Multibody Systems (MBS)

A set of bodies interconnected through mechanical joints which allow combined movements among them. Computational kinematics studies the movement of MBS from different approaches. Modular approach (Group Equations): divide a MBS into a set of modules whose kinematics can be solved in a hierarchical order facilitates parallelism.

Example: Stewart Platform

For each group one system of equations.
Six simultaneous systems.

Stewart Platform: Computation

Algorithm 1 Schematic of the Group Equations method for the Stewart Platform

For number of external iterations do
  Solve kinematics of terminal
  for all structural components do
    Solve kinematics of structural component
    for number of internal iterations do
      Solve kinematics of structural component
    end for
  end for
Several simultaneous equations.
Sparse matrices.

Parallel Simulator for Multi-Body Systems Based on Group Equations

Menu and Toolbar

Work Area

Graph View

Desktop application with three parts:

• Menu and Toolbar: access to the functionalities of the simulator for managing models. Models can be simulated (Training): generation of a database with information of execution time for the possible combinations of the nodes of the directed graph and their assignment to the computing units.
• Work Area: to manage a particular model. Groups of functions are created (nodes) and their connections are established. Each group contains a set of basic routines directly implemented or from basic libraries (MKL, M27, cuBLAS...). The scenarios represent the input for the simulation: matrix sizes and sparsity degree. A set of algorithmic parameters (OpenMP threads, number of GPUs...) whose values for low execution time are determined in the simulations.
• Graph View: shows the graph of the model (in the example, Stewart platform).

Adaptation for auto-tuning linear algebra routines

Linear algebra routines decompose in computations with smaller matrices.

Strassen multiplication

LU factorization

Application of the simulator to LAR auto-tuning

• Generate the model (or models) to experiment with (example: Strassen’s multiplication).
• Generation of the tree of possible groupings of nodes.
• Simulation and storage of the results for posterior use for other problem sizes. Branches of the tree can be selected, as also subsets of the computing units.

Further improvements

• Facilitate the generations of alternative models for a problem, and the selection of the best model.
• Reuse of the information generated for a routine or model in other routines which use them.
• Integration of the MBS in an auto-tuning system to generate auto-tuned libraries.