SICSA MultiCore Challenge 2014

Hans-Wolfgang Loidl
<hwloidl@macs.hw.ac.uk>

**School of Mathematical and Computer Sciences,
Heriot-Watt University,
Edinburgh**

August 2014

Motivation

- **With multi-cores parallel computing has become mainstream**
- **Many models for parallel computation have been developed.**
- **But: most uses of these for high-performance computation have been on super-computers or compute clusters.**
- ***Which models are well-suited for (clusters of) multi-core machines?***

Methodology

- *Aim:* Test effectiveness of existing models of parallelism on state-of-the-art multi-cores
- *Method:* Choose a common, representative application, implement and measure it
- *Outcome:* Assessment of
  - effort in parallelisation,
  - performance,
  - scalability etc

Previous Phases of the Challenge

Two challenges so far:

- **Phase I: Concordance application (text processing; linear algorithm)**
- **Phase II: n-body problem (numerical computation; quadratic algorithm)**

Results have been published in a special issue of Concurrency: Practice and Experience (Vol 26, Iss 4). See `http://onlinelibrary.wiley.com/doi/10.1002/cpe.v26.4/issuetoc` **for full details.**

Phase III: Maximum Clique

Specification:

- ***Def:*** **A clique of a graph (V, E) is a subset of vertices V, where every vertex in this subset is adjacent to every other in the subset.**
- ***Given:*** **A graph with vertices V and edges E.**
- ***Find:*** **A largest clique in the given graph (V, E).**

NB: **This problem is NP-hard. The algorithm is symbolic rather than numerical.**

Details are on the Wiki:

`http://www.macs.hw.ac.uk/sicsawiki/index.php/Challenge_PhaseIII#Maximum_Clique`

Motivations for choosing Maximum Clique

- The sequential algorithms have been widely studied, and have been implemented independently multiple times.
- Bit-parallel versions of these algorithms are well understood.
- It is exponential and irregular, and contains enough complications to make it a realisic problem, but it is not too complicated.
- These algorithms have been thread-parallelised independently with considerable success at least twice.
- We can describe the simple algorithm in one page.

Motivations for choosing Maximum Clique

- There is a good set of standard benchmark instances to work with. This set includes different families of graphs with different properties.
- We know a good way of checking that sequential algorithm has probably been implemented correctly (counting the number of colourings performed).
- Ties in with the Graph500 benchmark suite (maximum independent set is just maximum clique on the complement graph).

Activities during the Hackathon

- Short introductory talks on problem, algorithm, and languages.
- Identifying interest areas and hacking tasks.
- Define groups to work on concrete tasks.
- Summarise outcomes
- Further activities (hacking, collaboration, publication)