Lecture 6
Parallel Programming Models

Daniel Sanchez and Joel Emer

6.888 Parallel and Heterogeneous Computer Architecture
Spring 2013
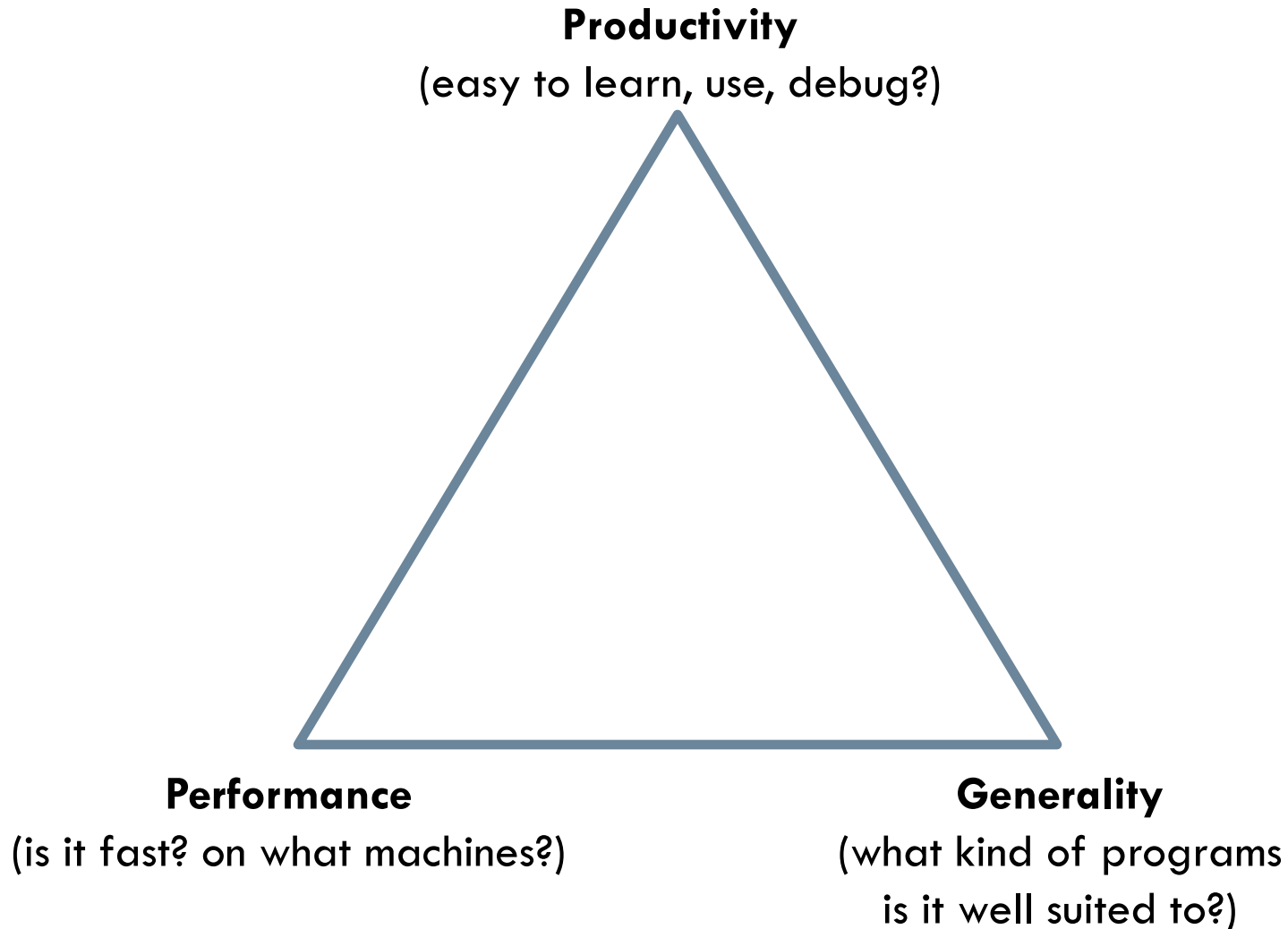
Massachusetts Institute of Technology

CSAIL

Recap: Message-passing Discussion

- Network speed/latency

  - Memory bus vs I/O bus

- Messaging overheads: Buffering, copying, protection

  - OS-level vs user-level messaging

  - Protocol overheads vs network complexity

- Synchronization overheads: Synchronous vs asynchronous

  - Polling vs interrupts?

Recap: Shared Memory Discussion

- UMA Scalability?

- NUMA Scalability?

- Cache coherence, consistency, atomic operations
  - Complexity?
  - Alternatives?

- Cost?

- What kind of applications need shared memory?

Programming Model Tradeoffs

Productivity
(easy to learn, use, debug?)

Performance
(is it fast? on what machines?)

Generality
(what kind of programs
is it well suited to?)

Programming Model Discussion

- Many dimensions to compare programming models
  - Syntax, type systems, imperative/functional/declarative…

- This discussion:
  - Interaction between architecture and programming model
  - Types of parallelism exploited
    - Data, task, pipeline
  - Scheduling features
    - Adaptive, locality-aware, focus on communication
  - Productivity and generality
    - New or extension, ease of use & debug, elegance, safe parallelism
    - Does it restrict algorithms? data structures? type of parallelism?
    - At a high level: Is it worth it? Successful?