University of Tartu, Estonia
Institute of Computer Science
Distributed Systems Group

http://ds.cs.ut.ee/

Prof. Eero Vainikko, PhD (eero.vainikko@ut.ee)
Ulrich Norbisrath, PhD (ulno@ulno.net)
Satish Srirama, PhD (satish.srirama@ut.ee)
Oleg Batrashev, MSc
Georg Singer, MSc, MBA
Vladimir Šor, MSc
Artjom Lind, BSc
Toomas Laasik, BSc
Lauri Tulmin, BSc

1 DOUG

Very many problems and real applications in science and engineering need to solve large systems of linear equations: Ax=b.

Often the problem can be even as large that it does not fit into the memory of a single computer RAM, also the solution process is numerically very intensive. Furthermore, matrix A is sparse (contains mostly zeros), so only non-zero values should be stored.

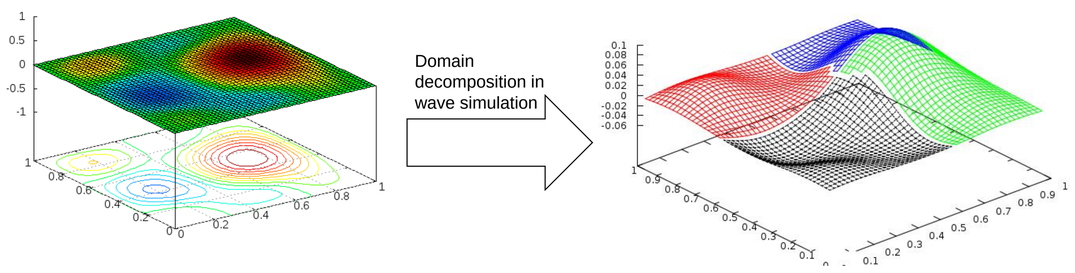
There are many ways to solve linear systems: direct solvers, like Gaussian elimination, or iterative solvers, like Preconditioned Conjugate Gradient (PCG) method or MultiGrid method. Direct solvers, unlike iterative solvers, usually eliminate matrix sparsity blowing up memory usage and moreover are difficult to parallelize.

In collaboration with the University of Bath (UK) we are developing a fast parallel black-box iterative solver - DOUG, which uses different techniques to shorten computation time, including:

• Domain decomposition methods, like graph partitioning through aggregation, to efficiently divide initial domain between processors, especially if matrix A has highly variable coefficients

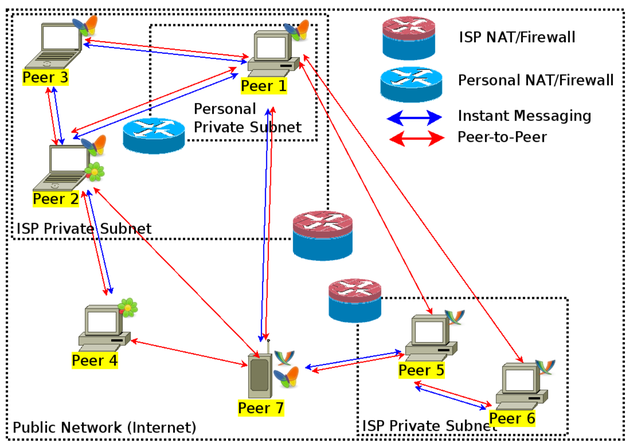
• Different 1-Level and 2-Level preconditioners for PCG

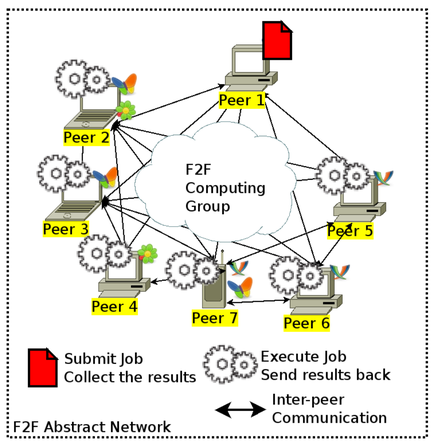
Currently we are in a process of re-engineering the whole code and also adding some new parallel preconditioners. The new implementation will be based on very fast Fortran95 components glued together with high level Python code and SciPy package.



2 Friend to Friend (F2F) Computing

Friend to Friend (F2F) Computing is a new paradigm for distributed computing, merging ideas from Peer-to-Peer, High Performance Computing, and social networks in instant messaging. The framework allows users to set up small computation Grids (spontaneous Grids) based on mutual trust in current instant messaging systems. Small research groups and companies will be able to combine their computational power with their friends or colleagues and be able to solve their computationally intensive problems much faster. The ease of administration and scalability is achieved due to Peer-to-Peer manner of forming Grids. Analogue to Peer-to-Peer the here applied paradigm is called Friend-to-Friend computing as a contraction of Grid, Peer-to-Peer, and friends relating to the spontaneously formed social networks.





3 PS3 Computing

Besides gaming Playstation 3 is a powerful computing platform. Its performance and flexibility lie between ordinary computer and massively parallel computation units like video cards or FPGA boards. Programming entry level is relatively easy and by adding explicit thread parallelization, you can utilize all the power.

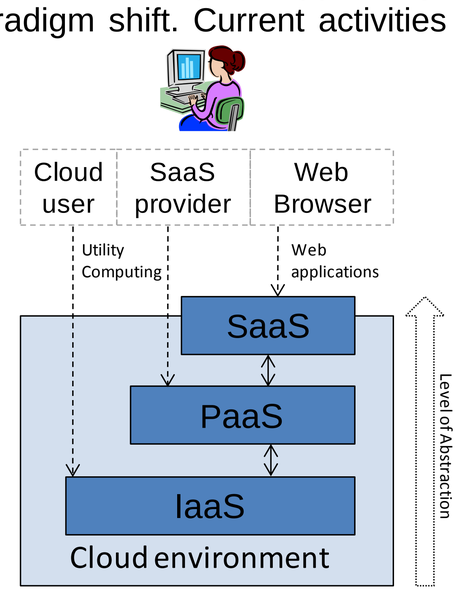
Main work on this platform includes topics like how to parallelize problems in given constraints, how to implement them and solving real problems from areas like physics, engineering and biology. Currently the most studied problem in our group is linear equation system solver for the Playstation 3 platform.

4 Scientific Computing on the Cloud

Cloud computing is a new style of computing in which, typically, real-time scalable resources are provided “as a service” over the Internet to the users. Servers within the cloud can be physical machines or virtual machines. The main goal of “SciCloud” project is to study the scope of establishing private clouds at universities. With these clouds, students and researchers can efficiently use the already existing resources of university computer networks, in solving computationally intensive scientific, mathematical, and academic problems.

The research topics of the project cover both technological and economical aspects. Technological research looks into architectural needs or shifts that occur when applications are moved to the cloud. Economic research of the domain tries to come out with a structured conceptual model of all the attributes that are required in order to assess computing tasks and selecting suitable IT package. We are also interested in the real world use cases, that can be tried out on existing cloud platforms like Amazon EC2/S3, Google App engine, Salesforce.com, etc, presenting the paradigm shift. Current activities of the project include:

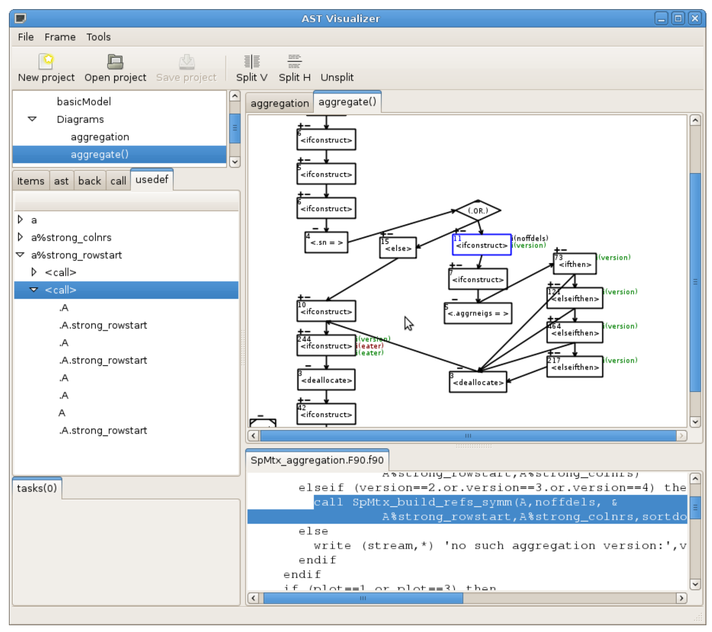
• Establishing private clouds with Hadoop and Eucalyptus
• Migrating scientific computing frameworks to cloud infrastructure
  - Reducing scientific computing problems to MapReduce algorithm
  - Adapting embarrassingly parallel computations to cloud platforms
• Real world use case analysis
  - Developing social networks and mashup applications with cloud application services
  - Finding parallelizable algorithms for batch processing



5 Fortran AST Visualizer (ASTVIS)

There are many legacy implementations of scientific algorithms which are written by individuals years ago without maintainability considerations. The goal of the project is to provide a visual tool that aids developers in understanding existing Fortran codes (program comprehension) using program analysis algorithms and diagram manipulation. The list of features currently include:

• build AST (abstract syntax tree) from a Fortran code
• navigate the AST and find function and variable references
• control-flow code analysis
• data-flow code analysis
  - reaching definitions and live variables analyses
  - definition-use chains
• several diagram types
  - function call graphs
  - control flow diagrams with dynamic (un)folding



The planned features are:

• interprocedural data-flow analysis diagrams
• data dependency analysis
• code slicing
• concepts and concept diagrams

6 Java UDP Hole Punching (JPunch)

This Java library is for establishing Peer-to-Peer connections between two random peers in the Internet. The library supports simple Network Address Translators traversal and reliable data transmission over UDP protocol. The goal of the project is to combine the most efficient Peer-to-Peer technologies into one library. Current work in the domain involves Improving design, implementing more traversal techniques and rewriting in C (CPunch).