Yahya Alhinai

EE5371

December 4, 2019

Problem 1:

1. **Full system simulation:**

   - We use full system simulation if we want an exact report of everything in a certain system that is being conducted on. Following table explain the features and the problem of this simulation technique:

Advantage	Disadvantage
Full level of details	Time consumption
	Unnecessary log of details
	Might take time to sort all desired information

2. **Simulation using benchmark programs with reduced input sets:**

   - If we know exactly which parameter are predominantly affecting a system or we are interested in known the details of certain parameters in such a system, then the most suitable simulation technique would be a benchmark with reduced input sets. The table below explain its pros and cons:

Advantage	Disadvantage
Only log in desirable details	Time consumption
Faster than full system simulation	Modifying the original system to monitor certain parameters
Easy to sore the obtain results	

3. **One of the sampling-based simulations:**

- In general, we use sampling-based simulations if we are interested in a statistical representation that describe any system with a certain desire accuracy. The following table shows the advantages and disadvantages of all sampling-based simulations:

Advantage	Disadvantage
Faster than the previous two simulation technique	Statistical representation not an exact representation
The accuracy of sampling based is very high given the time invested	
Can be reformed to new methods in order to suit different systems	

- Because sampling-based simulations have very versatile nature, several new methods under sampling-based simulations has been invented to suit different systems. In the paper "ESESC: A Fast Multicore Simulator Using Time-Based Sampling" the authors have presented two sampling-based simulations:

1. **Time-Based Sampling (TBS):**

Advantage	Disadvantage
maintain the progressed time by avoid divergence of progress among threads	Higher change of collecting error due to cycle dependency sampling
sampling within fixed length in number of cycles	
Woks in single-core as well as multi core system	

2. **Instruction-Based Sampling (IBS):**

Advantage	Disadvantage
Only log in desirable details	Cannot avoid divergence of progress among threads
sampling within fixed number of instructions	Only work in a single-core system
Lower change collecting error in comparison to TBS	

Problem 2:

The designers of the Itanium made a mistake in including If-conversion in the processor. The implementation of If-conversion would have given the promised results of up-to 35% speeding up have it been implemented on processors of early 90s. The problem was that the processors' architecture, at the time If-conversion was researched, were observing a grate alteration with the merge of multi-core processors and corresponding in a change in the compiler's interpretation as well. The designers of the Itanium made a mistake in including If-conversion in the processor because it was designed to work on older processors' architecture, and it was tested by testbenches that were not prepared to accommodate the new changes in the processors' functionally. This is described in the paper "*Statistical Performance Comparisons of Computers*" as incorrect comparisons that can affect research or acquisition decisions; therefore, the consequences are significant.

One of the main problems the author of "*Thinking Methodically about Performance*" discussed in his paper is the inability to reproducible in a lab environment with the components in isolation. This might have been a great contributor to the overly optimistic produced results. That been said, researchers in this area should have predicted the change in the industry and shift their time and effort to deal with the bottleneck issues that are not affected by the changes in processors' architecture like speeding up disk access or improving miss/hit rates of caches. This is saying that has If-conversion been carried on being implemented on processors to come, it would have a bad effect on the performance.