β-test- ParProc-TC v1.0

Parallel Processing Training

Parallel Processing Training Courseware *v*-1.0

Designed for Testing, Benchmarking & Performance Activities

(Draft Copy: To be Reviewed on September 01, 06)

Document Title	Parallel Processing Training Courseware (**β-test-ParProc-TC-*v*1.0**)
Users	Betatesting Group Members of NPSF, C-DAC,Pune
Source	Parallel Computing Workshops conducted by Betatesting Group, & Author's Experience on Parallel Processing Projects, Other Web-sites
Date:	August 01, 2006
No. of Pages	8
Author	VCV.Rao
Contact	betatest@cdac.in

Beta (β)-Testing Group, NPSF

Beta (β) -testing Group, National PARAM Supercomputing Facility, Centre for Development of Advanced Computing (C-DAC)

Non-Disclosure

This material is developed and provided by the Centre for Development of Advanced Computing (C-DAC), Pune, India. Licensor grants to Licensee, an Indian Government agency or a non-profit educational institution ("Licensee"), a royalty free, non-transferable license to use the copy of the product as per terms and conditions of this License Agreement. Licensee is permitted to use the product solely for Licensee's own internal research purposes, and alter, modify, adapt the product for Licensee's own internal research purposes.

© Copyright 2006
Centre for Development of Advanced Computing (C-DAC)

1. Objective

The complete courseware forms a single concentrated course on parallel computing, which is a continuously evolving High Performance Computing resource at NPSF. The courseware can be easily tailored to the developer, testing group, user community to extract performance of large scale applications and gives a strong foundation on programming models for Benchmarking computing systems in the range of *teraflop* to *petaflop* computing systems. The philosophy is to introduce new functionality and concepts to solve a design, implementation or analysis of problem in this courseware. The aim is to design Parallel Processing training courseware to build expertise for *Testing and Benchmark large Parallel Computing Platforms.* The complete training program gives a strong foundation for Testing, Benchmarking large message passing clusters in the range of *teraflop* to *petaflop* computing systems.

The courseware is useful for beginners, middle level and advanced level parallel computing users. The courseware is designed as a standard training courseware for classroom adoption in Computer Science or Computer Engineering or High Performance Application groups at Research & Development HPC organizations or M.Tech/M.S courses at graduate level programme.

The foundation is focused on quickly adapt to developments in High Performance Computing discipline which include Parallel Computers Architecture, Parallel Processing, Distributed Computing, Tuning and Performance of Parallel Programs using different programming paradigms.

The course has been classified into three-tier level, focusing on various aspects of HPC and each tier module gives an overview of topics, which benefit the on-going project activities. The duration of each tier may suffice depends upon on-going project activities. The course curriculum is designed for Beta-testing group for *Testing and Benchmark large Parallel Computing Systems project activities.*

Each module may focus on theory and laboratory session as per requirements of project activities. However, some groups in C-DAC may require detailed contents of modules, pertaining to the on-going research projects. Suggested below are three ***short-term*** courses or preliminary/exhaustive training programme of HPC module offerings. Various parts of the courseware and details of modules are summarized below.

For Parallel Programming, all the users should start with material covered in Part –I. Logically the courseware work should flow from top to bottom as per the numbers indicated in shown in figure 1 and figure 2. For specific project work, it is necessary for the course participant to refer advanced books on Parallel Processing or visit important web sites.

2. An Overview of β-test-ParProc-TC -*v*1.0 Courseware

The ***β*-test-ParProc-TC** *v*1.0 courseware can be grouped into *ten* parts and each part has sufficient number of modules. Each module contents can be covered in classroom lectures and the Hands-on Session can be done on PARAM series of clusters or your own Parallel Computing systems.

Figure 1 illustrates the various parts of the courseware and Figure 2 illustrates the various modules of each part and its relations with other modules. A brief summary of various parts of the courseware and flow of several modules in each part are described in Figure 1 and Figure 2.

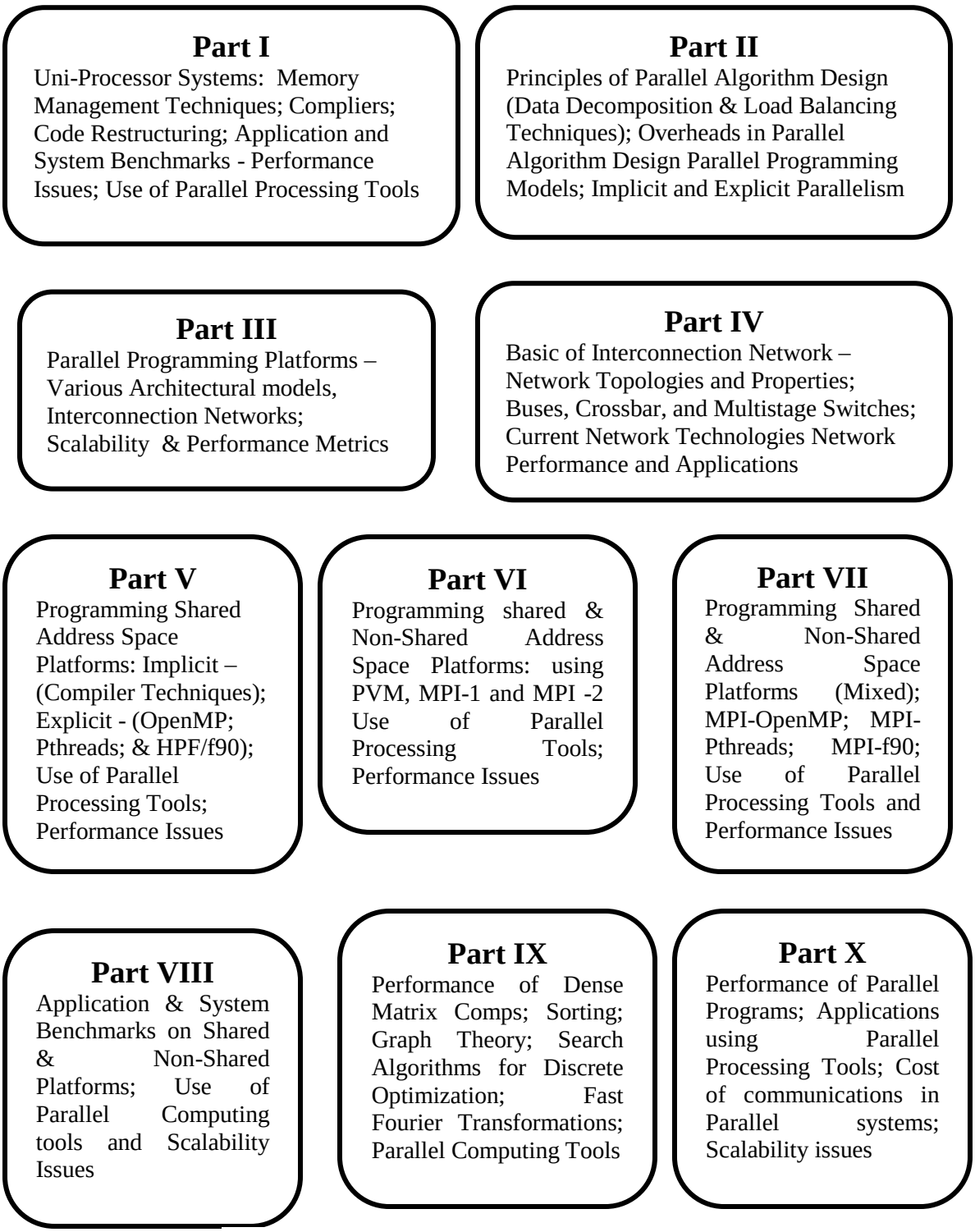

Figure 1. Representation of Courseware Contents

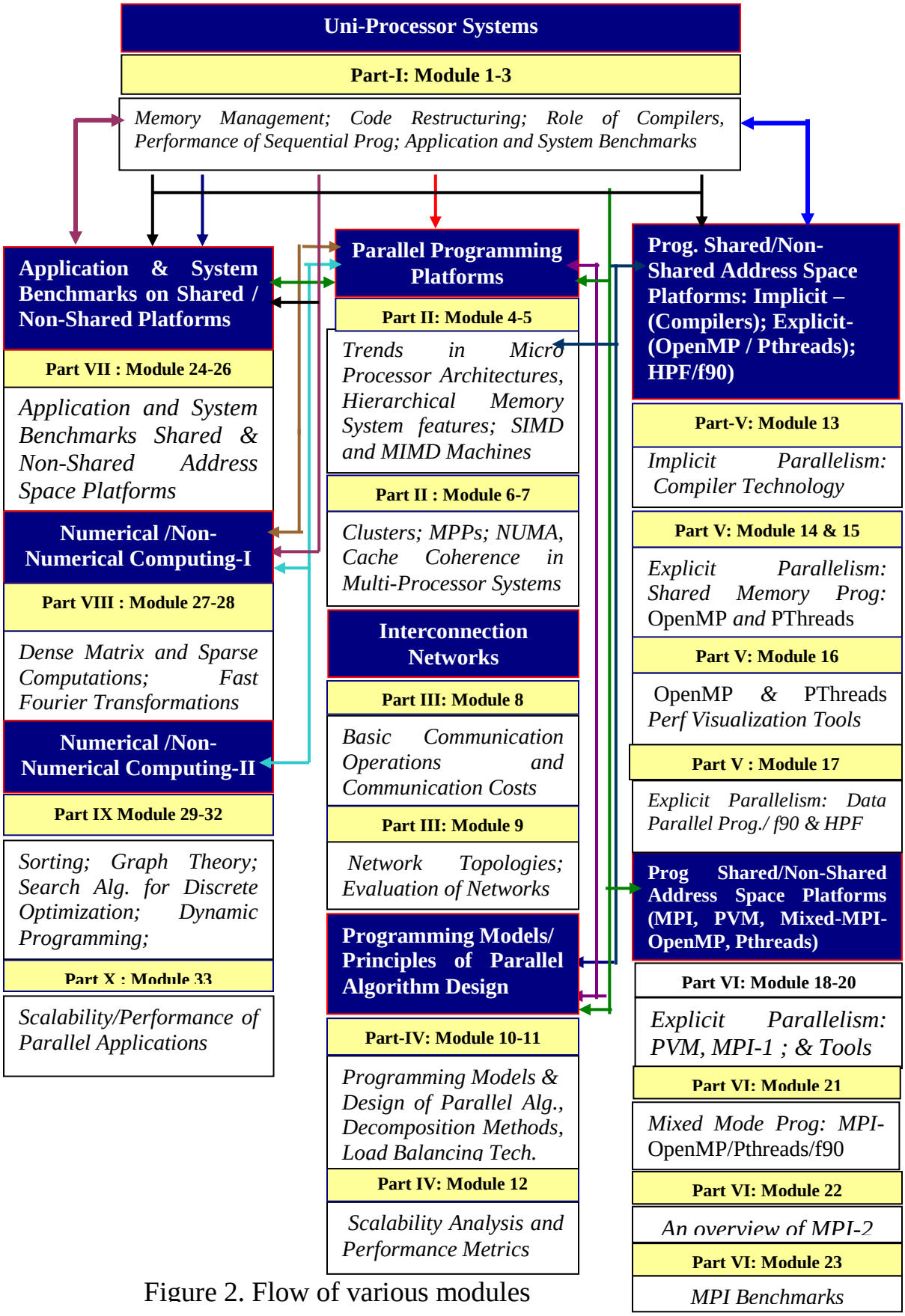

Figure 2. Flow of various modules

The courseware can be easily tailored to the developer, testing group, user community to extract performance of large scale applications and gives a strong foundation on programming models for Benchmarking computing systems in the range of *teraflop* to *petaflop* computing systems. The philosophy is to introduce new functionality and concepts to solve a design, implementation or analysis of problem in this courseware. To understand about the sorts of problems, beginners, developers encounter, when they begin thinking in parallel and writing parallel programs using different programming paradigms, the course contents may help.

To understand about the sorts of problems, beginners, developers encounter, when they begin thinking in parallel and writing parallel programs using different programming paradigms, the course contents may help and it is designed for short, mid and long-term duration of time schedule.

Most importantly, the courseware has sufficient number of programming assignments, which should play a central role to make strong foundations on Parallel Processing. Several modules defined in the courseware can be grouped together as per requirements of members who opt for short, mid, and long term time duration.

Figure 1 illustrates the various parts of the courseware and Figure 2 illustrates the various modules of each part and its relations with other modules. The solid arrow from Module A to Module B indicates Module B depends heavily upon material presented in Module A. For mid term and long-term courseware schedule, Module A and Module B have mutual relations, while performance of parallel programs is considered on target architecture.

All course contents given in each part is covered in numerical order, you will satisfy all requirements for Testing, Benchmarking, and Performance of large Parallel Computing Systems. However, you would like your members to start programming in C or Fortran with MPI or OpenMP as quickly as possible, you may wish to skip Part III or cover only two or three modules of the courseware. Definitely there is weak dependence across several modules of different parts and judicious choice should be taken up, which merely depends upon the duration of courseware i.e. Short, Mid or Long term.

If you wish to focus on performance of applications on *teraflop* to *petaflop* computing platforms, you may wish to skip Part III or cover only two or three modules, with more emphasis on Part IX Modules, focusing on parallel algorithms.

If you would like to start by having your members programming numerical computation algorithms with focus on performance, you can jump to Part X (Scalability and Performance) after covering important modules in remaining parts.

To get exposure to new functionality 'just in time' and performance of applications on *petaflop* computing systems, one can jump to various modules in Part VI & Part VII with strong foundation on Part III modules. A pre-requisite for this is to go through all modules as defined in short term course.

Most importantly, if you wish to get expertise on Benchmarks, programming various algorithms using MPI /OpenMP, you can work on Part VIII immediately after completion of necessary modules in other Parts.

3. Time Duration

The **short-term** course is focused on identifying suitable modules from Part-I to Part-X and quickly learns to write parallel programs with focus on algorithm design and performance. The emphasis is to design methodology to develop MPI/OpenMP programs that solve a series of progressively more difficult programming problems. Also, the focus for analyzing and predicting the performance of parallel systems is most important and it is addressed. The modules on architectural models, and important algorithms on numerical and non-numerical computations can be taken up as per requirements of user.

The **mid-term** course has more than enough material to write complex parallel programming, focusing on MPI, Pthreads, OpenMP and mixed MPI/OpenMP on different models of Parallel Computers. Special emphasis can be given to Application and System Benchmarks. Most importantly, use of Parallel Processing tools can be taken up which is required to know about the profiling a program, debugging the programs, visualization of programming behaviour.

The **long-term** course provides complete one semester course in parallel programming, focusing on turnkey projects. Even though the parallel programming is more demanding and it is hard to write programs, several algorithms for numerical and non-numerical computations can be taken up.

Suggested below is time duration for courseware program and equal weightage is given for the theory and hand-on on Parallel Computing systems.

- Tier - 1: Short term course and the time duration is 30 Calendar days
- Tier - 2: Mid term course and the time duration is 45 Calendar days.
- Tier - 3: Long term course and the time duration are 60 Calendar days.

Most importantly, the courseware has sufficient number of programming assignments, which should play a central role to make strong foundations on Parallel Processing.

.