Version: 28-Mar-17

ASME Technical Committee on Multibody Systems and Nonlinear Dynamics

The 2017 Summer School on "Multibody Systems and Nonlinear Dynamics"

Overview: The *Summer School on "Multibody Systems and Nonlinear Dynamics"* is organized and run by the ASME Technical Committee on Multibody Systems and Nonlinear Dynamics (TC-MSND).

Location of 2017 Edition: Campus of the University of Wisconsin-Madison, USA.

Participants: Undergraduate, graduate, and post-doc students.

Rationale: (a) Expose talented students to MSND research topics; (b) get students involved in TC-MSND activities; and, (c) encourage participation in yearly ASME MSNDC conference.

Timeframe: July 23-August 5, 2017; i.e., the two weeks prior to the 2017 ASME MSNDC Conference.

Number of participants: 10.

Fees: None.

Expenses: ASME and TC-MSND will cover costs for room and meals. Some travel costs might also be covered.

2017 Summer School Structure:

- Mornings: lectures on advanced computing techniques relevant in the context of computational dynamics. Emphasis will be placed on GPU computing, parallel computing using multi-core architectures, and parallel computing on supercomputers. See curriculum at end of this document.
- Afternoons: research seminar. Each participating student to give a one hour seminar pertaining his/her research. Also, time set aside for completion of work assigned during morning lecture.

2017 Instructors: Dan Negrut and Radu Serban

Application Material: Applicants to submit (*i*) a school transcript; (*ii*) a one page essay providing the reasons why they should be selected; and, (iii) a one page support letter from a sponsor.

Application Deadline. May 1, 2017.

Application Process. Email your application material as one PDF doc to negrut@wisc.edu with a cc to your application sponsor. Applicants will be informed of selection process outcome by May 15, 2017.

Summer School Schedule

Morning sessions: 9:00 am – noon
Afternoon sessions: 1:30pm – 5:00pm

Day 1

Morning [Overview of Computational Dynamics]

- History of computational dynamics
- Motivation: Overview of problems and challenges in computational dynamics
    - Classes of problems (FEA, CFD, MBD, DEM)
    - Underlying mathematical models (PDE, ODE, DAE, DVI)
    - Numerical techniques (linear and nonlinear solvers, time integration, optimization)

Afternoon

- Hands-on
    - Building and using Chrono
    - Multibody dynamics demos
    - FEA demos
- Student research presentation

Day 2

Morning [Intro to Computing]

- Introduction: Example use of advanced computing in multibody dynamics
- Sequential computing, the hardware/software interface:
    - The processor: the fetch-decode-execute cycle, registers, CU, ALU
    - Memory issues: caches, virtual memory, cache coherence, memory models
- Quick overview of trends in parallel computing (multi-core and GPU computing); Top500 list

Afternoon

- Hands-on
    - Accessing Euler
    - Compiling and debugging on Euler
    - Timing an Application
    - Work on assignment (I/O operations)
- Student research presentation

Day 3

Morning [GPU computing with CUDA (1)]

- NVIDIA's CUDA intro: computation model and execution configuration
- CUDA memory allocation
- CUDA Memory model: registers and global, constant, texture, shared, local memories
- CUDA execution scheduling; thread divergence

Afternoon

- Hands-on
    - Timing a GPU application

  - Vector addition and Dot product
  - Large, tiled matrix-matrix multiplication with and without shared memory
  - Work on assignment: Parallel reduce operation
- Student research presentation

Day 4

Morning [GPU computing with CUDA (2)]]

- Advanced CUDA features
  - CUDA Streams
  - CUDA Optimization rules of thumb
  - CUDA profiling and debugging
- CUDA productivity tools
  - CUDA library landscape
  - `thrust` library

Afternoon

- Hands-on
  - Reduction and prefix scan with `thrust`
  - Profiling a dot product using `nvpp`
  - Work on assignment: Array processing using `thrust`
- Student research presentation

Day 5

Morning [Multicore computing with OpenMP (1)]

- Introduction to OpenMP
- Parallel regions and work-sharing (parallel for, sections, tasks)
- Data environment and variable sharing
- Synchronization

Afternoon

- Hands-on
  - Dot product
  - Integral evaluation using OpenMP
  - Work on assignment: Matrix Convolution, comparison w/ GPU implementation
- Student research presentation

Day 6

Morning [Multicore computing with OpenMP (2)]

- Advanced topics (performance)
- Profiling with Visual Studio and GCC
- Parallelization examples

Afternoon

- Hands-on
  - Improving performance of Fibonacci code

  - Monte-Carlo simulation
  - Work on assignment: parallel scan with OpenMP
- Student research presentation

Day 7

Morning [Distributed parallel computing with MPI (1)]

- Introduction to message passing and MPI
- Point-to-point communication
- Collective communication

Afternoon

- Hands-on
  - Ping-pong – estimate bandwidth
  - Quadrature – Simpson's rule
  - Work on assignment: Simple Jacobi iteration (Laplace equation)
- Student research presentation

Day 8

Morning [Distributed parallel computing with MPI (2)]

- MPI derived types
- Other programming paradigms for distributed parallel computing
  - Brief overview of Charm++

Afternoon

- Hands-on
  - Determine the eager/rendezvous threshold
  - Work on assignment: Scaling analysis for Jacobi iteration (Laplace equation)
- Student research presentation

Day 9

Morning [Parallel computational dynamics]

- Overview of Chrono::Parallel
  - Design, features, validation studies, sample simulations
- Case study: parallel collision detection
  - Binning approach for broad-phase collision detection
  - Implementation using thrust algorithms
- Hybrid MPI-OpenMP in co-simulation of vehicle-terrain interaction
  - Code design
  - Sample analyses

Afternoon

- Student research presentations (two of them)

Day 10 (morning only)

Program wrap up, round table discussion (1 hour).