**摘要**

本研究探討一個以區塊鏈為基礎的排隊系統，該系統在不同的使用者優先權與不耐煩組合下，建構交易處理過程的數學模型，並引入 ON/OFF 運作狀態，以模擬區塊生成與共識階段的隨機可用性。系統由兩個串接的佇列構成：顧客佇列，客戶在此等待被打包進區塊；區塊佇列，已打包的客戶在此等待共識處理。系統採用部分批次服務機制進行區塊生成，允許最大數量的客戶打包成一個區塊。在 OFF 狀態期間，區塊生成與共識作業會暫停，然而顧客到達仍持續進行，若顧客佇列尚有容量，則新到達的客戶仍可進入排隊。

為深入描繪系統的行為與結構特性，本研究設計四種情境模型：（1）無不耐煩的單一類型客戶、（2）無不耐煩的雙類型客戶、（3）具不耐煩行為的單一類型客戶、（4）具不耐煩行為的雙類型客戶。針對每一情境，我們建構對應的多維馬可夫鏈來描述系統狀態，推導平衡方程並透過反覆計算求得穩態機率分布，進而計算多項效能指標，包括吞吐量、阻塞機率與平均等待時間。此外，本研究也探討不同參數對系統效能的影響。

為驗證理論分析結果的正確性，我們以 C++ 語言實作離散事件模擬，忠實重現各情境中的事件邏輯與服務規則。模擬結果不僅驗證了解析趨勢，也揭示多項值得注意的系統行為，包括非搶先式優先排程中公平性與效率的權衡、不耐煩機制帶來的效能提升，以及部分批次機制與 ON/OFF 動態對整體系統表現的影響。

**關鍵字：區塊鏈、非搶佔優先權、不耐煩、區塊生成、共識、部分批量服務、開/關機制**

# Abstract

This research investigates a blockchain-based queuing system that models the transaction process under varying combinations of customer priority and impatience, while incorporating ON/OFF operational states to reflect the stochastic availability of block generation and the consensus phases. The system consists of two sequential queues: a customer queue, where customers wait to be grouped into a block, and a block queue, where grouped customers await consensus. A partial batch service mechanism is employed for block generation, allowing up to a maximum number of customers to be batched into a block. During OFF periods, block generation and consensus operations are suspended, although customer arrivals continue and are admitted into the customer queue if queue capacity is available.

To capture the behavioral and structural characteristics of the system, four distinct scenarios are modeled: (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. For each scenario, a multi-dimensional Markov chain is constructed to describe the system state. Balance equations are derived and solved iteratively to obtain the steady-state distribution, from which key performance metrics, including throughput, blocking probability, and average waiting times, are calculated. The effects of various parameters on the performance metrics are explored.

To validate the analytical results, a discrete-event simulation is implemented in C++, faithfully replicating the event logic and service rules of each scenario. The simulation confirms the analytical trends and highlights several notable system behaviors, including the trade-off between fairness and efficiency in non-preemptive priority settings, the performance benefits introduced by impatience mechanisms, and the system-level effects of partial batch size and ON/OFF dynamics.

**Keywords: blockchain, non-preemptive priority, impatience, block generation, consensus, partial batch service, ON/OFF mechanism**