EE5902 CA component topics

Sample/Suggested topics from which you may choose:

- Energy efficient scheduling for SoC / NoC based applications;
- Decentralized Load Balancing Algorithms - Implementation/Performance Evaluation Studies
- Load balancing algorithms on Cloud [no survey allowed]
- FPGA-based Simulations for large-scale computing problems - Resource Allocation strategies [Maximum 2 groups allowed]
- FPGA based computing – MP scheduling problems
- FPGA – Routing problems simulation [no survey allowed]
- NoC – Routing Strategies & Performance Evaluation [no survey allowed]
- Implementation of RoR model for Arbitrarily partitionable tasks [Maximum - 1 Group]
- Parallelization of certain applications - Examples - **FMM** parallelization; Linear Algebra Computations, Solving 3D-Laplace Equations, Parallel suffix array implementations; [Maximum **1 Group per application**];
- Large-Scale Parallelization for Data Mining Problems (applications - Astronomical, molecular biology, etc) on Multiprocessor Architectures [Maximum 3 groups allowed]
- Large-Scale Parallelization for Bioinformatics Problems (String Processing) on Multiprocessor – Longest subsequence matching and related algorithms;
- Architectures - Implications on data partitioning & Solution accuracy [Maximum 2 groups allowed]
- Handling Non-linear computations on multiprocessors - Use of divisible load analysis]
- Algorithms for Scheduling tasks under Resource Reclamation in Multiprocessor Systems
- Energy Aware Scheduling for Embedded CPUs - Specifically DAG scheduling on Embedded CPUs (Maximum 3 Groups allowed)
- Energy Aware Scheduling for Arbitrarily Partitionable Tasks (Maximum 1 Group allowed)
- Battery-Aware Scheduling for Embedded CPUs (Maximum 2 Group2 allowed)
- Large Scale Optimization - Multiprocessor Solution approach for iterative Computations – model in which several processors update the same variables [Maximum 2 groups]
- Multi-Core - Architecture & Performance Simulations; Running a large-scale problem on Multi-Cores (duo/quad cores)
- Moving Codes to Data!
- Tightly Coupled Multiprocessor Architectures – Multi-core architectures
- Scheduling Divisible tasks on FPGAs;
- Security-Aware Scheduling on Loosely Coupled Network Systems;
- Processors in IoT elements – Computing and security issues
- Edge/Fog Computing – Load distribution strategies involving compute elements
- FHE Computing on Multiprocessors

NOTE: Some topics to AVOID:

Certain topics may have overlaps with other modules. As an example, real-time systems, multimedia networking, computer networks courses do have some degree of overlap. Plus, this year, we will avoid certain other topics in the multiprocessor domain, as last year this has been taken up by more than 35% of the students. In view of all this, avoid the following topics in RED below:

Deadlock detection
Survey on Fault-Tolerant Consensus
Scheduling Imprecise Computations on Multiprocessor Systems - Design and Simulation for large-scale compute intensive workloads on SLTN and Bus
Mutual Exclusion problem;
Real-Time Scheduling algorithms (EE5903!!)
Deadlock Detection Mechanisms;
Network Routing/routing in loosely coupled systems related problems and issues;
Disk/Memory/Cache related problems and issues + Replacement Algorithms;
Survey on RAID architectures
Cloud, Grid & Cluster computing (out of scope of this course)
Evaluating Pipeline Architecture Performance – Branch and Data Dependencies
Optimal Pipelining – Discussion on different models and factors involved in determining the optimal performance [maximum 2 groups]
Parallelization - Image Processing applications;
IBM Cell Processor - Modeling and use for completely partitionable
IBM Cell Broadband Engine - Dynamic Programming Iterations - Efficiency/Response time/Speed-up quantification
Scheduling - DAGs, Indivisible tasks, Divisible tasks & real-time scheduling problems;
Memory Issues in Multi-Core and/or Multiprocessor Architectures
Performance issues – A comparative study between HiPC Platforms versus Loosely-Coupled Architectures(Cloud/Grid/Clusters)
Resource Optimization - A Static & Dynamic Resource-On-Reservation Model [Specific Algorithm will be given and this study involves Implementation/Performance Evaluation - Only 1 Person]