

Parallel Processing

MT 2014_HK1_2017_2018

Ho Chi Minh City University of Technology
<http://www.cse.hcmut.edu.vn/~nam>

References

1. *Parallel Computing – theory and practice*, Michael J. Quinn, McGRAW-HILL, 1994.
2. *Parallel Programming: Techniques and Applications Using Networked Workstations and Parallel Computers*, Barry Wilkinson and Michael Allen, Second Edition, Prentice Hall, 2005.
3. *Distributed Algorithms*, Nancy Lynch, Morgan Kaufmann, 1997.
4. *Scalable Parallel Computing: Technology, Architecture, Programming*, Kai Hwang & Zhiwei Xu, McGRAW-HILL, 1997.
5. *Introduction to Parallel Computing*: https://computing.llnl.gov/tutorials/parallel_comp/#Designing
6. *Open MP*:
7. *MPI*: <http://www.mcs.anl.gov/research/projects/mpi/tutorial/>
8. *Xeon Phi Programming*:
9. *GPU Programming*
10. *Hadoop*:
11. *Spark*:
12. *Parallel Computing - theory and practice*: http://www.cs.cmu.edu/afs/cs/academic/class/15210-f15/www/tapp.html#_preface

Lectures

Lectures (90 min)	Week	Topics	References	Notes
Lecture 1	W1	Introduction	[1]	
Lab 1	W2	Lab: Introduction		Chung Thành Minh
Lecture 2	W2	Abstract machine models & Multithreading – PRAM & BSP – Multithreading	[1][2]	
Lab 2	W3	Lab: Multithreading		Chung Thành Minh
Lecture 3	W3	MPI	[7]	
Lab 3	W4	Lab: MPI – Point-to-point communication		Diệp Thanh Đăng
Lecture 4	W4	Parallel machine architectures: – Flynn classifications – Pipeline, Processor array, Multiprocessor, Data flow computer – Processor organizations	[1][2]	
Lab 4	W5	Lab: MPI – Collective communication		Diệp Thanh Đăng
Lecture 5	W5	Speedup: – Amdahl – Gustafson	[1]	
Lab 5	W6	Lab: OpenMP		Chung Thành Minh
Lecture 6	W6	OpenMP	[6]	
Lab 6	W7	Lab: Xeon Phi 1		Chung Thành Minh
Lecture 7	W7	Map/Reduce	[10]	
No Lecture & Lab	W8	Midterm Break		
Exam	W9	Midterm Exam		
Lab 7	W10	Lab: Xeon Phi 2		Chung Thành Minh
Lecture 8	W10	Parallel & distributed computing techniques (1)	[2]	
Lab 8	W11	Lab: GPUs 1		Ng. Mạnh Thìn
Lecture 9	W11	Parallel & distributed computing techniques (2)	[2]	Ref
Lab 9	W12	Lab: GPUs 2		Ng. Mạnh Thìn
Lecture 10	W12	Designing parallel programs (1)	[12]	Ref
Lab 10	W13	Lab: Hadoop 1		Ng. Mạnh Thìn
Lecture 11	W13	Designing parallel programs (2)	[12]	Ref
Lab 11	W14	Lab: Hadoop 2		Ng. Mạnh Thìn
Lecture 12	W14	Parallel algorithms (1)	[2]	

Lab 12	W15	Lab: Algorithm 1		Ng. Mạnh Thìn
Lecture 13	W15	Parallel algorithms (2)	[4]	Ref
Lab 13	W16	Lab: Algorithm 2		Ng. Mạnh Thìn
Lecture 14	W16	Review		
Lab 14	W16	Lab: Algorithm 3		Ng. Mạnh Thìn

HPC Lab

- SuperNode-I: 5 nodes x (2 CPUs x 8 cores, 32GB RAM), Infiniband 30 Gbps
- SuperNode-XP 24 nodes x (2 CPUs x 12 cores, 2 Xeon Phi x 61 cores, 512/256/128 GB RAM, 1 TB HD/SSD), Infiniband 56 Gbps
- IBM system: 8 nodes x (2 CPUs x 16 cores, 10 Gbps Ethernet)

Evaluation

- Midterm exam (tuần thứ 8): 10%
- Final exam: 30%
- Project: 30%
- Lab: 30%

Contact:

- Thoại Nam: namthoai@hcmut.edu.vn
- **Đại diện lớp (Lớp trưởng): Hồ Bảo Quốc (baoquoc29@gmail.com)**

Đề Project môn XLSS&HPB

TNMT 2014_HK1_2017_2018

Quy định:

1. Mỗi nhóm tối đa 3-4 sinh viên (trùng với nhóm seminar).
2. Các nhóm đăng ký danh sách và đề tài cho Trưởng lớp, **hạn cuối 01/09/2017**:
 - Ghi rõ họ tên, MSSV, email của các thành viên trong nhóm
3. Nộp báo cáo 2-4 trang mô tả nội dung đề tài thực hiện, nguồn dữ liệu, dự kiến kết quả, **hạn cuối 15/09/2017**
4. Nộp báo cáo sơ bộ về tiến độ và kết quả đạt được, **hạn cuối 15/10/2017**
5. Nộp báo cáo cuối kỳ, **hạn cuối 01/12/2017**, tất cả các nhóm phải nộp:
 - Báo cáo tối đa 8 trang A4
 - Mã nguồn.

Đề 1: Viết chương trình giải bài toán Association Rules dùng multi-thread trên Xeon Phi dùng cơ chế Offload và OpenMP

Lý thuyết:

- Tìm hiểu giải thuật cho Association Rules
- Tìm hiểu cách lập trình trên Xeon Phi dùng cơ chế Offload và OpenMP.

Hiện thực:

- Viết chương trình
- Vẽ biểu đánh giá hiệu năng (speedup) cho từng trường hợp với số lượng threads khác nhau
- Thử nghiệm sử dụng 2 Xeon Phi cards.

Đề 2: Viết chương trình Association Rules dùng multi-thread trên Xeon Phi dùng OpenCL

Lý thuyết:

- Tìm hiểu giải thuật cho Association Rules
- Tìm hiểu cách lập trình trên Xeon Phi dùng OpenCL.

Hiện thực:

- Viết chương trình
- Vẽ biểu đánh giá hiệu năng (speedup) cho từng trường hợp với số lượng threads khác nhau
- Thử nghiệm sử dụng 2 Xeon Phi cards.

Đề 3: Viết chương trình Association Rules dùng multi-thread trên Xeon Phi dùng Intel Cilk Plus và Intel TBB (Thread Building Blocks).

Lý thuyết:

- Tìm hiểu giải thuật cho Association Rules
- Tìm hiểu cách lập trình trên Xeon Phi dùng Intel Cilk Plus và Intel TBB (Thread Building Blocks).

Hiện thực:

- Viết chương trình
- Vẽ biểu đánh giá hiệu năng (speedup) cho từng trường hợp với số lượng threads khác nhau
- Thử nghiệm sử dụng 2 Xeon Phi cards.

Đề 4: Viết chương trình nhân ma trận kích thước 1.000x1.000, 10.000x10.000 và 100.000x100.000 (có trao đổi hàng/cột giữa các bộ xử lý) trên hệ thống máy tính ảo có giao tiếp 1Gbps, 10 Gbps (Gigabit Ethernet), 40 Gbps (Infiniband).

Lý thuyết:

- Tìm hiểu và viết chương trình nhân ma trận dùng MPI có trao đổi hàng cột
- MPI One-Sided Communication

Hiện thực:

- Viết chương trình
- Vẽ biểu đồ đánh giá hiệu năng (speedup) cho 2 trường hợp dùng hệ thống máy tính vật lý và máy tính ảo với số lượng máy tính khác nhau
- So sánh trường hợp giao tiếp thông thường và cách sử dụng One-Sided Communication.

Bài 5: Viết chương trình so trùng ảnh dùng “The Skein Hash Function Family” trên Hadoop

Lý thuyết:

- Tìm hiểu Hadoop
- Tìm hiểu “The Skein Hash Function Family”
- Giải pháp loại các ảnh trùng nhau trong một tập ảnh lớn

Hiện thực:

- Viết chương trình
- Đánh giá hiệu suất trên hệ thống thực.

Đề 6: Viết chương trình Association Rules trên GPUs

Lý thuyết:

- Tìm hiểu về lập trình GPU (CUDA)
- Tìm hiểu về giải thuật cho Association Rules.

Hiện thực:

- Viết chương trình
- Đánh giá hiệu năng (speedup) với số lượng core khác nhau.

Đề 7: sinh viên có thể đề xuất bài toán để giải như K-means, SVM (Support Vector Machines), bài toán trên Graph...

Tham khảo

Video

High Performance Computing made easy, <http://www.hpc.uva.nl/>

Algorithms on Xeon Phi (XP)

1. Strassen algorithm: https://www.singularis-lab.com/docs/materials/07_Shapovalov_Strassen_CKA.pdf
2. Fast Smith-Waterman: <http://xsw.sdu-hpcl.org>
3. PCIT algorithm: <https://utexas.influent.utsystem.edu/en/publications/optimizing-the-pcit-algorithm-on-stampedes-xeon-and-xeon-phi-proc>
4. Breadth-first search: <http://www.dislab.org/docs/bfs-phi-paper-eng.pdf>
5. Graph coloring:
http://www.sandia.gov/~egboman/papers/Deveci_coloring_ipdps16.pdf
6. Pattern matching: <http://sbac.lip6.fr/2014/session%206/1-BitParallel.pdf>
7. Sort: <http://cass-mt.pnnl.gov/docs/ia3-2013/2-3.pdf>
8. OpenFoam:
<https://www.nersc.gov/assets/Uploads/IXPUGISC15OpenFOAMTCSV6.pdf>
9. Kalman Filter:
<https://facultystaff.richmond.edu/~ggilfoyl/research/keegan2014DNP.pdf>
10. SU2: http://stanford.edu/~economon/docs/SU2_IPCC_SciTech2015_final.pdf
11. Conjunction Gradient method:
https://en.wikipedia.org/wiki/Conjugate_gradient_method

Deep learning on XP

1. Deep Neural Networks for Financial Market Prediction:
https://www.researchgate.net/publication/281685181_Implementing_Deep_Neural_Networks_for_Financial_Market_Prediction_on_the_Intel_Xeon_Phi