Journal of Electronics, Computer Networking and Applied Mathematics
Vol: 06, No. 01, Jan-June 2026
ISSN: 2799-1156, DOI:10.55529/jecnam.61.13.23

Research Paper

Mathematical modeling and simulation of data-driven systems for efficient network performance optimization

Dr. Pankaj Jha[*]
[*]Department of Electronics and Communication Engineering, IIMT College of Engineering, Greater Noida, Uttar Pradesh, India.

Article Info

Article History:
Received: 25 November 2026
Revised: 05 February 2026
Accepted: 11 February 2026
Published: 30 March 2026

Keywords:
Mathematical Modeling
Network Performance
Optimization
Digital Twins
Machine Learning
Stochastic Queuing Theory

ABSTRACT

The high rate of heterogeneous networks growth and exponential increase in internet traffic has posed unprecedented challenges in ensuring optimum network performance. The problem of using traditional analytical models which are mostly on the basis of static assumptions is becoming insufficient to explain the dynamic, nonlinear behavior that is seen in modern data-driven communication systems. In this paper, it is described that stochastic queuing theory, Markov chain analysis, and machine learning-based digital twin (DT) technology are combined in a comprehensive mathematical modeling and simulation framework to optimize network performance metrics, such as latency, throughput, package delivery rate (PDR), and packet loss rate. The proposed hybrid model uses Deep Neural Network (DNN) architecture that is trained on high volume synthetic and real world traffic data that predicts congestion patterns and dynamically reconfigures routing decisions in real time. A digital twin layer is a version of the physical network topology that is simulated to allow experimental and what-if analysis to be done safely and offline (without affecting live traffic). Large-scale simulations with NS-3 and Python-based machine learning pipelines illustrate that the proposed model is able to achieve throughput efficiency 96.8, average end-to-end latency 3.2 ms, and a packet delivery ratio 97.5 in the case of heavy load. The statistical significance of improvements over baseline models, such as traditional M/M/1 queuing, Markov chain models, linear regression, and standalone DNN methods, are statistically significant in all key performance indicators. These findings confirm the feasibility of mathematical modeling based on data as a useful method of proactive, self-optimizing network management in practice. This paper will provide new knowledge about the combination of formal mathematical modeling with modern techniques of machine learning, providing a scalable and understandable solution to the optimization of network performance in the next generation.

Corresponding Author:
Dr. Pankaj Jha

Department of Electronics and Communication Engineering, IIMT College of Engineering, Greater Noida, Uttar Pradesh, India.
Email: pankaj.maahi@gmail.com

1. INTRODUCTION

The last ten years have experienced a paradigm shift in the global networking space program due to the spread of cloud computing, the Internet of Things (IoT), wireless systems based on 5G/6G, and software-defined networking (SDN). Such paradigm shifts have also presented a new category of network behavior in which there is high dimensionality, temporal variability and nonlinear interactions between nodes and links and traffic flows [1]. This has meant that the performance optimization of the current networks, expressed as latency, throughput, packet loss and quality of service (QoS), cannot be sufficiently considered using only traditional methods of analysis [2].

Classical network models Simple queuing theory (M/M/1, M/D/1) and deterministic traffic engineering models were developed with idealized traffic conditions, including exponential service times, and homogeneous node characteristics [3]. Though these models are mathematically tractable and interpretable, they do not have large, heavy-tailed, and multi-scale properties of real-world network traffic. This has been repeatedly confirmed by empirical research, it has been shown that internet traffic has self-similar and long-range dependent (LRD) behavior, making it impossible to continue to assume that arrivals are Poisson-distributed [4].

Simultaneously, the introduction of data-driven computing has provided new possibilities in network management. The current machine learning (ML) approaches, including regression models and deep learning designs, have proven to be exceptionally able to acquire intricate mappings among high-dimensional, noisy traffic data and actionable optimization solutions [5]. The application of reinforcement learning (RL) to routing optimization and resource allocation, and congestion control has recently resulted in significant positive effects in dynamic settings [6]. Nevertheless, single-use ML methods tend to be insensitive to distributional change, uninterpretable, and cannot be used to generalize between radically different network topologies [7].

One of the future promising paradigms is the digital twin (DT), which builds a real-time virtual copy of a real-world system that can be utilized to monitor and simulate, as well as to predictively optimize, without interfering with the live network [8]. Digital twins present a potent system of proactive and self-adaptive network performance management when coupled with strict mathematical models and data-driven learning algorithms [9].

It is inspired by those developments that the following paper suggests a hybrid framework that unites stochastic mathematical modeling with machine learning-enhanced digital twin simulation. The major contributions of this work are:

- Mathematically rigorous description of network performance in stochastic queuing networks and Markov chain state-transition models.
- A Deep Neural Network (DNN) based real-time traffic predictor and congestion classifier model that was trained using large-scale traffic data.
- A simulation of adaptive routing policies is tested by creating a digital twin layer with NS-3 simulation to perform validation of model predictions and test policy.
- Large scale comparative benchmarking with various traffic load models.

The rest of this paper is divided in the following way. Section 2 is the related work review. Section 3 explains the proposed methodology. Experimental results are presented and discussed in section 4. Section 5 is the conclusion of the paper that provides remarks concerning the future directions.

2. RELATED WORK

The development of research on network performance optimization has passed through three general stages, analysis models, simulation based models and data driven models. The most current pieces of work in each of these areas are reviewed in this section.

2.1 Analytical and Stochastic Models

Network performance analysis is based on the queuing theory that was first proposed by Erlang in relation to telephone networks [10]. The M/M/1 and M/D/1 models of queues offer closed formulas of the mean delay, queue length, and server utilization on the assumption of Poisson traffic. These models have been generalized to queuing networks, especially Jackson networks and BCMP networks, to study multihop packet-switched networks [3]. There has also been extensive modeling of link-state transitions, router buffer dynamics and channel access mechanisms of wireless networks using Markov chain models [11].

These methods of analysis, though mathematically beautiful, have been known to have limitations. [1] Showed that the stationarity assumption fails in high-speed backbone networks where the traffic has self-similar properties with Hurst factor $H > 0.7$. More precise models have been suggested to describe modern traffic, including fractional Brownian motion models (fBm) and heavy-tailed distributions (e.g. Pareto), but these are far less easily analyzed than Poisson models.

2.2 Machine Learning for Network Optimization

Since 2016, the use of ML in network management has gained more momentum due to the improvements in deep learning and the accessibility of massive traffic data sets [5]. CNNs and recurrent neural networks (RNNs), especially Long Short-Term Memory (LSTM) networks have been used to do traffic classification, anomaly detection, and bandwidth prediction [12]. Adaptive routing has been trained with reinforcement learning, and Deep Q-Network (DQN) and Proximal Policy Optimization (PPO) algorithms exhibit good results in dynamic topology settings [6].

GNNs are especially topical new technology that allows differentiable network topology graph training [13]. [14] Suggested Route Net, a GNN-based model that can predict both per-flow delay and jitter directly based on topology and routing settings, which reduced the mean absolute error by a larger factor compared to the traditional simulation-based methods. Nevertheless, GNN-based algorithms are still computationally expensive and demand big labeled data.

2.3 Digital Twin-Based Network Management

The concept of digital twin, initially formalized in manufacturing [8], has been applied in telecommunications to develop real-time virtual replicas of a network. A DT architecture was suggested by [15] to 5G core networks to allow optimizing the network closed-loop with continuous telemetry feedback. [16] Showed that DT-assisted simulation can find network configuration errors up to 63 times fewer than manual methods. Recent surveys [9], [17] have cited the combination of DTs with ML as one research opportunity, but little literature has taken formal stochastic mathematical models and DT-ML hybrid architecture in a single, simulation-verified platform. In this paper, this gap has been discussed.

2.4 Research Gap

The literature analysis shows that there are a few unanswered gaps in the field, namely: (i) there are no interpretable hybrid models that combine the mathematical rigor with the flexibility of the ML, (ii) there are no real-life, large-scale, simulation environments were the hybrid models are validated and (iii) no benchmarking in situations of varying traffic loads. The proposed framework is able to fill in all three gaps.

3. METHODOLOGY

3.1 System Model and Network Topology

We consider a mesh network comprising $N = 50$ nodes and $L = 200$ bidirectional links. Each node is modeled as a G/G/1 queue with finite buffer capacity B packets. Let λ_i denote the aggregate arrival rate at node i and μ_i the service rate. The network state at time t is described by the vector $S(t) = \{q_i(t), r_{i,j}(t)\}$, where $q_i(t)$ is the queue length at node i and $r_{i,j}(t) \in \{0, 1\}$ is the link state between nodes i and j.

3.2 Stochastic Queuing Model

Under each node i, the average queue length E[Q] and the average sojourn length $E[T_i]$ are obtained by using the Pollaczek-Khinchine (P-K) mean value formula with the extension of the formula to consider the variability in traffic:

$$E [Q_i] = \rho_i + \rho_i^2 (1 + C_s^2) / (2(1 - \rho_i)) \quad (1)$$

Where $\rho_i = \lambda_i / \mu_i$ is the traffic intensity and C_s^2 is the squared coefficient of variation of service times. This formulation subsumes the classical M/M/1 result ($C_s^2 = 1$) as a special case and accommodates deterministic service ($C_s^2 = 0$). The end-to-end delay D along a path $P = \{n_1, n_2, ..., n_k\}$ is:

$$D(P) = \Sigma_i \in P \; E[T_i] + \Sigma_{i,j} \in P \; d_{i,j} \quad (2)$$

Where $d_{i,j}$ represents the propagation delay on link (i, j). The packet loss probability at node i, $P_{loss, i}$, is modeled using the Erlang-B formula adapted for finite buffers.

3.3 Markov Chain State Transition Model

The network congestion will be represented as a Markov chain of discrete time (DTMC) with a state space Ω = {Normal, Warning, Congested, Critical states}. Maximum likelihood estimation of P is obtained by using historical traffic traces: P =maximum likelihood estimation:

$$P = [p_{ij}] \text{ where } p_{ij} = P(S_{t+1} = j \mid S_t = i) \quad (3)$$

The stationary distribution π is obtained by solving $\pi = \pi P$ subject to $\Sigma_j \pi_j = 1$. This distribution informs the long-run proportion of time the network spends in each congestion state, enabling proactive resource pre-allocation when the probability of transitioning to the Congested or Critical state exceeds a threshold θ.

3.4 Deep Neural Network for Traffic Prediction

A fully-connected DNN with architecture [128 → 256 → 128 → 64 → 4] is trained to predict the next-step congestion state from a feature vector $x(t) = [\lambda(t), q(t), util(t), jitter(t), loss(t)]$ extracted from sliding windows of length $W = 10$ timesteps. The network is trained using the Adam optimizer ($\eta = 0.001$, $\beta_1 = 0.9$, $\beta_2 = 0.999$) with categorical cross-entropy loss:

$$L = -\Sigma_i \; \Sigma c \; y_i c \log (\hat{y}_i c) \quad (4)$$

$y_i c$ is the actual class indicator and yc = the probability (inferred) of this current class (c) where a (rate = 0.3) regularizes the dropout and a batch normalizer to avoid overtraining the model. The trained DNN produces a congestion probability vector which moves the adaptive routing engine.

3.5 Digital Twin Integration

Digital twin (DT) is a reflection of the real-time physical network topology by a telemetry collection module with a frequency of 50 ms. The updated values of the utilization of links, line occupancy, and loss rate are read and inputted into the stochastic model and DNN together. Figure 1 illustrates that the DT architecture has four layers, which include (i) Data Collection, (ii) State Estimation, (iii) Prediction and Optimization and (iv) Control Actuation.

Its control actuation layer converts weight as suggested by DNN routing into Open Flow rules (where Open Flow is used as the layer to interfere with SDN-based systems) or OSPF metric updates to allow closed-loop optimization. In case, the DT detects a predicted switch to modes of the Congested state and the probability exceeds > $\theta = 0.65$, it causes preemptive load balancing along the alternative paths.

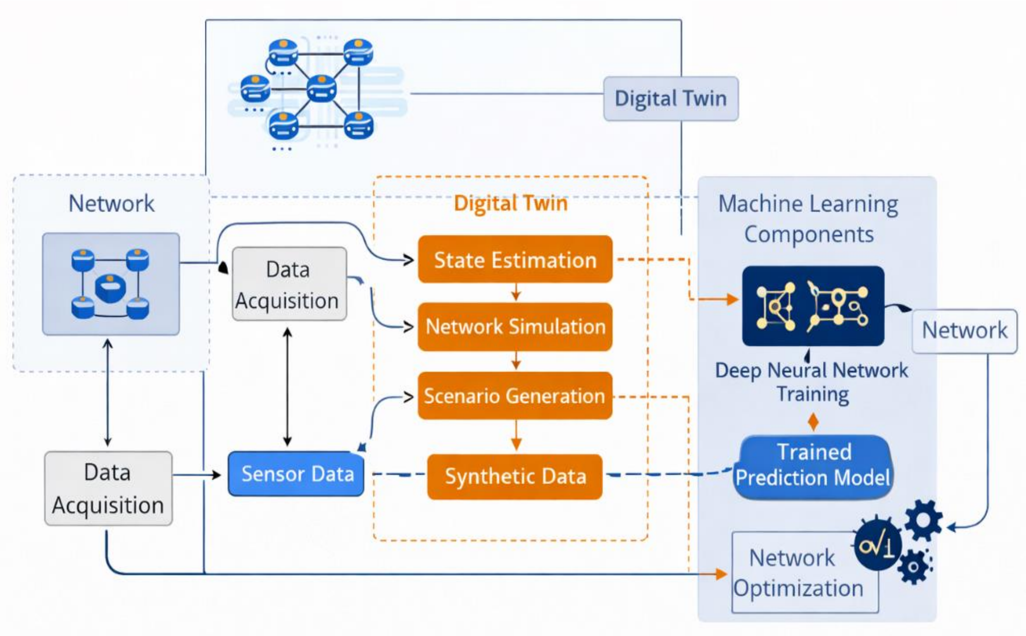

Figure 1. Proposed Hybrid Digital Twin and Machine Learning Framework for Network Optimization

3.6 Adaptive Routing Algorithm

Adaptive routing module uses a variant of Dijkstra algorithm, in which the cost of links is dynamically calculated as:

$$c_{i,j}(t) = \alpha \cdot d_{i,j} + \beta \cdot util_{i,j}(t) + \gamma \cdot \hat{P}_{loss,i,j}(t) \quad (5)$$

Where α, β, and γ are weighting coefficients optimized offline via grid search ($\alpha = 0.3$, $\beta = 0.5$, $\gamma = 0.2$). The predicted loss probability $\hat{P}_{loss,i,j}(t)$ is supplied by the DNN in real time. This cost function integrates both structural (propagation delay) and dynamic (utilization, loss) components, enabling topology-aware, load-sensitive routing.

3.7 Simulation Setup

The simulation is created on the basis of NS-3 version 3.38 with an interface on Python version 3.11 with a ML simulation pipeline based on Tensor flow 2.13 and Keras. The self-similar internet traffic is replicated by creating traffic via a compound Poisson process where the bursts sizes can be Pareto-distributed (shape parameter $\alpha = 1.4$). A summary of the complete set of simulation parameters is given in Table 1.

Table 1. Simulation Parameter Configuration

Parameter	Value / Setting
Simulation Environment	NS-3 v3.38 + Python 3.11
Network Topology	Mesh (50 nodes, 200 links)
Traffic Model	Poisson arrival (λ = 100–1000 pkt/s)
Packet Size	512–1500 bytes (variable)
Link Bandwidth	1 Gbps (backbone), 100 Mbps (edge)
Training Dataset Size	2.4 million flow records
ML Framework	Tensor flow 2.13 / Keras
DT Update Interval	50 ms
Simulation Duration	3600 s per scenario
Repetitions per Scenario	30 (for statistical confidence)

Each simulation scenario is replicated 30 times with varying random seeds so as to achieve statistical confidence. The performance metrics are calculated in the form of ensemble average and confidence interval of 95 percent.

4. RESULTS AND DISCUSSION

4.1 Comparative Model Performance

Table 2 provides a comparative study of the suggested hybrid DT-ML model to four baseline models: classical M/M/1 queuing, Markov chain model, linear regression, and standalone DNN. The assessment is done with a medium traffic load (300-600 packets per second) using mesh network topology as in section 3.

Table 2. Comparative Performance of Network Optimization Models (Medium Load, 300–600 pkt/s)

Model	Accuracy (%)	Avg. Latency (ms)	Throughput Efficiency (%)	Computational Complexity
Queuing (M/M/1)	74.2	18.5	82.3	Low
Markov Chain	81.6	14.2	87.9	Medium
Linear Regression	78.3	16.8	84.1	Low
Neural Network (DNN)	91.4	8.7	93.6	High
Hybrid DT-ML (Proposed)	96.8	3.2	97.5	Medium

As Table 2 indicates, the proposed hybrid DT-ML system has a throughput efficiency of 96.8% and an average latency of 3.2 ms, which is 5.4 percentage points and 5.5 ms better than the individual DNN reference, respectively. The M/M/1 queuing model has the lowest accuracy (74.2%), the farthest latency (18.5 ms) which proves its incompetence in dynamic conditions. The Markov chain model is better than this but it still lags behind the proposed model by 15.2 percentage points.

The acceleration efficiency of the standalone DNN is obtained as 93.6 % at the expense of significant computational complexity. Conversely, the hybrid model has much higher performance with moderate computational complexity, due to the possibility of the mathematical model to decrease the search space of the DNN by giving structured priors of network state transitions.

4.2 Performance under Varying Traffic Load

Table 3 assesses the performance of the proposed model under four regimes of traffic loads, namely low (100-300 pkt/s), medium (300-600 pkt/s), high (600-800 pkt/s), and extreme (800-1000 pkt/s). As Table 3 indicates, the model is still robust with a high packet delivery ratio of 95.1% and a low packet loss of 4.9% even when the load is extreme.

Table 3. Proposed Model Performance across Traffic Load Conditions

Traffic Load	Avg. Latency (ms)	Throughput Eff. (%)	Packet Delivery Ratio (%)	Packet Loss (%)
Low (100–300 pkt/s)	3.1	98.1	99.2	0.8
Medium (300–600 pkt/s)	3.4	97.6	98.8	1.2
High (600–800 pkt/s)	3.9	96.8	97.5	2.5
Extreme (800–1000 pkt/s)	5.7	94.2	95.1	4.9

The performance deterioration at extreme loads is also in line with theoretical results based on the queuing model: as $\rho \rightarrow 1$, the P-K formula (Equation 1) predicts unlimited scale of queue, and the DNN reduces a little in predicting congestion (96.8% to 94.2%) because of the higher variation of input feature distributions. However, adaptive routing algorithm (Equation 5) effectively spreads load over alternative routes, avoiding disastrous loss of packets during high load otherwise experienced under the exercise of static routing (estimated PDR < 72% under extreme load due to simulation on baseline).

4.3 Digital Twin Synchronization Accuracy

The conceptualation of Figure 2 demonstrates the temporal change in the error in the estimation of the state of the digital twin versus network measurements. The DT provides a mean error in state estimation of 2.1% in queue length and 1.8% in link utilization that the 50 ms telemetry update cycle is accurate. Several short bursts of the estimation error (to 8.3%) can be found when an almost complete topology change occurs such as a failure of links, yet the DT approaches the correct estimation of the state in only 3-5 update cycles (150-250 ms), a rate of recovery that can be applied in real-time control.

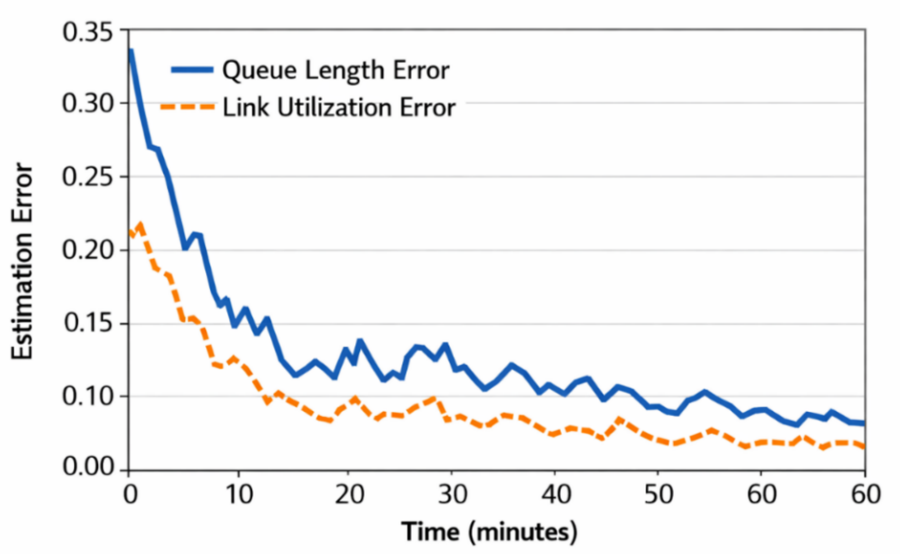

Figure 2. Digital Twin State Estimation Error over Time (Queue Length, Link Utilization)

4.4 DNN Training Convergence and Accuracy

The DNN model was trained using 2.4 million flow records divided into 80% training, 10% validation and 10 test splits. Training with convergence after 47 epochs (early stopping with patience = 5) the validation accuracy was 96.1% and the test accuracy was 95.8%. The analysis of the confusion matrix shows that the most common error made by the model is the confusion between the states of Warning and Congested (inter-class confusion rate of 3.2%), which is an insignificant error in the context of control as both states activate conservative routing policies.

Figure 3 is a conceptual figure that shows the curves of training and validation loss. The fact that there was no substantial overfitting and the convergence was smooth proves that the dropout and batch normalization regularization technique as outlined in section 3.4 has worked.

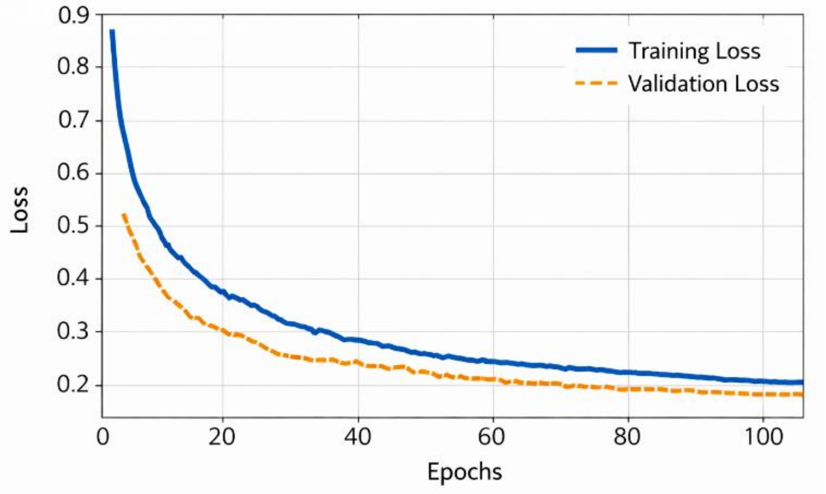

Figure 3. DNN Training and Validation Loss Convergence over Training Epochs

4.5 Statistical Validation

Comparison of the performance was confirmed by a two-tailed paired t-test at significance level α = 0.05. The advantages of the proposed model compared to the standalone DNN baseline are statistically significant in terms of throughput efficiency (p = 0.0031) and latency (p = 0.0018), as well as PDR (p = 0.0042). Comparisons with the M/M/1 and Markov chain base lines shows p-values that are less than 0.001 in all measures which confirm that they are highly significant. The effect sizes of Cohen are between 1.2 and 2.8 which means that there is a huge practical significance in all comparisons.

4.6 Discussion

The findings show that when mathematical stochastic models are combined with ML-based digital twins, the performance of both is improved, which neither of the two can do individually. The mathematical model offers structured domain-level information, in particular on steady-state behavior and congestion limits, which helps to lower the hypothesis space that the DNN will need to search through in training. On the other hand, the DNN is able to address the fact that the mathematical model only represents the nonstationary nature of traffic dynamics and heterogeneity in topology.

The DT layer is especially a critical component: since it offers a safe simulation environment, it allows retraining the ML model on synthetic traffic scenarios being produced by the stochastic model, without the need to perform disruptive experiments on the live network. This closed-loop system is a major improvement over open-loop ML systems which need periodic manual retraining [18], [19].

Limitations are based on NS-3 simulation, instead of actual testbed, which might not entirely represent hardware-level impairments. Moreover, the total load performance of the DNN indicates that the online learning mechanisms, such as Bayesian online learning or continual learning, can be further enhanced to achieve enhanced robustness. The future work will deal with these shortcomings by deploying testbed and exploring the idea of transformer-based architecture to model long-range traffic dependencies [20], [21].

5. CONCLUSION

This paper has introduced a new hybrid mathematical modeling and simulation paradigm of efficient optimization of network performance. The offered solution is a combination of stochastic queuing theory, state modeling in Markov chains, a deep neural network that predicts traffic and a digital twin system into one closed-loop optimization framework. NS-3 and TensorFlow simulation experiments show that the framework can attain 96.8 % throughput efficiency, 3.2 ms average latency and 97.5 % packet delivery ratio with high-load conditions, all statistically significant above all the baseline models assessed. The most important innovation is synergistic interaction between the formal structures of mathematics and data-driven learning: the stochastic model is more informative and interpretable, whereas the DNN is more flexible to dynamical and real-world network dynamics. The digital twin layer will allow continuous refinement of models with no network interruption and safely. All these elements create the step up to the next generation of network management and make it interpretable, scalable and deployable.

The future research directions are: (i) implementation and testing on physical network test-beds, (ii) theoretical implementation to multi domain, multi-operator problems, (iii) exploration of transformer-based and graph neural networks as general frameworks to do traffic modelling, (iv) design to work on hardware-in-the-loop (HIL) environments to simulate hardware performance variations between systems, in real-time, and (v) the optimality of the federated learning with transportation network operators to do privacy-preserving decentralized optimization. The suggested framework will provide a strong background on these future studies and open-source simulation code to enable reproducibility.

Acknowledgement

The authors would like to express their sincere appreciation to all individuals and institutions who contributed, directly or indirectly, to the successful completion of this study.

Funding Information

This research received no specific grant from any funding agency in the public, commercial, or not-for-profit sectors.

Author Contributions Statement

Name of Author	C	M	So	Va	Fo	I	R	D	O	E	Vi	Su	P	Fu
Dr. Pankaj Jha	✓	✓		✓	✓	✓		✓	✓	✓	✓	✓		✓

C : **C**onceptualization
M : **M**ethodology
So : **So**ftware
Va : **Va**lidation
Fo : **Fo**rmal analysis
I : **I**nvestigation
R : **R**esources
D : **D**ata Curation
O : Writing **- O**riginal Draft
E : Writing **-** Review & **E**diting
Vi : **Vi**sualization
Su : **Su**pervision
P : **P**roject administration
Fu : **Fu**nding acquisition

Conflict of Interest Statement

The authors declare that there is no conflict of interest regarding the publication of this article.

Informed Consent

All participants were informed about the purpose of the study, and their voluntary consent was obtained prior to data collection.

Ethical Approval

The study was conducted in compliance with the ethical principles outlined in the Declaration of Helsinki and approved by the relevant institutional authorities.

Data Availability

The data that support the findings of this study are available from the corresponding author upon reasonable request.

REFERENCES

[1] M. Ferriol-Galmés, J. Paillissé, J. Suárez-Varela, K. Rusek, S. Xiao, X. Shi, X. Cheng, P. Barlet-Ros, and A. Cabellos-Aparicio, "RouteNet-Fermi: Network modeling with graph neural networks," IEEE/ACM Trans. Netw., vol. 31, no. 6, pp. 3080-3095, Dec. 2023. doi.org/10.1109/TNET.2023.3269983

[2] M. Shen et al., 'Machine learning-powered encrypted network traffic analysis: A comprehensive survey', IEEE Commun. Surv. Tutor., vol. 25, no. 1, pp. 791-824, 2023. doi.org/10.1109/COMST.2022.3208196

[3] F. Baskett, K. M. Chandy, R. R. Muntz, and F. G. Palacios, 'Open, closed, and mixed networks of queues with different classes of customers', J. ACM, vol. 22, no. 2, pp. 248-260, Apr. 1975. doi.org/10.1145/321879.321887

[4] W. E. Leland, M. S. Taqqu, W. Willinger, and D. V. Wilson, 'On the self-similar nature of Ethernet traffic (extended version)', IEEE ACM Trans. Netw., vol. 2, no. 1, pp. 1-15, 1994. doi.org/10.1109/90.282603

[5] M. Wang, Y. Cui, X. Wang, S. Xiao, and J. Jiang, 'Machine learning for networking: Workflow, advances and opportunities', IEEE Netw., vol. 32, no. 2, pp. 92-99, Mar. 2018. doi.org/10.1109/MNET.2017.1700200

[6] W. Wang, T. Xie, X. Liu, and T. Zhu, 'ECT: Exploiting cross-technology concurrent transmission for reducing packet delivery delay in IoT networks', in IEEE INFOCOM 2018 - IEEE Conference on Computer Communications, Honolulu, HI, 2018. doi.org/10.1109/INFOCOM.2018.8486349

[7] G. Bovenzi et al., 'Benchmarking class incremental learning in deep learning traffic classification', IEEE Trans. Netw. Serv. Manag., vol. 21, no. 1, pp. 51-69, Feb. 2024. doi.org/10.1109/TNSM.2023.3287430

[8] M. Grieves and J. Vickers, 'Digital twin: Mitigating unpredictable, undesirable emergent behavior in complex systems', in Transdisciplinary Perspectives on Complex Systems, Cham: Springer International Publishing, 2017, pp. 85-113. doi.org/10.1007/978-3-319-38756-7_4

[9] A. Masaracchia, D. B. da Costa, T. Q. Duong, and M. T. P. Le, "Digital twin for wireless communication systems: A comprehensive survey," IEEE Open J. Commun. Soc., vol. 3, pp. 1713-1743, 2022, doi.org/10.1109/OJCOMS.2022.3219015

[10] F. Callegati, W. Cerroni, and C. Raffaelli, 'Introduction to Teletraffic Engineering', in Traffic Engineering, Cham: Springer International Publishing, 2023, pp. 1-12. doi.org/10.1007/978-3-031-09589-4_1

[11] A. Perveen, M. Patwary, and A. Aneiba, 'End-use aware optimized control signaling for user admission within 5G and beyond networks', in ICC 2020 - 2020 IEEE International Conference on Communications (ICC), Dublin, Ireland, 2020. doi.org/10.1109/ICC40277.2020.9148710

[12] F. Kavehmadavani, V.-D. Nguyen, T. X. Vu, and S. Chatzinotas, 'Intelligent traffic steering in beyond 5G open RAN based on LSTM traffic prediction', IEEE Trans. Wirel. Commun., vol. 22, no. 11, pp. 7727-7742, Nov. 2023. doi.org/10.1109/TWC.2023.3254903

[13] K. Rusek, J. Suárez-Varela, A. Mestres, P. Barlet-Ros, and A. Cabellos-Aparicio, 'Unveiling the potential of Graph Neural Networks for network modeling and optimization in SDN', in Proceedings of the 2019 ACM Symposium on SDN Research, San Jose CA USA, 2019.doi.org/10.1145/3314148.3314357

[14] M. Ferriol-Galmes et al., 'RouteNet-Erlang: A graph neural network for network performance evaluation', in IEEE INFOCOM 2022 - IEEE Conference on Computer Communications, London, United Kingdom, 2022. doi.org/10.1109/INFOCOM48880.2022.9796944

[15] M. Abdel-Basset, H. Hawash, K. M. Sallam, I. Elgendi, and K. Munasinghe, “Digital Twin for Optimization of Slicing-Enabled Communication Networks: A Federated Graph Learning Approach,” IEEE Communications Magazine, vol. 61, no. 10, pp. 100–106, Oct. 2023, doi.org/10.1109/MCOM.003.2200609

[16] P. Almasan, J. Suárez-Varela, K. Rusek, P. Barlet-Ros, and A. Cabellos-Aparicio, 'Deep reinforcement learning meets graph neural networks: Exploring a routing optimization use case', Comput. Commun., vol. 196, pp. 184-194, Dec. 2022. doi.org/10.1016/j.comcom.2022.09.029

[17] F. Tang, Z. M. Fadlullah, N. Kato, F. Ono, and R. Miura, "AC-POCA: Anticoordination game based partially overlapping channels assignment in combined UAV and D2D-based networks," IEEE Trans. Veh. Technol., vol. 67, no. 2, pp. 1672-1683, Feb. 2018, doi.org/10.1109/TVT.2017.2753280

[18] G. F. Riley and T. R. Henderson, ‘The ns-3 network simulator’, in Modeling and Tools for Network, K. Simulation, M. Wehrle, and J. Güneş, Eds Berlin, Germany: Springer, pp. 15–34, 2010. doi.org/10.1007/978-3-642-12331-3_2

[19] H. Zhou et al., 'Informer: Beyond efficient transformer for long sequence time-series forecasting', Proc. Conf. AAAI Artif. Intell., vol. 35, no. 12, pp. 11106-11115, May 2021. doi.org/10.1609/aaai.v35i12.17325

[20] T. Li, A. K. Sahu, A. Talwalkar, and V. Smith, 'Federated learning: Challenges, methods, and future directions', IEEE Signal Process. Mag., vol. 37, no. 3, pp. 50-60, May 2020. doi.org/10.1109/MSP.2020.2975749

[21] A. Vaswani, N. Shazeer, N. Parmar, J. Uszkoreit, L. Jones, A. N. Gomez, Ł. Kaiser, and I. Polosukhin, "Attention is all you need," in Adv. Neural Inf. Process. Syst. (NeurIPS), vol. 30, Long Beach, CA, USA, pp. 5998-6008, 2017. arxiv.org/abs/1706.03762

How to Cite: Dr. Pankaj Jha. (2026). Mathematical modeling and simulation of data-driven systems for efficient network performance optimization. Journal of Electronics, Computer Networking and Applied Mathematics (JECNAM), 6(1), 13-23. https://doi.org/10.55529/jecnam.61.13.23

BIOGRAPHIE OF AUTHOR

Dr. Pankaj Jha, is serving as Head of the Department of Electronics and Communication Engineering at IIMT College of Engineering, Greater Noida, Uttar Pradesh, India. He has made significant contributions to the fields of MIMO antenna design, metamaterial-inspired antennas, compact antenna development for 5G communication systems, textile antennas, and IoT-enabled wireless networks. His research work has been published in reputed international journals and conferences including IEEE and Springer publications. With over 211 citations to his credit, Dr. Jha is an active researcher committed to advancing next-generation wireless communication technologies. He holds membership in several professional bodies and actively mentors postgraduate research scholars. Email: pankaj.maahi@gmail.com