Contribution ID: **73**   Type: **Poster**

High Performance Computing in Global Optimization

Monday, 11 April 2011 09:00 (8 hours)

URL

http://www.mii.lt/index.php?siteaction=personnel.view&id=442&lang=en

Conclusions

High performance computing is important for global optimization. It allows faster solution of practical problems. Even more important is that it allows solution of larger problems which solution would take non-reasonable amount of time using desktop computers.

Overview

Many problems in engineering, physics, economy and other fields are reduced to global minimization with many local minimizers. Global optimization problems are classified difficult in the sense of the algorithmic complexity theory. Therefore global optimization algorithms are computationally intensive, and solution time crucially depends on the dimensionality of a problem. When computing power of usual computers is not sufficient to solve a practical problem, high performance computing is helpful.

Impact

The developed global optimization algorithms have been applied to solve various problems: multidimensional scaling, optimal placement of piles in grillage-type foundations, estimation of model parameters and others. High performance computing enabled solution of larger problems.

Using parallel branch and bound algorithm for multidimensional scaling with city block distances global optimization problems with non-differentiable objective function and 27 variables have been solved exactly, what would take at least a month with a desktop computer.

Larger problems have been solved (although without guarantee of global optimality) with parallel genetic algorithm. Record-breaking solutions have been found for multidimensional scaling of Morse code confusion data with 72 variables using high performance computing and the developed parallel genetic algorithm.

Description of the work

For various problems we apply both deterministic global optimization algorithms (branch and bound algorithm with bounds estimated using interval arithmetic, Lipschitz condition, statistics, ad hoc techniques) and metaheuristic algorithms (genetic and memetic algorithms to mention a few).

Although covering, selection, branching and bounding rules differ in different branch and bound algorithms,

the structure of the algorithm remains the same. This allows implementation of generalized branch and bound templates. Standard parts of branch and bound algorithms are implemented in the template, only specific rules should be implemented by the user. Templates ease implementation of branch and bound algorithms for combinatorial optimization and for covering methods of continuous global
optimization. Parallel versions of algorithms can be obtained automatically using sequential program and parallelization techniques implemented in the template.
Genetic algorithms simulate evolution (selection, mutation, crossover) in which a population of solutions evolves improving function values. Genetic algorithms are suitable to parallelization, for example implementing multiple populations.

Primary author: Dr ŽILINSKAS, Julius (Vilnius University)

Presenter: Dr ŽILINSKAS, Julius (Vilnius University)

Session Classification: Posters

Track Classification: Poster