Parallel Evolutionary Optimization under MATLAB on standard computing networks

Hartmut Pohlheim
DaimlerChrysler AG
Research and Technology
Berlin, Germany
hartmut.pohlheim@daimlerchrysler.com

Sven Pawletta, Andreas Westphal
University of Rostock
Institut for Automatic Control
Rostock, Germany
sven.pawletta@etechnik.uni-rostock.de

Overview

• Motivation
• Tools
• Experiments
• Summary and Outlook
Motivation

- MATLAB: de-facto standard in research and development
- open and extensible architecture
- use of Evolutionary Algorithms (EA) for many difficult optimization tasks (GEATbx)

Problem:
- when optimizing large and complex systems
  - one run could take more than a day
  - but: evolutionary operations take only 2-10 minutes
  - all other time is spent evaluating the objective function

  parallel execution of objective function evaluation
  - utilize inherent parallelism of Evolutionary Algorithms
  - distribute objective function calls of single generation to multiple slaves (master slave system - global population model)
DP Toolbox

Structure

Distributed and Parallel Application Toolbox (DPTbx)

Tool for distributing computing tasks
- interface between MATLAB and PVM (multi level abstraction)
- runs on WinNT, Sun Solaris and LINUX (even mixed)
- application interface for straightforward application
- full integration into GEATbx

Evolutionary Optimization (GEA Toolbox)

Application Interface (Master and Slave Function)

DP Toolbox (high-level interface)

DP Toolbox (low-level interface)

Parallel Virtual Machine (PVM)
**Experiments**

**Calculation Time vs. Slaves**
- linear reduction of overall calculation time (if objective function call is expensive)
- inclusion of slow slaves only useful for many tasks

**Distribution of tasks**
- powerful machines calculate most tasks
- slow machines: no tasks at end of distribution (other slaves finish these jobs earlier)

---

**Parallel Evolutionary Optimization (MATLAB)**

H. Pohlheim, S. Pawletta, A. Westphal, GECCO’99
Summary

Advantages

• powerful tool for distribution of time intensive (evolutionary) experiments to multiple machines
  → use of standard tools: MATLAB and PVM
  → integration of DPTbx and GEATbx by application interface
  → efficient and robust distribution of tasks

Advantages:

→ straightforward application in everyday optimization tasks
→ reduces long computing times considerably
→ employs existent standard computer equipment

• application to real world problems proved successful
  → Genetic and Evolutionary Algorithm Toolbox for Matlab
    http://www.geatbx.com/index.html
  → Distributed and Parallel Application Toolbox for Matlab
    http://www-at.e-technik.uni-rostock.de/dp/
Application of parallel evolutionary optimization

- start of virtual machine
  
  ```matlab
  maschines = {'plota', 'galeili', 'plota', 'plunck'};
dpgeamain('start', maschines);
  ```

- start evolutionary optimization
  
  ```matlab
  OBJ_F = 'objchopper';
  [xnew, GeaOpt] = geamain2(objfun, GeaOpt, VLUB);
  ```

- distribution of tasks by master function inside EA
  
  ```matlab
  ObjVOff = dpgeamain('master', OBJ_F, Chrom,...
                        ObjFunAddPara);
  ```

- closing the slaves at end of optimization
  
  ```matlab
dpgeamain('kill');
  ```