



## ECE 254B: Parallel Processing

Behrooz Parhami: 2008/12/13 || E-mail: [parhami@ece.ucsb.edu](mailto:parhami@ece.ucsb.edu) || Other contact info at: [Bottom of this page](#)

Go up to: [B. Parhami's course syllabi](#) or [his home page](#)

### Background and history of ECE 254B

The graduate-level course ECE 254B was created by Dr. Parhami, shortly after he joined UCSB in 1988. It was first taught as ECE 594L (Special Topics in Computer Architecture: Parallel and Distributed Computations) in the spring quarter of 1989. A year later, it was converted to a regular graduate course (ECE 251). In 1991, Dr. Parhami led an effort to restructure and update UCSB's graduate course offerings in the area of computer architecture. The result was the creation of the three-course sequence ECE 254A/B/C to replace ECE 250 (Advanced Computer Architecture) and ECE 251. The three new courses were designed to cover high-performance uniprocessing, parallel computing, and distributed computer systems, respectively. In 1999, based on a decade of experience in teaching ECE 254B, Dr. Parhami published a textbook on parallel processing; see the course syllabus below.

[Link to previous offerings of ECE 254B](#)

### ECE 254B: Fall Quarter 2008

**This area reserved for important course announcements:**

**2008/12/13:** This course is now over and the final grades have been reported to the registrar's office. There will be no further updates to this Web page. The

**2008/12/04:** Updated PPT and PDF presentations for Parts V and VI of our textbook are now available at the book's website. Suggested practice problems from Chapters 17 and 18 appear after HW4. A note on the Google platform has been added near the end of this page. The following office hours have been added during the finals week: M 12/8, 3:00-5:00; T 12/9, 10:00-1:30. The open-book final will be held in our regular classroom on W 12/10, 8:30-11:00. Course grades will be available on Sat. 12/13.

**2008/11/16:** Homework 4 has been posted below. Exclusions and sample problems for the final exam have also been posted. Updated PPT and PDF presentations for Part IV of the textbook are now available at our textbook's website.

**2008/11/02:** Homework 3 and a note on Problem 5.11 (HW2) have been posted below. Updated PPT and PDF presentations for Part III of the textbook are now available at our textbook's website.

**2008/10/24:** A note on Problem 7.14 (HW2) has been posted below, along with examples representing the types of problems you might see on the midterm exam.

**2008/10/22:** The presentation files for Part II of the textbook have undergone minor revision (addition of slide #116 and correction of an error in slide #83)

**2008/10/13:** Homework 2 has been posted below. Please see the notes added at the end of HW1 regarding the solutions to Problems 1.G and 3.I. Updated PPT and PDF presentations for Part II of the textbook are now available at our textbook's website.

**2008/09/29:** Homework 1 has been posted below. Updated PPT and PDF presentations for Part I of the textbook are now available at the book's website.

**2008/08/27:** The required and recommended textbooks for this course are now available at the UCSB bookstore. Based on past experience, used copies of the required textbook can be purchased at significant discount (at times, 50% or more) from various on-line retailers, compared with its price at the campus bookstore.

**Course:** ECE 254B – Advanced Computer Architecture: Parallel Processing, University of California, Santa Barbara, Spring 2006, Enrollment Code 48439

**Catalog entry:** **254B. Advanced Computer Architecture: Parallel Processing.** (4) PARHAM, B. *Prerequisite:* ECE 254A. *Lecture, 4 hours.* The nature of concurrent computations. Idealized models of parallel systems. Practical realization of concurrency. Interconnection networks. Building-block parallel algorithms. Algorithm design, optimality, and efficiency. Mapping and scheduling of computations. Example multiprocessors and multicomputers. (S)

**Instructor:** Behrooz Parhami, Room 5155 Harold Frank Hall (Engineering I), Phone 805-893-3211, [parhami@ece.ucsb.edu](mailto:parhami@ece.ucsb.edu)

**Meetings:** TR 10:00-11:30, Phelps Hall, Room 1431

**Consultation:** Open office hours, held in Room 5155 Harold Frank Hall – T 12:00-1:30, R 2:00-3:30

**Motivation:** The ultimate efficiency in parallel systems is to achieve a computation speedup factor of  $p$  with  $p$  processors. Although often this ideal cannot be achieved, some speedup is generally possible by using multiple processors in a concurrent (parallel or distributed) system. The actual speed gain depends on the system's architecture and the algorithm run on it. This course focuses on the interplay of architectural and algorithmic speedup techniques. More specifically, the problem of algorithm design for "general-purpose" parallel systems and its "converse", the incorporation of architectural features to help improve algorithm efficiency and, in the extreme, the design of algorithm-based special-purpose parallel architectures, are dealt with. The foregoing notions will be covered in sufficient detail to allow extensions and applications in a variety of contexts, from network processors, through desktop computers, game boxes, Web server farms, multiterabyte storage systems, and mainframes, to high-end supercomputers.

**Prerequisite:** Basic computer architecture at the level of ECE 154; an advanced architecture course, such as ECE 254A, would be helpful but can be waived.

**References:** *Required text* – Parhami, B., [Intro. Parallel Processing: Algorithms and Architectures](#), Plenum, 1999. Click on the title to view detailed information about the textbook, its errata, and downloadable presentation files used in course lectures.

*Recommended text, optional* – Herlihy, M. and N. Shavit, *The Art of Multiprocessor Programming*, Morgan Kaufmann, 2008. Because the focus of our course is on architecture and its interplay with algorithms, this recommended text, which deals primarily with software and programming topics, constitutes helpful supplementary reading.

*Journals* – *IEEE Trans. Parallel and Distributed Systems*, *IEEE Trans. Computers*, *J. Parallel & Distributed Computing*, *Parallel Computing*, *Parallel Processing Letters*. Also, *IEEE Computer* and *IEEE Concurrency* (the latter ceased publication in late 2000) for broad introductory papers.

*Conferences* – Int'l Symp. Computer Architecture (ISCA, since 1973), Int'l Conf. Parallel Processing (ICPP, since 1972), Int'l Parallel & Distributed Processing Symp. (IPDPS, formed in 1998 by merging IPPS/SPDP, which were held since 1987/1989), and ACM Symp. Parallel Algorithms and

Architectures (SPAA, since 1988).

**Electronic Resources at UCSB**

<http://www.library.ucsb.edu/eresources/databases/> (electronic journals, collections, etc.)

<http://www.library.ucsb.edu/subjects/engineering/ece.html> (research guide in ECE)

**Evaluation:** Students will be evaluated based on these three components with the given weights:

25% -- Homework: see the “deadlines” column in course calendar for schedule.

25% -- Closed-book midterm exam: see the course calendar for date and coverage.

50% -- Open-book final exam: see the course calendar for date and coverage.

**Research:** Students interested in the research option, in lieu of the final exam, will cooperate in investigating a well-defined problem relating to interconnection networks for parallel processing. Please see the instructor for details and follow the [research paper guidelines](#) to format your research report. Students who obtain results of publishable quality will receive an "A" for the course, regardless of homework and midterm exam grades.

**Calendar:**

Course lectures, homework assignments, and exams have been scheduled as follows. This schedule will be strictly observed. Textbook chapters are specified for lectures and exams. It is a good idea to look over the covered chapter(s) *before* each lecture if possible.

Day & Date	Chap	Lecture Topic	Deadlines, Links, Notes
R 09/25/08	1	Introduction to parallel processing	<a href="#">Homework requirements</a>
T 09/30/08	2	A taste of parallel algorithms	<a href="#">HW#1</a> posted (chaps. 1-4)
R 10/02/08	3-4	Complexity and models	
T 10/07/08	5	PRAM model and basic algorithms	Research project defined
R 10/09/08	6	More shared-memory algorithms	HW #1 due
T 10/14/08		Shared memory implementation	<a href="#">HW #2</a> posted (chaps. 5-8)
R 10/16/08	7	Sorting and selection networks	Research subprojects assigned
T 10/21/08	8	Other circuit-level examples	
R 10/23/08	9	Sorting on a 2D mesh or torus	HW #2 due
T 10/28/08	10	Routing on a 2D mesh or torus	
R 10/30/08	1-10	Midterm exam: closed book (10:00-11:45 AM)	Note the extended time
T 11/04/08	11	Other mesh/torus algorithms	<a href="#">HW #3</a> posted (chaps. 9-12)
R 11/06/08	12	Mesh variations and extensions	
T 11/11/08		Holiday (Veterans' Day): No lecture	
R 11/13/08	13	Hypercubes and their algorithms	HW #3 due
T 11/18/08	14	Sorting and routing on hypercubes	<a href="#">HW #4</a> posted (chapters 13-16)
R 11/20/08	15	Other hypercubic architectures	Project plan and references due
T 11/25/08	16	Other interconnection networks	
R 11/27/08		Holiday (Thanksgiving): No lecture	
T 12/02/08	17	Emulation and scheduling	HW #4 due
R 12/04/08	18	Data storage, input, and output	HW #4 due
W 12/10/08	1-18	Final exam: open book, 8:30-11:00 AM	Research report due by 5:00 PM

**Homework: General Requirements**

Solutions should be brought to class on the due date and handed in before the lecture begins. Late homework will not be accepted, so plan to start work on your assignments early. Use a cover page that includes your name, course and assignment number for your solutions. Staple the sheets and write your name on top of every sheet in case sheets are separated. Some cooperation is permitted, but plagiarism or direct copying will have severe consequences.

**ECE 254B f2008, Homework #1: Fundamental concepts (chapters 1-4, due R 10/9/08)**

Do the following problems from the textbook or defined below: 1.C [below, 15 pts.], 1.G [below, 20 pts.], 2.9 [20 pts.], 2.11 [15 pts.], 3.3 [20 pts.], 3.1 [below, 10 pts.]

**Problem 1.C: Amdahl's law** A multiprocessor is built out of 5-GFLOPS processors. What is the maximum allowable sequential fraction of a job in Amdahl's formulation if the multiprocessor's performance is to exceed 1 TFLOPS?

**Problem 1.G: Top-performing supercomputers** Slide 12 in the lecture presentation for Part I of the textbook contains a table listing the three highest-performing supercomputers, as of early 2005. Construct an updated table that lists the current top three supercomputers, presenting at least the same information that is contained in the aforementioned table. You can obtain the required information via <http://www.top500.org/>. By averaging the peak performance of the top three supercomputers from 2005 and 2008, calculate an annual rate of growth in performance over the past three years. Compare the calculated annual rate of performance growth for the top supercomputers to the corresponding rate for microprocessors (Fig. 1.1 in the textbook) and discuss.

**Problem 3.I: Solving recurrences** Solve the recurrence  $T(m) = T(m/2) + \log^k m$ , obtaining the constant for the leading term and the order of remaining terms.

**Note on Problem 1.G:** The presentation for Part I of the textbook has been updated with a slide that lists the top 3 supercomputers, as of June 2008.

**Note on Problem 3.I:** The solution handed out stated that Theorem 3.1 is applied. The solution is in fact obtained via unrolling, which leads to  $T(m) = T(1) + \log 2 + \log 4 + \dots + \log^k(m/2) + \log^k m$ . Assuming  $T(1) = 0$  and taking the logarithm base to be 2, we have  $T(m) = 1^k + 2^k + \dots + (\log m - 1)^k + (\log m)^k$ . The latter is the sum of the  $k$ th powers of natural numbers, for which formulas are available. Using such a formula, we find  $T(m) = (\log^{k+1} m)/(k+1) + (\log^k m)/2 + o(\log^k m)$ . Finding the sum of the  $p$ th powers of consecutive integers from 1 to  $n$  is discussed in <http://maa.mc.edu/proceedings/spring2002/doucet.jahroni.pdf>, among other sources.

#### ECE 254B f2008, Homework #2: Shared-memory and circuit models (chapters 5-8, due R 10/23/08)

Do the following problems from the textbook or defined below: 5.11 [20 pts.], 5.15 [20 pts.], 6.1 [10 pts.], 6.14 [corrected below, 15 pts.], 7.14 [25 pts.], 8.C [below, 10 pts.]

**Problem 6.14: Corrected introduction** Consider the butterfly memory access network depicted in Fig. 6.9, but with only 8 processors and 8 memory banks, one connected to each circular node in columns 0 and 3, respectively.

**Problem 8.C: Distributed dictionary machine** The  $p$  processors of a linear array hold  $n$  records in sorted order such that the leftmost processor holds the records with the smallest keys. As the processors compute, they produce new records or cause records to be deleted. Discuss in high-level terms a protocol to be followed by the processors for insertion and deletion of records such that the sorted order of the records is maintained and the processor loads (number of records held) are roughly balanced over the long run.

**Note on Problem 5.11:** In the solution to part a, the "then" part of the second "if" statement should also include the assignment statement "sum := 0".

**Note on Problem 7.14:** Mr. Aleksey Khiverenko noted that the latency of the 9-sorter in Fig. 7.5 of the textbook is not the claimed 9 levels, but 10 levels. Thinking that I might have copied the figure incorrectly, I consulted the original source of the figure (Knuth's Vol. 3 of *The Art of Computer Programming*, 1973), which shows the same design (I have not yet checked newer editions of the book to see if the error has been corrected). A Google search led to <http://www.cs.chalmers.se/~ms/Charme03.pdf>, which contains the correct design: The comparator between the third and seventh lines from the top should be drawn between the third and fifth lines from the top, thus creating a seventh 3-sorter near the right end of the diagram.

#### ECE 254B f2008, Homework #3: Mesh and mesh-based architectures (chapters 9-12, due R 11/13/08)

Do the following problems from the textbook or defined below: 9.5 [20 pts.], 9.11 [15 pts.], 10.3 [15 pts.], 10.C [below, 20 pts.], 11.Ba [below, 20 pts.], 12.8 [10 pts.]

**Problem 10.C: Mesh routing for matrix transposition** An  $m \times m$  matrix is stored on an  $m \times m$  processor array, one element per processor. Consider matrix transposition (sending each element  $a_{ij}$  from processor  $ij$  to processor  $ji$ ) as a packet routing problem. (a) Find the routing time and required queue size in each processor assuming store-and-forward row-first routing, stating all your assumptions. (b) Discuss the routing time with column-first wormhole routing, assuming 20-flit packets.

**Problem 11.Ba: Matrix algorithms on linear arrays** A  $b$ -banded or  $b$ -diagonal matrix is a matrix for which all the entries not contained in a single band of  $b$  diagonals are zero. Show how to multiply an  $m$ -vector by an  $m \times m$   $b$ -diagonal matrix in  $O(m)$  steps on a  $b$ -cell linear array.

#### ECE 254B f2008, Homework #4: Interconnection networks (chapters 13-16, due T 12/02/08)

Do the following problems from the textbook or defined below: 13.6 [10 pts.], 13.14 [20 pts.], 14.13 [20 pts.], 14.15 [20 pts.], 15.3 [10 pts.], 16.J [below, 20 pts.]

**Problem 16.J: Hoffman-Singleton graph** A Hoffman-Singleton graph is built from 10 copies of a 5-node ring as follows. The rings are named  $P_0$ - $P_4$  and  $Q_0$ - $Q_4$ . The nodes in each ring are labeled 0 through 4 and interconnected as a standard (bidirectional) ring. Additionally, node  $j$  of  $P_i$  is connected to node  $ki + 2j \pmod{5}$  of  $Q_k$ . (a) Determine the node degree of the Hoffman-Singleton graph. (b) Determine the diameter of the Hoffman-Singleton graph. (c) Prove that the Hoffman-Singleton graph is a Moore graph. (d) Show that the Hoffman-Singleton graph contains many copies of the Petersen graph.

An  $m \times m$  matrix is stored on an  $m \times m$  processor array, one element per processor. Consider matrix transposition (sending each element  $a_{ij}$  from processor  $ij$  to processor  $ji$ ) as a packet routing problem. (a) Find the routing time and required queue size in each processor assuming store-and-forward row-first routing, stating all your assumptions. (b) Discuss the routing time with column-first wormhole routing, assuming 20-flit packets.

**Sample problems from Chapters 17-18** Because no homework has been assigned based on the material in Chapters 17 and 18, the following problems are recommended as practice problems for the final exam: 17.5, 17.9ab, 17.13, 18.11, 18.A (defined below).

**Problem 18.A: Reliability of disk arrays** Assume that each disk has a mean time to failure (MTTF) of 40 years. (a) What is the MTTF for a 512-disk system with no redundancy? State all your assumptions. (b) Ignoring controller failures, what is the MTTF for a RAID5 configuration in which parity is applied across the entire set of 512 disks and the rebuild time is 0.5 hour?

#### ECE 254B f2008, Reading list and sample problems for midterm

Remember that the midterm exam ends at 11:45 AM (not 11:30 as our normal classes). The following sections are excluded from the first 10 chapters of the textbook to be covered in the midterm exam: 2.6, 3.5, 4.5, 4.6, 6.5, 7.6, 8.5, 9.6.

Sample problems: The following problems from the textbook are representative of what you might expect in the midterm exam: 2.3, 3.5, 5.5 (with  $i + s$  corrected to  $j + s$ ), 7.6a, 10.7a. Collectively, these 5 problems might form a 105-minute exam, although the problem statements might change a bit for a closed-book exam.

#### ECE 254B f2008, Reading list for final exam

Remember that the final exam begins at 8:30 (not 8:00, which is given in the Schedule of Classes). The following sections are excluded from the first 18 chapters of the textbook to be covered in the final exam: 2.6, 3.5, 4.5, 4.6, 6.5, 7.6, 8.5, 9.6, 11.4, 11.6, 12.6, 16.6, 17.2, 18.6.

Sample problems: The following problems from the textbook are representative of what you might expect in the final exam: 2.2, 7.9, 13.11, 14.2, 16.2, 17.13. Collectively, these 6 problems might form a 2.5-hour open-book exam.

#### A note on the Google platform

This is in response to a question raised in class with regard to the hardware platform used by Google. Google uses 0.2-0.5M servers within its hardware architecture (there is disagreement about the exact number). Servers, which are chosen based on cost-effectiveness rather than raw performance, are divided into classes dedicated to bot dispatching, query distribution, index access, document storage, etc. Queries are sometimes broken into subqueries that are sent to different servers to reduce the latency. The following two websites provide good overviews and point to more detailed info:

<http://highscalability.com/google-architecture>

[http://en.wikipedia.org/wiki/Google\\_platform](http://en.wikipedia.org/wiki/Google_platform)

#### ECE 254B f2008, Research Project Notes

None at this time.

Return to: [Top of this page](#) || Go up to: [B. Parhami's course syllabi](#) or [his home page](#)

---

Dr. Behrooz Parhami, Professor  
[Dept. Electrical & Computer Engineering](#)  
[University of California, Santa Barbara](#)  
[Santa Barbara, CA 93106-9560 USA](#)



Office phone: +1 805 893 3211  
Departmental fax: +1 805 893 3262  
Office: Rm 5155 Harold Frank Hall  
Deliveries: Rm 4155 Harold Frank Hall

E-mail: [parhami at ece.ucsb.edu](mailto:parhami@ece.ucsb.edu)

<http://www.ece.ucsb.edu/~parhami/>