Programming Languages

Discussion lead by Charles Leiserson and Umut A. Acar.
Notes taken by Umut A. Acar.

July 12, 2012

Participants

- Umut Acar
- Bob Harper
- Amir Kamil
- Charles Leiserson
- Cyrus Omar
- John Reppy
- Vijay Saraswat
- Danny Sleator
- Fujao Zhao
- Zhunping Zhang

General Remarks

- Difference between models and languages/semantics: it is hard to separate these two. The community would likely benefit from discussing them together.
  (For example: models discussion talked a lot about costs, machines.)

Research Directions

- (Weak) Memory models
- Resilience (fault tolerance)
- Immutable, Purely Functional Parallel Languages
- Domain Specific Languages
- Shared memory (is Evil)
- Memory abstractions
- Encapsulating non-determinism, concurrency
- Cost Semantics
- Heterogeneity
- Synergy of linguistics mechanism
- Abstractions for parallelism
- Verifications
- Semantically tractable abstractions
- Program synthesis
- Parallelism for non-experts
- Parallelism without concurrency

Theme: Parallelism for Non-Experts

Non-experts should be able to

- Performance
- Productivity
- Debugging/release engineering.
- Performance modeling, cost semantics, reasoning about performance.

Question: *How to achieve this theme and the subgoals.*

Semantics and Programming Languages for Parallelism

Provide semantics for parallelism and more generally performance that isolates concerns of scheduling, concurrency, memory model, heterogeneity, data distribution, resilience from the programmer

while still providing

performance, productivity, debugging, verification, predictions about performance, correctness.

How can we achieve these goals?

We need research on the following topics:

- Declarative and purely functional programming.
- Purely functional algorithms.
- Efficient, effective resilience.
- Memory abstractions, including shared memory.
- ...

Research Directions

- (Weak) Memory models
- Resilience (fault tolerance)
- Immutable, Purely Functional Parallel Languages
- Domain Specific Languages
- Shared memory (is Evil)
- Memory abstractions
- Encapsulating non-determinism, concurrency
- Cost Semantics
- Heterogeneity
- Synergy of linguistics mechanism
- Abstractions for parallelism
- Verifications
- Semantically tractable abstractions
- Program synthesis
- Parallelism for non-experts
- Parallelism without concurrency