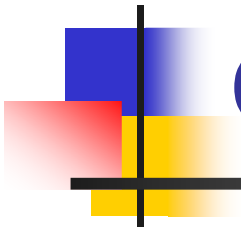


Simulation of Memory Management Using Paging Mechanism in Operating Systems

Tarek M. Sobh and Yanchun Liu

Presented by: Bei Wang

University of Bridgeport

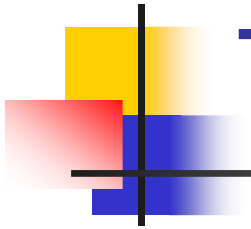
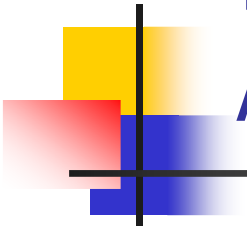


Table of Content

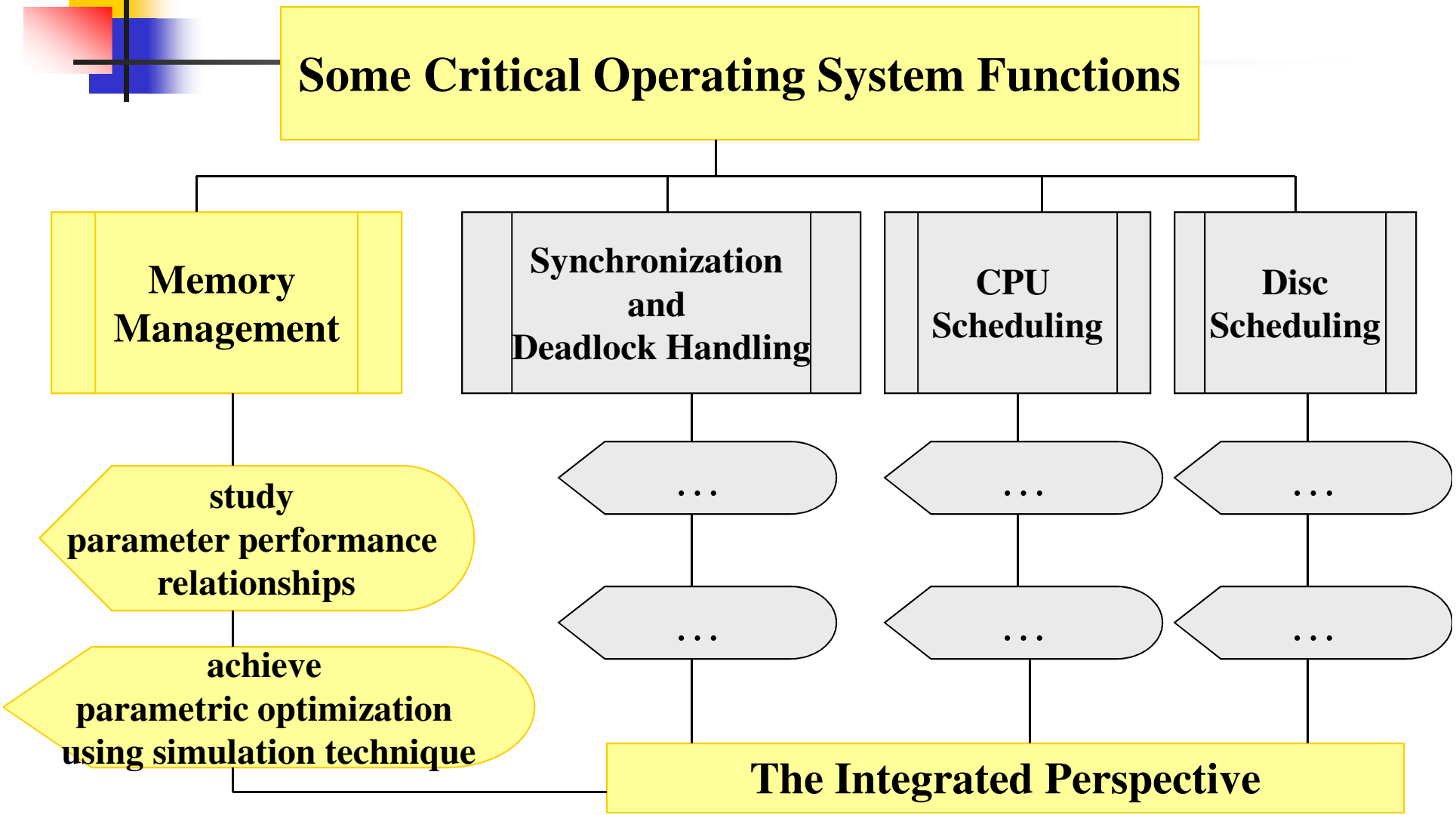


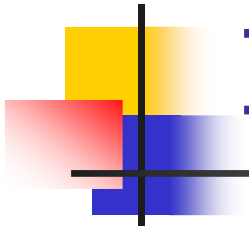
Parametric Optimization

An Alternative Approach of OS Study

- What is the critical OS function?
- What are the parameters involved?
- How to measure the performance?
- What is the relationship between parameter and performance?
- How to achieve optimization using simulation techniques?

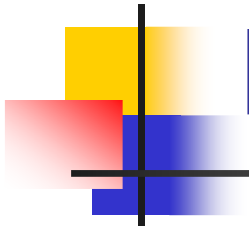
Parametric Optimization of Some Critical Operating System Functions





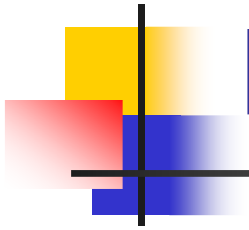
Introduction

- Multi-Process OS
- **Memory Management**
- **Paging Mechanism**
- **CPU Scheduling**



Memory Management

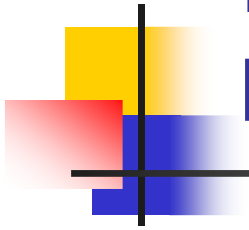
- Keep track of memory in use
- Memory allocation
- Manage swapping between main memory and disk



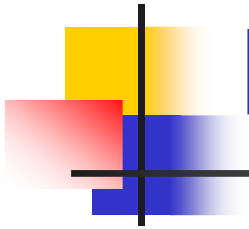
Memory Management (Cont.)

- Three disadvantage related to memory management are
  - Synchronization
  - Redundancy
  - Fragmentation

Memory Management (Cont.)
Parameters involved

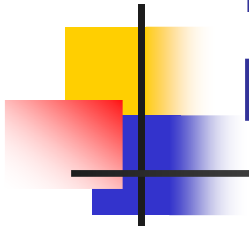


- Memory Size
- Disc access time (transfer time, latency and seek)
- Time slot for RR
- Compaction thresholds (percentage and hole size)
- RAM access time
- Fitting Algorithm
- Disc Scheduling algorithm choice (FIFO, SSTF, SCAN, LOOK, etc)
- Disc Structure and Capacity (Surfaces/tracks/etc.)
- Disc writing mechanism (where to write back processed pages)



Paging

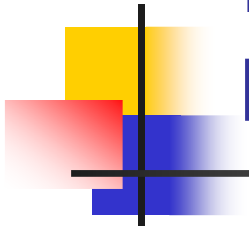
- Paging entails division of physical memory into many equal-sized frames
- When a process is to be executed, its pages are loaded into any available memory frames



Paging
Parameters Involved

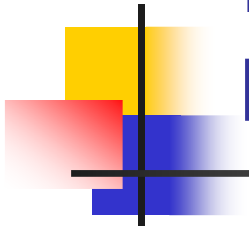
The parameters involved in this memory management scheme are:

- Page Size
- Page Replacement Algorithms, such as First-In-First-Out, Least-Recent-Used, Least-Frequently-Used and Random



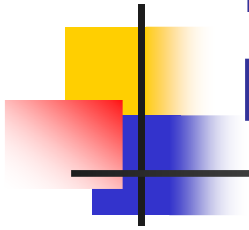
Paging
Effect of Page Size

- Large page size: internal fragmentation
- Small page size: requires large amounts of memory space to be allocated for page tables and more memory accesses potentially
- Finding an optimal page size: not easy, dependent on the process mix and the pattern of access.



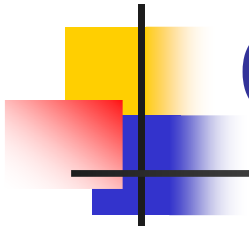
Paging
Effect of Page Replacement Algorithms

- LRU, FIFO, LFU and Random replacement are four of the more common schemes in use
- LRU is often used and is considered to be quite good
- LRU may require substantial hardware assistance



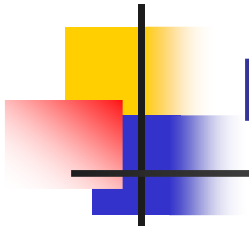
Paging
Performance Measures

- Average Waiting Time
- Average Turnaround Time
- CPU utilization
- CPU throughput
- Replacement ratio *(The ratio of number of page replacement to total number page accesses)*



CPU Scheduling

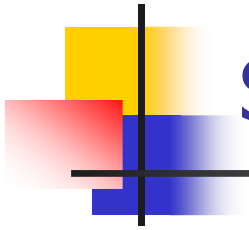
- Round Robin Mechanism
- Scheduling Criteria



CPU Scheduling
Round Robin Mechanism

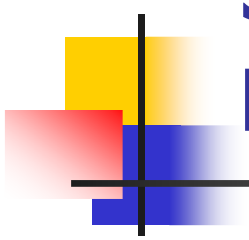
- Timesharing systems: a small unit of time – a time quantum is used
- Ready queue: circular queue
- CPU scheduler: traverses the ready queue, allocating the CPU to each process for a time interval of up to 1 time quantum

CPU Scheduling
Scheduling Criteria



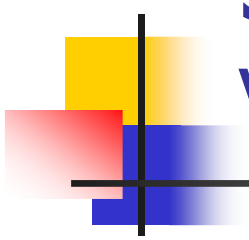
- CPU utilization: 40 percent (lightly loaded) to 90 percent (heavily used)
- Throughput: The number of processes that are completed per time unit.
- Turnaround time: The interval from the time of submission of a process to the time of completion.
- Waiting time

Simulation Specifications
Methodology



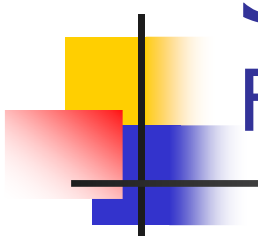
- 4 page replacement algorithms
- Randomizer: page access pattern
- dynamic algorithm: number of memory pages to be assigned to a process
- Analyze the collected data and examine their inter-relationship

Simulation Specifications
Variable parameters



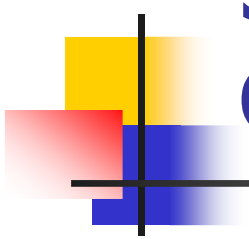
- Disc access time (seek + latency + (job size (in bytes)/500000) ms, where, seek and latency are variable parameters)
- Round Robin time Slot (a variable parameter, multiple of 1ms)

Simulation Specifications
Fixed parameters



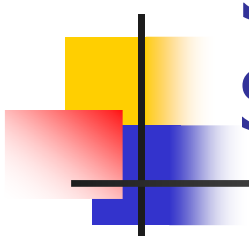
- Disc configuration (8 surfaces and 300 tracks/surface).
- Process sizes range (20KB to 2MB)
- Disc writing mechanism
- Disc capacity (512 MB, initially 50% full with jobs)
- Memory Size (32MB)
- RAM Access Time (14ms)
- Process execution times (2 ms to 100ms)

Simulation Specifications
Other Parameters

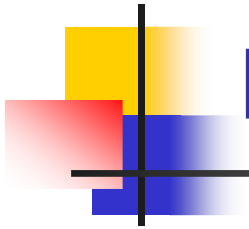


- Page access: random generator
- Timing wheel data structure
- CPU Round Robin fashion: as long as there are processes in the first level of the queue

Simulation Specifications
Simulation goal



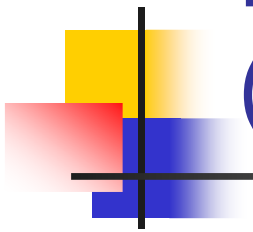
- The goal is to optimize some of the following performance measures such as:
  - Average waiting time
  - Average turnaround time
  - CPU utilization
  - Maximum turnaround time
  - Maximum waiting time
  - CPU throughput



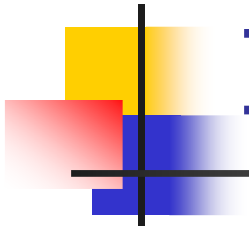
Memory Management Paging Module

- Disk: m processes are created (50% full)
- Page assignment: pages in memory proportional with process size
- Place new page in transfer queue from disk to memory
- Processor execute a chosen process: RR
- Move finished process from memory to disk (FCFS)
- Simultaneous execution of processes and transfer between disk and memory
- Page fault: a page is not available in the memory

Memory Management Paging Module (Cont.)

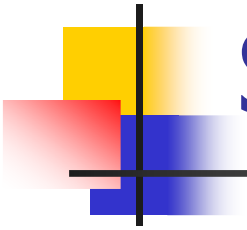


- Page sequences to be fetched from memory are generated randomly using the following mechanism: no new page is requested if a previously requested page is in transfer
- Remove page which belongs to current process: 4 algorithms, FIFO queue
- The current process transfers to a wait state: caused the page fault
- The simulation ends when all the processes finish execution and the queue is free.



Implementation Framework

- Process control block
- Queue
- Main memory
- Disk Drive
- CPU
- Simulator

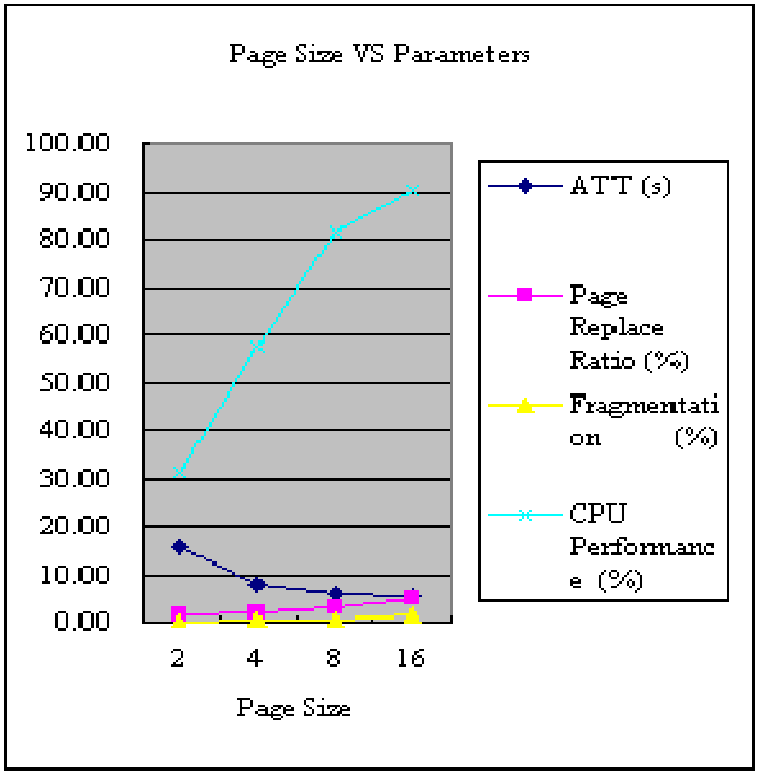


Simulation Results

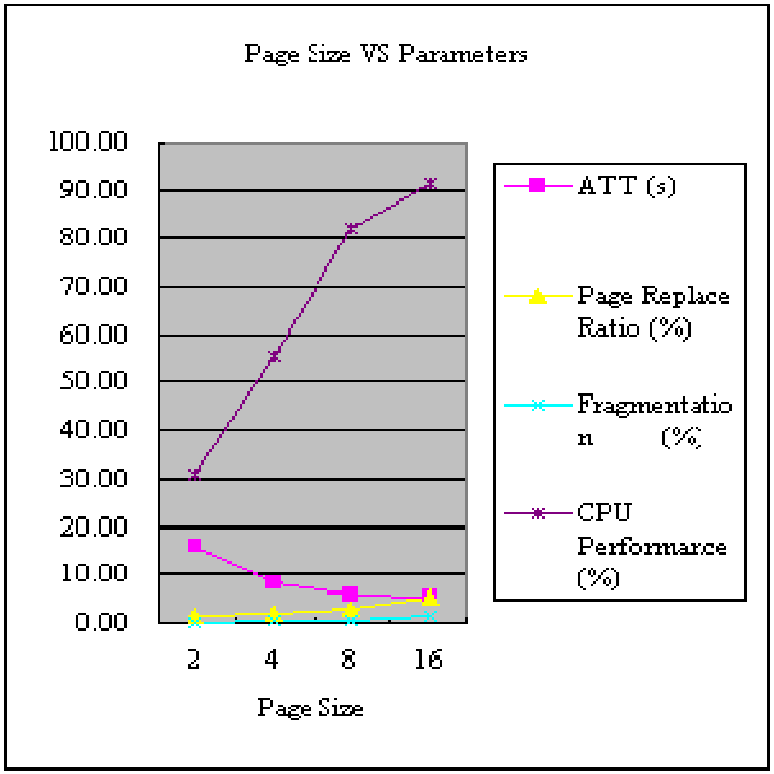
- Different combinations of parameters
- Eliminate the worst performing parameter combinations
- For example, if the simulation shows that a large time slot is superior to small ones, only large time slots are used in the simulation.

Simulation Result

Parameters change according to page sizes



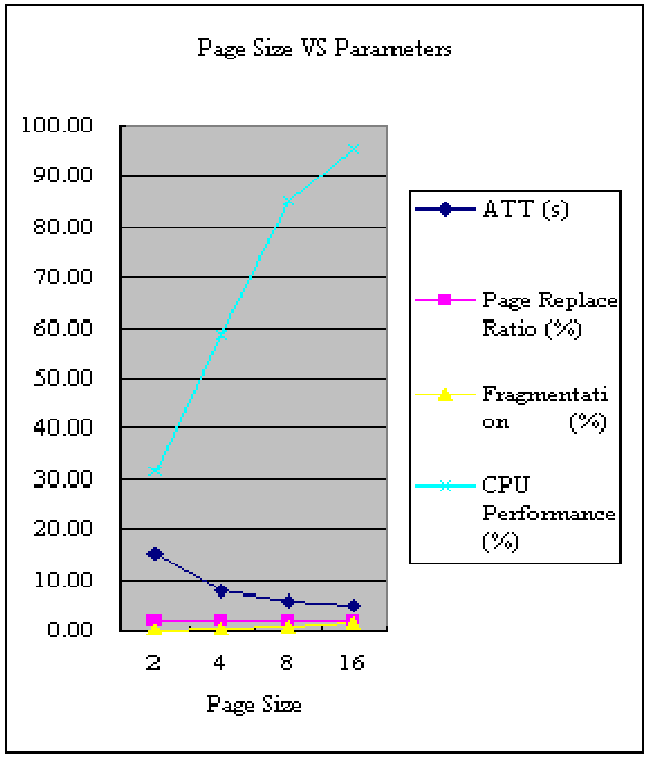
FIFO/Time Slot 8



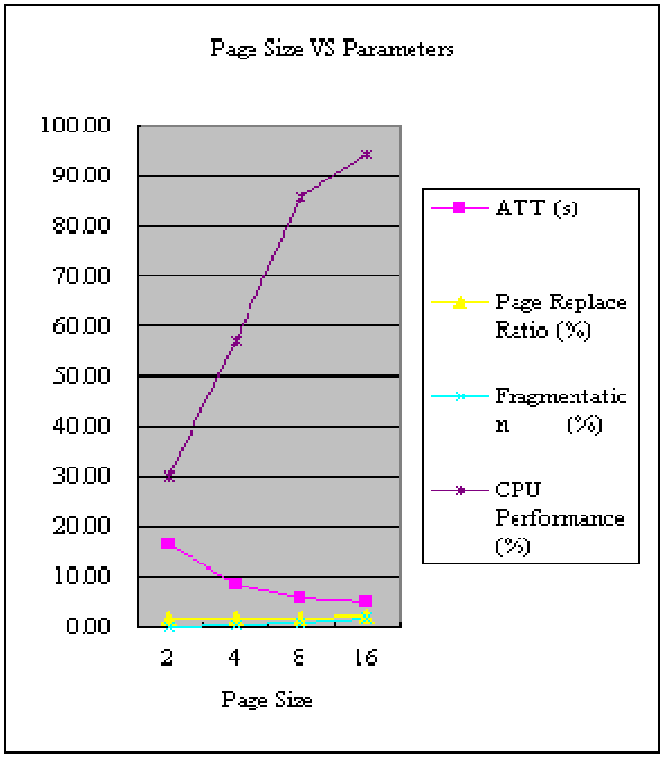
FIFO/Time Slot 4

Simulation Result

Parameters change according to page sizes (Cont.)



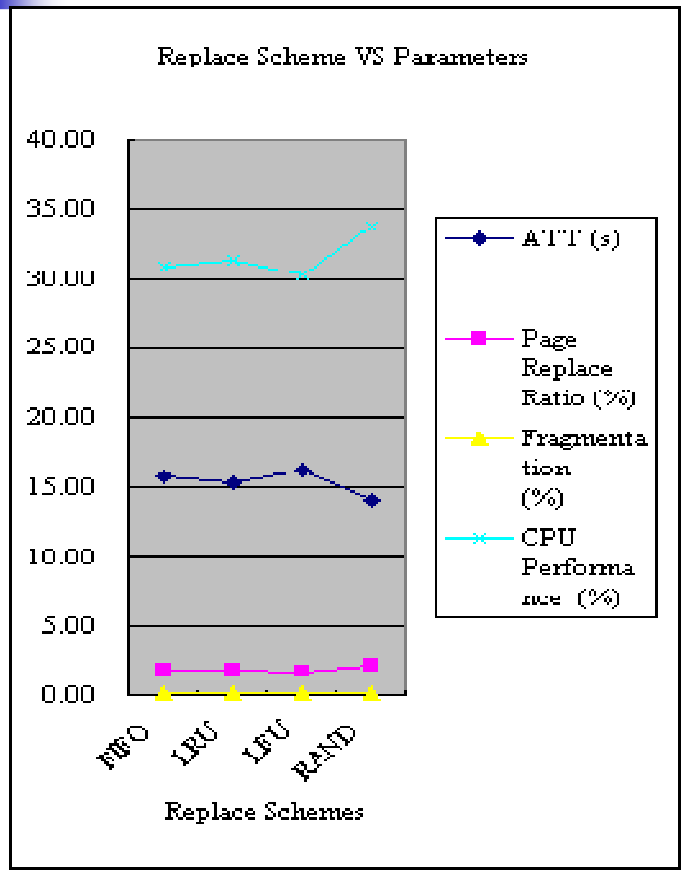
LRU/Time slot 8



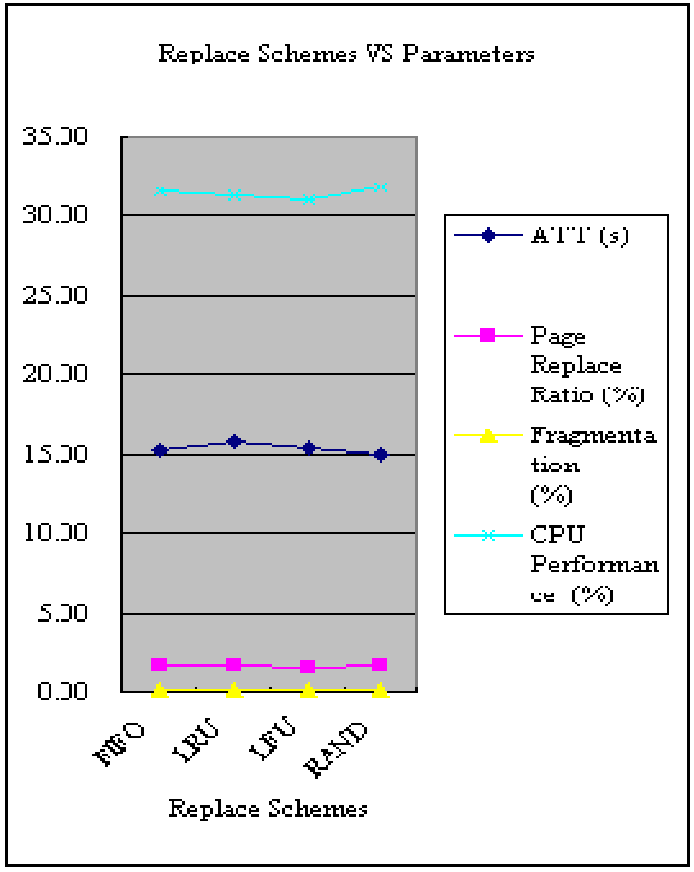
LRU/Time slot 4

Simulation Result

Parameters change according to page replacement schemes



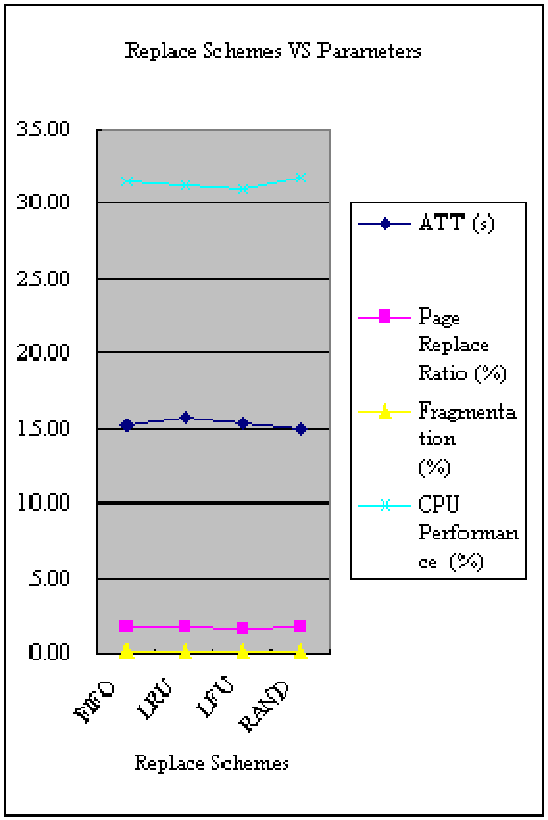
Page Size 2KB/Time slot 6



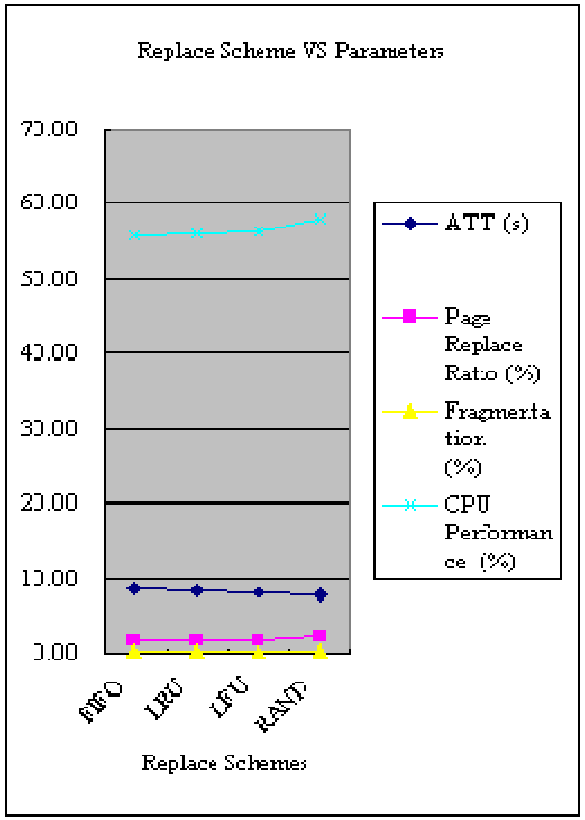
Page Size 2KB/Time slot 12

Simulation Result

Parameters change according to page replacement schemes (Cont.)



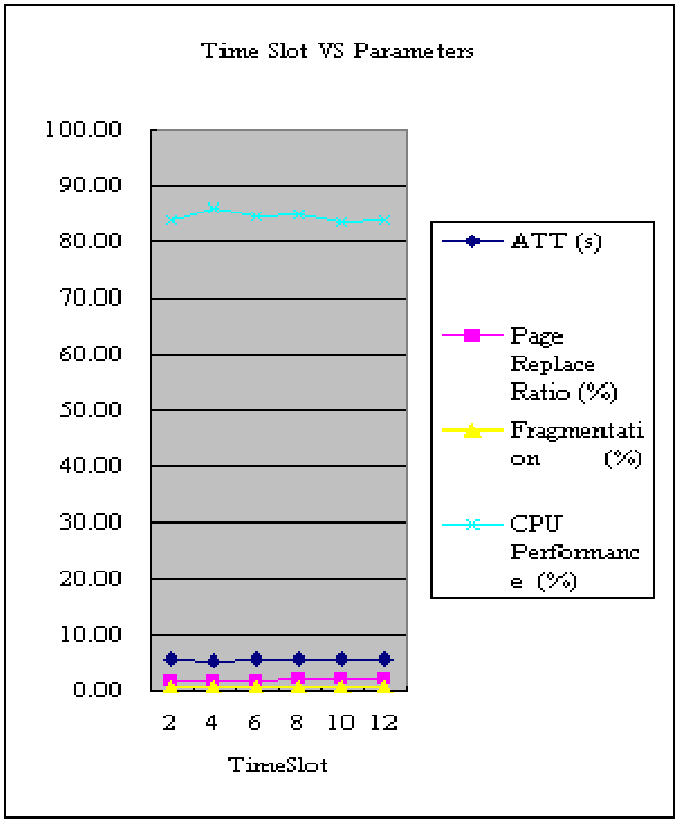
Page Size 4KB/Time slot 6



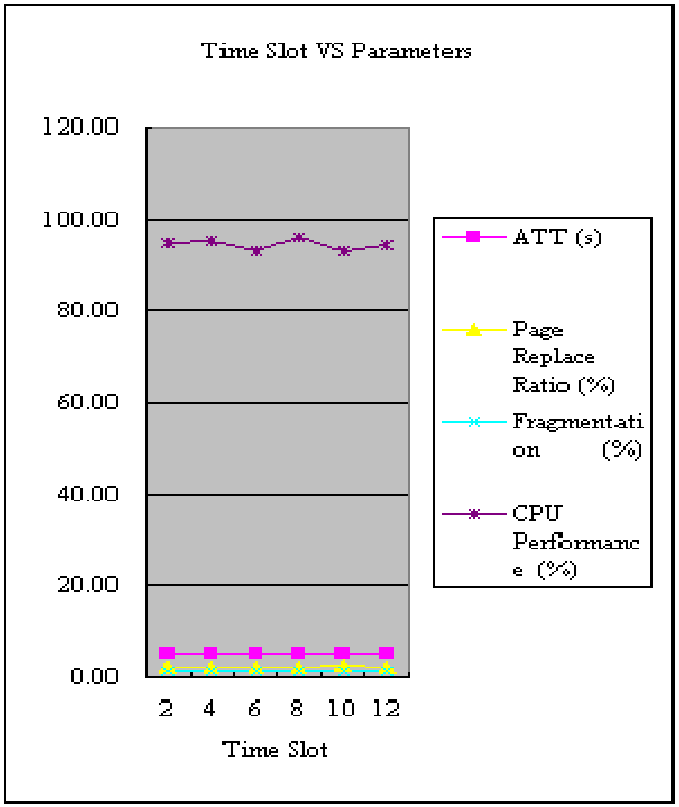
.Page Size 4KB/Time slot 12

Simulation Result

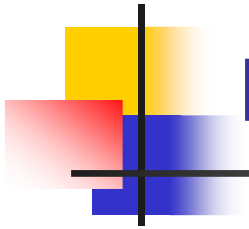
Effects of different time slots on different parameters



Page Size 8KB/FIFO



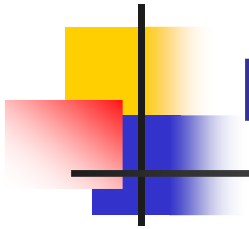
Page Size 16KB/RAND



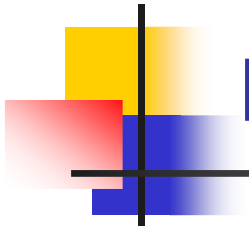
Conclusion
Parameter Analysis

- Page Size
- Page Replacement Algorithm
- Round Robin Time Slot
- Best Combination of parameters

Conclusion
Parameter Analysis (Cont.)

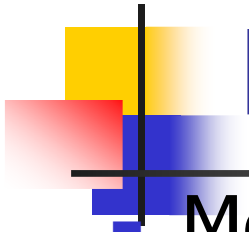


- Smaller page: more references in memory → longer ATT
- Smaller page: less internal fragmentation, more disk access time
- Large page: degeneration to continuous memory scheme; shorten ATT and increase CPU performance



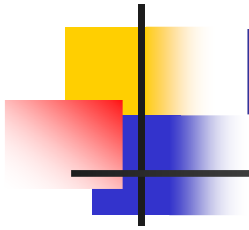
Conclusion
Parameter Analysis (Cont.)

- Random replacement performs best
- Page replacement ratio of LFU: high if page size >= 4KB
- Small RR time slot: higher context switch time, low CPU utilization, high turnaround time and waiting time



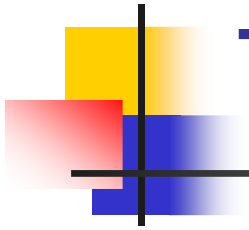
Future Work

- Modify to serve a specific platform or system
- Test the parameters in extremely multiplexed systems
- Some other parameters could also be simulated
- For example, the disk drive searching mechanism affects the turn around time of a process



References

- Tarek M.Sobh & Abhilasha Tibrewal , 2002. Parametric optimization of some critical operating system functions-an alternative approach to the study of operating system design, Bridgeport, CT, University of Bridgeport, Department of Computer Science and Engineering
- Wenle Zhao, 1998. Non-Platform Based Operating System Optimization , Bridgeport, CT, University of Bridgeport, Department of Computer Science and Engineering
- Avi Silberschatz, Peter Gal ,1999, Applied operating system concepts, John Wiley & Sons, Inc.
- Abraham Silberschatz, Peter Baer, 1999, Operating System Concepts (5th ed.).New York: John Wiley & Sons, Inc.
- Andrew S.Tan, 1987. Operating systems: design and implementation . New Jersey: Prentice-Hall, Inc.



Thank You