UDK 004.75:519.876.2

Khoshaba O.M.

REAL-TIME PERFORMANCE MANAGEMENT OF PRIVATE SOLANA DEPLOYMENTS USING MODEL PREDICTIVE CONTROL

Vinnitsia National Technical University

Анотація.

Приватні розгортання високопродуктивної інфраструктури блокчейну все частіше служать внутрішніми субстратами транзакцій та аудиту для корпоративних систем, де передбачуваний час підтвердження та обмежена поведінка хвоста при піковому навантаженні часто є цінніші, ніж пікова пропускна здатність. У тезах представлено інженерну концепцію управління продуктивністю приватного кластера Solana з використанням адаптивного циклу керування, реалізованого на вхідному шлюзі. Підхід розглядає шлях обробки транзакцій від початку до кінця як макромасштабний процес черг та регулює продуктивність, сприйняту користувачем, за допомогою прогнозного керування моделлю (MPC) в умовах багатовхідного-багатовихідного (MIMO) середовища. Вхідні дані керування вибрані таким чином, щоб бути практичними в реальних розгортаннях без нав'язливих інструментів валідатора, і включають формування доступу до шлюзу та політику пріоритетної плати на основі параметрів бюджету обчислень, тоді як вихідні дані включають затримку підтвердження з високим процентилем, медіанну затримку підтвердження, підтверджену пропускну здатність, виражену в кількості транзакцій за секунду, та коефіцієнт успішності, що відображає своєчасне досягнення цільового статусу зобов'язання. Оскільки ефективна пропускна здатність сервісу та моделі конкуренції змінюються з часом, метод включає онлайн-ідентифікацію системи за допомогою легкої рекурсивної оцінки з процедурами постійного збудження, придатними для вікон введення в експлуатацію. Оптимізатор горизонту відступу забезпечує досягнення цілей рівня обслуговування для затримки хвоста та надійності за допомогою м'яких обмежень та вводить явний показник вартості, щоб перешкоджати економічно неефективним стратегіям «ескалації комісій». Оскільки показники виробництва ще недоступні, у тезах визначено план оцінки на основі сценарно-орієнтованих режимів робочого навантаження (базовий рівень, збудження, стресове навантаження), перевірок прогнозної узгодженості та критеріїв операційної прийнятності, виражених через частоту порушень цілей затримки хвоста, стабільність пропускної здатності та контрольовані витрати на комісії. Запропонована розробка забезпечує практичний зв'язок між інженерією управління та блокчейн-операціями для приватних розгортань Solana, що дозволяє систематичне налаштування та вимірюване управління послугами за допомогою легких вимірювань.

Ключові слова: *приватний кластер Solana, управління продуктивністю, прогнозне керування моделлю, багатовхідне багатовихідне керування, затримка "хвоста", підтвердження транзакцій, керування допуском*

шлюзу, онлайн-ідентифікація системи, моделювання на основі черг, пріоритетні комісії, цілі рівня обслуговування.

Abstract.

Private deployments of high-throughput blockchain infrastructure increasingly serve as internal transaction and audit substrates for enterprise systems, where predictable confirmation time and bounded tail behaviour under bursty load are often more valuable than peak throughput. This paper presents an engineering concept for performance management of a private Solana cluster using an adaptive control loop implemented at the ingress gateway. The approach treats the end-to-end transaction processing path as a macro-scale queueing process and regulates user-perceived performance using Model Predictive Control (MPC) in a Multi-Input Multi-Output (MIMO) setting. Control inputs are selected to be practical in real deployments without intrusive validator instrumentation and include gateway admission shaping and a priority-fee policy based on compute-budget parameters, while outputs include high-percentile confirmation latency, median confirmation latency, confirmed throughput expressed as transactions per second, and a success ratio reflecting timely attainment of the target commitment status. Because effective service capacity and contention patterns vary over time, the method incorporates online system identification using a lightweight recursive estimator with persistent excitation procedures suitable for commissioning windows. The receding-horizon optimiser enforces service-level objectives for tail latency and reliability through soft constraints and introduces an explicit cost proxy to discourage economically inefficient "fee escalation" strategies. As production metrics are not yet available, the paper defines an evaluation plan based on scenario-driven workload regimes (baseline, excitation, stress), prediction-consistency checks, and operational acceptance criteria expressed through violation frequency of tail-latency objectives, throughput stability, and controlled fee expenditure. The proposed design provides a practical bridge between control engineering and blockchain operations for private Solana deployments, enabling systematic tuning and measurable service governance with lightweight measurements.

Keywords: *private Solana cluster, performance management, model predictive control, multi-input multi-output control, tail latency, transaction confirmation, gateway admission control, online system identification, queueing-based modelling, priority fees, service-level objectives.*

In private deployments, Solana-based infrastructure is increasingly used as an internal transaction substrate for enterprise services, where the decisive quality attributes are not peak throughput but predictable confirmation time and controlled tail behaviour under bursty load. In such environments, the operational target can be expressed as a soft real-time service-level objective: occasional deadline misses may be tolerated, but the proportion of late confirmations must remain below an agreed threshold while avoiding economically inefficient overpayment through priority fees. This motivates a control-oriented view of a private Solana cluster and its ingress layer as a constrained, time-varying processing pipeline whose user-perceived performance can be shaped by gateway-level actions.

The problem addressed in these theses is the design of a practical method for performance management of a private Solana deployment that (i) measures performance using lightweight observables available from a Remote Procedure Call (RPC) interface and gateway-side logs, (ii) captures the end-to-end dynamics at an aggregated time scale suitable for feedback control, and (iii) selects gateway-level actuation policies that keep tail confirmation latency within target bounds with a controlled violation rate. The aim of this work is to substantiate an engineering architecture based on Model Predictive Control (MPC) to regulate percentile confirmation latency and reliability indicators, while balancing throughput targets and the

economic cost proxy induced by priority-fee policies. The object of research is the end-to-end transaction submission and confirmation process in a private Solana cluster (ingress gateway plus validator pipeline). The subject of research is a control-oriented model of that process and a closed-loop policy that manipulates gateway-admissible inputs to regulate measurable outputs.

The relevance of the topic follows from two practical facts. First, user experience is dominated by tail latency rather than averages; therefore, percentile-based constraints (e.g., a high percentile of confirmation latency) are more appropriate control targets than mean delay under congestion. Methods that explicitly enforce tail-latency objectives in shared backends show that "average-first" control can mask intermittent but severe quality-of-service breakdowns, hence the need to treat tail behaviour as a first-class objective [10]. Second, Solana exposes economically meaningful levers for traffic classes under contention, including base fees and optional prioritisation fees, which can be set through the compute budget mechanism by selecting a compute unit limit and a compute unit price (in micro-lamports per compute unit) [2]. Commitment semantics also matter: applications may target a particular commitment level (processed, confirmed, finalised) as their definition of "done", and each level corresponds to a different confirmation guarantee [3]. In private clusters, "confirmed" is often a pragmatic feedback target because it is meaningful to applications yet sufficiently responsive for operational control.

The methodological foundation is an aggregated-time, multi-input, multi-output (MIMO) control loop deployed at the gateway. The controlled outputs are defined over fixed sampling windows and include: a high-percentile confirmation latency (tail latency), a median confirmation latency for robustness diagnostics, confirmed throughput, expressed as transactions per second, and a success ratio, defined as the fraction of submitted transactions that reach the target commitment level within a specified time budget. The controlled inputs are chosen to be implementable without invasive validator instrumentation or node restarts. The primary input is admission shaping at ingress (for example, rate limiting or an adaptive acceptance fraction per traffic class). A secondary input is the priority-fee policy, implemented as a rule that maps recent performance to compute unit prices and, possibly, to compute unit limits for specific transaction types, consistent with Solana's fee model [2]. A third input can be the offered-load setpoint of a workload generator used during commissioning and controlled experiments. These inputs form a coupled MIMO problem because admission affects both backlog and the completion-time distribution, and priority fees affect the scheduling share and confirmation dynamics under contention.

A key constraint is computational feasibility in real time: measurement aggregation, model update, and optimisation must finish within each sampling period. Therefore, the model is intentionally macro-scale and control-oriented rather than a faithful micro-level simulator. Internal contention drivers (such as account-lock hotspots, compute-heavy instructions, or background traffic from other services) are treated as disturbances that manifest as time-varying effective service capacity and variable delay. This abstraction aligns with the operational goal: it is not necessary to predict internal states precisely; instead, it is necessary to predict the near-future evolution of observable outputs over a short horizon and to compute admissible gateway actions that keep the tail-latency objective within a tolerance envelope.

The modelling choice is a discrete-time MIMO prediction model that relates current outputs to a short history of past outputs and inputs. In practice, an autoregressive model with exogenous inputs, or an equivalent state-space form, is suitable because it supports straightforward receding-horizon prediction. Given that effective capacity and contention patterns drift over time, online identification is required. A lightweight recursive estimator, such as recursive least squares with a forgetting factor, can update model parameters at each sampling step with modest computational overhead, enabling adaptation to regime changes. To avoid ill-conditioning and ensure informative data, the commissioning procedure must include persistent excitation. In engineering terms, this is implemented as controlled perturbations of a single input channel at a time (e.g., small step changes or pseudo-random switching of the admission setpoint) while keeping the other channels fixed during an off-peak window on the private cluster.

The MPC component uses the identified prediction model to forecast multiple outputs across a finite horizon and computes a short sequence of future inputs, applying only the first control action at each step (receding-horizon control). The design explicitly assigns roles to outputs. Throughput and median latency are treated primarily as optimisation objectives (for stability and responsiveness), while tail latency and success ratio are treated as constraints, initially in soft form to accommodate measurement noise in percentile estimation and unmodelled disturbances. To prevent a degenerate strategy in which the controller simply increases priority fees whenever latency rises, the optimisation includes an explicit cost proxy for fee expenditure. This is justified by the broader observation that transaction-fee mechanisms act as allocation mechanisms for scarce resources and can shape congestion outcomes in non-trivial ways; modern analyses of multidimensional fee markets formalise why fee parameters should be treated as part of the control surface rather than as static configuration [7], and mechanism-design perspectives clarify how fees interact with incentive and allocation properties under demand spikes [8]. Accordingly, the controller's goal is not "lowest latency at any cost", but latency governance under a constrained economic envelope.

The measurement and data-collection layer is designed to be lightweight and deployable. Gateway logs capture submission timestamps, transaction identifiers, admission decisions, and the fee policy parameters applied. Confirmation timestamps and commitment status are obtained through RPC-based queries, using the same commitment semantics documented for Solana clients [3]. Understanding confirmation and expiration behaviour is essential because expired transactions represent a distinct failure mode that can bias latency statistics if not handled correctly. The Solana guidance on confirmation and expiry provides an operational basis for distinguishing late confirmations from expirations and for interpreting observed status transitions [4]. In addition, public network performance reporting illustrates practical metrics used by ecosystem operators (uptime, throughput characteristics, and performance under stress) that can guide the selection of monitoring dashboards even in private deployments [5]. While a private cluster differs from the public mainnet, the measurement patterns are transferable across them.

Because the theses are positioned as an engineering concept rather than a completed production study, the "results" component is expressed as explicit acceptance criteria and an evaluation plan. The primary performance criterion is compliance with the chosen tail percentile's service-level objective: across sliding windows, the proportion of windows in which tail latency exceeds the deadline must remain below a specified tolerance. Secondary criteria include maintaining throughput within a target band, limiting the fraction of expirations or status failures, and controlling fee expenditure (e.g., bounding the average compute unit price or the total prioritisation fees per unit of confirmed throughput). The evaluation protocol includes three scenario classes. The baseline regime uses step changes in the offered load to estimate the system response and calibrate the sampling and horizon lengths. The excitation regime applies controlled perturbations to online identify and validate predictions. The stress regime uses bursty load patterns to validate constraint handling during fast overload episodes and to test the controller's ability to restore compliance without oscillation. Prediction quality is assessed using one-step-ahead and multi-step prediction errors for each output, as well as the stability of identified parameters in steady-state regimes. The practical deliverable is a repeatable commissioning workflow that produces a tunable controller configuration rather than a one-off benchmark.

The main limitations follow from the abstraction. Percentiles are non-linear functionals of the latency distribution and can be noisy for small sample sizes; therefore, early deployments should prefer conservative sampling windows and soft constraints. Model mismatch can be significant when the transaction mix changes abruptly or when internal contention sources arise, so the adaptation rate of the online estimator must be tuned to balance responsiveness with noise amplification. Finally, fee levers represent an economic control surface whose effects depend on cluster policy and contention conditions; thus, the system must be monitored for unintended "cost inflation" without proportional performance gains, and the cost proxy in the objective must be calibrated to the operator's priorities.

In conclusion, these theses substantiate a practical pathway for adaptive performance management of private Solana deployments by implementing a MIMO MPC loop at the gateway with lightweight measurements. The key contribution is an integrated design in which admission shaping and priority-fee policy are treated as explicit inputs, and tail confirmation latency, throughput, and reliability are treated as outputs governed by service-level objectives. Future work should prioritise robust MPC variants for improved uncertainty handling, richer disturbance proxies derived from low-cost telemetry (e.g., transaction mix indicators), automated weight selection based on business cost sensitivity, and deeper validation across representative private-cluster workloads. An academic analysis of Solana's transaction network properties can further inform scenario design and stress testing in controlled settings [6].

LIST OF REFERENCES

1. Zou D., Lu W., Zhu Z., Lu X., Zhou J., Wang X., Liu K., Wang K., Sun R., Wang H. OptScaler: A Collaborative Framework for Robust Autoscaling in the Cloud [Electronic resource] // Proceedings of the VLDB Endowment (PVLDB). - 2024. - Vol. 17, No. 12. - P. 4090–4103. - DOI: 10.14778/3685800.3685829. - URL: https://doi.org/10.14778/3685800.3685829 (accessed: 27.01.2026).

2. Solana Foundation. Transaction Fees [Electronic resource]. - URL: https://solana.com/docs/core/fees (accessed: 27.01.2026).

3. Anza. Solana Commitment Status [Electronic resource]. - URL: https://docs.anza.xyz/consensus/commitments/ (accessed: 27.01.2026).

4. Solana Foundation. Transaction Confirmation & Expiration [Electronic resource]. - URL: https://solana.com/developers/guides/advanced/confirmation (accessed: 27.01.2026).

5. Solana Foundation. Network Performance Report: July 2023 [Electronic resource]. - URL: https://solana.com/news/network-performance-report-july-2023 (accessed: 27.01.2026).

6. Alizadeh S., Khabbazian M. Solana’s transaction network: analysis, insights, and comparison [Electronic resource] // EPJ Data Science. - 2025. - Vol. 14. - Article 48. - DOI: 10.1140/epjds/s13688-025-00561-x. - URL: https://doi.org/10.1140/epjds/s13688-025-00561-x (accessed: 27.01.2026).

7. Diamandis T., Evans A., Chitra T., Angeris G. Designing Multidimensional Blockchain Fee Markets [Electronic resource] // 5th Conference on Advances in Financial Technologies (AFT 2023). Leibniz International Proceedings in Informatics (LIPIcs). - 2023. - Vol. 282. - P. 4:1–4:23. - DOI: 10.4230/LIPIcs.AFT.2023.4. - URL: https://doi.org/10.4230/LIPIcs.AFT.2023.4 (accessed: 27.01.2026).

8. Roughgarden T. Transaction Fee Mechanism Design [Electronic resource]. - arXiv preprint, 2021. - arXiv:2106.01340. - URL: https://arxiv.org/abs/2106.01340 (accessed: 27.01.2026).

9. Rawlings J. B., Mayne D. Q., Diehl M. Model Predictive Control: Theory, Computation, and Design. - 2nd ed. - [Electronic resource]. - Nob Hill Publishing, 2022. - URL: https://sites.engineering.ucsb.edu/~jbraw/mpc/ (accessed: 27.01.2026).

10. Ma L., Liu Z., Xiong J., Jiang D. QWin: Enforcing Tail Latency Service Level Objectives at Shared Storage Backend [Electronic resource]. - arXiv preprint, 2021. - arXiv:2106.09206. - URL: https://arxiv.org/abs/2106.09206 (accessed: 27.01.2026).

11. Alom I., Ferdous M. S., Chowdhury M. J. M. BlockMeter: An Application Agnostic Performance Measurement Framework for Private Blockchain Platforms [Electronic resource]. - arXiv preprint, 2022. - arXiv:2202.05629. - URL: https://arxiv.org/abs/2202.05629 (accessed: 27.01.2026).

Хошаба Олександр Мирославович — канд. техн. наук, доцент кафедри програмного забезпечення, Вінницький національний технічний університет

Khoshaba Oleksandr M. — Cand. Sc. (Eng) Assistant Professor of the Department of Software Engineering, Vinnytsia National Technical University, Vinnytsia