ICW – CDA3101 – Performance Computation – Right-click Information	MSS	25 Feb 2013

Conceptual Background

Computer Performance Simulation allows you to estimate the time it will take for a computer program to run on a processor (e.g., a CPU), given the following parameters:

(a) the number of instructions in the program [called *Instruction Count* or *IC*],
(b) the number of CPU clock cycles per instruction [called *Cycles Per Instruction* or *CPI*], and
(c) the number of seconds per clock cycle (the reciprocal of the *CPU Clock Frequency*).

Historical Background

Alan Turing possibly coined the term "simulation" to describe a universal machine executing a state transition table that itself described the state transitions, inputs and outputs of a (simulated) discrete-state machine. A Turing machine that can simulate any other Turing machine is called a universal Turing machine [1]. The Church–Turing thesis states that Turing machines indeed capture the informal notion of effective method in logic and mathematics, and provide a precise definition of an algorithm or "mechanical procedure" [2].

The history of practical computer simulation began during World War II while John von Neumann and Stanislaw Ulam investigated neutron behavior [3]. Probably the first end-to-end performance simulation of a computer central processing unit (CPU) was constructed by Harwood Kolsky and John Cockeon the IBM Stretch project beginning in August 1957 [4]. By the early 1960s, there were numerous computer simulation efforts under way at IBM and many other companies.

[1] F. C. Hennie and R. E. Stearns. *Two-tape simulation of multitape Turing machines*. JACM, 13(4):533–546, 1966.

[2] http://en.wikipedia.org/wiki/Turing_machine (accessed 25 Feb 2013).

[3] http://www.uh.edu/~lcr3600/simulation/historical.html (accessed 25 Feb 2013).

[4] http://archive.computerhistory.org/resources/text/IBM/Stretch/102636400.txt (accessed 25 Feb 2013).

Economic Background

The history of computer simulation began during World War II while John von Neumann and Stanislaw Ulam investigated neutron behavior [1]. Random trial experimentation was too expensive, and theory was too complex to produce a closed-form analysis. As the basic equations and data about the occurrence of events in a sequence were known, the probabilities of separate events could be merged stepwise to predict the outcome of a sequence of events. Following von Neumann's and Ulam's remarkable success in simulation of neutron behavior, the technique of computer simulation soon became popular and found many applications in the business and industry [1].

Therefore, computer simulation of a physical process is useful when (a) it is too expensive to run the physical process repeatedly varying input parameters, and (b) a closed-form analytical result is not achievable due to theoretical complexity or intractability.

[1] http://www.uh.edu/~lcr3600/simulation/historical.html (accessed 25 Feb 2013).

Math Example

To understand the CPU performance equation, we first consider its variables and units:

IC = instruction count, in units of *instructions*

CPI = cycles required to execute an instruction, in units of *cycles / instruction*

ClockPeriod = seconds per CPU clock tick = 1/clock_frequency, in units of *seconds* / *cycle*

So, IC · CPI · ClockPeriod = instructions · cycles/instruction · seconds/cycle = units of seconds.
