

Urban Traffic Big Data Intelligent Scheduling Model Combining Optimal Transport Theory and Sinkhorn Algorithm

Chengshuang Miao

Hope College, Southwest Jiaotong University, Chengdu 610400, Sichuan, China

Corresponding author: Chengshuang Miao (e-mail: qimakanluori@163.com)

ABSTRACT In urban traffic big data scheduling, this paper proposes an intelligent scheduling model combining Optimal Transport theory and the Sinkhorn algorithm to address insufficient flow optimization, with consideration of traffic resource allocation in connected transportation environments supported by wireless sensing, road-side communication, and electromagnetic information infrastructure. In the specific implementation, starting from the OD matrix, traffic demand is modeled as a probability distribution, and the optimal transport plan for traffic allocation is obtained by solving the Wasserstein distance. Then, entropy regularization is introduced, and the Sinkhorn algorithm is employed to obtain a fast-converging transport solution under large-scale data conditions. Next, a real-time traffic data input mechanism is integrated to dynamically update the transportation cost matrix and recalculate the scheduling plan. Finally, a multilevel scheduling strategy is applied based on global optimization, and different priorities and local constraints are set in combination with regional functions to improve the model's adaptability to complex traffic scenarios. Experimental results show that the scheduling time of the proposed method under high traffic density is $132.5 \text{ ms} \pm 5.6 \text{ ms}$, the CPU utilization rate is $52.9\% \pm 3.1\%$, and the model has high scheduling efficiency. Under the condition of more than 2,100 vehicles/hour, the traffic flow balance is maintained at 0.7, and the congestion index is only 1.32, indicating an obvious traffic optimization effect.

INDEX TERMS urban traffic scheduling, optimal transport, sinkhorn algorithm, flow distribution optimization, wireless sensing

I. INTRODUCTION

JUST as a complex transportation network requires precise coordination to maintain its operational stability, modern urban transportation systems must now address the growing travel demand and unbalanced resource allocation that threaten the stability of urban operations. Driven by the rapid development of big data and artificial intelligence technologies, traffic scheduling is shifting from an experience-driven to a data-driven model.

Mathematical optimization theory and efficient computational methods can achieve accurate modeling and dynamic control of traffic flow distribution in complex road networks [1]. A scheduling framework with dynamic response and system-level balanced distribution capabilities is the key to improving the rationality of traffic flow distribution and system operation efficiency.

Existing scheduling approaches struggle to achieve an optimal equilibrium between global optimization objectives and localized responsiveness, leading to inconsistent performance under dynamic traffic scenarios. This instability

is particularly pronounced under highly uneven traffic flow distributions, where inadequate modeling of spatial resource flows produces suboptimal path allocations, compromising overall system stability [2]. A further limitation is the prevalent reliance on static cost functions during route planning, which inadequately incorporates real-time traffic data. Consequently, scheduling solutions become misaligned with actual traffic fluctuations, diminishing system responsiveness [3]. Additionally, current algorithms lack robust mechanisms for rapid adaptation to emergencies or unexpected traffic disruptions, preventing timely correction of transmission paths and resource distributions. This deficiency risks localized congestion propagation and reduced network-wide traffic efficiency. While prior research has employed probability distributions and distance metrics for traffic modeling, computational complexity and convergence limitations persist in the solution process, hindering real-time applicability in urban road networks [4]. Therefore, enhancing model scalability and robustness without sacrificing scheduling precision emerges as a critical challenge.

This challenge motivates the present work's integration of optimal transport theory with the Sinkhorn algorithm to develop a more efficient intelligent scheduling framework. This study using multi-dimensional flow allocation logic to construct an intelligent traffic dispatch model based on optimal transport theory and the Sinkhorn algorithm, thereby enhancing flow optimization and dispatch efficiency in complex urban traffic systems. This paper models traffic demand as a probability distribution and establishes an optimal transmission (OT) model based on the Wasserstein distance, thereby achieving overall optimal allocation of traffic flows. On this basis, an entropy regularization term is introduced, and the Sinkhorn algorithm is employed for efficient computation, thereby addressing the computational challenges posed by large-scale OD matrices. Unlike previous static scheduling methods, this paper designs a dynamic update mechanism that integrates a real-time traffic data input interface. This allows the transportation cost matrix to be adaptively adjusted based on changing road conditions, thereby ensuring the timeliness and accuracy of the scheduling solution. In addition, considering the differences in functional attributes and traffic pressures in different areas within the city, a multi-level scheduling strategy is adopted, and local constraints are set in combination with regional priorities to enhance the flexibility and adaptability of the model in complex traffic scenarios. The main contributions of this paper are: 1) applying OT theory and the Sinkhorn algorithm system to urban traffic scheduling modeling, expanding its scope of applicability in real-world traffic systems; 2) constructing an intelligent scheduling framework that includes dynamic updates and multi-level control, improving the model's responsiveness to emergencies and non-uniform traffic flows; 3) empirically verifying the superiority of the proposed method in multiple quantitative indicators, providing new theoretical support and practical paths for future intelligent traffic scheduling.

II. RELATED WORK

Numerous studies have applied classical algorithms to traffic path planning and flow optimization. Dijkstra's algorithm reliably finds shortest paths in static networks but ignores global traffic distribution, often causing local overloads [5,6]. Genetic algorithms (GAs) excel in multi-objective optimization yet suffer from slow convergence and parameter sensitivity, limiting real-time use [7,8]. Ant colony optimization (ACO) adapts dynamically but is sensitive to initial pheromone settings and prone to local optima in large networks [9,10].

Simulated annealing (SA) can escape local extrema, but its temperature schedule compromises computational efficiency, failing real-time demands [11,12]. Overall, existing methods struggle with high computational costs, slow response, and poor scalability under large-scale, dynamic traffic conditions.

Crucially, they often inadequately model global traffic flow, hindering a balance between efficiency and fairness—

highlighting the need for a new approach that ensures both computational efficiency and global optimality in modern urban traffic scheduling.

OT theory has been introduced into data flow and trajectory scheduling due to limitations of traditional algorithms in high-dimensional data [13]. Zhang Q proposed a distributed OT algorithm for industrial IoT, reducing average delay by 87% and increasing transmission rate and spectrum efficiency by 157% and 98% respectively [14]. OT describes resource flows from a probability distribution perspective, suitable for abstracting traffic demand into an OD matrix and measuring scheduling cost differences using Wasserstein distance. Studies show OT improves route rationality and system load balancing [15]. Liu J developed a fast Wasserstein-1 distance algorithm, completing calculations in seconds on a single CPU [16]. However, directly solving Wasserstein distance is computationally complex in large urban networks, hindering real-time scheduling. Researchers have used entropy regularization to reduce complexity, developing the efficient Sinkhorn algorithm [17,18], which iteratively adjusts scaling factors to approach the optimal solution with good stability and convergence [19,20]. Nevertheless, existing research is mostly theoretical, with limited application in traffic scheduling. It lacks integration with real traffic data and dynamic updates, limiting model generalization. Most methods focus on system-level balanced distribution, ignoring regional functional differences. Therefore, this paper designs a multi-level dispatch model with dynamic updates and regional priority constraints to enhance adaptability and optimization in complex traffic environments.

III. CONSTRUCTION OF TRAFFIC FLOW SCHEDULING MODEL

A. CONSTRUCTING TRAFFIC FLOW DISTRIBUTION MODEL BASED ON OPTIMAL TRANSPORT THEORY

Starting from the urban traffic OD matrix, traffic demand in each region is modeled as a probability distribution, with a target distribution defined for the desired state. By constructing a transportation cost matrix and computing the Wasserstein distance, the optimal transport plan between source and target distributions is obtained. In this model, "transportation" in this model is not physical displacement, but an abstract representation of path allocation for OD traffic under a given road network topology. The transport cost matrix is constructed from the shortest path distance (or travel time) of the actual road network, implicitly including road connectivity and capacity constraints; therefore, all transfers of "probabilistic quality" are constrained by the real road network structure, rather than unconstrained free flow. As shown in Figure 1, the intelligent scheduling model decomposes the OD matrix into source and target distributions—representing current origins and ideal traffic demand—then minimizes flow transmission costs via OT theory to yield a globally optimal allocation. A dynamic adjustment mechanism updates the cost matrix with real-

time traffic data, enabling the plan to adapt to congestion or road changes. The resulting optimized flow distribution balances network load and alleviates localized bottlenecks.

1) Traffic Demand Modeling and Target Distribution

Definition

This study models urban traffic demand using an origin-destination (OD) matrix, where each element represents flow from a source to a destination region. To enable optimization, traffic demand is converted into a probability density function: for a given time interval, the demand from source region T_i to destination region T_j defines this distribution, effectively capturing the time-varying nature of urban traffic when integrated with real-world flow data. A target distribution is also defined to represent the ideal traffic flow state—ideally minimizing congestion and ensuring balanced utilization across the network. This target is derived from network capacity, road constraints, regional functions, and either historical data analysis or dynamic road condition simulations.

Establishing this desired distribution is essential for guiding optimal path planning and mitigating traffic bottlenecks. The target distribution in this study is not arbitrarily set, but rather generated through capacity-weighted normalization based on the current road network topology, the real-time capacity of each road segment, and historical OD patterns. Specifically, its construction process ensures that the total demand does not exceed the maximum carrying capacity of the road network and is consistent with existing path connectivity, thereby guaranteeing that the distribution can be achieved through a set of legal paths.

Specifically, the target distribution ν_j is defined as a normalized vector proportional to the road network capacity:

$$\nu_j = \frac{c_j}{\sum_k c_k} \quad (1)$$

where c_j is the real-time traffic capacity of region j (dynamically estimated by the number of lanes, signal timing, and historical saturation), ensuring that the ideal state reflects the current network capacity limit rather than a subjective setting.

2) Wasserstein Distance and OT Scheme

After determining the traffic demand distribution in the source and target areas, the next task is to find the most appropriate traffic allocation solution using OT theory. To achieve this goal, the Wasserstein distance is used to measure the difference between the source and target distributions. The calculation of the Wasserstein distance essentially seeks a transmission method that minimizes both traffic flow and transmission path costs. Specifically, the Wasserstein distance is expressed as follows:

$$W(p, q) = \inf_{\gamma \in \Gamma(p, q)} \int_{X \times Y} c(x, y) \gamma(dx, dy) \quad (2)$$

p and q represent the traffic demand distributions of the source and target regions, respectively. $\gamma(dx, dy)$ is the set of all possible joint distributions. $c(x, y)$ is the transport cost function, representing the cost required to transport the demand from source region x to target region y . The minimization objective is to find the optimal transport path that minimizes the integral of the cost function. Notably, although this model uses a probability distribution to represent macroscopic traffic flow, its output OT scheme only serves as a guide for path allocation ratios. In actual scheduling, physical constraints such as lane capacity and signal timing are still strictly followed. In specific implementation, this ratio will be passed as input to downstream microscopic simulation modules, which will then complete vehicle-level allocation under the premise of satisfying vehicle conservation and discrete event rules.

B. APPLYING THE SINKHORN ALGORITHM TO ACHIEVE APPROXIMATE OPTIMAL SOLUTION CALCULATION FOR LARGE-SCALE DATA

1) Application of Entropy Regularization and Construction of Optimization Problem

When applying the optimal transport problem to large-scale data, the solution cost is often very high in terms of time and computational resources. To alleviate this problem, an entropy regularization strategy is used to reduce the computational cost while maintaining the numerical stability of the optimal transport problem. By introducing the entropy regularization term into the optimal transport problem, a constrained linear programming problem is obtained. The regularization term is expressed as follows:

$$\lambda H(\gamma) = \lambda \sum_{i,j} \gamma_{ij} \ln(\gamma_{ij}) \quad (3)$$

γ_{ij} represents the amount of transmission between the source and target regions; $H(\gamma)$ is the entropy of the transmission matrix; and λ is the entropy regularization coefficient. Entropy regularization smooths the transport matrix by dampening extreme source–destination transfers, distributing flow more evenly and avoiding numerical instability and slow convergence in the original optimal transport problem. It transforms the problem into a tractable, entropy-regularized form solvable via the Sinkhorn algorithm, minimizing Wasserstein distance while maximizing matrix smoothness. This not only stabilizes optimization but also drastically reduces computation time without sacrificing solution quality for large-scale data. For each regularization parameter λ , the algorithm iterates until convergence or a maximum iteration limit, recording per-iteration error, Wasserstein distance, and execution time to assess convergence behavior and computational cost.

2) Iterative Solution Process of Sinkhorn Algorithm

Applying entropy regularization transforms the optimal transport problem into a more computationally tractable

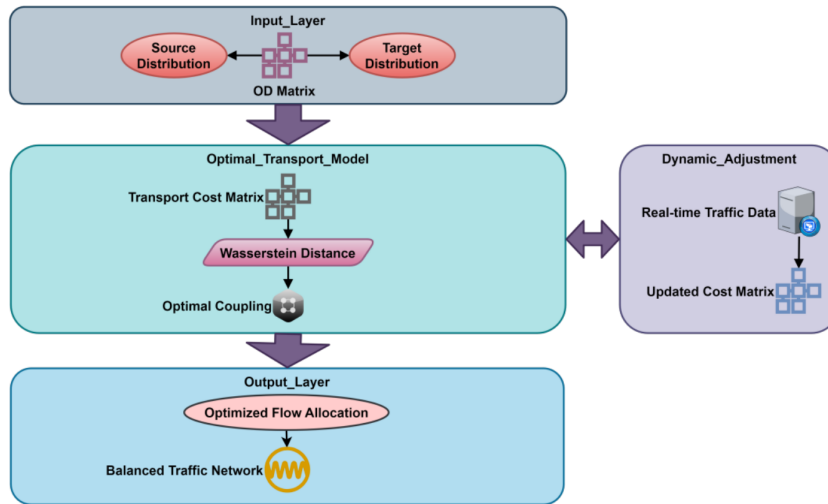


FIGURE 1. Traffic flow distribution

form. To solve for the resulting transport matrix, the Sinkhorn algorithm is employed, generating a solution that approximates the true optimum. This algorithm achieves successive approximation by iteratively updating the matrix's row and column scaling factors. Specifically, each iteration alternates between a row normalization step and a column normalization step. The transport matrix computed by the Sinkhorn algorithm is defined by the following formula:

$$\gamma_{ij} = \frac{p_i u_i c_{ij} v_j}{q_j} \quad (4)$$

p_i and q_j are the demand distributions of the source and target regions, respectively. u_i and v_j are the row and column normalization factors, respectively. c_{ij} is the transmission cost between the source and target regions. In each iteration, the update process is as follows. First, updating the transmission amount of each row by updating the row normalization factor of current matrix. Next, updating the transmission amount of each column by updating the column normalization factor. Repeating the above update process until the row and column normalization error of transfer matrix is small enough. In Table 1, λ is the entropy regularization coefficient that smooths the transport plan and accelerates convergence. ϵ is the accuracy threshold: the algorithm stops when the absolute difference between successive iterations falls below ϵ . The maximum iteration count prevents excessive computation in slow-convergence cases. Row and column normalization factors—both in $[0, 1]$ —scale the transfer matrix to match the source and target distributions, respectively. The convergence error sets the allowable discrepancy between row and column marginals; the algorithm terminates when this error drops below the threshold.

C. INTEGRATING REAL-TIME DATA INPUT AND DYNAMIC SCHEDULING MECHANISM

1) Real-Time Data Stream Access and Road Section Status Extraction

The dynamic nature of urban traffic demands dispatching models that rapidly adapt to changing conditions.

Real-time data—such as road congestion, traffic flow, vehicle speed, and travel demand—collected via sensors, cameras, and GPS devices, enable accurate and responsive scheduling. These multi-source data not only reveal current traffic volumes but also detect incidents, signal states, and emergency impacts. To derive road congestion levels, the system first fuses and preprocesses raw data to remove noise and ensure reliability. Traffic status is then converted into dynamic weights reflecting road segment load: high density or low speed increases transmission cost (indicating congestion), while sparse traffic lowers it. By continuously updating these costs, the model responds promptly to traffic fluctuations, overcoming the rigidity of static scheduling methods that fail in emergencies.

2) Dynamic Scheduling and Optimized Path Recalculation

When road conditions change, the transport cost matrix must be updated in real time. Upon receiving new road status data from the monitoring system, the matrix is immediately refreshed by recalculating each element based on current congestion levels, ensuring costs accurately reflect real-time traffic. For instance, congestion or accidents sharply increase the cost of affected segments, encoding traffic burden directly into the cost structure and influencing downstream scheduling decisions.

Following this update, a new dispatch plan must be rapidly generated by feeding the revised cost matrix into the optimization model. The Sinkhorn algorithm's iterative update mechanism is key to efficiently computing the new optimal transport plan.

$$\gamma_{ij}(t) = \frac{p_i u_i(t) c_{ij}(t) v_j(t)}{q_j} \quad (5)$$

TABLE 1. Sinkhorn algorithm iteration parameter configuration

Parameter	Description	Default Value	Range
λ	Entropy regularization coefficient, controls smoothness and convergence rate	0.1	[0, 10]
ϵ	Precision threshold for iteration stop criteria	1×10^{-6}	$[1 \times 10^{-7}, 1 \times 10^{-5}]$
Max Iterations	Maximum number of iterations for Sinkhorn algorithm	500	[100, 1000]
u_i	Scaling factor applied to each row of the transportation matrix	0.5	(0, 1)
v_j	Scaling factor applied to each column of the transportation matrix	0.5	(0, 1)
Convergence Error	Maximum allowable error for row and column normalization, controlling solution accuracy	1×10^{-5}	$[1 \times 10^{-6}, 1 \times 10^{-4}]$

$u_i(t)$ and $v_j(t)$ are factors that normalize rows and columns based on the latest traffic conditions, and $c_{ij}(t)$ is the updated transmission cost. Each iteration uses the updated cost matrix to adjust the traffic flow distribution, allowing the OT plan to quickly adapt to the new traffic environment. Ultimately, through multiple rounds of row and column normalization, the Sinkhorn algorithm can quickly approach the new optimal solution, achieving dynamic scheduling and real-time optimization.

When traffic conditions on a road section change, the dispatch system can adjust promptly, ensuring not only the timeliness of the dispatch plan but also a balance between traffic flow and system stability. This dynamic dispatch approach gives the dispatch system great flexibility and adaptability in complex urban traffic environments, enabling it to cope with various traffic conditions and improving dispatch efficiency and reliability.

D. BUILD A MULTI-LEVEL DISPATCHING STRATEGY TO DEAL WITH COMPLEX TRAFFIC SCENARIOS

The city is partitioned into high-, medium-, and low-priority functional zones, each assigned dispatch priorities based on traffic demand and flow characteristics. As shown in Figure 2, the multi-level dispatch framework achieves system-level balanced distribution by applying local constraints per zone. A dynamic priority adjustment module uses real-time traffic data to elevate priorities in high-demand areas during peaks or special events. Continuous data updates refine the transportation cost matrix and dispatch strategies, enabling precise route planning and ensuring efficient, stable urban traffic operations.

1) Urban Functional Area Division and Priority Setting

Urban traffic mobility varies significantly across functional areas, necessitating region-specific scheduling priorities. Areas like transportation hubs, commercial districts, and hospitals—characterized by high traffic density, limited road capacity, or critical socioeconomic functions—are assigned higher priority to address congestion proactively. In contrast, low-demand residential zones or off-peak road segments receive lower priority, enabling efficient resource allocation. In this model, regional priorities—determined by functional attributes and real-time traffic conditions—are integrated

into a global optimization framework. Higher-priority regions are scheduled first, with dynamic resource reallocation (e.g., increased capacity during peak hub demand) preventing localized congestion. This approach ensures fairness across regions while enhancing overall system efficiency and avoiding global performance degradation.

2) Application of Local Constraints and Hierarchical Optimization

Local constraints are often introduced as additional constraints to the system-level balanced distribution problem to ensure that the global optimization problem balances global efficiency with specific local needs.

When designing constraints, the traffic flow restrictions, capacity, and priority of each area should be considered. For example, a formal constraint to limit the maximum capacity of an area can be introduced as shown below:

$$\sum_{i,j} \gamma_{ij} \leq C_{\max} \quad (6)$$

$C_{\max}$ represents the maximum capacity of the road segment. The scheduling plan must not exceed load capacity, as traffic exceeding the allowable limit is prohibited from entering the road segment. This can avoid the traffic jam. At the same time, the system must also consider these local constraints when performing the global optimization and adjust the flow distribution plan online to make sure that the system can meet the local needs and enhance the overall traffic flow.

IV. SCHEDULING PERFORMANCE EVALUATION

A. EXPERIMENTAL DATA AND SETTINGS

The experimental data are derived from a real-world OD matrix and real-time monitoring data from a city's traffic management center. It covers traffic flow during peak hours and includes vehicle travel records for approximately 5,000 major road sections. "5,000 road segments" refers to directed links in the main road network, corresponding to approximately 2,500 intersections, which is consistent with the modeling conventions for core areas of medium-sized cities. During data preprocessing, the original OD matrix is normalized, and outliers and missing data are removed to ensure input quality. The experimental environment is

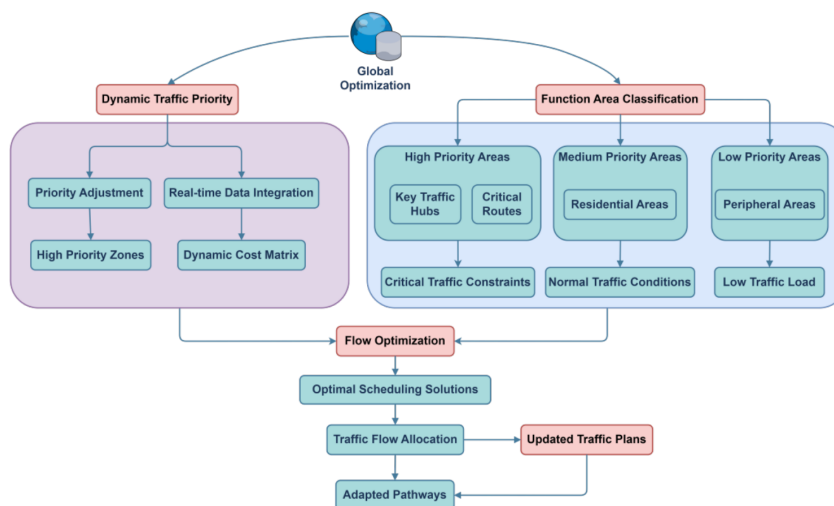


FIGURE 2. Multi-level traffic scheduling framework

based on an Intel Xeon Gold 6248R processor and 128 GB of memory, utilizing multi-threaded parallel computing for improved efficiency. All algorithms run on the same hardware platform without GPU acceleration or dedicated optimization libraries enabled. Therefore, performance differences mainly reflect the computational characteristics of the algorithms themselves. To simulate different traffic density scenarios, the study divides traffic density into three groups: low ($< 1,000$ vehicles/km), medium ($1,000\text{--}3,000$ vehicles/km), and high ($> 3,000$ vehicles/km), “High density ($> 3,000$ vehicles/km)” refers to the average cross-sectional traffic density of the road network (i.e., total traffic flow divided by total length), not the density per lane. As well as different traffic flow ranges (< 300 , $300\text{--}600$, $600\text{--}900$, $900\text{--}1,200$, $1,200\text{--}1,500$, $1,500\text{--}1,800$, $1,800\text{--}2,100$, and $> 2,100$ vehicles/hour). The corresponding test dataset is generated by adjusting the scaling factor of the OD matrix. The initial value of the transportation cost matrix is calculated based on historical road network travel times and is dynamically updated to reflect real-time congestion conditions. In the comparative experiment, Dijkstra, GA, ACO, and SA all use the same data input and hardware configuration to ensure fairness. It should be clarified that this paper uses algorithms such as Dijkstra as benchmarks to evaluate the impact of each method on global traffic distribution under the same input (OD matrix) and output (path allocation scheme) formats, rather than equating their original design goals with global optimization. Dijkstra represents a typical local shortest path strategy, often used as the default routing rule in large-scale scheduling; by comparing it with Dijkstra, the practical value of introducing global coordination mechanisms (such as optimal transmission) in alleviating local congestion can be highlighted.

B. ANALYSIS OF OT COST CHANGES UNDER TRAFFIC DEMAND

The source and destination demands are normalized and scaled to the range $[0, 1]$. Next, a transportation cost matrix is constructed to calculate the difference between the source and destination regions. Combining the demand differences with the cost model, the Wasserstein distance is used to measure the optimal transport solution. Under different demand configurations, the optimal transport cost is calculated for each scenario, and a three-dimensional surface plot is plotted to illustrate the impact of demand changes on costs.

Figure 3 illustrates the impact of changes in traffic demand in the source and target regions on the optimal transport cost. The X-axis and Y-axis represent the traffic demand in the source and target regions, respectively, both normalized to the range $[0, 1]$, while the Z-axis represents the corresponding normalized optimal transport cost. By calculating the source-target demand difference and the transport cost matrix, the data demonstrates the transport costs under different demand configurations. As demand in the source and target regions increases, especially when the demand difference is significant, the optimal transport cost increases significantly, approaching 1. This phenomenon arises from the fact that when there is a significant disparity between the demand in the source and target areas, the system must mobilize more resources to distribute traffic, increasing transmission costs. This imbalance not only complicates resource scheduling but also necessitates more efficient traffic flow to reduce system burden. This analysis demonstrates that demand imbalance is a key factor affecting traffic scheduling costs, further emphasizing the importance of demand balancing and efficient resource allocation in urban traffic scheduling.

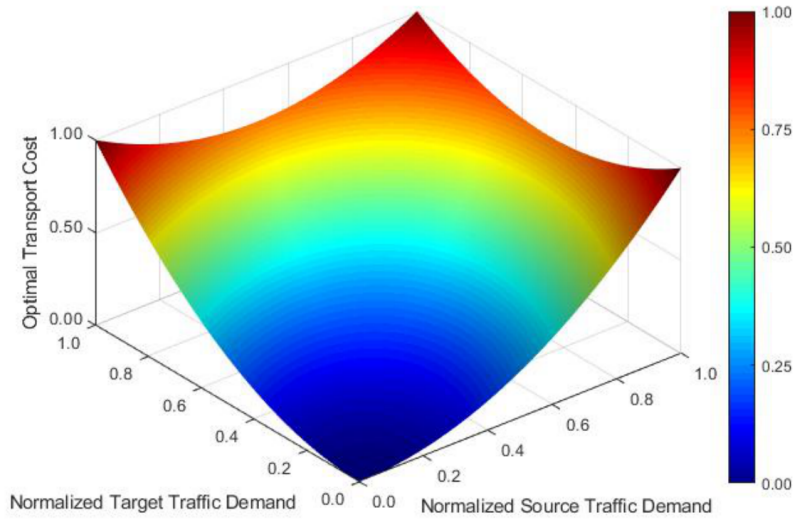


FIGURE 3. Changes in OT cost under traffic demand

C. SCHEDULING EFFICIENCY EVALUATION

To validate the computational efficiency advantages of the proposed Sinkhorn algorithm in dynamic traffic scenarios, this study designs comparative experiments under various density conditions, selecting the Dijkstra algorithm, GA, ACO, and SA as benchmark methods. The evaluation process is strictly divided into three scenarios: low-density (< 1,000 vehicles/km), medium-density (1,000–3,000 vehicles/km), and high-density (> 3,000 vehicles/km). Real-world traffic loads are simulated by manipulating road network size and traffic flow distribution. Quantitative metrics focused on dispatch time (the total time from input data to solution generation) and resource usage (measured by peak memory and CPU usage).

Table 2 compares the scheduling efficiency of the Sinkhorn algorithm with traditional path planning algorithms (Dijkstra, GA, ACO, and SA) in different traffic density scenarios through systematic experiments.

The experimental data clearly demonstrates significant differences in computational speed and resource consumption among the algorithms. The Sinkhorn algorithm demonstrates excellent computational efficiency under low, medium, and high density conditions, achieving a time of $132.5 \text{ ms} \pm 5.6 \text{ ms}$ in high density. Its scheduling time is shorter than that of the Dijkstra algorithm, and it achieves significant performance improvements over the GA, ACO, and SA algorithms. In terms of resource usage, Sinkhorn's CPU utilization remains consistently low, reaching $52.9\% \pm 3.1\%$ at high density, significantly lower than the comparison algorithms. Regarding memory management, Sinkhorn's peak memory usage is lower than that of traditional methods. This advantage becomes more pronounced as traffic density increases, validating the effective control of computational complexity achieved by its entropy regularization mechanism. Notably, traditional algorithms exhibit high

resource consumption in various scenarios. All data are obtained in a strictly controlled experimental environment, and the standard deviation range demonstrates Sinkhorn's more stable performance, providing critical support for its practical application in urban-scale transportation systems.

D. TRAFFIC OPTIMIZATION EFFECT COMPARISON

To evaluate the performance of different algorithms for traffic flow optimization, this study selects several different traffic flow ranges (< 300, 300–600, 600–900, 900–1,200, 1,200–1,500, 1,500–1,800, 1,800–2,100, and > 2,100 vehicles/hour) and compares the optimization results of the Sinkhorn algorithm with those of other classic algorithms. The evaluation primarily considers two quantitative indicators: traffic flow balance and congestion index. Traffic flow balance is measured by calculating the variance of traffic flow across each road segment. The formula is:

$$J = \frac{\sum_{i=1}^n (f_i - \bar{f})^2}{n} \quad (7)$$

f_i is the traffic flow of the i -th road section, $\bar{f}$ is the average flow, and n is the number of road sections. The congestion index is measured by the ratio of the flow of the most congested road section in the road network to the average flow of the entire network. The formula is:

$$Y = \frac{\max(f_1, f_2, \dots, f_n)}{\bar{f}} \quad (8)$$

$(f_1, f_2, \dots, f_n)$ represents the traffic flow of each road section. Figure 4 shows the performance of five optimization algorithms in terms of traffic flow balance and congestion index under different traffic flow conditions. The horizontal axis represents the traffic flow category, ranging from low flow (< 300 vehicles/hour) to high flow (> 2,100 vehicles/hour), while the vertical axis corresponds to the traffic

TABLE 2. Comparison of scheduling efficiency of multiple algorithms

Algorithm	Traffic Density	Scheduling Time (ms)	CPU Usage (%)	Peak Memory (MB)
Sinkhorn	Low	42.3±2.1	18.5±1.2	156±8
	Medium	87.6±3.8	34.7±2.4	298±12
	High	132.5±5.6	52.9±3.1	412±19
Dijkstra	Low	68.9±3.5	25.3±1.8	203±11
	Medium	142.7±6.2	48.6±3.3	387±16
	High	278.4±9.7	76.2±4.5	598±24
GA	Low	215.4±8.3	62.4±4.2	354±15
	Medium	498.1±14.6	85.7±5.9	621±28
	High	864.3±21.9	94.1±6.7	892±37
ACO	Low	183.2±7.1	58.7±3.8	312±14
	Medium	421.5±12.4	79.3±5.1	587±26
	High	702.8±18.5	91.5±6.3	843±34
SA	Low	198.6±7.9	60.2±4.0	337±13
	Medium	467.3±13.8	83.6±5.6	605±27
	High	809.5±20.2	93.8±6.5	876±39

flow balance and congestion index, respectively. As traffic flow increases, the balance of each algorithm generally decreases. The Sinkhorn algorithm performs relatively stable across the entire traffic range, maintaining a high balance, reaching 0.7 when the flow rate is > 2,100 vehicles/hour. This can be attributed to entropy regularization, which enforces smoother and more uniform traffic distributions. In contrast, the Dijkstra and SA algorithms experience a significant drop in balance during high-traffic intervals. This is because these algorithms lack a global perspective when handling large-scale traffic, which can easily lead to localized congestion. The genetic algorithm and ant colony algorithm (ACO) demonstrate some adaptability, but still struggle under extreme traffic conditions. In a comparison of congestion indices, the congestion index of each algorithm increases with increasing traffic. The Sinkhorn algorithm maintains a low congestion index, demonstrating its effectiveness in alleviating traffic congestion in high-volume traffic conditions. The index is only 1.32 when traffic volumes exceeded 2,100 vehicles per hour. The Dijkstra algorithm exhibits a higher congestion index in high-volume traffic conditions, which is related to its overreliance on local traffic conditions during optimization, potentially leading to lower efficiency in managing large-scale traffic flows. The ACO algorithm performs well, showing a slight advantage over other algorithms in high-volume traffic conditions. It possesses strong adaptability and distributed optimization properties, but it also has shortcomings compared to the Sinkhorn algorithm.

E. SYSTEM TOTAL TRAVEL TIME AND RESOURCE UTILIZATION ANALYSIS

This study constructs a dynamic road network model based on the high-fidelity urban traffic simulation platform (VIS-SIM) to simulate vehicle behavior under real-world road conditions. The experiment uses a fixed two-hour time window, injects scheduling solutions generated by different

algorithms, and calculates the system's total travel time (the cumulative time for all vehicles traveling from their starting point to their destination) and road resource utilization (segment saturation). Data collection covers both network-level resource utilization indicators (average speed, percentage of congested sections) and microscopic vehicle trajectories. Millisecond-level data synchronization is achieved using the TraCI aggregation tool. Congested sections are defined as those with a saturation greater than 85% and a speed less than 20 km/h. To eliminate random interference, each experiment is repeated five times to calculate the mean and standard deviation.

Table 3 compares the comprehensive performance of five traffic scheduling algorithms in dynamic urban road networks, revealing the fundamental differences between these methods in terms of system efficiency and resource allocation. The Sinkhorn algorithm, leveraging its global optimization capabilities based on optimal transport theory, significantly reduces total travel time and road saturation, achieving mean values of 1,248 hours and 68.5%, respectively. This is due to its balanced spatial distribution of traffic flow, which avoids the resource contention issues often associated with local optima in traditional path planning algorithms. The high congestion rate and low average speed of the Dijkstra algorithm reflect the limitations of its static shortest path strategy—it cannot adapt to dynamic traffic demand, resulting in overload on key sections of road. Although ACO, GA, and SA alleviate some of these problems through swarm intelligence or random search, their inefficient exploration of the solution space still leads to suboptimal scheduling. Sinkhorn's entropy regularization mechanism further optimizes computational efficiency, enabling real-time response while maintaining high accuracy. Other algorithms, however, struggle to balance speed and quality due to computational complexity and convergence stability. This comparison highlights the necessity of a scheduling framework that integrates mathematical opti-

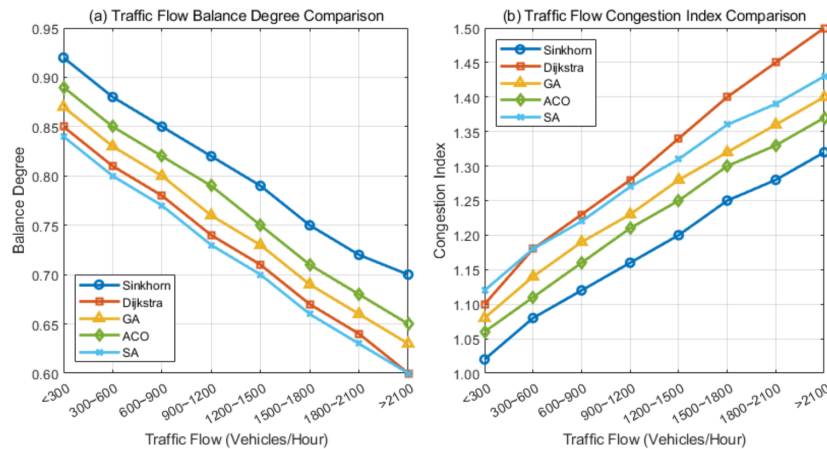


FIGURE 4. Traffic flow balance and congestion index

mization and dynamic adjustment in complex transportation systems.

F. SYSTEM STABILITY ASSESSMENT

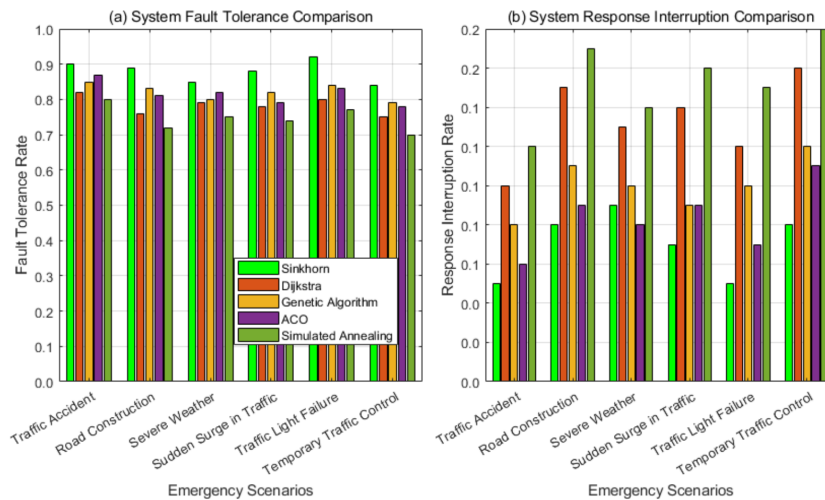
To comprehensively evaluate the proposed dispatch system's robustness in handling emergency scenarios, the study compares the Sinkhorn algorithm with four other algorithms (Dijkstra's algorithm, genetic algorithm, ACO, and SA) in response to various complex traffic conditions. Evaluation metrics include system fault tolerance and response interruption rate. The system fault tolerance measures the system's ability to maintain effective dispatch after an emergency, while the response interruption rate indicates the frequency of system interruptions during emergency response. The system's fault tolerance and response interruption rate are quantified for each emergency scenario. These evaluation metrics compare the strengths and weaknesses of different algorithms in handling emergencies and reveal their adaptability to dynamic traffic environments.

Figure 5 compares the system stability of five scheduling algorithms under different traffic emergency scenarios. Each algorithm performs differently in different scenarios, reflecting its adaptability to complex and dynamic traffic environments. In terms of fault tolerance, the Sinkhorn algorithm demonstrates strong system stability in most emergencies, maintaining a fault tolerance of 0.84 or above. This is particularly true in the case of traffic light failures, where the fault tolerance is high, demonstrating its superior optimization capabilities and robustness to system failures. This is related to the smoothness and global optimality of its transmission matrix calculation, which allows it to better adapt to sudden traffic changes and road bottlenecks. Traditional algorithms have low fault tolerance in these complex scenarios. Regarding response interruption rates, traditional algorithms have high interruption rates in different scenarios, indicating that recovery is slow in these special cases and they are unable to quickly respond to dynamic changes. The Sinkhorn algorithm, on the other hand, performs better under variable traffic patterns, achieving a low interruption

rate of less than 0.1. This demonstrates its resistance to local optima in complex environments, improving system recovery efficiency. These results demonstrate the strengths and weaknesses of different algorithms in responding to emergencies and reveal the limitations and potential of various scheduling strategies in practical applications.

TABLE 3. Analysis of total travel time and resource use

Algorithm	Total Travel Time (hours)	Road Saturation (%)	Congested Road Ratio (%)	Average Speed (km/h)
Sinkhorn	1,248±32	68.5±2.3	12.7±1.5	45.2±1.8
Dijkstra	1,572±41	82.4±3.1	24.6±2.2	38.7±2.1
ACO	1,463±37	75.8±2.8	19.3±1.9	41.5±1.9
GA	1,521±39	78.6±3.0	21.8±2.1	39.6±2.0
SA	1,487±38	76.2±2.9	20.1±2.0	40.8±1.9

**FIGURE 5.** System stability under different emergency traffic scenarios

V. CONCLUSION

Drawing upon multi-dimensional flow coordination principles, this paper proposes an intelligent urban traffic scheduling model that integrates optimal transport theory with the Sinkhorn algorithm to ensure seamless flow distribution. By representing traffic demand as a probability distribution and minimizing Wasserstein distance, the method globally optimizes traffic flow allocation. Entropy regularization and the Sinkhorn algorithm enable efficient, accurate computation on large-scale, high-dimensional data, overcoming traditional methods' scalability limits. A dynamic update mechanism adapts to real-time road condition changes, while a multi-level scheduling strategy enhances responsiveness to regional characteristics and emergencies. Experiments show that under high traffic density ($> 2,100$ vehicles/hour), the system achieves a dispatch time of $132.5 \text{ ms} \pm 5.6 \text{ ms}$, CPU utilization of $52.9\% \pm 3.1\%$, maintains a traffic flow balance of 0.7, and keeps the congestion index low at 1.32. Compared to classical algorithms, the approach significantly improves dispatching efficiency, flow balance, and system stability. By adopting a parallel coordination mechanism for large-scale flow allocation, this approach resolves the computational bottleneck of optimal transport in intelligent transportation and offers scalable support for city-wide real-time optimization and coordinated control.

AUTHOR CONTRIBUTIONS

Chengshuang Miao designed, collected and analyzed the data, and drafted the manuscript. Chengshuang Miao conducted the study, critically revised the manuscript for important intellectual content, and gave final approval of the version to be published. Chengshuang Miao participated fully in the work, take public responsibility for appropriate portions of the content, and agreed to be accountable for all aspects of the work in ensuring that questions related to the accuracy or integrity of any part of the work are appropriately investigated and resolved.

CONFLICTS OF INTEREST

The author declares no conflict of interest.

FUNDING

This research received no external funding.

ACKNOWLEDGEMENTS

Not applicable.

REFERENCES

- [1] M. Kii, R. Isikawa, and Y. Kometani, "Toward a carbon neutral urban transportation system in Japan," *IATSS Research*, vol. 47, no. 2, pp. 171-178, 2023, doi: 10.1016/j.iatssr.2023.01.001.
- [2] J. Zhu, N. Xie, Z. Cai, W. Tang, and X. Chen, "A comprehensive review of shared mobility for sustainable transportation systems," *International Journal of Sustainable Transportation*, vol. 17, no. 5, pp. 527-551, 2023, doi: 10.1080/15568318.2022.2054390.

- [3] S. Lv, S. Chen, Z. Wei, and H. Zhang, "Power-transportation coordination: Toward a hybrid economic-emission dispatch model," *IEEE Transactions on Power Systems*, vol. 37, no. 5, pp. 3969-3981, 2021, doi: 10.1109/TPWRS.2021.3131306.
- [4] C. Chen, Y. Liu, L. Chen, and C. Zhang, "Bidirectional spatial-temporal adaptive transformer for urban traffic flow forecasting," *IEEE Transactions on Neural Networks and Learning Systems*, vol. 34, no. 10, pp. 6913-6925, 2022, doi: 10.1109/tnnls.2022.3183903.
- [5] B. Li, J. Hou, X. Wang, Y. Ma, D. Li, T. Wang, et al., "High-resolution flood numerical model and dijkstra algorithm based risk avoidance routes planning," *Water Resources Management*, vol. 37, no. 8, pp. 3243-3258, 2023, doi: 10.1007/s11269-023-03500-5.
- [6] L. S. Jabbar, E. I. Abass, and S. D. Hasan, "A modification of shortest path algorithm according to adjustable weights based on Dijkstra algorithm," *Engineering and Technology Journal*, vol. 41, no. 2, pp. 359-374, 2023, [Online]. Available: <https://etj.uotechnology.edu.iq>.
- [7] T. Mao, A. S. Mihăită, F. Chen, and H. L. Vu, "Boosted genetic algorithm using machine learning for traffic control optimization," *IEEE Transactions on Intelligent Transportation Systems*, vol. 23, no. 7, pp. 7112-7141, 2021, doi: 10.1109/ITITS.2021.3066958.
- [8] J. Mi, Y. Zhao, H. Zhang, P. Zhang, W. Cheng, H. Wang, et al., "On dynamic path planning based on the DBSCAN-AGA algorithm," *International Journal of Vehicle Systems Modelling and Testing*, vol. 19, no. 2, pp. 128-151, 2025, doi: 10.1504/IJVSMT.2025.147352.
- [9] B. E. Akilo, S. A. Oyedotun, G. P. Oise, O. C. Nwabuko, and N. B. Unuigbokhai, "Intelligent traffic management system using ant colony and deep learning algorithms for real-time traffic flow optimization," *Journal of Science Research and Reviews*, vol. 1, no. 2, pp. 63-71, 2024, doi: 10.70882/josrar.2024.v1i2.52.
- [10] S. Liao, Y. Wu, K. Ma, and Y. Niu, "Ant colony optimization with look-ahead mechanism for dynamic traffic signal control of IoV systems," *IEEE Internet of Things Journal*, vol. 11, no. 1, pp. 366-377, 2023, doi: 10.1109/IJOT.2023.3286799.
- [11] H. Wang, P. Zhang, and W. Wang, "Research on robot path planning based on simulated annealing algorithm," *Journal of Artificial Intelligence Practice*, vol. 6, no. 7, pp. 29-36, 2023, doi: 10.23977/jaip.2023.060705.
- [12] W. Shi, Z. He, W. Tang, W. Liu, and Z. Ma, "Path planning of multi-robot systems with boolean specifications based on simulated annealing," *IEEE Robotics and Automation Letters*, vol. 7, no. 3, pp. 6091-6098, 2022, doi: 10.1109/LRA.2022.3165184.
- [13] D. Wang, J. Tian, H. Zhang, and D. Wu, "Task offloading and trajectory scheduling for UAV-enabled MEC networks: An optimal transport theory perspective," *IEEE Wireless Communications Letters*, vol. 11, no. 1, pp. 150-154, 2021, doi: 10.1109/LWC.2021.3122957.
- [14] Q. Zhang, Y. Jiang, X. Ge, Y. Huang, and Y. Liu, "Distributed data flow scheduling optimization in industrial internet of things based on optimal transport theory," *IEEE Internet of Things Journal*, vol. 10, no. 14, pp. 12961-12974, 2023, doi: 10.1109/IJOT.2023.3256357.
- [15] W. Lee, W. Li, B. Lin, and A. Monod, "Tropical optimal transport and Wasserstein distances," *Information Geometry*, vol. 5, no. 1, pp. 247-287, 2022, doi: 10.1007/s41884-021-00046-6.
- [16] J. Liu, W. Yin, W. Li, and Y. T. Chow, "Multilevel optimal transport: a fast approximation of Wasserstein-1 distances," *SIAM Journal on Scientific Computing*, vol. 43, no. 1, pp. A193-A220, 2021, doi: 10.1137/18M1219813.
- [17] S. Eckstein and M. Nutz, "Quantitative stability of regularized optimal transport and convergence of sinkhorn's algorithm," *SIAM Journal on Mathematical Analysis*, vol. 54, no. 6, pp. 5922-5948, 2022, doi: 10.1137/21M145505X.
- [18] Z. Goldfeld, K. Kato, G. Rioux, and R. Sadhu, "Limit theorems for entropic optimal transport maps and Sinkhorn divergence," *Electronic Journal of Statistics*, vol. 18, no. 1, pp. 980-1041, 2024, doi: 10.1214/24-EJS2217.
- [19] R. J. Berman, "The Sinkhorn algorithm, parabolic optimal transport and geometric Monge-Ampère equations," *Numerische Mathematik*, vol. 145, no. 4, pp. 771-836, 2020, doi: 10.1007/s00211-020-01127-x.
- [20] M. Khamlich, F. Pichi, and G. Rozza, "Optimal Transport-Inspired Deep Learning Framework for Slow-Decaying Kolmogorov-Width Problems: Exploiting Sinkhorn Loss and Wasserstein Kernel," *SIAM Journal on Scientific Computing*, vol. 47, no. 2, pp. C235-C264, 2025, doi: 10.1137/23M1604680.