

Performance Assessment of CPU and Disk Components Using Discrete-Time Event Simulation

Evan Smtih
CS4328 Operating Systems
Mina Guirguis, Ph.D.

March 26, 2024

1 Abstract

This report presents the findings from a series of experiments conducted to assess the performance of CPU and Disk components of an operating system under varying workloads using a discrete-time event simulator. The objective was to understand how different rates of process arrivals (denoted by λ) affect key system performance metrics.

2 Running the Simulator

The simulation can be executed directly from the command line by providing specific arguments related to the simulation's configuration. Follow these steps to run the simulator:

1. Open your terminal or command prompt.
2. Navigate to the directory where the simulation scripts are located.
3. Download the get-pip script using curl:

```
[user]$ curl https://bootstrap.pypa.io/get-pip.py -o get-pip.py
```

4. Run the get-pip script:

```
[user]$ python get-pip.py
```

5. Install the required dependencies using the following command:

```
[user]$ python -m pip install matplotlib pandas
```

6. Run the simulation using the following command format:

```
[user]$ python main.py <lambda> <average_CPU_service_time> <average_Disk_service_time>
```

3 Introduction

The discrete-time event simulation project aims to evaluate the performance of two critical operating system components: the CPU and Disk. By simulating different workload intensities, this study provides insights into the system's behavior under various conditions.

4 Methodology

The simulator was designed to model the process flow within an operating system, focusing on the interaction between processes and the CPU/Disk components. Processes arrive at an average rate following a Poisson distribution, and service times are exponentially distributed.

5 Implementation Details

The simulation is structured around several key classes: CPU, Disk, Process, Event, and Simulator. These components work together to mimic the scheduling and processing of tasks within an operating system environment.

6 Results

This section presents the simulation results, showing how system performance metrics vary with different process arrival rates (λ).

6.1 Average Turnaround Time

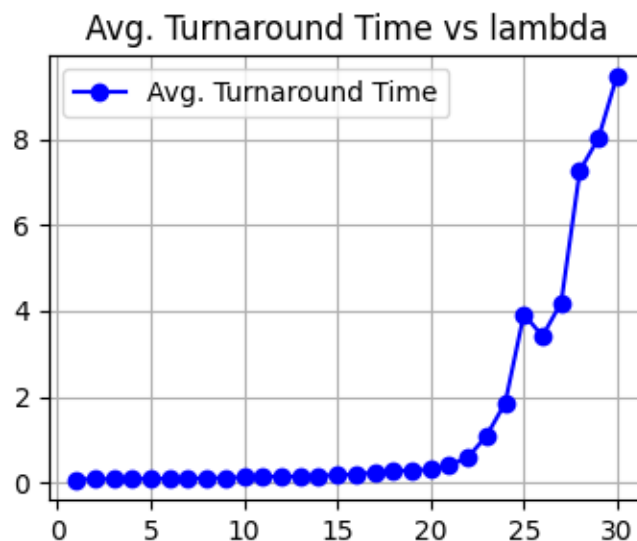


Figure 1: Average Turnaround Time vs. λ

6.2 Average Throughput

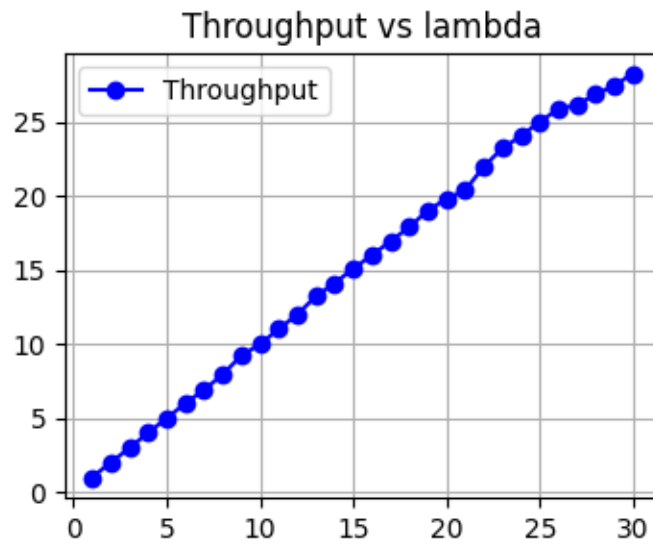


Figure 2: Average Throughput vs. λ

6.3 CPU Utilization

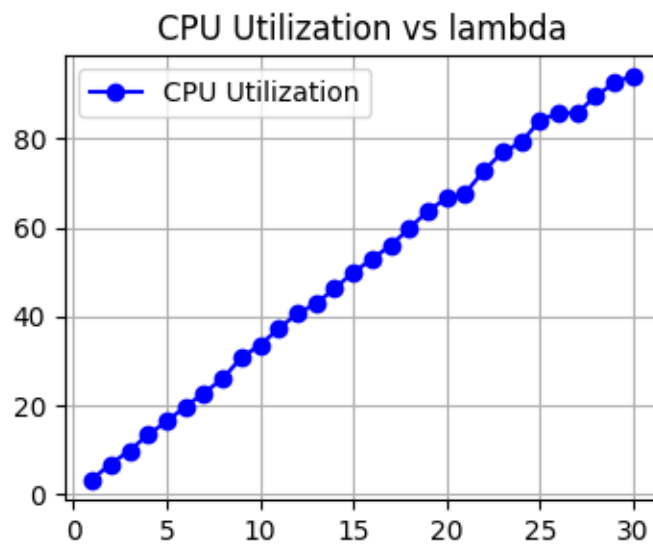


Figure 3: CPU Utilization vs. λ

6.4 Disk Utilization

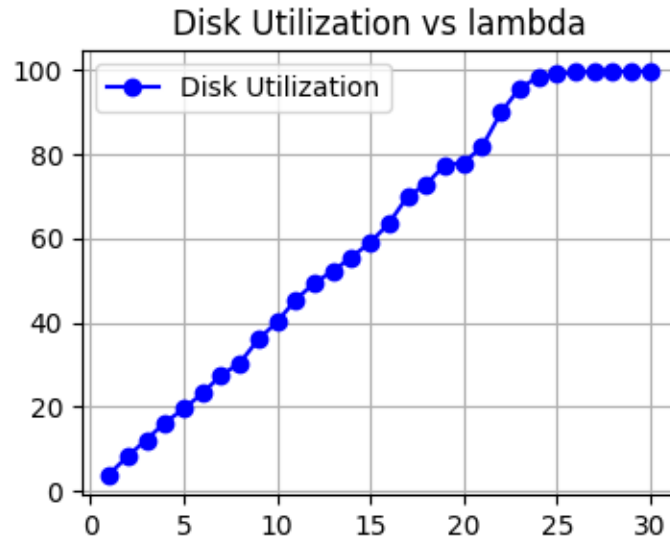


Figure 4: Disk Utilization vs. λ

6.5 Average Number of Processes in Ready Queue

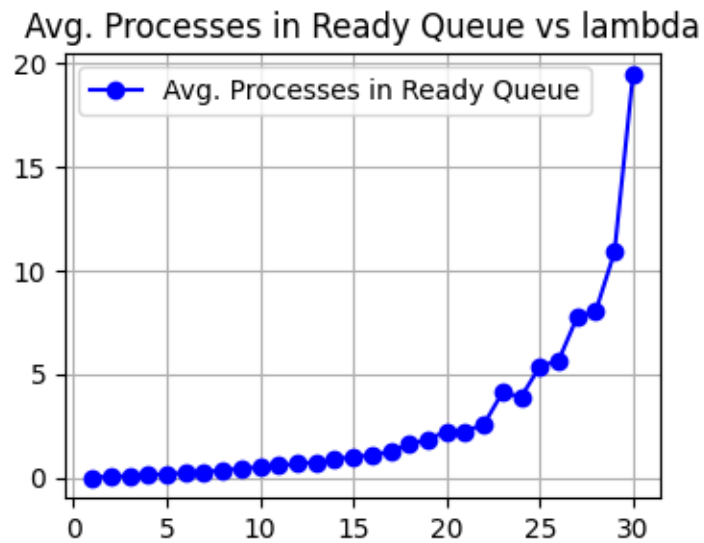


Figure 5: Average Processes in Ready Queue vs. λ

6.6 Average Number of Processes in Disk Queue

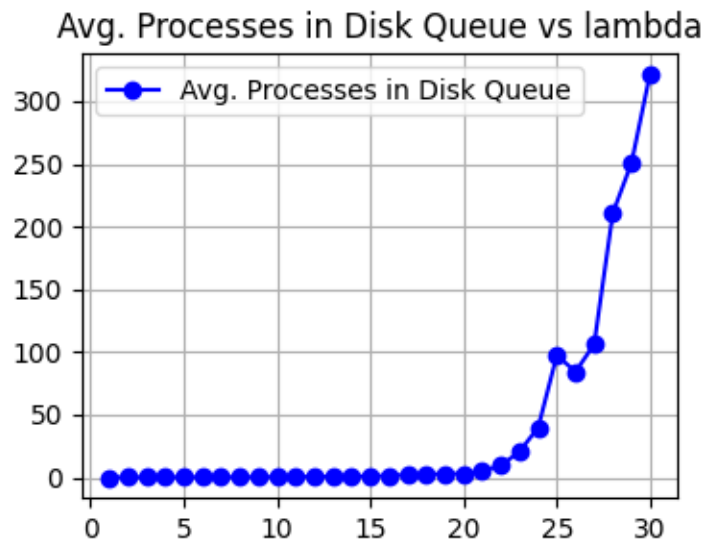


Figure 6: Average Processes in Disk Queue vs. λ

7 Analysis

The plots reveal a direct relationship between the process arrival rate and system performance metrics. As λ increases, both the average turnaround time and system utilization rates exhibit noticeable trends, indicative of the system's capacity and efficiency in processing tasks.

8 Conclusion

The discrete-time event simulation project has successfully demonstrated the impact of varying workload intensities on the performance of CPU and Disk components. These findings provide valuable insights into the design and management of operating systems to ensure optimal performance under different conditions.