# Analytical Model

In this chapter, we are going to present four different scenarios for modeling blockchain-based systems: (1) single-class customers without impatience, (2) two-class customers without impatience, (3) single-class customers with impatience, and (4) two-class customers with impatience. Each of these scenarios is built upon a queuing-based abstraction of the blockchain process and aims to capture distinct behavioral features related to customer priority and abandonment. In all cases, as shown in Figure 3‑1, the system is composed of two queues with limited capacity: the customer queue, which temporarily holds customers before block generation, and the block queue, which represents the stage where customers participate in the consensus protocol after being grouped into a block.

Assume that the arrivals of customers follow a Poisson process, where the arrival rate is denoted by . In the multi-class scenarios, we further distinguish between HP and LP customers, whose respective arrival rates are and , so that the total arrival rate satisfies . After arriving at the customer queue, customers wait for the block generation process, that determined by an exponential distribution and occurs at a rate of (or and in the two-class case). Once a block is formed, a group of customers is transferred to the block queue, where the consensus process is carried out at a service rate denoted by (or and depending on customer class) which is determined by an exponential distribution.

In scenarios that involve impatience, we assume that customers may abandon the system while waiting in the customer queue if their waiting time exceeds a certain threshold. The impatience threshold is modeled as an exponential random variable with a rate for single-class customers, and rates and for HP and LP customers, respectively. Once a customer enters the block queue, impatience is no longer considered. In addition, we consider the operational reliability of the system by incorporating the possibility of the system state alternating between ON and OFF periods. During ON periods, both block generation and consensus operations are allowed to proceed, while during OFF periods, these operations are suspended. The durations of both ON and OFF periods are exponentially distributed. The transition rates between the two states are given by (ON to OFF) and (OFF to ON) respectively.