研究方向进行了展望。

关键词 序列比对,并行处理,异构,GPU,FPGA,Cell BE,MIC

中图分类号 TP301 **文献标识码** A

Advances in Biological Sequence Alignment Parallel Processing Based on Heterogeneous Systems

ZHU Xiang-yuan¹ LI Ren-fa² LI Ken-li^{2,3} HU Zhong-wang¹

(School of Computer Science, Zhaoqing University, Zhaoqing 526061, China)¹

(College of Information Science and Engineering, Hunan University, Changsha 410082, China)²

(National Super-Computer Center in Changsha, Changsha 410082, China)³

Abstract Sequence alignment is a fundamental operation in Bioinformatics. In recent years, there have been extensive studies and rapid progresses in biological sequence alignment parallel processing technologies due to its wide use, high computational complexity and large-scale data. This paper first analyzed the new development of high performance computing in sequence alignment, and then classified the up-to-date researches based on the applied architectures. Their implementation and performance were compared and analyzed in detail. It was pointed out that problems such as memory access control, synchronization, data transferring and scalability of algorithms are the key techniques to the study of biological sequence alignment parallel processing. Finally, some future directions of research were given.

Keywords Sequence alignment, Parallel processing, Heterogeneity, GPU, FPGA, Cell BE, MIC

生物信息学中,序列作为生物信息的载体,通常包括DNA(脱氧核糖核酸)、RNA(核糖核酸)和蛋白质。序列比对(Sequence Alignment)是以序列为研究对象,通过比较序列中各字符间的对应关系或字符的对比排列,发现序列之间的相似性,辨别序列之间的差异,从而推测其结构、功能以及进化上的联系。序列比对是生物序列分析领域最重要的研究方向之一,已被广泛应用于进化分析、功能预测、相似性搜索、生物制药、疾病诊断与治疗等方面^[1,2]。

在高吞吐量的基因时代,随着大规模测序工作的日益成熟,序列比对的方式已发生巨大变化。过去几十年,比较一条给定基因的所有直系同源基因仅需要与几千条序列进行比对;而在未来几十年,可能需要比对几十亿条相关联的序列,这使得序列比对越来越复杂、耗时,现有的一些序列比对算法已经不适用于处理大规模序列数据。另一方面,多核计算机^[3,4]、集群^[5-8]、云计算^[9]和超级计算机等高性能计算平台

的快速发展给高效处理大规模数据带来了巨大的机遇和挑战。序列比对应用的广泛性、计算的复杂性以及海量的数据特征,对计算机性能提出了越来越高的要求,生物序列比对迫切需要高性能计算的支持,序列比对并行处理研究已成为生物计算和计算机科学领域具有相当重要性和迫切性的研究内容。

20世纪90年代,GPU(Graphic Processing Unit)^[10]、FPGA(Field Programmable Gate Array)^[11]、Cell BE(Cell Broadband Engine)^[12]和众核芯片^[13,14]等的快速发展,使得序列比对并行处理逐渐成熟异构混合体系结构采用通用多核微处理器CPU与定制加速协处理器(GPU、FPGA等)相结合,注重发挥非传统体系结构的优势,由CPU完成序列比对流程控制工作,协处理器并行地处理数据密集型的序列比对计算。目前序列比对综述研究主要包括:(1)阐述序列比对方法、分类及经典算法;(2)介绍GPU、FPGA等技术;(3)分析经典S-W

本文受国家自然科学基金重点项目(61133005,61432005),国家自然科学基金青年基金(61402400),广东省自然科学基金(2014A030310288)资助。
朱香元(1974—),女,博士,主要研究领域为并行算法与体系结构、高性能计算、计算生物学,E-mail:hnzxy@hnu.edu.cn;李仁发(1957—),男,博士,教授,主要研究领域为嵌入式计算、无线网络、网络与数字媒体;李肯立(1971—),男,博士,教授,主要研究领域为并行与分布式处理、生物计算机、组合最优化;胡忠望(1965—),男,教授,主要研究领域为计算机网络与安全。

序列比对算法^[2]基于 GPU、FPGA 异构平台的并行处理；(4) 从性能、功耗、可扩展性等方面对序列比对并行处理技术进行比较, 总结未来的研究方向。然而, 当前高性能计算能力与计算机体系结构正处于快速发展与变革期, 随着 2013 年我国设计与制造的 CPU+MIC(Many Integrated Core)异构体系结构的超级计算机“天河二号”成为世界上最快的计算机, 各种新型并行体系结构不断涌现。现有研究未能对序列分析领域高性能计算发展现状进行全面总结、分析与比较, 对并行处理用于解决序列比对问题时不同程度上遭遇的协同性差、性能优化及可扩展性等问题的论述不够; 且现有研究仅论述了 S-W 算法在相应平台上的并行处理方法, 未涉及近几年权威学术会议及刊物讨论的其他成果。

本文总结了基于异构系统生物序列比对并行处理技术的最新研究成果。第 1 节阐述序列比对领域高性能计算的新发展。第 2 节对序列比对并行处理技术进行分类, 通过比较分析得出实现异构系统序列比对的难点。最后总结全文并针对未来可能的发展与研究方向给出预测和建议。

1 序列比对领域高性能计算新的发展

近年来, 随着人们对生物序列认识的逐渐加深, 越来越多的蛋白质三维结构及其功能被人们所认识, 新的序列结构信息、功能信息、进化关系等也加入到序列比对模型中。作为生物信息学领域重要的基础性研究工作, 序列比对具备的海量生物数据、高计算复杂性等计算特征对计算机性能提出越来越高的要求, 采用高性能计算平台实现对海量生物序列比对的高效处理, 已成为生物信息学和计算机学科的重要交汇点和挑战性课题。

虽然基于通用微处理器结构的计算平台能够提供很强的峰值计算能力, 但由于在计算和存储方面的固有局限性, 通用平台不能较好地满足序列比对领域的计算需求, 实际应用效率较低。在计算方面, 随着生物数据库的信息量每年以指数方式增长, 对序列比对所需要的运算时间、精度和存储提出了

更高的要求, 通用计算平台难以提供灵活、高效的支持。在存储访问方面, 通用计算平台大多采用固定的存储访问控制方式, 难以适应序列比对领域多维度的访存请求, 导致访存效率低下。

异构体系结构顺应了超大规模集成电路芯片容量快速增长的趋势, 同时能够满足应用程序特征日趋多样化的发展需求。如图 1 所示, 异构体系结构是在一个计算系统结构内集成多个不同结构的处理器, 其中通用处理器适合完成较复杂的控制和事务管理任务, 协处理器集成了大量计算和存储资源, 适合作为面向特定应用的算法加速器, 执行大规模数据密集型的并行计算任务^[11]。

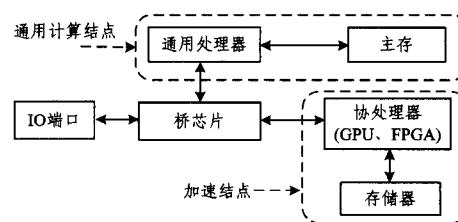


图 1 异构系统体系结构

2 基于异构系统的序列比对并行处理

2.1 基于 GPU 的并行序列比对

GPU 是高并行度、多线程、拥有强大浮点计算能力和极高存储器带宽的多核处理器。其强大的浮点运算能力, 不仅提升了图形计算效率, 还大大拓宽了在通用计算 GPGPU (General Purpose GPU) 领域的道路。自 2006 年最早提出 GPU 加速 S-W 算法以来, 研究人员一直致力于 GPU 在序列比对领域的应用研究, 如表 1 所列。目前基于 GPU 的序列比对方法研究主要集中于 S-W、MUMmer 和 BLAST 算法。除文献^[27]采用 AMD GPU 以外, 其他均采用 NVIDIA GPU。使用的编程语言主要有 GLSL (GL shading language)、Brook+ 和 CUDA。可分为多 GPU、单 GPU 并行序列比对方法。

表 1 基于 GPU 的并行序列比对方法

文献	软件名称	GPU 型号	待比序列数(长度)	搜索数据库(序列)	编程语言	参照算法	加速比
[15]	S-W	GTX 7800	1 (16,384)	蛋白质序列集	GLSL	OSEARCH34	4.69(无回溯) 3.46(回溯)
						SSEARCH34	2.01(无回溯) 1.49(回溯)
[16]	S-W	GTX 6800	8(63-4092)	Swiss-Port 46.3	GLSL	OSEARCH	7.5
		GTO 7900				SSEARCH	4.6
[17]	S-W	2 GTX 8800	5(63-511)	Swiss-Port 2006.12	CUDA	OSEARCH	18
			11(143-567)			SSEARCH	30
[18]	CUDASW++	GTX295 (2 G80)	25(144-5478)	Swiss-Port 56.6	CUDA	BLAST	2.4
						SWPS3 x86/SSE2	1.46-2.74
						SWPS3 Cell BE	4.70-4.73
						SW-CUDA	6.24-10.27
[19]	CUDASW++ 2.0	GTX 280	20(144-5478)	Swiss-Port 56.6	CUDA	NCBI-BLAST 2.2.19	3.08-9.55
[20]	S-W	2 GTX 295 (4 G80)	—	SSCA #1 benchmark	CUDA	CUDASW++	1.74-1.77
						S-W(CPU version)	45

(续表)

文献	软件名称	GPU 型号	待比序 列数(长度)	搜索数据库 (序列)	编程 语言	参照算法	加速比
[21]	DOPA(S-W)	GTX 275	14(144-5478)	Swiss-Port 2010.10	CUDA	SSEARCH(SSE2)	2.14
						BLAST	1.8
						SWPS3	1.43
						CUSASW++2.0	1.13
[22]	S-W	GTX 285	5(7680- 731068)	4 条序列集	CUDA	S-W (CPU version)	120
[23]	MUMmerGPU	GTX 8800	—	C. briggsae Chr. III(Sanger) L. monocytogenes (454) S. suis (Illumina/Solexa)	CUDA	MUMmer	3.47-3.79
[24]	MUMmerGPU 2.0	GTX 8800	—	同上	CUDA	MUMmerGPU 1.0	1.5-4.0
[25]	GPU-BLAST	Fermi C2050	51(2-4998)	BLAST Databases	CUDA	NCBI-BLAST	3.0-4.0
[26]	CUDA-BLASTP	GTX 295	5(127-2026)	GenBank Nonredundant Protein Database	CUDA	NCBI-BLASTP 2.2.22	4.8-6.1
						SSEARCH35 (SSE2)	7.2-21.1
[27]	BLASTP	ATI HD4850	10(81-7980)	Swiss-Prot 57.3	Brook+	NCBI-BLASTP	6-9

2.2 基于 FPGA 的并行序列比对

定制计算 (Custom Computing) 又称为可重构计算 (Reconfigurable Computing), 是异构体系结构研究领域正在探索的另一重要发展方向。FPGA 是现场可编程门阵列的简称, 是实现可重构计算的关键部件。FPGA 是一种超大规模集成电路芯片, 通过消息共享内存与主机通信。与通用微处理器相比, FPGA 器件具有支持级联并行、可重构和功耗低等优点。自 1998 年首次应用 FPGA 加速序列比对以来, 已有许多基于 FPGA 的序列比对并行处理研究被提出, 归纳其共同特征如下:

(1) 使用线性脉动阵列处理器: 脉动阵列处理器是指一组以同一步调运行的处理器单元。正如血液从心脏规则地流出, 数据以时钟频率流经脉动式处理单元。

(2) 得分矩阵计算: 在 S-W/N-W 算法中, 计算得分矩阵 W 存在着数据依赖, 即 $W(i, j)$ 的计算依赖于相邻元素 $W(i-1, j)$, $W(i, j-1)$ 和 $W(i-1, j-1)$ 。而矩阵中处于同一条反对角线上的元素不存在数据相关, 即 $W(i-1, j)$ 和 $W(i, j-1)$ 互不依赖。可从矩阵左上角开始, 沿反对角线方向, 采用波前 (wavefront) 方法对得分矩阵实现并行处理。

(3) 序列最大相似度: 序列比对分为计算和回溯两个步骤。计算是指将两条序列 S 和 L 的比对分值填充于得分矩阵, 得出其最大相似度。而要获得序列的比对结果, 则必须启动回溯过程。不包含和包含回溯步骤的时间复杂度分别为 $O(m+n)$ 和 $O(mn)$, 其中 m 和 n 分别为 S 和 L 的长度。由于指数级的时间复杂度极大地限制了待比序序列长度, 因此, 许多算法均以获取序列最大相似度得分为目标, 而忽略序列的比对结果。

(4) 划分技术: 待比序序列的字符通常直接存储在处理单元 (PE) 中。每个 PE 存储一个字符, 这样 PE 阵列能计算得分矩阵中的一列。由于 PE 阵列的规模远远小于实际的序列长度, 为了比对任意长度的序列, 采用划分技术在 PE 中存储几个字符, 或将得分矩阵的计算划分为多个阶段, 每个阶段仅比对待比序序列的一部分。

表 2 列举了基于 FPGA 异构系统的并行序列比对方法。从表 2 可以得出, 除文献 [33] 采用 SIMD (Single Instruction

Multiple Data) 阵列外, 其他研究均采用一个或多个 FPGA 脉动式阵列实现并行序列比对。目前基于 FPGA 的并行序列比对主要集中于 S-W、DIALIGN、HMMER、ClustalW 和 MUSCLE 算法的研究。

表 2 基于 FPGA 的并行序列比对方法

文献	加速技术	基础算法	比对 目标	划分	加速比
[28]	SAMBA	S-W	得分	是	13-83
[29]	Virtex II 1000(XC2V1000) Virtex II 6000(XC2V6000)	S-W	得分	否	500
[30]	Xilinx XV1000-4	S-W	得分	否	5.6-55.6
[31]	FPGA 脉动阵列	S-W	比对	是	57-85.9
[32]	Virtex II 6000(XC2V6000)	S-W/Gotoh	得分	是	125-170
[33]	SIMD 阵列	S-W	得分	是	99
[34]	FPGA 脉动阵列	S-W/Gotoh	得分	是	185-250
[35]	Altera EP1S30	Gotoh	得分	是	352
[36]	SRC-6 Cray-XD1	S-W	得分	是	98
[37]	Xilinx Virtex-II 6000 Xilinx Virtex-II 4000	Gotoh	得分	否	6-16.9
[38]	Xilinx Virtex II Pro XC2VP70	S-W/N-W	比对	是	186-510
[39]	Xilinx XC2V6000-4	S-W/N-W	比对	是	2
[40]	Xilinx XC5VLX330T	S-W	得分	是	32
[41]	Virtex4 XC4VLX100	S-W	比对	否	16
[42]	Stratix II EP2S180F1508I4	DIALIGN	得分/ 比对	是	383.4/ 141.38
[43]	Xilinx Spartan3-4000	Hmmsearch	比对	是	71
[44]	Xilinx XC2V6000	ClustalW	得分/ 比对	是	50.9/ 13.3
[45]	Xilinx Virtex-4 FX100	MUSCLE	比对	是	150

2.3 基于 Cell BE 的并行序列比对

Cell BE 是由 STI (SONY、Toshiba、IBM) 设计的一种典型的异构多核处理器。Cell BE 处理器由 1 个通用的控制处理单元 PPE (PowerPC Processing Element)、8 个协处理器单元 SPE (Synergistic Processing Element)、1 个连接 XDR 主存的高带宽内存接口、1 个连接其他 Cell BE 的一致性接口和 1 个输入输出接口组成, 它们通过环形片上单元互连总线 EIB (Element Interconnect Bus) 连接。操作系统运行于 PPE 上, PPE 负责对 SPE 进行调度, 而 SPE 则负责主要的计算工作。SPE 直接访问本地存储 (Local Storage, LS), 它通过 DMA

(Direct Memory Access)方式访问主存,由 DMA 负责将指令和数据传输到 LS。

2007年,Sachdeva等^[46]第一次将 Cell BE 异构系统应用到序列比对领域,提出 FASTA 和 ClustalW 算法的 Cell BE 加速方法。文中不仅详细阐述了将 FASTA 和 ClustalW 程序中最耗时部分映射到 SPU(Synergistic Processing Units)上的策略和实现方法,还分析了 Cell BE 不支持相关序列比对算法的硬件细节。2009年,Sachdeva等^[47]在文献^[46]的基础上进行扩展,提出 HMMER 算法在单 SPU 上的加速方法。

表 3 从 Cell BE 型号、加速的基础算法、搜索的数据库、待比对序列的数目及长度、加速比等 5 个方面总结了基于 Cell BE 的并行序列比对方法。

表 3 基于 Cell BE 的并行序列比对方法

文献	Cell BE 型号	基础算法	搜索数据库(序列)	待比对序列数目及长度	加速比
[48]	Playstation3	S-W	Swiss-Port55.1	17(144-4061)	2
[49]	Playstation3	S-W BLASTP	Swiss-Port55.2 GenBank Protein	18(63-852) 100(1-2000)	30 3.57
[50]	QS20 (16 SPEs)	Global Spliced Syntenic	— Phytoene Lycopersicum	2(2048 * 2048) 2(1792 * 872) 2(1408 * 1408)	6.5-8 10 10
[51]	QS21 (16 SPEs)	ClustalW	BioPerf	B class;66 (平均 1082, 最长 3270) C class;318 (平均 1043, 最长 5196)	3.8
[52]	Playstation3	距离矩阵计算	6 个序列集	8000(平均 73-825)	2

2.4 异构系统序列比对并行处理技术分析

CPU 与协处理器之间的高效协同计算是发挥异构平台性能的关键因素^[53,54]。鉴于协处理器的计算能力和执行特点,结合现有的序列比对并行处理研究,分别分析 CPU+GPU、FPGA、Cell BE 异构平台序列比对并行处理技术的难点。

2.4.1 基于 GPU 的并行序列比对技术难点

虽然 GPU 具有强大的计算能力和高带宽,但是硬件结构固定且不能在运算粒度、存储调度等方面适应序列比对具体应用的特点,限制了其实际计算性能。

(1) 负载均衡

基于 GPU 的序列比对算法中存在两种负载不平衡现象,一是 GPU 计算线程之间的负载不平衡,二是 CPU 与 GPU 之间的负载不平衡。GPU 线程间负载不平衡可能由某些序列比对算法本身造成,如 S-W 算法,其得分矩阵副对角线上的元素可并行地计算。以线程总数 32 为例,副对角线上每次参与计算的并行线程数量分别为 1, 2, 4, ..., 32, 30, ..., 1, 实际线程利用率仅为 50%,在并行计算过程中导致了大量计算资源的闲置。

(2) 任务同步

某些序列比对算法本身存在数据依赖关系,如 S-W 算法。GPU 由于缺乏全局同步机制,需要使用同步操作来协调各线程之间的数据通信,因此对需要频繁同步的序列比对算法实际加速性能较低。

(3) 访存控制

GPU 采用分级存储器模型,包括片和片外存储器。片

上存储器具有较高的访问速率,而访问片外存储器具有很长的延迟。如何将序列数据及中间结果有效地分配到这些存储器中,以加速数据的访问、减少访问片外存储器的长延迟操作,是设计基于 GPU 加速的序列比对算法的技术难点之一。

2.4.2 基于 FPGA 的并行序列比对技术难点

虽然 FPGA 细粒度的位级并行能力能较好地适用序列比对应用的计算特征,可重构特性使用户能针对应用需求定制最优的计算和存储结构,但设计基于 FPGA 的并行序列比对仍存在以下技术难点。

(1) 计算方法多样性

FPGA 算法加速的难点之一是如何充分理解算法的应用特点,将其转换为定制计算结构。由于序列比对领域的计算方法多、典型算法多,不可能为每种算法和同一算法下不同规模的应用设计定制的计算结构,因此要通过提取公共计算特征,对典型序列比对算法提出通用化的并行策略和计算结构。

(2) 存储访问

尽管 FPGA 的存储容量迅速增加,但目前顶级 FPGA 芯片的存储容量仍不足 4MB,远不能满足序列比对算法 $O(n^2)$ 的存储需求,而片外存储器的访问延迟将导致并行处理效率下降。因此,有限的存储容量和存储带宽是实现 FPGA 硬件加速的主要障碍,需要以存储为核心设计高效的计算和存储访问策略,缓解“存储墙”导致程序执行性能下降的问题。此外,由于 FPGA 工作频率不到 CPU 的 1/10,需要采用高效的流水和并行策略才能获得高于 CPU 的性能。如何在资源受限的情况下实现资源的优化配置,以及通过高效的访存调度实现数据在计算阵列中的合理流动和有效并行,是设计基于 FPGA 序列比对并行算法的关键和难点^[22]。

2.4.3 基于 Cell BE 的并行序列比对技术难点

结合 Cell BE 计算特性,分析基于 Cell BE 的并行序列比对技术的难点如下。

(1) 有限的本地内存

Cell BE 处理器的计算能力主要来自于协处理器单元 SPE,而每个 SPE 中仅有 256kB 可直接寻址的片上本地存储 LS,且 SPE 只能直接访问 LS 中的数据和代码。当 SPE 上运行的代码和数据大小超过 256kB 时,需将部分数据和代码放在片外主存,必要时通过 DMA 操作从片外主存中获取。在序列比对应用领域,许多代码均超过了 256kB 的限制。同时,256kB 本地存储能力也限制待比对的序列最大长度为 2048 个字符(氨基酸或核苷酸)。比对超过 2048 个字符的序列时,需采用有效的划分机制。所以,必须使用较少内存的数据结构,通过在程序中显式地对存储结构进行管理,才能避免因“存储墙”而带来的可编程性及性能等问题。

(2) 数据传输与协调

在 SPE 上,需要计算的序列数据都要从主存传输到 LS。如何最小化数据传输时间是 Cell BE 加速序列比对技术的难点之一。

在高性能计算领域,虽然世界上首台千万亿次系统 Roadrunner 超过 12 万处理器核是由通用多核处理器和 Cell BE 处理器混合组成,但自从 IBM 宣布不再生产用于科学计算的 Cell BE 处理器后,将 Cell BE 应用于序列比对领域的研究可能会慢慢消失。

与之相反,最新一期(2014 年 11 月)超级计算 TOP500

排名中,第二名和第六名均采用了 CPU+GPU 架构,在前 100 名中有 17 台这种结构的 GPU 集群。序列比对并行处理研究现状表明,应用 CPU+GPU 解决序列比对问题的相关研究仍十分活跃,并且取得了很大进展,其方法也趋于成熟,未来仍不失为解决序列比对问题的有效方法和手段。

作为协处理器的发明者,Intel 公司于 2012 年推出了新一代众核协处理器 MIC。随着采用 CPU+MIC 异构体系结构的超级计算机“天河二号”成为世界上最快的计算机,以及在前 100 名中出现 9 台采用这种结构的 MIC 集群,CPU+MIC 异构协同计算已成为高性能计算的重要发展趋势。

目前,CPU+GPU/CPU+MIC 异构计算架构逐渐成为高性能计算领域的主流。当前流行的 CPU+GPU 异构协同计算主要是以 CPU 为主体,在其上运行逻辑计算部分,而 GPU 主要负责密集型任务的计算,CPU 和 GPU 是属于不同指令集下的异构。与此不同的是,CPU+MIC 异构协同计算则是基于 x86 架构的异构计算,其基本指令与 CPU 一样,采用 x86 指令集并有部分指令扩展;而且,MIC 作为协处理器,其编程模式与主处理器高度一致、协同工作,某些情况下,可看作一个独立的计算节点使用,甚至在一定程度上可认为 MIC 上的每个“核”都是独立的节点,可将 MIC 作为一个超小型的集群。另一方面,GPGPU 的核心比较轻量级,线程在核上的切换速度快到可以忽略不计,所以在使用 GPGPU 编程时,成千上万个线程可同时工作。应用于序列比对领域,GPGPU 适合并行度极高而分支较少的应用;相比之下,MIC 的核心略显笨重,单核的功能性比 GPGPU 强大得多,它更适合于并行度较高、逻辑比较复杂的序列比对应用。

3 进一步的研究工作

随着生物信息呈指数级别增长,高效、快速而准确地处理序列数据带来了基因时代研究新型序列比对方法的需求。我们认为可从以下 3 个方面对序列比对并行处理研究进行深入探讨。

3.1 面向序列比对的异构多/众核并行计算系统协同并行计算

目前序列比对并行处理研究均针对某一种单纯并行计算平台,例如 CPU+GPU,其研究主要集中在把序列比对任务分别调度到 CPU 和 GPU 上实现。然后,随着 CPU 的多核化和 GPU 通用计算化,另外 Intel 公司新开发的众核处理器 MIC 协处理器芯片,由于核心数达到六十多个,其并行计算能力得到迅速提升,CPU+GPU/CPU+MIC 异构计算架构逐渐成为高性能计算领域的主流。

当前尚未见针对 CPU/GPU/MIC 计算特性以及特定异构平台对典型序列比对应用的研究,这还处于早期探索阶段,其原因与大规模异构并行系统的序列比对计算面临的一些重大技术挑战,如异构并行编程技术、任务调度、存储优化和性能优化等有关。CPU+MIC 异构协同计算由于基于 x86 架构,其编程模式与主处理器高度一致,且 MIC 的单核功能性强大,更适合于解决并行度较高、逻辑比较复杂的序列比对问题。基于 CPU+MIC 异构众核系统的序列比对并行处理研究将会成为序列比对并行处理的研究热点。

3.2 大规模序列比对并行算法研究

近年来,Gen Bank 中的数据每 9~12 个月就要翻一番,可扩展性已成为设计序列比对并行算法所追求的重要目标之

一。目前序列比对并行处理的大部分研究工作是将已有比对算法向异构平台移植,而针对异构平台设计的全新算法较少。随着计算生物学的快速发展,需要设计高效的新型算法用于解决大规模序列比对问题。

(1)提高运算速度。这一问题不仅包括更快地得到所需要的比对结果,还将得到一些算法先前由于运行速度的瓶颈而得不到的比对结果,并解决这些算法的可扩展性问题。

(2)解决异构系统存储空间的瓶颈问题。解决这一问题将大大加强大规模生物序列比对问题的求解能力。

另外,平台的异构性、海量并行度、协处理器计算能力差异、CPU 与协处理器之间数据传输速度差异为设计大规模序列比对并行算法提出了新的挑战。

3.3 适合序列比对的异构计算系统性能优化

对于异构系统序列比对并行处理程序而言,影响其性能的因素主要有:计算的组织方式、寄存器和缓存使用、存储访问、同步、数据传输等。为开发高效的序列比对并行计算程序,需全面考虑协处理器的计算特性,针对序列比对并行应用程序计算特征,设计相应的任务划分和数据流组织方式,以保证 CPU 与协处理器之间负载均衡,降低各种交互开销,进而获得性能提升。

结束语 随着半导体和制造工艺的飞速发展以及处理器体系结构的不断演化,GPU、FPGA、MIC 等协处理器的性能受到了高性能计算业界的广泛重视。序列比对是生物信息学中重要而基础性的研究工作,如何充分发挥协处理器硬件加速技术的优势,准确而快速地通过序列比对判断序列的相似性,进而为发现序列间的进化关系,预测基因功能、揭示新陈代谢和生物进化规律提供重要信息,已成为生物信息学中重要而具有挑战性的任务之一。

在研究者和实践者的共同努力与推动下,如今生物序列比对充分借助异构协同并行计算的发展,已取得不少成果。本文总结并根据异构系统体系结构组织序列比对的研究成果,重点分析实现硬件加速序列比对的难点,希望能为该领域的进一步研究提供有益的参考。基于异构系统的序列比对并行处理研究是开放的课题,新型并行算法研究、软/硬件高效协同方式、并行性能优化和大规模可扩展算法研究等方面都值得进一步研究。

参考文献

- [1] Needleman S B, Wunsch C D. A general method applicable to the search for similarities in the amino acid sequence of two proteins [J]. *Journal of Molecular Biology*, 1970, 48(3): 443-453
- [2] Pearson WR, Lipman DJ. Rapid and sensitive protein similarity searches [J]. *Science*, 1985, 227(4693): 1435-1441
- [3] Zhu Xiang-yuan, Li Ken-li, Salah A. A Data Parallel Strategy for Aligning Multiple Biological Sequences on Multi-Core Computers [J]. *Computers in Biology and Medicine*, 2013, 43(4): 350-361
- [4] Zhu Xiang-yuan, Li Ken-li, Salah A, et al. Cluster-Distribute-Align-Merge: A General Algorithm to Speed Up Multiple Sequence Alignment on Multi-Core Computers [J]. *Journal of Computational and Theoretical Nanoscience*, 2014, 11(4): 1000-1006
- [5] 张法,乔香珍,刘志勇. 基于 Smith-Waterman 算法的并行分而

- 治之生物序列比对算法[J]. 中国科学 E 辑技术科学, 2004, 34(2):190-199
- [6] Kim T, Joo H. Clustalxseed: a GUI-based grid computation version for high performance and terabyte size multiple sequence alignment [J]. BMC Bioinformatics, 2010, 11(38):467
- [7] Krishna M R, Benedict P, David H. Meta-Alignment with Crumble and Prune Partitioning very large alignment problems for performance and parallelization [J]. BMC Bioinformatics, 2011, 12(5):1447-1448
- [8] 刘立芳, 霍红卫, 王宝树. PHGA-COFFEE: 多序列比对问题的并行混合遗传算法求解[J]. 计算机学报, 2006, 29(5):727-733
- [9] Paolo D T, Miquel O, Fernando G, et al. Cloud-Coffee: implementation of a parallel consistency-based multiple alignment algorithm in the T-Coffee package and its benchmarking on the amazon elastic-cloud [J]. Bioinformatics, 2010, 26(15):1903-1904
- [10] Zhu Xiang-yuan, Li Ken-li, Salah A, et al. Parallel Implementation of MAFFT on CUDA-Enabled Graphics Hardware [J]. IEEE/ACM Transactions on Computational Biology and Bioinformatics, 2015, 12(1):205-218
- [11] 夏飞. 生物序列分析算法硬件加速器关键技术研究[D]. 长沙: 国防科学技术大学, 2011
- [12] Yang S, Gregory M S, Ali A. Performance Analysis of IBM Cell Broadband Engine on Sequence Alignment [C]//Proc of IEEE AHS'09. San Francisco, California: IEEE, 2009:439-446
- [13] David D, Francisco J E, Pilar H, et al. Parallelizing and optimizing a bioinformatics pairwise sequence alignment algorithm for many-core architecture [J]. Parallel Computing, 2011, 37(45):244-259
- [14] 叶笑春, 林伟, 范东睿, 等. 蛋白质序列比对算法在众核结构上的并行优化[J]. 软件学报, 2010, 21(12):3094-3105
- [15] Liu Y, Huang W, Johnson J, et al. GPU accelerated Smith-Waterman [M]//Computational Science—ICCS 2006. 2006:188-195
- [16] Liu W G, Bertil S, Gerrit V, et al. Streaming algorithms for biological sequence alignment on GPUs [J]. IEEE Trans on Parallel and Distrib Systems, 2007, 18(9):1270-1281
- [17] Manavski S A, Valle G. CUDA compatible GPU cards as efficient hardware accelerators for Smith-Waterman sequence alignment [J]. BMC Bioinformatics, 2008, 9(1):1-9
- [18] Liu Y, Maskell D, Schmidt B. CUDASW++: Optimizing Smith-Waterman sequence database searches for CUDA-enabled graphics processing units [J]. BMC Research Notes, 2009, 2(1):1-10
- [19] Liu Y, Schmidt B, Maskell D. CUDASW++ 2.0: Enhanced Smith-Waterman protein database search on CUDA-enabled GPUs based on SIMT and virtualized SIMD abstractions [J]. BMC Research Notes, 2010, 3(11):93
- [20] 林江, 唐敏, 童若锋. GPU 加速的生物序列比对[J]. 计算机辅助设计与图形学学报, 2010, 22(3):420-427
- [21] Saeed A K, Poole S, Perot J B. Acceleration of the Smith-Waterman algorithm using single and multiple graphics processors [J]. Journal of Computational Physics, 2010, 229(11):4247-4258
- [22] Laiq H, Marijn K, Zaid A. DOPA: GPU-based protein alignment using database and memory access optimizations [J]. BMC Research Notes, 2011, 4(1):1-11
- [23] Michael C S, Cole T, Arthur L D, et al. High-throughput sequence alignment using Graphics Processing Units [J]. BMC Bioinformatics, 2007, 8(1):141-144
- [24] Trapnell C, Schatz M C. Optimizing data intensive GPGPU computations for DNA sequence alignment [J]. Parallel Computing, 2009, 35(8):429-440
- [25] Panagiotis D V, Nikolaos V S. GPU-BLAST: using graphics processors to accelerate protein sequence alignment [J]. Bioinformatics, 2010, 27(2):182-188
- [26] Weiguo L, Bertil S, Wolfgang M W. CUDA-BLASTP: Accelerating BLASTP on CUDA-Enabled Graphics Hardware [J]. IEEE/ACM Transactions on Computational Biology And Bioinformatics, 2011, 8(6):1678-1684
- [27] 裴颂文, 王心怡, 韦刚, 等. 基于多核流处理器的 BLAST 并行化算法研究[J]. 系统仿真学报, 2011, 23(10):2065-2069
- [28] Lavenier D. Speeding up Genome Computations with a Systolic Accelerator [J]. SIAM News, 1998, 31(8):6-7
- [29] Puttegowda K, Worek W, Pappas N, et al. A Run-Time Reconfigurable System for Gene-Sequence Searching[C]//Proc of the 16th Int Conf on VLSI Design. New Delhi, India: IEEE, 2003:561-566
- [30] Marongiu A, Palazzari P, Rosato V. A Specialized Hardware Device for the Protein Similarity Search [J]. Concurrency and Computation: Practice and Experience, 2004, 16(9):917-931
- [31] 汪冬, 唐志敏. Smith-Waterman 算法在脉动阵列上的实现及分析[J]. 计算机学报, 2004, 27(1):12-20
- [32] Oliver T F, Schmidt B, Maskell D L. Hyper Customized Processors for Bio-Sequence Database Scanning on FPGAs [C]//Proc of ACM/SIDA FPGA '05. Monterey, California: ACM, 2005:229-237
- [33] Bias A D, Dahle D M, Diekhans M, et al. The UCSC Kestrel Parallel Processor [J]. IEEE Trans on Parallel and Distributed Systems, 2005, 16(1):80-92
- [34] Zhang P, Tan G, Gao GR. Implementation of the Smith-Waterman Algorithm on a Reconfigurable Supercomputing Platform [C]//Proc of ACM HPRCTA '07. Reno, Nevada: ACM, 2007:39-48
- [35] Jiang X, Liu X, Xu L, et al. A Reconfigurable Accelerator for Smith-Waterman Algorithm [J]. IEEE Trans on Circuits and Systems II, 2007, 54(12):1077-1081
- [36] Abouellail M, El-Araby E, Taher M, et al. DNA and Protein Sequence Alignment with High Performance Reconfigurable Systems [C]//Proc of NASA/ESA Conf on Adaptive Hardware and Systems. Edinburgh, UK: IEEE, 2007
- [37] Harris B, Jacob A C, Lancaster JM, et al. A Banded Smith-Waterman FPGA Accelerator for Mercury BlastP [C]//Proc of Int Conf on Field Programmable Logic and Applications. Amsterdam, Netherlands: IEEE, 2007:765-769
- [38] Tom V T, Martin C H. Families of FPGA-based accelerators for approximate string matching [J]. Microprocessors and Microsystems, 2007, 31(2):135-145
- [39] Khaled B, Ying L, AbdSamad B. A highly Parameterized and efficient FPGA-based skeleton for pairwise biological sequence alignment [J]. IEEE Trans on Very Large Scale Integration, 2009, 17(4):561-570

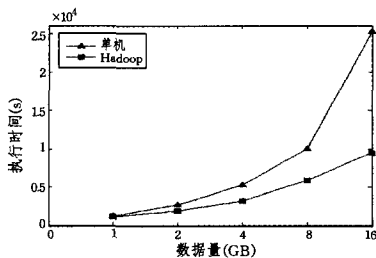


图4 单机和 Hadoop 环境下执行效率的对比

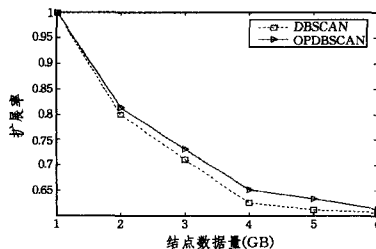


图5 DBSCAN 算法与 OPDBSCAN 算法扩展率的对比

结束语 本文在研究 DBSCAN 算法的基础上,提出了改进的 OPDBSCAN 算法。OPDBSCAN 算法基于 Hadoop 平台实现,利用交叠分区并行获取局部自适应的 Eps 参数解决了 Eps 参数敏感问题,再通过 MapReduce 并行实现 DBSCAN,提高算法效率。最后实验证明,改进后的算法能比原算法更快、更好地实现聚类。

参考文献

[1] 李爱国,库向阳. 数据挖掘原理、算法及应用[M]. 西安:西安电子科技大学出版社,2012:29-35
 [2] Ekan Y J, Pallickara S. MapReduce for data intensive scientific analysis[C]// eScience 2008; Proceedings of the Fourth IEEE,

International Conference on eScience. Piscataway; IEEE Press, 2008,277-284

[3] Dean J, Ghemawat S. MapReduce; Simplified data processing on large clusters [J]. Communications of the ACM, 2008, 51(1): 107-113
 [4] Ester M, et al. A density based algorithm for discovering clusters in large spatial databases with noise[C]// Proc of 2nd Inter Conf Knowledge Discovering in Databases and Data Mining (KDD-96). Portland; AAAI Press, 1996
 [5] 侯荣涛,朱斌. 基于 DBSCAN 聚类算法的闪电临近预报模型[J]. 计算机应用, 2012, 32(3): 847-851
 [6] 李莉平,沈俊媛. 基于数据挖掘的 DBSCAN 算法及其应用[J]. 科技创业月刊, 2009(8): 134-135
 [7] 黄毅磊. DBSCAN 算法及在城市网格化管理中的应用[D]. 上海:上海交通大学, 2010
 [8] 夏鲁宁,荆继武. SA-DBSCAN: 一种自适应基于密度聚类算法[J]. 中国科学院大学学报, 2009(4): 530-538
 [9] 陈刚,刘秉权,吴岩. 一种基于高斯分布的自适应 DBSCAN 算法[J]. 微电子学与计算机, 2013, 30(3): 27-30
 [10] 周水庚,周傲英,金文,等. FDBSCAN: 一种快速 DBSCAN 算法[J]. 软件学报, 2000, 11(6): 735-744
 [11] Ankerst M, Breunig M, Kriegel H-P, et al. Optics: Ordering points to Identify the Clustering Structure[C]// Alex D, Christos F, Shahram G, eds. Proc ACM SIGMOD Int Conf on Management of Data. Philadelphia; ACM Press, 1999: 49-60
 [12] 周水庚,周傲英. 基于数据分区的 DBSCAN 算法[J]. 计算机研究与发展, 2000, 37(10): 1153-1159
 [13] 孙凌燕. 基于密度的聚类算法研究[D]. 太原:中北大学, 2009
 [14] 熊忠阳,吴林敏,张玉芳. 针对非均匀数据集的 DBSCAN 过滤式改进算法[J]. 计算机应用研究, 2009, 26(10): 3721-3723

(上接第 395 页)

[40] Yoshiki Y, Hung K T, Wayne L. FPGA-Based Smith-Waterman Algorithm; Analysis and Novel Design [M]// Reconfigurable Computing: Architectures, Tools and Applications. 2011: 181-192
 [41] Nuno S, Nuno R, Paulo F. Hardware accelerator architecture for simultaneous short-read DNA sequences alignment with enhanced traceback phase [J]. Microprocessors and Microsystems, 2012, 36(2): 96-109
 [42] Azzedine B, Jan MC, Alba de M, et al. A Hardware Accelerator for the Fast Retrieval of DIALIGN Biological Sequence Alignments in Linear Space [J]. IEEE Transactions on Computers, 2010, 59(6): 808-821
 [43] Steven D, Patrice Q. Hardware Acceleration of HMMER on FPGAs [J]. J Sign Process Syst, 2010, 58(1): 53-67
 [44] Tim O, Bertil S, Darran N, et al. Using recon? gurable hardware to accelerate multiple sequence alignment with ClustalW [J]. Bioinformatics, 2005, 21(6): 3431-3432
 [45] Scott L, Quinn O S. Accelerated large-scale multiple sequence alignment [J]. BMC Bioinformatics, 2011, 12(29): 466
 [46] Vipin S, Michael K, Evan S, et al. Exploring the viability of the Cell Broadband Engine for bioinformatics applications[C]// Proc of IEEE IPDPS'07. California; IEEE, 2007
 [47] Vipin S, Michael K, Evan S, et al. Exploring the viability of the Cell Broadband Engine for bioinformatics applications [J]. Parallel Computing, 2008, 34(11): 616-626

[48] Adam S, Christian L, Philipp K, et al. SWPS3-fast multi-threaded vectorized Smith-Waterman for IBM Cell/B. E. and x86/SSE2 [J]. BMC Research Notes, 2008, 1(1): 107
 [49] Adrianto W, Bertil S, Huiliang Zh, et al. High performance protein sequence database scanning on the Cell Broadband Engine [J]. Scientific Programming, 2009, 17(12): 97-111
 [50] Abhinav S, Srinivas A. Parallel genomic alignments on the Cell Broadband Engine [J]. IEEE Transactions on Parallel and Distributed Systems, 2009, 20(10): 1600-1610
 [51] Hans V, Sean R, Koen D B. Accelerating multiple sequence alignment with the Cell BE processor [J]. The Computer Journal, 2010, 53(6): 814-826
 [52] Adrianto W, Chee K K, Bertil S. Multi-threaded vectorized distance matrix computation on the CELL/BE and x86/SSE2 architecture [J]. Bioinformatics, 2010, 26(10): 1368-1369
 [53] Aarti S, Chen C, Liu W G, et al. A Hybrid Computational Grid Architecture for Comparative Genomics [J]. IEEE Transactions on Information Technology in Biomedicine, 2008, 12(2): 218-225
 [54] Meng X D, Vipin C. A High-Performance Heterogeneous Computing Platform for Biological Sequence Analysis [J]. IEEE Transactions on Parallel and Distributed Systems, 2010, 21(9): 1267-1280
 [55] 卢风顺,宋君强,银福康,等. CPU/GPU 协同并行计算研究综述[J]. 计算机科学, 2011, 38(3): 5-9, 46