The Status of Parallel Processing Education *

Russ Miller†

August 2, 1995

Abstract

This report is concerned with the status of parallel processing education at the post-secondary level. The information collected suggests that *i*) courses focusing on parallelism are available in computer science departments of some four year colleges, *ii*) advanced graduate courses in parallelism are being regularly offered at university centers, *iii*) a smattering of courses devoted to parallelism are being offered to undergraduates at university centers, and *iv*) various aspects of parallelism have been incorporated into traditional graduate and undergraduate courses at both four year colleges and university centers.

The IEEE Technical Committee on Parallel Processing (TCPP) was formed in 1992 with the intent of providing a forum for scientists and engineers throughout the world to exchange ideas on parallel processing. As part of its mission, the TCPP is concerned with establishing guidelines for parallel processing education. This report concentrates on developing the basis for such an undertaking by considering the current status of parallelism in the post-secondary education process.

In October, 1992, the first Newsletter of the Parallel Processing Technical Committee was distributed to more than 4,000 members of the IEEE Computer Society. The newsletter contained a request for information regarding current and proposed mechanisms (new and revised courses, seminars, and so forth) for incorporating parallelism into the education process. Subsequently, direct mailings were sent to chairs of computer science and

*This work was partially supported by NSF grant IRI-9108288 and by a grant made available from New York State/United University Professionals.

†Department of Computer Science, State University of New York, Buffalo, New York 14260; miller@cs.buffalo.edu

engineering departments. The direct mail distribution list was based on various lists that were assembled at the State University of New York at Buffalo. Finally, a request for information was posted to the well-established newsgroup "comp.parallel."

This report contains information from more than 70 sites that offer educational opportunities involving parallelism, including undergraduate institutions, university centers, institutes, and corporations, from a variety of countries around the world. It is interesting to note that the majority of this information was obtained in response to the electronic posting.

The incorporation of parallelism into the education process is in a state of flux. In order to accommodate such an environment, it is expected that this report will continue to evolve, with existing entries being updated and new entries being created. In order to facilitate the dissemination of such a document, it has been decided that most recent version will be available via anonymous FTP in the directory users/miller on ftp.cs.buffalo.edu.

It is the intent that this report provide insight for the community at large in terms of developing individual curricula that incorporate parallelism. In addition, this report should be of use to high school and college students who are interested in continuing their education, and incorporating parallelism into that process. One of the early modifications planned for this report is to update it to include a contact at each institution, whether that is a member of the faculty, administrative assistant, or scientist. Further, in revising the information initially obtained for this report, a decision was made to eliminate the few references to texts and machines that were provided, due to the inconsistency of this information. Therefore, at some point in the future, we would like to provide such information in a uniform fashion.

After reviewing the information provided from various sites, the following trends appear to have emerged.

- At four year colleges, courses in parallelism are beginning to appear. Many of these colleges have access to either a small parallel machine (typically, transputer-based) or a small network of workstations. These machines are used for projects in the courses and for independent studies on parallelism.

- The trend at university centers in terms of graduate-level education appears to be to provide an array of courses, including parallel algorithms, parallel programming, and parallel architectures, to name a few. Typically, the students have access to machines either at the university or at one of the NSF sponsored sites. Typical machines that are utilized in the classroom include machines manufactured by Maspar, Thinking Machines (both the CM-2 and CM-5), Intel, and Ncube.

- The trend at university centers in terms of undergraduate education appears to be to include aspects of parallelism in a variety of traditional courses (e.g., algorithms, architecture, operating systems, and programming languages) and to provide at most one introductory course in parallel processing. However, it is interesting to note the one-year laboratory course in High Performance Scientific Computing being developed with NSF support at the University of Colorado at Boulder, which is described in more detail below.

Finally, we would like to mention an electronic facility called 'parlib,' which contains entries of 37 schools with a graduate emphasis on parallel processing. Parlib can be reached by sending mail to parlib@hubcap.clemson.edu.

Auburn University

The Department of Computer Science and Engineering offers the following courses for seniors and graduate students.

- CSE 526: Design of Software for Parallel Systems.

  An introduction to programming paradigms for parallel processing.

- CSE 533: Parallel Processing.

  An overview of current parallel processing architectures, algorithms, and programming paradigms.

- CSE 534: Distributed-Memory Multiprocessors I.

  An introduction to Hoare's CSP formalism for specifying concurrent systems.

The following course is offered to graduate students.

- CSE 634: Distributed-Memory Multiprocessors II.

  Students specify and implement a significant application on a multiprocessor system.

Brown University

The Computer Science Department offers a variety of graduate-level seminars. One that is offered every other year is entitled "Topics in Parallel and Distributed Debugging."

Bucknell University

The Computer Science Department offers the following advanced undergraduate course.

- CS366: Parallel Computation.

  Parallel models of computation, architectures, languages, and algorithms. Performance and programmability of vector and array processors, SIMD, MIMD, hypercube, and dataflow machines.

California Institute of Technology

Several courses are offered at Caltech that involve parallel programming. One three term course that is offered concentrates on parallel and distributed algorithms.

Clarke College

The Computer Science Department offers an undergraduate course in parallel programming, covering topics that include parallel architectures, SIMD and MIMD programming, parallel algorithms, as well as speedup and performance issues.

Colgate University

The Computer Science Department offers an advanced undergraduate course in parallel computing. In addition, aspects of parallelism have been incorporated into existing courses including Computer Organization, Programming Languages, Operating Systems, and Simulation.

Colorado State University

The Computer Science Department incorporates various aspects of parallelism into courses on Comparative Programming Languages, System Software, and Analysis of Algorithms. In addition, the following undergraduate course is offered.

- CS475: Parallel Programming.

  An introduction to parallel programming, considering MIMD (shared and distributed memory), SIMD, functional programming, and related issues.

The following graduate-level courses are also offered.

- CS557: Parallel and Distributed Algorithms.

  A more serious look at parallel and distributed algorithms and their analysis.

- CS570: Parallel Architectures.

  A consideration of a variety of systems containing multiple processors.

Connecticut College

An advanced undergraduate course in parallel processing is offered which provides a survey of the field and an introduction to parallel algorithms.

Cornell University

The Department of Computer Science incorporates aspects of parallelism into traditional courses in numerical algorithms and theory. In addition, the following courses are offered.

- CS 422/522: Parallel Scientific Computing.

  The focus is on parallel numerical algorithms.

- CS 612: Compilers for Parallel Architectures.

  The focus is on issues such as automatic parallelization and dependent versus independent computations.

- CS 683: Theory of Parallel Computing.

  The focus is on parallel complexity classes, parallel graph algorithms, lower bounds, models of parallel computation, and so forth.

Dartmouth College

A variety of parallel computing courses have been taught over the past several years. Currently, a course entitled "Parallel Computing" will be offered for beginning graduate students, with plans to offer the course yearly.

Drexel University

The Department of Mathematics and Computer Science offers a course in concurrent programming. This course contains concurrent process material from the standard operating systems course along with some additional material on parallel programming.

Eastern Connecticut State University

The Department of Mathematics and Computer Science offers a course in parallel processing, as well as a topics course in parallel and distributed processing.

Edinburgh Parallel Computing Centre

For the past eight years, the Centre has been running a Summer Scholarship Program. This program involves between 15 and 20 undergraduate and graduate students from all over the globe (including USA, Germany, Netherlands, Scotland, England, Bulgaris and Pakistan).

The course begins with a week of practicals, lectures and tutorials on the following subjects: C, Fortran, SIMD, MIMD, local software for message passing (CHIMP) and parallel utilities (PUL), presentation skills, documentation skills, 2D graphics, 3D graphics and EPCC work in general. The students then work on individual projects for up to eight weeks and then a final report is written which goes through a review cycle.

The students are treated as active members of EPCC and as such, are expected to present their skills and work during the summer to each other and the staff.

Gallaudet University

The Department of Mathematics and Computer Science offers an undergraduate course in High Performance Computing. This course is an introduction to the techniques and theory of computation at the upper limits of what is currently possible. This area of practice and study is sometimes called "supercomputing" and significantly involves "parallel processing." The course includes topics on UNIX, vector pipelining, parallel processes, (massively) parallel processors, and programming methods for parallel processes.

Gettysburg College

The Department of Computer Science incorporates aspects of parallelism into its Computer Organization and Numerical Analysis courses. In addition, a course entitled "Introduction to Parallel Processing" is offered, which includes a basic overview of the area, as well as a discussion of SIMD and MIMD machines.

Harvard University

A course entitled "Scientific Computation on Large Scale Parallel Computers" is taught regularly. This course provides an introduction to architecture and algorithms for parallel computers. The course gives an overview of vector and parallel architectures. Basic algorithms for matrix operations, the solution of linear systems of equations, sorting, the Fast Fourier Transform, the computation of eigenvalues and eigenvectors, and solution of partial differential equations are taught. The emphasis is on distributed memory architectures.

Reducing the need for data motion through proper data allocation and management of the data motion is critical for performance. The techniques for managing distributed memory hierarchies are new. Techniques for efficient data motion will be taught, both for common communication patterns, such as data replication/reduction and matrix transposition, and for general routing.

Hood College

The Department of Mathematics and Computer Science incorporates aspects of parallel computing in the following courses.

- CSCI 379: Analysis of Algorithms (undergraduate)
- CSCI 461/561: Computer Architecture (undergraduate/graduate)
- CSCI 464/564: Operating Systems (undergraduate/graduate)
- CSCI 508: Computer Organization (graduate)

In addition, CSCI 498/598 ("Parallel Computing") provides a comprehensive introduction to parallel architectures, programming paradigms, and applications; this course is open to graduate students and advanced undergraduates.

Illinois State University

An undergraduate special topics course is offered which provides a survey of parallel processing.

Iowa State University

The Department of Computer Science offers the following undergraduate course.

- Computer Science 425: High Performance Computing for Scientific and Engineering Applications.

  Introduction to high-performance computing using different architectures, programming on vector processors and parallel computers. Discussion of performance issues, sample applications from science and engineering, and practical issues in high-performance computing are covered.

In addition, the following graduate-level courses are offered.

- Computer Science 524: Computer System Architecture.

  Fundamentals of computer design, performance and cost, instruction set design, basic processor implementation techniques, pipelining, memory design, caches, I/O systems, multiprocessor systems, interconnection networks.

- Computer Science 612: Parallel and Distributed Algorithms.

  An advanced course in the theory of parallel and distributed computation. Models of parallel and distributed computation. Parallel sorting, graph, geometric, algebraic, and number theoretic-algorithms. The parallel computation thesis. NC- algorithms and P-complete problems. Agreement problems. Mutual exclusion and resource allocation. Implementations of wait-free registers. Network problems. Lower bounds and impossibility results. Fault-tolerance. Randomized computation.

- Computer Science 624: Advanced Topics in Computer Architecture.

  Shared-memory multiprocessors, distributed memory machines, systolic processing, neural networks, issues in high performance computing, comparative study of different architectures, performance models.

- Computer Science 625: Issues in Parallel Programming and Performance.

  Parallel solutions of numerical and non-numerical problems, performance and other computational issues in parallel programming.

- Computer Science 652: Topics in Distributed Operating Systems.

  Concepts and techniques for network operating systems: high-level languages and communication protocols, name and object management, concurrency control for consistent distributed data, design of reliable software, protection, performance analysis.

Macalester College

The Department of Mathematics and Computer Science offers a senior-level course entitled "Introduction to Parallel Processing." The course is an overview covering parallel architectures, parallel languages, and parallel algorithms.

Michigan State University

The Computer Science Department offers a variety of courses. The 400-level courses are intended for both undergraduate and graduate students, while the 800- and 900-level courses are intended for graduate students.

- CPS 474: Vector and Parallel Programming.

  Programming of high-performance computers. Hardware, algorithms, numerical accuracy, compilers. Vector, MIMD and SIMD machines.

- CPS 479: Software Tools for Concurrent Systems.

  Design, development and application of software tools for parallel and distributed systems. Program development, debugging, performance monitoring, simulation, data and control flow analysis, and visualization.

- CPS 812: Advanced Operating Systems.

  Advanced topics in parallel and distributed operating systems which include load sharing, scheduling, reliability recovery, security, memory management, distributed file systems, naming, distributed agreement, and object-oriented operating systems.

- CPS 820: Advanced Computer Architecture.

  Instruction set architecture, instruction pipelining, vector processors, cache memory, high bandwidth memory design, virtual memory, input and output, benchmarking techniques, new development of single CPU systems.

- CPS 822: Parallel Computer Systems.

  Massively parallel SIMD processors, multiprocessor architectures, interconnection networks, synchronization and communication, memory and address space management, process management and scheduling, parallel compilers and languages, performance evaluation.

- CPS 838: Design of Parallel Algorithms.

  Fundamental issues in the design of parallel algorithms. Material is drawn from current research topics including models of parallel computation and implementations of algorithms on SIMD and MIMD machines.

- CPS 910: Topics in Computer Networks and Distributed Systems.

  Advanced topics and recent developments in high-bandwidth computer networks, protocol engineering, and distributed computer systems.

- CPS 920: Topics in High Performance Computer Systems.

  Advanced topics in the design of high performance computer systems.

Michigan Technology University

A course is offered in parallel algorithms, which emphasizes principles used in the development of algorithms for parallel computers. The focus is on the following topics: programming paradigms, implementation, analysis and performance evaluation. Algorithms in the areas of scientific computation and nonnumeric processing are considered, as are software tools for automatic vectorization and parallelization, performance visualization, and debugging.

Mississippi State University

The following graduate-level courses are offered in either the Computer Science or Electrical Engineering Department.

- CS 8133: Designing Parallel Algorithms.

  Techniques for designing algorithms to take advantage of different parallel architectures. Includes techniques for parallelizing sequential algorithms, and techniques for matching algorithm to architectures.

- CS 8733: Advance Systems Programming.

  Concepts of multi-programming, multi-processing, time-sharing. Distributed operating systems concepts.

- CS 8793: Complexity of Sequential and Parallel Algorithms.

  Complexity of sequential algorithms, theory of complexity of parallel algorithms.

- EE 8063: Parallel Computing Architectures I.

  Study of hardware structures relevant to concurrent computing; evaluation and design methods associated with memory, pipelining, and multiple processors.

- EE 8073: Parallel Computing Architectures II.

  Study of communication structures and routing methods that are central to concurrent computing, multiple computers, and data flow machines.

Monclair State

The Department of Mathematics and Computer Science plans to incorporate parallelism into the Computer Architecture and Fundamentals of Programming Languages courses.

New Jersey Institute of Technology

The Department of Electrical and Computer Engineering offers the following courses.

- EE784: Digital Systems Architecture.

  Modern trends in computer architecture such as pipelining, array processing, and multiprocessing. Principles of parallel processing and examples of their implementation.

- EE785: Parallel Processing Systems.

  Concurrent programming environments, data decomposition and parallel algorithms (e.g., sorting, FFT, matrix operations, and Monte Carlo methods), embedding techniques, and performance evaluation.

The Computer and Information Science Department offers a variety of seminars, as well as the following courses.

- CIS 650: Computer Architecture.

  Parallel memory systems, pipelining and vector processing, SIMD and MIMD organizations and algorithms, and interconnection networks.

- CIS 668: Parallel Algorithms.

  Design of SIMD algorithms on linear-connected array, mesh, cube, perfect-shuffle computer, Omega, Bitonic sorting network, and so forth.

- CIS 750: Parallel Computer Systems.

  MIMD systems and design issues; cache-coherence, packet routing, multi-thread, data-flow, case studies and projects of parallel programming on MIMD machines.

Norfolk State University

Aspects of parallelism are incorporated into the Survey of Computing Languages course.

North Dakota State University

The Department of Computer Science offers a graduate course in parallel processing, which includes a presentation of parallel architectures and parallelizing compilers, as well as job scheduling and performance evaluation.

Northeastern University

The College of Computer Science offers two formal courses, one at the graduate level and one at the undergraduate level. Both courses focus on data parallel computing and the SIMD models.

Northwestern University

The Department of Electrical Engineering and Computer Science incorporate aspects of parallelism into a variety of courses. At the undergraduate level, material is included in the Computer Architecture sequence, from instruction-level parallelism to SIMD and MIMD architectures. The Operating Systems sequence devotes a large percentage of time to distributed systems, with some mention of multiprocessors. Recently, an undergraduate course on the Theory of Parallel Computation has been introduced, covering algorithms for interconnection networks and PRAMs.

At the graduate level, the department offers courses on Neural Networks and VLSI Algorithmics. Also, the graduate course in Physical Database Design incorporates material on parallel disk arrays.

There are also several graduate-level seminars, offered with varying levels of frequency. Among the topics offered are Parallel Algorithms, Shared-Memory Multiprocessors, VLSI Layout Algorithms, and High-Performance Scientific Computing.

Ohio University - Athens

The Computer Science Department offers courses in Concurrent Programming and Parallel Computing, both of which are cross-listed for graduate and undergraduate students. They also offer an advanced graduate course in Parallel Computing.

Oregon Graduate Institute of Science and Technology

The Computer Science and Engineering Department incorporates aspects of parallelism into their Issues in Contemporary Operating System Design, Contemporary High-Performance Architectures, and Highly Parallel Architectures courses. In addition, the following courses are offered.

- Compiling for High Performance Computers.

  Vector and parallel processors, superscalar and wide-instruction architectures impose special requirements on compilers in order to utilize these high-performance architectures effectively. This course explores techniques for automatic program analysis and code generation for such architectures. Topics include data and control flow computations, detection of parallelism, data dependence analysis, optimal register allocation, instruction scheduling and program restructuring transformations.

- Advanced Computer Architecture.

  This course covers cache memories, pipelined arithmetic units, multiple functional units, high performance processors, VLIW and superscalar processors, vector processing, interleaved memories, multiple processors, shared memory processors, message-passing multicomputers, distributed shared memory systems and latency hiding, massively parallel processors, performance evaluation, and compiler optimizations. Example systems will include the Sequent Symmetry, Intel iPSC, Cray Y-MP, IBM RS/6000 and Connection Machine.

- Scientific and Parallel Programming.

  This is a project-based course in parallel programming for scientific applications. Algorithms are expressed as message-passing programs. Experiments are performed on networks of workstations and on a distributed memory parallel computer. The course includes the following topics: solution of linear algebraic equations, random numbers and Monte Carlo methods, minimization of functions, combinatoric optimization, systems of ordinary differential equations, and systems of partial differential equations.

Oregon State University

The Computer Science Department offers the following undergraduate course.

- CS 475: Introduction to Parallel Computing.

  Theoretical and practical survey of parallel processing, including a discussion of parallel architectures, parallel programming languages, and parallel algorithms.

In addition, the following graduate courses are offered.

- CS 575: Introduction to Parallel Computing.

  Theoretical and practical survey of parallel processing, including a discussion of parallel architectures, parallel programming languages, and parallel algorithms.

- CS 576: Parallel Programming Languages.

  In-depth study of languages used to program parallel computers, including low-level languages, machine-dependent languages, higher-level languages, and trade-offs between portability, programmability, and performance.

- CS 577: Parallel Architectures.

  In-depth study of advanced concepts in parallel computer architectures; design, management, and modeling of memory hierarchies, pipelined, and multiprocessor systems; SIMD, MIMD, and Data-Flow machines. Interconnection networks and conflict free access in parallel memory systems. Studies of recent research work in parallel processing.

- CS 578: Parallel Algorithms and Theory.

  Design and analysis of parallel algorithms. PRAM and its variants. Parallel computation thesis. Parallel complexity classes. Analysis of interconnection networks.

- CS 579: Topics in Computer Architecture and Parallel Processing.

  Current topics in advanced computer architecture and parallel processing.

Parasoft Corporation

Parasoft offers classes on parallel processing which cover general descriptions of parallel processing hardware, a review of parallel and distributed processing operating systems, APIs, theory of data distribution, parallel and distributed programming models, debugging and performance analysis.

Pennsylvania State University

The following graduate-level courses are offered in the Department of Computer Science and Engineering.

- CmpSc 538: Parallel Algorithms.

  Computational aspects of VLSI: synthesis/analysis of efficient parallel and distributed algorithms; computational structures; models of parallel computers and their interrelationships.

- CmpSc 561: Sequential and Parallel Complexity Theory.

  Models of sequential and parallel computers; relationships between complexity measures; simulations and universality; resource-bounded hierarchies; lower-bound techniques.

- CmpSc 574: Contemporary Computer Architectures.

  Contemporary computer organizations and operations; parallel systems, pipelined, array, multiprocessor; shared memory and message passing architectures; data flow architectures.

Polytechnic University

Parallelism is the underlying theme of the advanced graduate-level architecture course, which covers pipelined and superscalar, vector, SIMD and MIMD machines. In addition, special topics courses are occasionally offered on parallel software environments and parallel algorithms.

Purdue University

The Department of Computer Science offers the following courses.

- CS 590V: Parallel and Vector Computations.

  This course studies the experimental and theoretical evaluation of architectures, algorithms and software systems for high speed computation. The topics covered include: Cray system organization and instructions set analysis, bus and distributed memory architectures, massively parallel systems and transputer boards, mapping and allocation algorithms, linear system solvers, PDE solvers, combinatorial and graph algorithms, parallel back-propagation and neural networks, parallel and vector extensions of procedural languages, automatic parallelization methods, parallel data-flow and functional languages, performance evaluation, debugging and monitoring tools, ELLPACK, SIMPLEX, PARAGRAPH. Programming projects to provide hands on experience on several machines are available.

- CS650: Computational Aspects of Parallel Processing.

  This course focuses on the design and implementation of parallel algorithms on different parallel models, the theory of parallel computation, and other theoretical issues. The focus is on general paradigms, with specific algorithms covered to illustrate important general principles. Topics include parallel models of computation, including PRAMs, arrays, meshes, and trees, simulation results, parallel algorithm design techniques, an introduction to parallel complexity theory, lower bound techniques, processor-time and area-time tradeoffs, and VLSI layout techniques.

The Electrical Engineering School introduces aspects of parallelism in its dual-level (advanced undergraduate / graduate) course in Computer Architecture. In addition, the following graduate-level courses are offered (EE 595S is a dual-level course).

- EE 595S: Programming Parallel Machines.

  This course studies how to program parallel processing systems. Three types of parallel architectures are considered: SIMD (synchronous), MIMD (asynchronous), and mixed-mode (SIMD/MIMD hybrid). A variety of machines that represent these classes are used in the course.

- EE 666: Advanced Computer Systems.

  This course studies theoretical aspects of advanced computer systems where multiprocessing is used. Topics include the design, architecture, and performance evaluation of multiprocessor memories, interconnection networks, and computation pipelines. Also included are the topics of scheduling, synchronization, resource allocation, load-balancing, partitioning, and deadlock avoidance in multiprocessors. Also covered are the design and analysis of parallel algorithms, programming languages, and automatic approaches to parallelism detection/exploitation for concurrent computation.

- EE 667: Parallel Processing.

  The design of large-scale parallel processing systems is studied. Synchronous (SIMD) and asynchronous (MIMD) machine organizations are discussed. Single-stage and multistage interconnection networks for such systems are examined in depth. Case studies of existing and proposed systems are presented.

- EE 660: Parallel Algorithms.

  The purpose of this course is to introduce students to the fundamental theory of parallel algorithms and to explore parallel algorithm principles via the study of important example algorithms.

- EE 663: Compiler Coder Generation, Optimization, and Parallelization.

  This course presents the concepts needed to design and implement production quality code generators for any of the the more popular languages and families of computer architecture, including various pipelined and macro-parallel machines. Flow analysis and concurrency detection, as well as optimizations and loop and irregular code parallelizations, are covered in detail.

- EE 695A: Parallel and Distributed Computation for Optimization and Control.

  The purpose of the course is to introduce techniques for solving large-scale optimization and control problems in a parallel / distributed computing environment. Emphasis is placed on design and analysis of distributed numerical algorithms for nonlinear problems. The issue of synchronous versus asynchronous computation is an underlying theme throughout the course.

- EE 695F: Algorithm-Specific Parallel Processing.

  An introduction to the theory and fundamental techniques for the design of algorithmically specialized computing structures. The course provides a rigorous framework for modeling and transforming algorithms in order to derive optimal space/time mappings of algorithms into VLSI architectures and/or parallel processors. Applications of the course content include high-level VLSI design and optimizing compilers.

- EE 695G: Distributed Computing Systems.

  This course discusses the design issues of distributed computing systems (DCS). The general theory of distributed transaction management, reliability, and resource management is discussed. Various algorithms and specification methodologies for DCS are introduced. A general coverage of three major areas of DCS, namely: distributed operating systems, distributed databases, and distributed AI, is provided. The discussion is augmented with various case studies.

Randolph-Macon College

The Department of Computer Science incorporates aspects of parallelism into undergraduate courses on Computer Organization and Analysis of Algorithms.

Rice University

A multidisciplinary effort involving parallelism involves the Department of Computational and Applied Mathematics and the Department of Electrical and Computer Engineering. The following are upper-division undergraduate courses.

- C&AM 420: Introduction to Computational Science.

  Vector and parallel architectures, parallel numerical algorithms, analysis and enhancement of performance, scientific visualization, use of programming tools and environments.

- COMP 421: Operating Systems and Concurrent Programming.

  Introduction to the design, construction, and analysis of concurrent programs with an emphases on operating systems, including file systems, schedulers, and memory allocators.

The following are graduate-level courses, some of which may be taken by upper-division undergraduates.

- COMP 515: Advanced Compilation for Vector and Parallel Processors.

  Advanced compilation techniques for vector and parallel computer systems, including the analysis of program dependence, program transformation to enhance parallelism, compiler management of the memory hierarchy, interprocedural data flow analysis, and parallel debugging.

- COMP 520: Distributed Systems.

  Multiprocess structuring, and interprocess communication. File access and memory management. User interfaces.

- COMP 525: Advanced Computer Architecture.

  Design issue of pipelined, vector, and multiprocessor architectures. Development of performance evaluation techniques to model and simulate configuration of concurrent architectures.

- COMP 551: Advanced Numerical Linear Algebra.

- C&AM 571: Advanced Integer and Combinatorial Optimization.

  Emphasis on methods of solving various optimization problems with discrete components.

Rochester Institute of Technology

The Computer Science Department offers the following courses.

- ICSS-531/ICSG-735: Introduction to Parallel Computing.

  A study of the hardware and software issues in parallel computing. Topics include an introduction to the basic concepts, parallel architectures, parallel algorithms, parallel languages, network topology, coarse versus fine grained parallelism, applications, parallel programming design and debugging.

- ICSS-532/ICSG-715: Parallel Algorithms and Program Design.

  A study of the principal trends in parallel algorithm design, through the analysis of algorithms used in various areas of application. Specific techniques that have gained widespread acceptance will be highlighted. The course will investigate the interplay between architecture and algorithmic structure and will discuss the effect that these issues have on the complexity and efficiency of parallel algorithms.

St. Bonaventure University

The Department of Computer Science offers a senior-level undergraduate course on parallel computation, which presents a survey of parallel architectures and examines algorithms for MIMD and SIMD architectures.

St. John Fisher College

A course in Parallel Processing is offered every other year. This course includes a history of parallelism, SIMD and MIMD models, fundamental algorithms such as sorting and matrix transpose, and related architectures including systolic arrays and data flow machines. In addition, aspects of parallelism are incorporated into traditional courses in Data Structures and Programming Language Concepts.

San Jose State University

The Mathematics and Computer Science Department offers the following course, which is cross-listed for both undergraduate and graduate students.

- CS 159: Introduction to Parallel Processing.

  This course is designed to provide a broad background on parallel computing, covering architectures, programming, and algorithms. The architecture portion provides a short overview of some of the major issues, including shared-memory, distributed-memory, and pipelined vector processor machines. Some measures of performance are introduced and Amdahl's law is discussed. The programming portion examines communication and synchronization of processes as well as the ability to protect shared variables. Some of the language extensions used to resolve these issues are presented. The algorithms section considers some of the areas in which parallel algorithm development has been effective, including matrix algorithms, linear recursion, sorting and searching, and a class of algorithms designed specifically for distributed processor networks (explorer-echo algorithms).

Sangamon State University

The Computer Science Program offers an undergraduate course on Parallel Processing, and incorporates aspects of parallelism into courses on Operating Systems, Programming Languages, Compiler Construction, Database, Computer Architecture and Networking.

State University of New York at Buffalo

The Department of Computer Science offers the following graduate courses.

- CS515: Introduction to Parallel Computing.

  General introduction to parallel computing, with an emphasis on different approaches to programming a variety of shared memory, distributed memory, SIMD, and MIMD parallel machines. Fundamental issues about parallel architectures, languages, and algorithms are included.

- CS606: Parallel Computer Architecture.

  The organization of parallel computer systems. Different parallel architectures, such as SIMD and MIMD parallel computers, shared memory and distributed memory computers, as well as dataflow architectures are presented.

- CS622: Parallel and Distributed Operating System Concepts.

  Theoretical results in performance evaluation of computer systems introduced through synthetic and analytic models of system structure; techniques for verifying properties of concurrent processes; issues in multiprogramming operating system mechanism such as naming and binding, protection, reliability, and recovery.

- CS633: Parallel Algorithms.

  The design and analysis of parallel algorithms. Time and space complexity of problems from graph theory, computational geometry, combinatorial algorithms, and image processing. Models of computation include pram, mesh, pyramid, mesh-of-trees, and hypercube.

- CS643: Compilers for Parallel Computers.

  Implementation of programming languages on parallel machines: programming constructs for implicit and explicit parallelism, dependency analysis, partitioning, optimization, and code generation for parallel machines.

State University of New York at Geneseo

The Department of Computer Science offers an advanced undergraduate course in parallel computing every other year.

State University of New York at Oswego

The Department of Computer Science offers an undergraduate course in Parallel and Distributed Systems. The course contains a survey of parallel hardware and software computing technology, and an introduction to parallel programming. Topics include an introduction to parallel architectures, parallel algorithms, parallel languages, parallel algorithm design, analysis and debugging.

Swiss Scientific Computing Centre

The Research and Development Section of the Swiss Scientific Computing Centre (CSCS) has recently introduced a summer student internship program which focuses on research in parallel computational science. The program involves approximately ten students (both graduate and undergraduate) from around the world.

A week of lectures and exercises provides an introduction for students to the parallel architectures used at CSCS and the software used to program them. Individual students are then given projects and assigned advisors; the project work goes on for up to ten weeks.

Internship students are treated as active researchers at CSCS, and are expected to participate in project meetings, and also to write a technical report and give a presentation to the staff at CSCS when their projects are completed.

Taylor University

The Department of Computing and System Sciences incorporates aspects of parallelism into its undergraduate Computer Organization, Algorithms, and Language Structures courses.

Texas Institute of Technology

The following graduate courses are offered.

- CS 5365: Principles of Multiple-Processor Systems.

  Comprehensive introduction to the field of parallel and distributed systems. Algorithms, architectures, networks and systems. Theory and applications.

- CS 5379: Parallel Processors and Processing.

  Theory, architectures, and algorithms for design and implementation of parallel computing systems. Operating system requirements for parallel computing. Current approaches and constraints.

United States Air Force Academy

An undergraduate special topics course is offered which provides a survey of parallel processing.

Universidade Federal de Mato Grosso do Sul

The Department de Computacao e Estatistica offers a class in parallel processing to undergraduates in computer science.

University of Arizona

The Department of Computer Science incorporates aspects of parallelism into its undergraduate Operating Systems course and also offers the following course to both graduate and undergraduate students.

- 522: Principles of Concurrent Programming.

  Fundamental concepts of concurrent programming, synchronization mechanisms based on shared variables and message passing, systematic development of correct programs, paradigms for distributed programming.

University of Arkansas

The Department of Electrical Engineering and Computer Systems Engineering offers two graduate courses in Parallel Processing. The first is a course in Parallel Programming. The second is a course in Computer Architecture covering memory system design, pipeline techniques, SIMD and MIMD computers, multiprocessor control, and RISC multiprocessor design.

University of Bergen, Norway

The Computer Science Department offers an advanced undergraduate course covering PRAMS, networks for parallel computers, parallel permutations procedures, gray codes, and structures sparse matrices.

University of California, Berkeley

The Computer Science Department incorporates aspects of parallelism into courses on Advanced Programming Language Implementation, Computer Networks, Advanced Topics in Operating Systems, and Advanced Programming Languages. In addition, the following graduate-level courses are offered.

- CS 258: Parallel Processors.

  Parallelism, its representation; models and their properties. Parallelism detection, scheduling of parallel processes. Principle of pipeline computation, classification, scheduling; current trends in pipeline architectures. Computer network, deadlock, protocol, routing, global information management, security, distributed operating systems and databases.

- CS 267: Applications of Parallel Computers.

  Parallel architectures, parallel languages and programming systems, applied mathematics, algorithms and applications in scientific computing including PDEs, linear algebra, and sorting.

- CS 273: Foundations of Parallel Computation.

  Shared memory models of parallel computation: parallel algorithms for linear algebra, FFT, sorting, recurrences, and graph problems; interconnection network models; algorithm design for networks like hypercubes, shuffle-exchange, trees, meshes and butterflies; systolic arrays; and message routing.

University of California, Irvine

The Department of Information and Computer Science offers the following graduate-level courses.

- 222: Parallel Computer Architectures and Languages.

  Introduction to the principles of parallel processing. Surveys fundamental organizations of multiprocessor architectures and examines their programmability. The focus is on developing software for such machines.

- 228: Parallel Algorithms and Architectures.

  Definition of parallel processing systems. Theory and application of arrays, trees, and hypercubes. Computational models and applications.

University of California, Los Angeles

The Computer Science Department offers a graduate course in Parallel Architectures which covers reasons for parallel computers, parallel algorithms and their representation, classification of algorithmic structures, execution models and computer organizations, granularity and coupling, performance evaluation measures, approaches for high speed, and general-purpose and special-purpose architectures. A course is also offered in Parallel Programming, which covers mutual exclusion and resource allocation in distributed systems, primitives for parallel computation, specification of parallelism, interprocess communication and synchronization. Finally, a section of the graduate Algorithms course covers distributed and parallel algorithms, including models of message-passing systems, distributed synchronization, fair-safe compilation, models of parallel algorithms, and so forth.

Two undergraduate courses are offered. The first course in Parallel Programming involves mutual exclusion and resource allocation in distributed systems, primitives for parallel computation, specification of parallelism, interprocess communication and synchronization, atomic actions, binary and multiway rendezvous, and so on. The second course in Parallel Languages and Compilers focuses on environments for portable parallel programming, process synchronization algorithms, heterogeneous parallel computing, distributed shared memory, concurrent data structures, consistency models, programming shared memory architectures, and more.

University of California, San Diego

The Department of Computer Science and Engineering offers the following undergraduate course.

- CSE 174: Introduction to Parallel Computation.

  An introduction to the major issues in parallel computation. A survey of parallel hardware, software, algorithms and models is given.

At the graduate level, the department runs a weekly seminar on parallel computation, offers topics courses involving parallelism, and offers the following courses.

- CSE 260: Graduate Parallel Computation.

An overview of parallel hardware, software, algorithms and models. Topics include: parallel programming models and taxonomies, commercially available parallel hardware, parallel programming environments and tools, heterogeneous computing, and so forth.

- CSE 261: Parallel and Distributed Computation.

  The concentration is on developing easy-to-parallelize numerical algorithms for optimization. Topics selected from: iterative methods for linear and non-linear equations, network problems, and so forth.

- CSE 262: Systems Support for Parallel Scientific Computation.

  Issues arising out of the design of software support for concurrent numerical computation. Topics include software portability, load balancing and some numerical analysis.

- CSE 263: Parallel Algorithms.

  An introduction to parallel algorithms on the mesh, tree, hypercube, PRAM and related architectures. The algorithms include sorting, matrix algorithms, graph algorithms and FFT.

University of California, Santa Cruz

Aspects of parallelism are incorporated into traditional courses in Operating Systems and Programming Language Concepts. The following graduate course is also offered.

- CIS220: Parallel Processing.

  An introduction to massively parallel computing on array processors and multiprocessors. A study of special- and general-purpose parallel architectures and the applications they support. Networks and memory hierarchies for multiprocessors. Evaluation of programming methodologies. This course emphasizes architectures that scale to thousands of processing elements.

University of Central Florida

The Department of Computer Science offers an undergraduate concentration in parallel computing, in which the students must take Parallel Computing Concepts, Introduction to Parallel Algorithms, and two of Parallel

Computer Architectures and Organization, VLSI and Parallel Computation, Parallel Programming Languages, and Applications of Parallel Computing. Descriptions of these courses follow.

- COP 4932: Parallel Programming Concepts.

  Flynn's hardware taxonomy, measures of performance, parallel programming models, scheduling parallel programs, parallelizing serial programs, programming support environments and applications.

- COT 4932: Introduction to Parallel Algorithms.

  Models for parallel computation, parallel algorithms and implementation for classical problems such as selection, sorting, merging and graph problems, and concurrent access data structures.

- COP 4932: Parallel Programming Languages.

  Parallel programming models and languages: dataflow, functional, logic, object oriented, and constraint. Programming, debugging and performance monitoring tools, and basics of operating systems for parallel computers.

- CDA 4932: Parallel Computer Architectures and Organization.

  Sequential and parallel architectures, SIMD and MIMD architectures, interconnection networks, multiprocessor memory and I/O system organizations.

- CAP 4932: Applications of Parallel Computing.

  Survey of commercially available machines and their performance capabilities, numerical and scientific computing, and applications in non-numeric computing.

- CDA 4932: VLSI and Parallel Computation.

  Physics and technology of semiconductors, CMOS logic and circuit design, subsystems design, systolic and parallel algorithms.

In addition, the Department of Computer Science offers the following graduate-level courses.

- CDA 5110: Parallel Architecture & Algorithms.

General-purpose vs. special-purpose parallel computers; arrays, message-passing; shared memory; taxonomy; parallelization techniques; communication synchronization and granularity; parallel data structures; automatic program restructuring.

- CDA 6108: Selected Topics in Computer Architecture.

  Selected research papers on multiprocessors, data base machines, virtual machines, ultra-computer, connection machine, MPP, Butterfly architecture, data-flow architectures, object-based architectures, fault-tolerant architectures.

- COT 6415: Complexity of Parallel Computation.

  Theoretical models: justification and buildability. Inherent parallelism and communication costs. Lower and upper complexity bounds. Parallel computation thesis. NC, SC classes; paradigms of parallel algorithms.

University of Colorado

Funded by the National Science Foundation, a one-year course and supporting laboratory facility is being developed for undergraduate students, which focuses on high-performance scientific computing.

The objectives of the project are as follows.

- To teach undergraduate students how to use high-performance computing systems in scientific and engineering applications.
- To provide an advanced computing environment for undergraduate instruction in particular applications.
- To develop instructional materials: course notes, problem examples and exercises, manuals, and software.
- To export our instructional materials and experience to support undergraduate studies at other institutions.
- To encourage interdisciplinary studies for computer scientists and engineers.
- To encourage research scientists to become involved in undergraduate education.

- To prepare students for work in the field of computational science.

In the first semester, the concentration is on the analysis of the performance of a computer and the codes that run on it. As a case study, a problem in molecular dynamics is considered for a variety of architectures. For each architecture, the organization of the problem is studied with respect to achieving the best possible performance. This exercise is aimed at distinguishing the strengths and weaknesses of the different architectures. In the second semester, problems in heat conduction and image reconstruction are considered for the three machines.

The laboratory has been established, and the course was offered in academic year 1991-92 and academic year 1992-93. It makes use of DEC and SGI workstations, a Cray Y-MP, a CM-2, and an Intel iPSC/2 hypercube multiprocessor. The course begins with introductions to the machines and performance measurements. Students also study the computational and visualization tools Matlab, IDL, and AVS. The concepts and tools learned in the early chapters are then applied to the solution of numerical problems in molecular dynamics, advection, and tomography.

The machines chosen for this course are representative of MIMD and SIMD distributed-memory multiprocessors and vector computers. In the future, the course might make use of the CM-5, the Intel Paragon, the KSR1, a network of workstations connected by PVM, and possibly with downsizing materials, laptops or PC's.

Final drafts of course material are available by anonymous FTP in the directory /pub/cs/HPSC on cs.colorado.edu. It is expected that all chapters of the laboratory manual and accompanying software and tutorials will be complete by Fall 1993.

University of Connecticut

The Department of Computer Science and Engineering offers the following undergraduate course.

- CSE 228: Introduction to Parallel Systems

  An introduction to the theory of parallel computation, parallel architectures and parallel algorithms.

In addition, the Department of Computer Science and Engineering offer the following courses for graduate students.

- CSE 330: Computer Networks and Distributed Systems

  A survey of networked/distributed architectures, concentrating on the distributed model. Performance analysis of distributed systems.

- CSE 332: Parallel Processing

  Models of parallel and distributed architectures, performance analysis and algorithms.

- CSE361: Computational Complexity of Sequential and Parallel Algorithms

  The theory and design of parallel algorithms; parallel models.

University of Delaware

The Department of Computer and Information Sciences offers the following courses.

- CIS 662: Computer Systems Architecture.

  Recent advances in computer architecture and their impact on architectural design. Multicomputer and multiprocessor architectures. Parallel machines.

- CIS 649: Parallel Computing for Applications.

  Principles of parallel computation with special emphasis on methods suited to applications programming. Techniques for the development, analysis, testing, and validation of parallel programs.

- CIS 809: Parallel Algorithms.

  Design and analysis of parallel algorithms. Emphasis on both specific important parallel algorithms and general techniques used in the design of parallel algorithms. Various complexity issues associated with parallel algorithm design.

- CIS 879a: Introduction to Parallel Computing.

  Introduction and overview of the field of parallel computing including a survey of different paradigms of parallelism. Measures of performance, parallel architecture overview, parallel programming paradigms, scheduling, automatic parallelization, debugging, parallel programming environments.

- CIS 879b: Optimizing and Parallelizing Compilers.

  Issues in building a highly optimizing compiler for various architectures. Sophisticated static program analyses, optimizing program transformations, advanced compiler techniques for improving the effective utilization of vector and parallel machines.

University of Edinburgh

The School of Engineering and Information Technology offers a parallel systems theme at the graduate level involving the following five core modules.

- The concurrent architectures module covers topics in concurrent computer architecture, including pipelining, multiple execution units, vector processing, data parallel (SIMD) and multiprocessor (MIMD) architectures.
- The computer communications module focuses on the problems involved with constructing communication networks.
- The distributed systems module examines issues in the design and implementation of distributed systems, such as communication primitives, naming, authentication, protection, load distribution, resource allocation and reliability.
- The principles of parallel computing module covers the theory of computation and its relationship to practice. Topics include parallel computation and their mappings to architectures and languages, parallel algorithm analysis, scalability of parallel systems, parallel complexity classes, communication and synchronization principles, memory management and shared memory simulation.
- The software for parallel computers module covers and introduction to parallel programming, parallel languages for message passing and shared memory, languages for SIMD machines, parallel algorithms and models of parallel computation.

University of Florida

The Department of Computer and Information Sciences offers a special topics course in parallel computer architecture which concentrates on design concepts in pipeline and parallel architectures. The first part of the course deals with the design of instruction-level parallelism, such as superscalar, VLIW, and superpipeline processors. The second part deals with basic architectural elements for parallel processing including interconnection networks, processing elements and memories (including caches). Practical design examples, such as existing parallel architectures of SIMD, MIMD, pipelined computers, and data flow architecture, are use to illustrate design decisions. The third part covers the execution environment for parallel tasks, including language constructs, parallel programming, compiler support for parallelism, operating system support, scheduling, synchronization, and load balancing. Performance evaluation of parallel systems is also covered.

In addition, a course focusing on the theory and practice of parallel algorithms is offered. This course covers topics including a taxonomy of parallel architectures, shared-memory and distributed memory models, programming paradigms and languages, and SIMD and MIMD algorithms for numerical and non-numerical applications. In addition, the course covers the basics of synchronization, interconnection networks, and performance issues.

University of Houston-Downtown

The Department of Computer and Mathematical Sciences offers an undergraduate course in Parallel Computing. In addition, aspects of parallelism are covered in the Advanced Numerical Methods course.

University of Idaho

The Computer Science Department incorporates aspects of parallelism into its graduate-level course in computational complexity. In addition, the following course is offered.

- CS 413/513: Concurrent Systems.

  This is a joint undergraduate/graduate course which serves as a survey of the computational issues surrounding concurrency. The focus of the course is on the application of formal methods to the programming of concurrent systems. Topics include the basics of parallel architecture, issues in distributed computing (e.g., mutual exclusion, termination detection, and readers-writers), and parallelization.

University of Illinois at Urbana-Champaign

The following computer science courses are offered.

- CS 328: Computer Networks and Distributed Systems.

  Introduction to concepts of transport connections and sessions; design issues in transport layer and session layer protocols, terminal and file transfer protocols, message handling protocols; methods to ensure network security and privacy; algorithms for deadlock detection, concurrency control and synchronization in distributed systems; models of distributed computation; networking facilities and resource control and management methods in network and distributed operating systems.

- CS 397: Introduction to Parallel Programs.

  Characterization of performance, array/data parallel programming, memory and interconnection network issues in array programming, shared-memory parallel programming, distributed-memory parallel programming, and description of real systems.

- CS 433: Theory of High-Speed Parallel Computation.

  Theoretical aspects of parallel and pipeline computation; time and processor bounds on classes of computations; data alignment network speed and cost bounds; conflict-free access memories; and overall computer system ideas.

- CS 454: Parallel Numerical Algorithms.

  Introduction of numerical algorithms for vector and parallel computers: parallel algorithms in numerical linear algebra (dense and sparse solvers for linear systems and the algebraic eigenvalue problem), numerical handling for ordinary and partial differential equations, and numerical optimization techniques.

University of Iowa

The Department of Computer Science incorporates aspects of parallelism into its Operating Systems and Introduction to System Software courses. In addition, the following graduate-level course is offered.

- 22C:132: Parallel Programming.

  The goal of this course is to teach students the current methodology for parallel program development and to present the reasoning logics (such as temporal) about the correctness of parallel programs.

University of Massachusetts, Dartmouth

The Computer and Information Science Department offers the following senior-level technical elective undergraduate course.

- CIS 422: Design of Parallel Algorithms

  Concepts of parallel computing: speedup laws, utilization, scalability. Parallel programming paradigms: process communication with channels, data sharing, synchronous parallelism. Fundamental parallel architectures and parallel algorithms, including sorting (bitonic), searching, matrix multiplication, prime number computation, connected components, minimal spanning tree, Jacobi relaxation, solution to linear equations. Synchronization techniques. Multi-Pascal software package and LINDA workstation-based programming environments are also introduced.

In addition, the following graduate-level courses are offered.

- CIS 574: Advanced Computer Architecture.

  Study of parallelization issues in uniprocessor systems (memory subsystem - interleaved memories, cache memories; I/O subsystem). Pipeline processing techniques and pipeline computers. Vector computer techniques and vector computers. Interconnection networks: design and performance-related parameters. MIMD vs. SIMD multiprocessors and multicomputers. Synchronization techniques. Scalability issue in parallel architectures and its relation to performance of parallel algorithms in massively parallel systems. Examples of parallel and massively parallel systems are discussed (CM-5, CRAY T3D, Kendall Square).

- CIS 525: Topics on Parallel Computations.

  Course entirely devoted to mathematical modeling of parallel and distributed systems (hardware and software) by means of high-level Petri nets. Special attention is devoted to Colored Petri Nets and their hierarchies. Mathematical and behavioral properties of this model are studied. Simulations of distributed software systems by means of Colored Petri nets are performed to check reachability, deadlock-freeness, and fairness properties. Elements of timed Petri nets are discussed.

- CIS 575: Parallel Algorithms and Parallel Architectures.

  Selected parallel algorithms are discussed: bitonic sort, homomorphisms of finite automata, multi-dimensional searches. Advanced techniques in parallel algorithms are discussed: parallel dynamic programming, actor-based object-oriented concurrent systems. Isoefficiency analysis of parallel algorithms has the central place in the whole course.

University of Minnesota

The following courses are offered.

- cs5151. Introduction to Parallel Computing I & II.

  The objective of the course is to introduce techniques for designing and analyzing parallel algorithms for a variety of problems. Both numerical as well as nonnumeric applications are covered. Suitability of different commercially available and/or practically feasible architectures for executing these parallel algorithms are considered.

- Sci 8001: Parallel and High Performance Computing.

  The objective of the course is to introduce parallel and high performance computing to persons carrying out graduate work involving scientific computations.

University of Nebraska-Omaha

The Department of Mathematics and Computer Science offers courses entitled “Introduction to Parallel Computing” and “Advanced Topics in Parallel Computing.”

University of Notre Dame

A graduate-level course covering principles of parallel computing is being introduced in the Department of Computer Science and Engineering, which will cover parallel architectures and software. In addition, a course on VLSI Processor Arrays covers the design of highly parallel, as well as pipelined, systolic and wavefront arrays for special scientific applications.

University of Oklahoma

The School of Computer Science offers the following courses.

- CS 4743: Large Scale Scientific Computing.

  This is a course on parallel and vector computing with emphasis on matrix problems.

- CS 5613: Interconnection Networks.

  This is a course on the analysis and the design of dyamic and static networks for shared and distributed memory organizations.

- CS 5413: Computational Complexity.

  This course focuses on parallel complexity issues and randomized algorithms.

University of San Francisco

The Department of Mathematics offers a basic introductory course in Parallel Processing, which focuses on algorithms for scientific computing on distributed memory machines. An advanced course devoted to a specific project is also offered.

University of Southern California

The following advanced graduate-level courses are offered in the Computer Science and Electrical Engineering Departments.

- CS 620: Design and Analysis of Parallel Computation.

  Routing and algorithm design for parallel machines. PRAM model, fixed connection models, circuit depth, randomization. Parallel algorithms including independent set, matching, evaluation, linear system, sparse techniques.

- EE 653: Data-flow Computers and Functional Programming.

  Programmability of general purpose multiprocessor systems, functional programming, data-flow multicomputers, existing prototypes, fault-tolerance.

- EE 657: Parallel Processing.

  Array processors, multiprocessors, pipeline processors; data flow computers; VLSI architectures; parallel numerical and non-numerical algorithms; mapping algorithms onto computer structures.

- EE 659: Interconnection Networks.

  Theory, design and analysis of interconnection networks for multiprocessor systems. Study of regular, Omega, binary n-cube, Delta, Gamma networks; control algorithms; mapping algorithms to architecture.

- EE 661: Parallel Knowledge Processing.

  Parallel approaches to knowledge representation and reasoning. Topics range from the analysis of parallelism to the design of special purpose computer systems.

University of Tennessee

The Department of Computer Science offers the following courses.

- CS 435/494: Microprocessors, Multimicroprocessor Systems, and LANs. The first third of the course deals with state-of-the-art microprocessors (e.g. 486, Pentium, and Alpha), reservation tables, and caches (including coherence problems in parallel systems). The middle third covers multimicroprocessor systems, including hypercubes, shared-memory machines, the Intel Paragon, the MasPar MP-1, and the PVM system. Students will write programs on two different systems.

- CS 535: Architecture.

  The emphasis is on parallel architectures. Students write programs on distributed-memory and shared-memory machines.

- CS 594: Algorithms for Parallel Architectures.

This course reviews various features of advanced architectures and discusses how they affect the potential performance of linear algebra algorithms. Recent techniques devised for utilizing advanced architectures will be considered more fully, especially the design of the Level 1, 2 and 3 BLAS. A new package, called LAPACK, which is intended to most fully exploit advanced computers, will be discussed. In addition, a programming environment that will aid in the design, development and understanding of parallel algorithms for high-performance computers will be described. The tools provide an environment for developing and analyzing new parallel algorithms in Fortran that require sophisticated synchronization at a large-grain level. This package provides portability of a user's code and also has a graphics postprocessor for performance analysis and debugging. Finally, the challenge facing designs of mathematical software in view of the development of massively parallel computer systems will be addressed.

- CS 594: Parallel Programming.

  This course covers the basics of parallel programming, fundamental theoretical models, and tools for evaluating and assimilate new technology in parallel computing. Students will gain practical hands-on experience with several types of parallel computers and several models of parallel programming, including message passing, shared memory, and data parallel models. Students will implement programs on each of these models.

- CS 690: Parallel Computing.

  Areas covered include a description of different types of parallel architectures, the development of parallel algorithms for solving important numerical problems, and software tools intended to aid implementation of parallel numerical computation.

- CS 593: Seminar in Parallel Computing.

  This is a survey of programming tools and techniques for parallel computing, with emphasis on software environments for distributed network computing, message-passing paradigms, object-oriented programming models, and programming tools for task partitioning and scheduling.

University of Wisconsin-Madison

The Computer Sciences Department incorporates aspects of parallelism into its graduate-level Operating Systems and Robotics courses. In addition, the following courses are offered for graduate students.

- CS757: Advanced Computer Architecture II.

  Topics include shared-memory MIMD machines, private-memory MIMD machines, SIMD machines, parallel software, interconnection networks, and cache coherence and dataflow.

- CS838-3: Parallel Computing and Computation Science.

  This is an experimental interdisciplinary course that introduces parallel programming models and computation science techniques to computer scientists, scientists, and engineers. Data parallel, shared memory, and message passing models are discussed.

Union College

The Department of Electrical Engineering and Computer Science offer a course on concurrent programming for graduate and upper-level undergraduate students.

Walla Walla College

Aspects of parallelism are presented in the introduction to programming paradigms course. In addition, there is a course that focuses on Parallel and Distributed Computation.

Wellesley College

An introductory course is offered that gives an in-depth presentation to parallel computing. It includes basic algorithmic techniques, interconnection networks, implementation decisions, and a case study of a problem in scientific computing.

Acknowledgments

I would like to thank my assistant, Jeannine Early, for learning how to deal with the LaTeX mail merge feature and getting the hard copy queries out in a timely fashion. I would also like to thank those who took time out of their schedules to respond to either the electronic or hard copy query, as well as to the members of the TCPP executive and advisory committees who reviewed an early draft of this manuscript. Finally, this work was performed while the author was on sabbatical at The Medical Foundation of Buffalo Research Institute.