Outline

- Performance Evaluation
  - Introduction
  - Performance Debugging for Distributed Systems of Black Boxes
- Q&A
Introduction

- Exploiting Parallelism with the Distributed System (Compiler or Library)
  - Autoparallelization
    - Heterogeneity
    - Variable latencies
  - Manual Computation Decomposition and Load Balancing (Distributed Memory)
    - Architecture independence
  - Data Allocation (Distributed Memory)
    - Maximizing locality
    - Minimizing communication
Introduction (Cont’d)

- Parallel Execution of Components
  - Load matching
  - Communication optimization
- Overlap of Communication with Computation (Distributed Memory)
  - Large and variable latencies
- Reuse of More Data in Local Memories (Distributed Memory)
- Spreading Computation Evenly across Processors (Distributed Memory)
  - Minimizing communication

- Task Parallelism
- Latency Hiding
- Latency Reduction
- Load balancing
Introduction (Cont’d)

- Performance Tools
  - Goal
    - User’s identifying and overcoming performance bottlenecks
  - Functionalities
    - Measurement
    - Analysis
    - Visualization
    - Engineering/Tuning
    - Estimation/Prediction

Via Instrumentation
To Identify Bottlenecks
Introduction

- **Critical Path**
  - **Longest Path** through the DAG
    - Corresponding to the longest path

- **Critical Path of a Program**
  - **Longest CPU or Communication**
  - **Weighted Path** of the PAG

---

A Graph Consisting of Nodes Representing Significant Events, and Arcs Indicating the Ordering of Events within a Process or Synch Dependencies between Events.