

Dynamic Analysis

17-654/17-754:
Analysis of Software Artifacts

Jonathan Aldrich



Part 1: Performance Analysis



- Your boss tells you, “make it run faster.”
What do you do?

Speedup Analysis Scenario



- Assumptions
 - Procedure x takes 75% of execution time
 - We can speed up x by a factor of 3
- How much faster will the program run?

22 March 2007

15-313: Foundations of Software Engineering
Static Analysis

3

Amdahl's Law



- P = % of program you can speed up
- S = speedup of that part of the program
- Maximum overall speedup:

$$\frac{1}{(1-P) + P/S}$$

↑ ↑
time to run time to run
rest of sped-up
program program part

22 March 2007

15-313: Foundations of Software Engineering
Static Analysis

4

80/20 rule (Pareto principle)



- 20% of the program takes 80% of the time
- No point in optimizing anything until you know what 20% is causing the problem!
 - Another way of stating Amdahl's law

22 March 2007

15-313: Foundations of Software Engineering
Static Analysis

5

Getting Good Performance



- What matters?
 - Algorithms
 - Parallelism
 - Cache performance
 - Caching
 - trade space for time
 - Doing less work
 - trade precision for time
- What doesn't?
 - Inlining vs. calls
 - field reads vs. accessors
 - Recursion vs. loops
 - Low-level arithmetic

What's the cost of optimization?

22 March 2007

15-313: Foundations of Software Engineering
Static Analysis

7