Computer Architecture

Performance

Debadatta Mishra, CSE, IITK

Recap (Performance Analysis)

- Performance analysis is crucial
  - To make observations
  - To perform root-cause analysis
  - To show the effectiveness of improved design
- Aspects of performance analysis
  - Correctness
  - Generality
  - Repeatability
  - Metric of comparison

Agenda: Analysis and reporting of performance

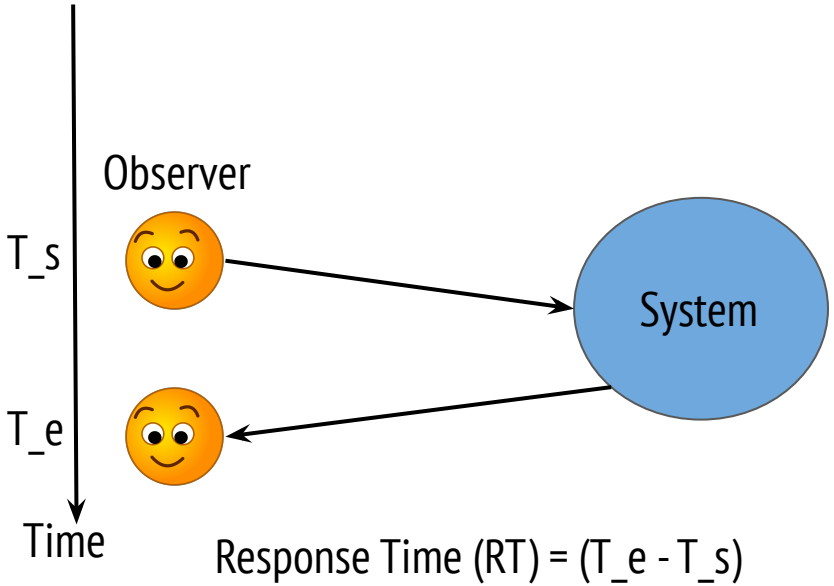
Performance Evaluation

- Metric: A measurement to compare two or more entities
- Examples:
  - Memory usage of a program
  - Time to load a web page
  - # of DB queries executed per second
  - # of cycles to add two numbers

- How to compare performance of two computer systems?

Response time

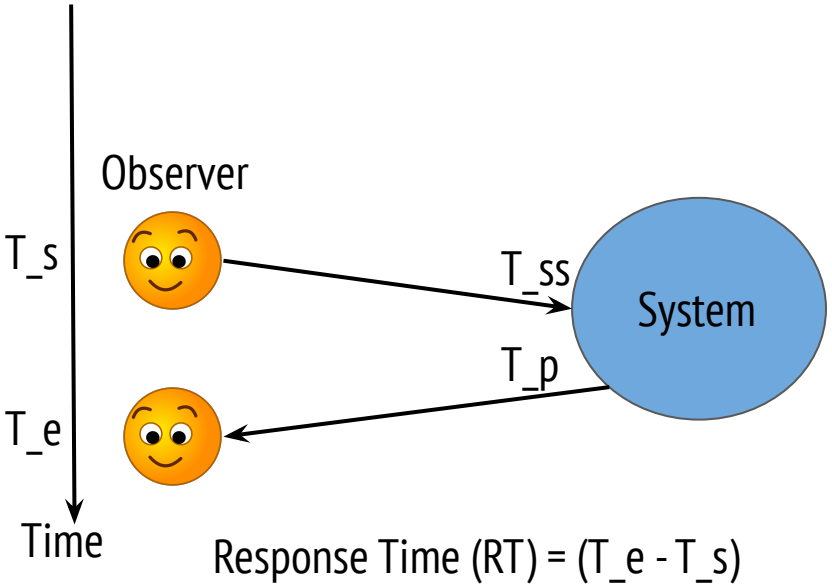
- Response time: Time taken to complete one task



- Consider two implementations of the system S_A and S_B. If $S_A < S_B$, S_A is better than S_B (True/False) ?

Response time

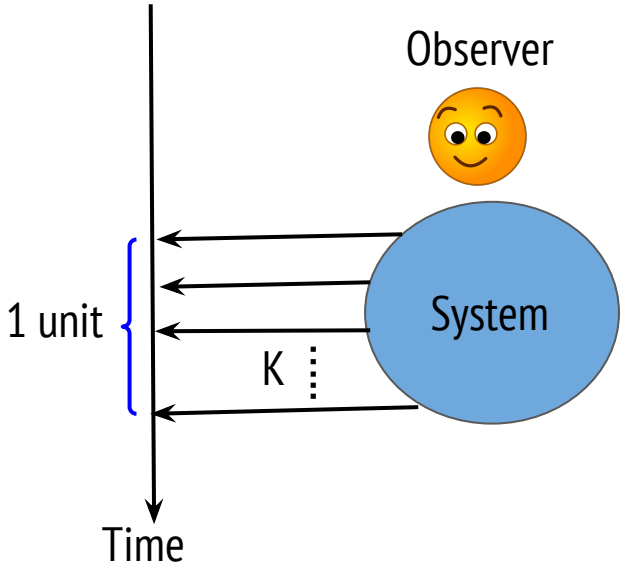
- Response time: Time taken to complete one task



- Consider two implementations of the system S_A and S_B. If $S_A < S_B$, S_A is better than S_B (True/False) ?
- True, assuming (T_ss -T_s) and (T_e-T_p) are same during the observations
- Another way to think: move the observer near the system boundary

Throughput

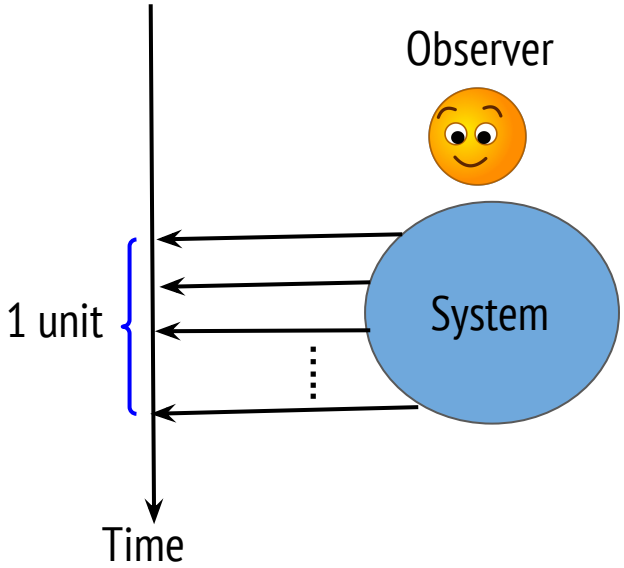
- Throughput: # of tasks completed per unit time



- How good a performance measure is throughput?
- Decreasing response time increases the throughput (True/False)

Throughput

- Throughput: # of tasks completed per unit time



- How good a performance measure is throughput?
- Good to measure system limits
- Decreasing response time increases the throughput (True/False)
- True, assuming enough load is generated

Response time vs. Throughput

Example: Computer-**A** is a 32-cores machine, takes 16 seconds to execute a give task **T**. Computer-**B** is a single core machine, 8 seconds to execute **T**.

- Which machine is better?
- Which processor is better?

Response time vs. Throughput

Example: Computer-**A** is a 32-cores machine, takes 16 seconds to execute a given task **T**. Computer-**B** is a single core machine, 8 seconds to execute **T**.

- Which machine is better?
- Which processor is better?

Response time: A = 16s B = 8s

Throughput: A = 2 tasks / sec B = 1/8 tasks/sec

Response time vs. Throughput

Example: Computer-**A** is a 32-cores machine, takes 16 seconds to execute a given task **T**. Computer-**B** is a single core machine, 8 seconds to execute **T**.

- Which computer is better?
- Which processor is better?

Response time: A = 16s B = 8s

Throughput: A = 2 tasks / sec B = 1/8 tasks/sec

- There is some unease concluding processor used in **B** is better
- Even more unease concluding computer **A** is better

Response time vs. Throughput

Example: Computer-**A** is a 32-cores machine, takes 16 seconds to execute a given task **T**. Computer-**B** is a single core machine, 8 seconds to execute **T**.

- Which computer is better?
- Which processor is better?

Response time: A = 16s B = 8s

Throughput: A = 2 tasks / sec B = 1/8 tasks/sec

- There is some unease concluding processor used in **B** is better (Does **T** represent all tasks? Is it CPU intensive?)
- Even more unease concluding computer **A** is better (+ Tasks independent?)

Response time vs. Throughput (Summary)

- For CS422, focus is more on response time (round trip latency)
  - Ensure all other aspects of the system are identical
  - Choice of workloads is crucial
  - Benchmarks are standard workloads

- Does not mean we are not worried about throughput!
  - Throughput can be improved introducing more parallelism (more cost!)
  - Improvement of throughput of a bottlenecked sub-system, can improve (amortize) the round trip latency of an encompassing system

Response time vs. Throughput (HW)

- With increased response time, throughput decreases (True/False)

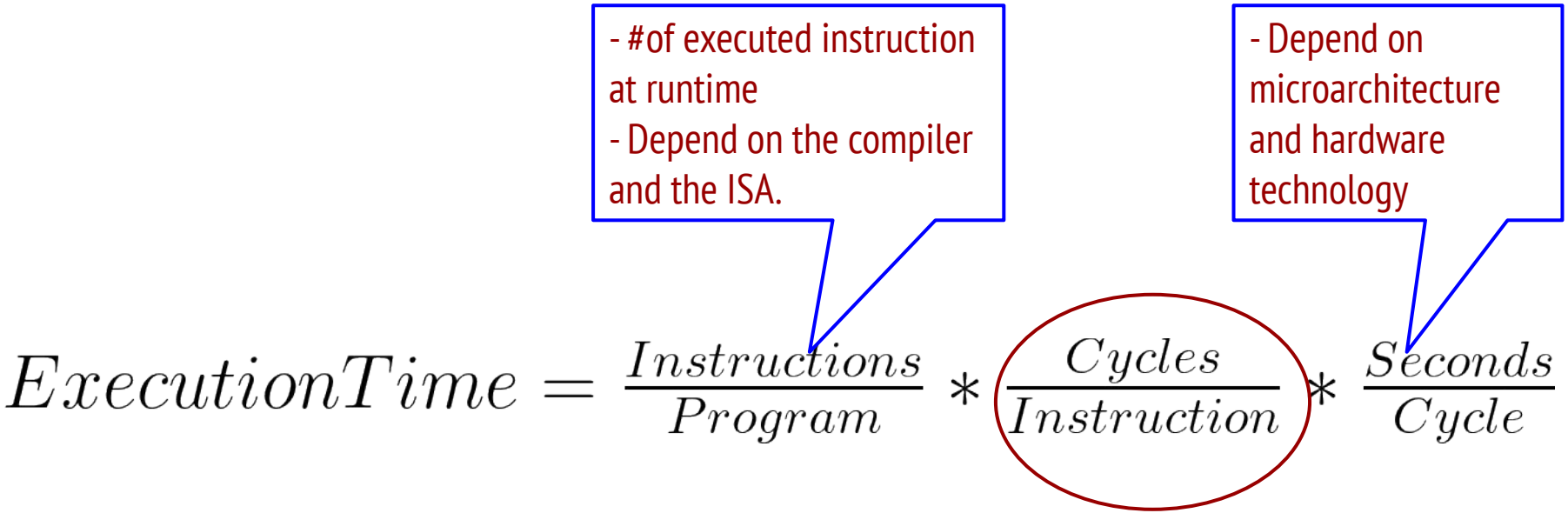
- If response time of a system increases, throughput of the system is bound to decrease (True/False)

- Increasing throughput of a system does not have any impact on response time (True/False)?

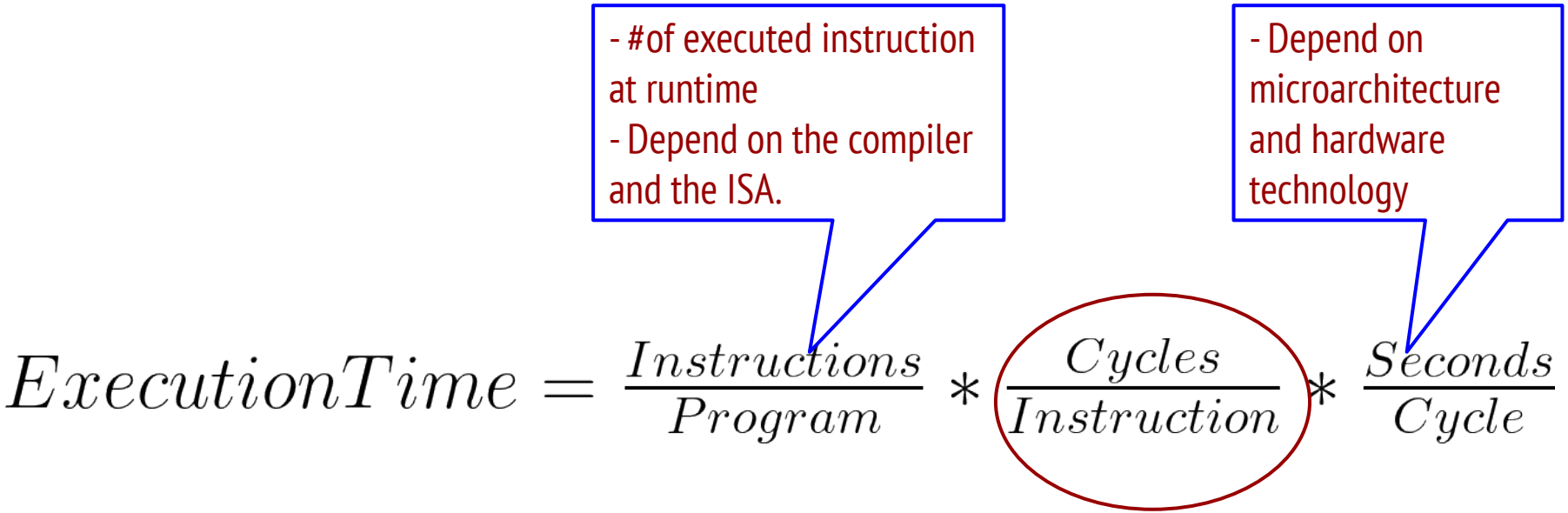
CPU performance equation

$$ExecutionTime = \frac{Instructions}{Program} * \frac{Cycles}{Instruction} * \frac{Seconds}{Cycle}$$

CPU performance equation

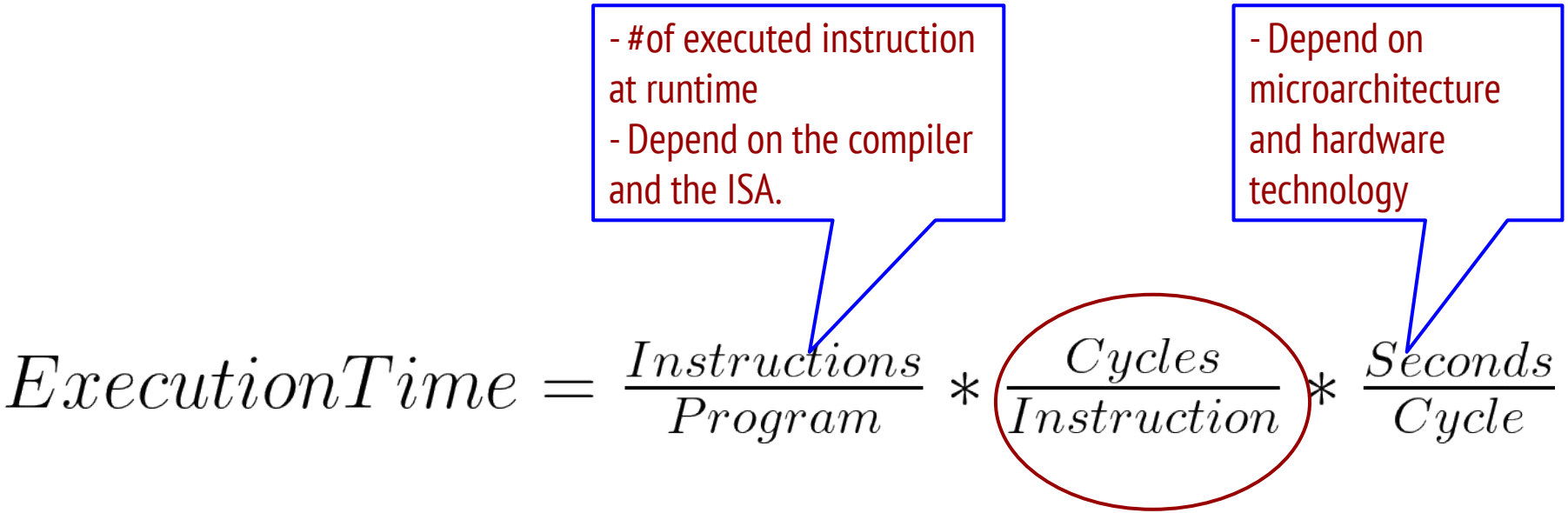


CPU performance equation



- Cycles per instruction (CPI) is a commonly used measure of performance
- What aspects determine CPI?

CPU performance equation



- Cycles per instruction (CPI) is a commonly used measure of performance
- What aspects determine CPI? (ISA, Micro-architecture, Organization)

Comparing performance

For a given task, computer **A** is **N** times faster than computer **B**

$$\frac{ExecutionTime_B}{ExecutionTime_A} = N$$

- Consider a special case where **A** and **B** only differ in terms of their implementations of the same ISA
- What will be the ratio of their CPIs?

Comparing performance

For a given task, computer **A** is **N** times faster than computer **B**

$$\frac{ExecutionTime_B}{ExecutionTime_A} = N$$

$$\frac{CPI_B}{CPI_A} = N = \frac{IPC_A}{IPC_B}$$

- Consider a special case where **A** and **B** only differ in terms of their implementations of the same ISA
- What will be the ratio of their CPIs? (N) IPC: Instructions per cycle

Summarizing Performance (HW/Piazza discussion)

- Representation of performance using a single point estimate (e.g., mean)
  - Arithmetic mean
  - Harmonic mean
  - Geometric mean
- Which one to use and when?

	Computer A	Computer B	Computer C
Program X	1	10	20
Program Y	100	50	20