Numerical Methods for Direct and Large-Eddy Simulation

Johan Ohlsson *johan@mech.kth.se*

Qiang Li *qiang@mech.kth.se*

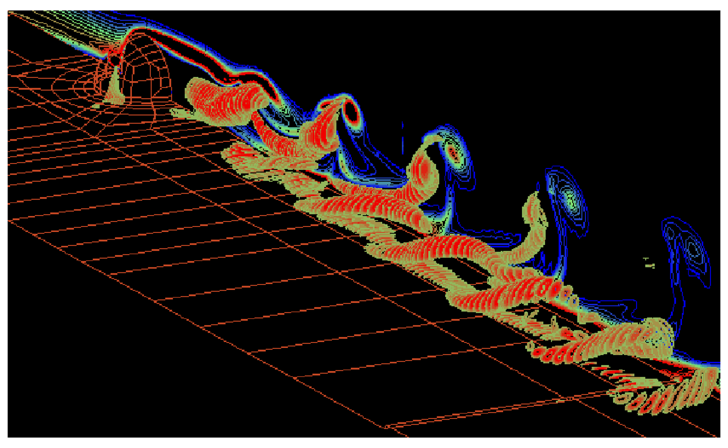
Murtazo Nazarov *murtazo@csc.kth.se*

Michael Stöckli *stockli@csc.kth.se*

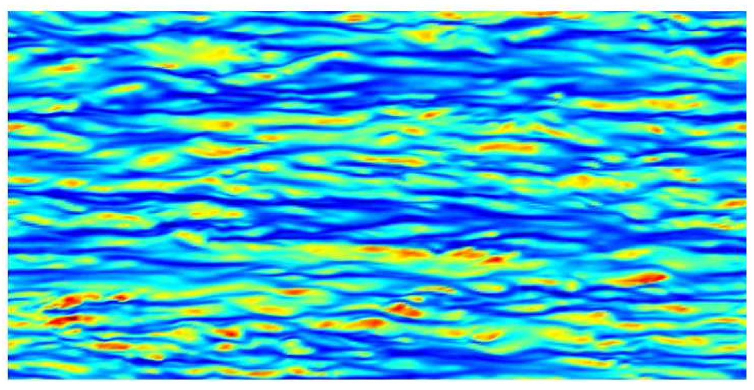
Linné Flow Centre

Spectral Element Method – Nek5000

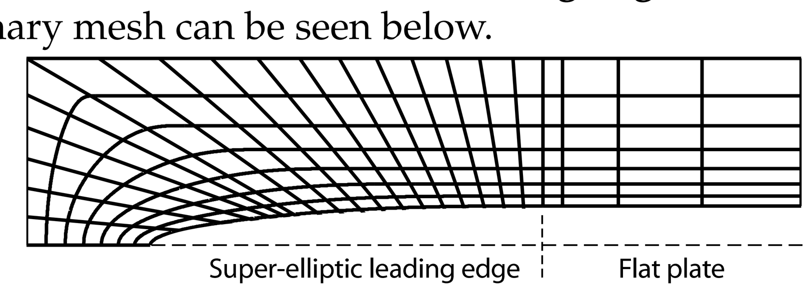
The Spectral Element Method (SEM) is a numerical method particularly well suited for transitional and turbulent flow simulations. The objective in fluid dynamic simulations is to solve the incompressible Navier-Stokes equations, which in the framework of SEM are formulated as a variational problem. SEM combines the accuracy of global methods (such as spectral methods) and the flexibility of methods based on a local approach (such as finite element (FEM), finite difference and finite volume methods). It is similar to a high-order FEM based on Lagrangian interpolation on the related Gauss-Lobatto quadrature grid. The solution to the PDE is expressed in a series of polynomial basis functions, which are orthogonal to each other. The method exhibits several favourable computational properties, such as the use of tensor products and naturally diagonal mass matrices, which makes it suitable for massively parallel implementations and large calculations.
Nek5000 by **Paul Fischer** at Argonne National Laboratory, USA, is the implementation of SEM that we use at Linné Flow Centre. It is specifically developed to run parallel simulations, where one example (simulation of transitional flow past a hemisphere) by Paul Fischer is shown below.



The motivation to start using this code is that it will open new possibilities in our flow research, since it enables the study of more complex geometries (compared to the flat-plate boundary layer simulations made so far with spectral codes). Still, the accuracy is essential in order to study the transition to turbulence, which excludes the use of low-order FEM. The code has recently been evaluated here at Mechanics by means of **turbulent** and **transitional** channel flow simulations. Streaks in a wall-parallel plane from one of the turbulent simulations at $Re_\tau = 590$ are shown below.

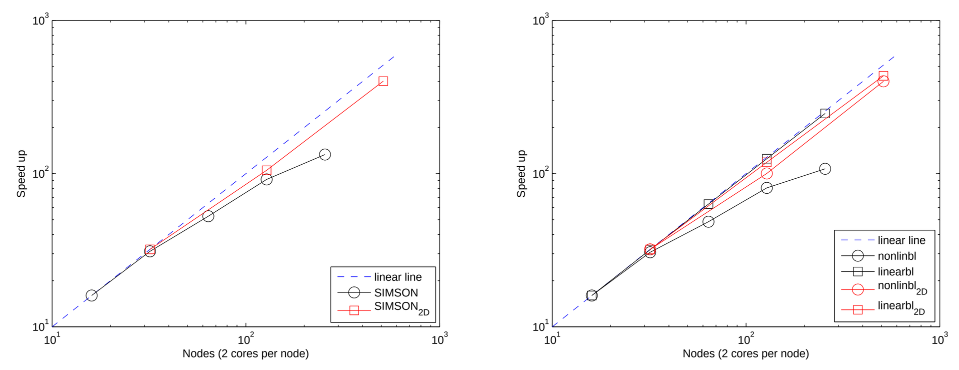


The performance of the code in terms of speed and scalability proved to be very good. The spectral code SIMSON (*see right*) performed about twenty times faster than Nek5000 in a turbulent channel flow simulation, which is considered to be very good for such a flexible code as Nek5000, where a wide range of geometries can be studied with very high accuracy. In addition, Nek5000 has shown an almost linear speed-up when runnning in parallel on thousands of cores.
Nek5000 is at the heart of many new projects within the Linné Flow Centre. After the successful validation of Nek5000 for the canonical turbulent channel flow (see above), one of the next steps is to run large-eddy simulations (LES) of a **turbulent diffuser**, aiming at increased accuracy which has not been possible with other methods in this geometry. This will include extensive implementations and validation work, since LES studies with SEM are rare. The transition to turbulence as well as the nature of the separation and Reynolds-number effects will be studied. Other projects with Nek5000 that have just started include external flows, *i.e.* the **leading edge**, where a preliminary mesh can be seen below.



A Pseudo-Spectral Code – SIMSON

Due to the rapid development of high performance computers during the last decades, highly resolved time-dependent numerical simulation (*e.g.* for direct numerical simulation, DNS) has become an important tool for turbulence research. An efficient numerical code to solve the Navier–Stokes equations for incompressible flows has been developed at KTH Mechanics for the last years, see the report by Chevalier *et al.*, 2007. The method is based on a standard Fourier/Chebyshev spectral discretization, leading to high accuracy and efficiency. The nonlinear terms are evaluated pseudo-spectrally in physical space to avoid the evaluation of convolution sums using FFTs. The code supports various wall-bounded shear flows, *i.e.* channel and boundary-layer geometries, in both temporal (periodic) and spatial (inflow-outflow) simulation setting. Perturbation mode plus linearized equations can be used in addition to solve for multiple passive scalars.
The code supports parallelisation with both shared memory (OpenMP) and distributed memory (MPI). With an older version of **SIMSON**, the domain is distributed in the spanwise direction only, leading to a restriction of the maximum possible cores to about 256. In order to fully exploit new massively parallel computers with many cores, *i.e.* more than 1000, a new parallelization in two dimensions has been implemented. The two parallelizations have been compared on the Blue Gene/L for a large case ($300 \cdot 10^6$ grid points). The left figure shows the total speed up; note that the global data transpose operation required every time step will inherently not scale linearly with the number of cores. Nevertheless, we obtain a speed up of about 130 (on 256 nodes) and 400 (on 512 nodes) for the old and new parallelization, respectively. The right figure compares the subroutines **nonlinbl** and **linearbl** representing the major computational effort of the code. One can observe that the subroutine **linearbl** scales almost linearly while **nonlinbl** clearly shows suboptimal performance for the old parallelization.



A General FEM Code – FEniCS

The FEniCS project (www.fenics.org) is a collection of software supporting general finite element (FEM) computation, of which the TACO research group (http://na37.nada.kth.se/tacowiki) at the Numerical analysis (NA) department is one of the main developers.
The process of assembling a discrete FEM system from a mathematical variational formulation is automated using FEniCS; with FIAT being a tabulator of general finite element basis functions, and FFC a compiler for variational forms generating efficient code for computation of finite element integrals. DOLFIN is the problem solving environment (PSE) of FEniCS using FFC/FIAT and state of the art packages for solution of discrete systems (such as PETSc and uBLAS) as computational back-ends. The FEniCS project also contain subprojects related to other aspects of FEM computation, such as pre- and postprocessing.
DOLFIN provides a PSE for simple construction of FEM solvers for general partial differential equations (PDE) on general geometric domains, using general FEM technology such as arbitrary high order Lagrange elements, mixed elements, discontinuous Galerkin methods, and adaptive mesh refinement. Today a prototype for parallel computation using DOLFIN runs on Blue Gene/L, and by summer 2008 large scale fully distributed parallel computation with MPI is expected to be in place. FEniCS is currently used in several courses given by the NA department.

Continuum Mechanics – Unicorn

Unicorn (www.fenics.org/wiki/Unicorn) is an implementation of a Unified Continuum Formulation (UCF) of continuum mechanics based on DOLFIN and a set of computational tools related to Arbitrary Lagrangian-Eulerian (ALE) moving mesh FEM, and duality based adaptive FEM computation. UCF refers to a formulation of the basic conservation laws for mass, momentum and energy for the whole computational domain seen as one single continuum, regardless of variations in phases over the domain. For example, the Unicorn solver of fluid-structure interaction (FSI) is based on solving one set of conservation laws for the combined fluid-structure continuum.
Unicorn is today a collection of FEM solvers for incompressible/compressible flow, and fluid-structure interaction, including contact/fracture, with the goal of being merged into one general FEM solver with minimal differences with respect to the different constitutive relations of different phases.
Unicorn is used within several applied projects of the TACO group related to aerodynamics of vehicles and aircrafts, aero-acoustics and computational medicine.