# Numerical Results

In this chapter, we present the numerical results derived from both the analytical model and the simulation model, covering four distinct blockchain queuing scenarios. Each scenario has two queues, which are customer queue and the block queue, operating under varying combinations of customer class structures and impatience behavior.

The first scenario models a single-class customer system without impatience, serving as a baseline configuration with First-Come-First-Served (FCFS) discipline and no abandonment. The second scenario introduces two classes of customers with non-preemptive priority, distinguishing high-priority and low-priority customers in both queuing and service procedures. In the third scenario, we revisit the single-class setting while incorporating impatience behavior, where customers may abandon the system after a random impatience threshold. Finally, the fourth scenario combines both priority and impatience, modeling a two-class customer system with distinct abandonment rates and non-preemptive priority rules.

Across all scenarios, the blockchain service is influenced by ON/OFF operational states and employs a partial batch policy during block generation. Performance metrics such as throughput, blocking probability, waiting time, and impatient rate (where applicable) are computed and compared under varying parameter configurations. These results provide a comprehensive view of how priority, impatience, and system reliability jointly impact the overall performance of blockchain-based queuing systems.

## Scenario 1: Single-Class Customer without Impatience

The default values are as provided as below: , , , . The maximum capacity of the system is , and the maximum block size is .

Figure 5‑1 to Figure 5‑6 show the relationship between various performance metrics and the block size. Both simulation results and analytical results are shown for comparison.

Figure 5‑1 illustrates the impact of the block size on the average waiting time in the system (). As the block size increases from 11 to 19, the average waiting time steadily decreases. This is because larger blocks allow more customers to be served per consensus cycle, thereby reducing the time customers spend waiting in the queue. Lastly, the analytical results are in good agreement with the simulation results.

Figure 5‑2 illustrates the impact of the block size on the average waiting time in the customer queue () and the block size . As increases from 11 to 19, the value of exhibits a steadily decreasing trend. This behavior reflects the system's ability to more efficiently admit and process queued customers when larger blocks are allowed, thereby reducing queue buildup and shortening the average waiting duration. The decline in becomes more gradual as increases, indicating diminishing marginal improvements as the block size grows. Lastly, the analytical results are in good agreement with the simulation results.

Figure 5‑3 illustrates the impact of the block size on the average waiting time in the block queue (). As the block size increases from 11 to 19, remains nearly constant, indicating that the consensus phase exhibits relatively stable processing dynamics regardless of the number of customers per block. This trend suggests that the duration each block spends in the block queue is primarily governed by the consensus rate itself, and is not significantly affected by moderate increases in block size. Lastly, the analytical results are in good agreement with the simulation results.

Figure 5‑4 illustrates the impact of the block size on the blocking probability (). The blocking probability initially decreases as increases from 11 to 13, reaching a minimum plateau between and . However, beyond this point, begins to rise again. This non-monotonic trend reflects a trade-off between two opposing effects: increasing block size helps reduce blocking by serving more customers at a time, but excessively large blocks may delay service initiation due to the need to accumulate enough customers, thereby increasing the chance of blocking under limited queue capacity. Lastly, the analytical results are in good agreement with the simulation results.

Figure 5‑5 illustrates the impact of the block size on the system throughput (). As the block size increases from 11 to 19, the throughput remains virtually constant. This indicates that the overall rate at which customers are served is primarily constrained by the system’s service capacity or external parameters such as arrival rate, rather than by the block size. As a result, adjusting does not significantly influence the long-term average number of customers processed per unit time. Lastly, the analytical results are in good agreement with the simulation results.

Figure 5‑6 illustrates the impact of the block size on on the average number of blocks generated per unit time (). The plot reveals that remains nearly constant across different values of . This suggests that the total system throughput is maintained at a stable level, and as block size increases, the system compensates by generating fewer blocks of larger size. The inverse relationship between block size and block generation frequency preserves a steady output rate in terms of overall customer processing. Lastly, the analytical results are in good agreement with the simulation results.

Figure ‑: effect of block size on average waiting time in the system

Figure ‑: effect of block size on average waiting time in the queue

Figure ‑: effect of block size on average waiting time in the system

Figure ‑: effect of block size on blocking probability

Figure ‑: effect of block size on throughput

Figure ‑: effect of block size on average block number per unit of time