Journal of Intelligence and Engineering Technology
Vol. 1, No. 1, 2026

Dynamic Task Prioritization for Edge AI in Smart Cities: Balancing Latency and Energy Efficiency

HAO, Zihe [1*]

[1] Northeastern University, US

** HAO, Zihe is the corresponding author, E-mail: zhihehao123@gmail.com*

Abstract: The growth of latency sensitive smart city applications is rapid nowadays. Therefore deploying microservice architecture in the heterogeneous edge cloud continuum has become a mainstream choice. However a structural challenge arises in orchestrating these coupled services modeled as Directed Acyclic Graphs. Exact algorithms like Branch and Bound struggle with computation in high concurrency scenarios. Meanwhile Deep Reinforcement Learning methods face the challenges of excessive training overhead and a lack of zero shot adaptation capability for topology changes. Recently popular quantum inspired algorithms often fail to satisfy strict predecessor constraints. This makes them unsuitable for use in dependent workflows. To alleviate this dilemma this paper proposes a Dependency Aware Quantum Inspired Scheduler. This framework utilizes a topological quantum coding scheme and dynamic dependency masks to integrate DAG constraints into the quantum search process. It also introduces an entropy weighted evolutionary rotation mechanism to accelerate the convergence of critical paths. After conducting extensive simulation experiments in city level environments we found that the scheduling success rate of DAQ Scheduler is 100% while the success rate of standard quantum inspired algorithms is only 58.1%. Compared with leading multi objective DRL baselines this method reduces the average makespan by 9.9%. This method provides near optimal scheduling solutions with millisecond level inference latency. It builds an efficient and lightweight paradigm for real time edge intelligence and balances theoretical optimality with engineering feasibility.

Keywords: Edge Computing, Microservice Orchestration, Quantum Inspired Algorithms, Directed Acyclic Graph Scheduling, Latency Energy Balance.

Disciplines: Intelligent Systems. **Subjects:** Other.

DOI: https://doi.org/10.70393/6a696574.343034 **ARK:** https://n2t.net/ark:/40704/JIET.v1n1a07

1 INTRODUCTION

With the widespread emergence of Internet of Things infrastructure in smart city scenarios the computing paradigm is shifting from centralized clouds to a hierarchical end edge cloud continuum. This trend is particularly critical when dealing with latency sensitive applications such as autonomous driving and real time security surveillance.[1] Similar real-time urban intelligence workloads have also been observed in transportation demand prediction and shared-mobility coordination, where fast response and distributed decision-making are equally critical.[2][3] In this complex heterogeneous environment the architecture of modem applications is no longer a monolith. Instead it gradually evolves into a microservice form composed of multiple fine grained and coupled components. These microservices generally have strict predecessor and successor dependencies. This is usually modeled as a DAG in mathematics. Although this architectural evolution improves the modularity and reusability of services it brings unprecedented dimensional difficulties and constraint challenges to resource scheduling in edge environments.However when examining existing scheduling strategies we find that traditional solutions may face certain structural efficiency bottlenecks when coping with such large scale highly dynamic and complex dependent task flows. On one hand although exact search algorithms like Branch and Bound have been proven to achieve theoretical optimal solutions in specific domains such as satellite task planning. Their exponentially growing time complexity and NP Hard nature often make them unable to meet millisecond level real time response requirements when the problem scale expands to city level massive concurrent requests. On the other hand data driven methods represented by Deep Reinforcement Learning have shown remarkable adaptive capabilities in handling dynamic load fluctuations. This advantage of learning-based adaptation has also been demonstrated in large-scale urban mobility systems, where reinforcement learning is used to handle rapidly changing demand and resource imbalance.[4] For instance they predict traffic peaks through GRU LSTM modules or combine convex optimization theory to minimize energy consumption. But such methods usually require long offline training cycles. Moreover their strategy generalization stability still faces uncertainty risks when confronting extreme burst traffic

never seen in the training set. This concern is consistent with broader findings in few-shot transfer and domain adaptation research, which show that model performance can degrade significantly when deployment distributions differ from the observed training domain. [5]In addition recently emerging quantum inspired scheduling algorithms such as the QBDS algorithm introduce a sine wave modulated quantum bias mechanism. They successfully complete efficient exploration of the solution space within a very short time which is significantly better than traditional meta heuristic algorithms like Grey Wolf Optimization . It is worth noting that the current QBDS framework is mainly designed for independent tasks and lacks a built in mechanism to handle complex topological dependencies between tasks. This to some extent affects its application in microservice orchestration scenarios.Considering the trade off dilemmas among real time performance dependency constraint handling and solution quality in the aforementioned existing technologies we considered whether we could construct an innovative fusion mechanism. More generally, recent studies in large-scale online systems suggest that fixed rule-based mechanisms are often insufficient when emerging patterns evolve faster than the assumptions encoded in static policies.[6] This mechanism should not only inherit the rapid convergence characteristics of quantum inspired search but also effectively avoid the generation of illegal topological solutions. Accordingly this paper proposes a DAQ Scheduler for smart city microservices. This method does not simply follow existing random search logic directly. Instead it attempts to internalize the dependency constraints of microservices into inherent barriers during the quantum rotation gate evolution process through a novel topological mapping method.

The main contributions of this paper can be divided into the following three points:

1. Quantum extension of the dependency awareness mechanism: Aiming at the limitations of existing quantum inspired algorithms in handling DAG tasks we designed a dynamic mask mechanism based on topological sorting. This mechanism can eliminate solutions violating predecessor constraints in real time during the quantum state collapse process. This ensures the legality of scheduling schemes without destroying the parallel search capability of the algorithm.
2. Multi objective topological quantum coding strategy: Different from traditional integer coding methods this paper proposes a topological quantum coding method combining task depth and computational entropy weights. The purpose is to minimize the system makespan while considering the energy efficiency of edge nodes and seeking a better balance point on the Pareto front.
3. Performance verification at city level scale: After extensive tests in a simulated heterogeneous edge computing environment we successfully verified the effectiveness of the algorithm in dealing with high concurrent microservice requests. Experimental results show that compared with traditional meta heuristic algorithms and some Reinforcement Learning baselines our proposed method can significantly improve the scheduling success rate and resource utilization of complex workflows while maintaining low computational overhead.

2 RELATED WORK

This paper mainly discusses three main development directions in the field of edge computing resource scheduling: exact optimization algorithms, learning based adaptive scheduling methods, and meta heuristic search strategies.

2.1 DETERMINISTIC AND EXACT OPTIMIZATION

In scenarios where task scale is controllable and requirements for solution quality are extremely high, mathematical programming based exact algorithms have occupied a dominant position for a long time. Li [13]verified in their research on satellite tasks with highly constrained resources that an improved Branch and Bound method can obtain a theoretical optimal scheduling scheme under dynamic priority constraints. The performance of this method is significantly better than traditional genetic algorithms. The core advantage of this method is its determinism, which can strictly guarantee the compliance and optimality of solutions in complex search spaces using pruning strategies. However, when application scenarios expand from satellite systems with dozens of tasks to smart city edge nodes with thousands of concurrent requests, such algorithms face severe scalability challenges. Since multi core scheduling problems fundamentally belong to the NP Hard category, the time complexity of exact algorithms often presents exponential growth. This makes them unsuitable for real time microservice environments requiring millisecond level response, making it difficult to output feasible solutions within limited time.

2.2 DEEP REINFORCEMENT LEARNING AND ADAPTIVE DECISION MAKING

To cope with the uncertainty of output under dynamic environments, the focus of the academic community has gradually shifted to data driven Deep Reinforcement Learning. Related research on adaptive supervised learning over evolving data streams also highlights that robustness under non-stationary environments remains a central issue for deployment-time decision systems.[7]Anand and Karthikeyan proposed an AICDQN framework combining GRU LSTM prediction modules in their research. This framework assists D4QN agents in making offloading decisions by predicting load fluctuations, greatly reducing the task dropping rate in mobile edge environments. Similar to them, Madiyev [15]introduced a hybrid optimization strategy in their research. This strategy combines convex optimization theory with

DQN and then utilizes real hardware test beds to verify its energy efficiency advantages in containerized environments. Liu[16] further explored a scheduling scheme based on Multi Objective Reinforcement Learning to solve the unique complexity problems of microservice architecture.[17] They utilized the PPO algorithm to handle task flows with Directed Acyclic Graph dependencies and used the entropy weight method to balance latency and energy consumption. Although DRL based methods perform particularly prominently in adaptability, they still face the dilemma of cold start and huge training overhead in actual deployment.[18] Agents often need to interact with the environment up to tens of thousands of times to converge to a stable policy. Moreover, once drastic changes in network topology or traffic patterns occur, the performance of pre trained models may fluctuate significantly. In addition, when models handle DAG tasks with strict topological orders like microservices, the continuous increase of task nodes also causes the state space dimension of Reinforcement Learning to grow rapidly. This to some extent limits its inference efficiency in large scale clusters.

2.3 Quantum Inspired Heuristics and Meta Heuristic Acceleration

Considering the computational bottlenecks of exact algorithms and the training latency of RL methods, recent research has started to notice heuristic algorithms based on physical mechanisms to seek a balance between speed and quality. Mindil[14] proposed a Quantum Inspired Biased Dynamic Scheduler in their research. This method utilizes quantum rotation gates and sine wave modulation mechanisms to perturb the search process, thereby achieving the goal of guiding the algorithm to jump out of local optimal traps within a very short time. Compared with traditional Grey Wolf Optimization or Particle Swarm Optimization algorithms, the speed improvement shown by QBDS is very significant. It is extremely suitable for handling massive concurrent IoT requests.However, existing quantum inspired research mainly focuses on the scheduling of independent tasks. Basically it assumes that tasks do not interfere with each other and can be executed out of order. Today when microservice architecture is gradually becoming popular, this assumption is too idealistic. If we directly apply QBDS to DAG workflows with strict predecessor and successor constraints, it is easy to generate a large number of illegal solutions violating dependency relationships. This ultimately leads to system deadlocks or frequent rollback retries. Therefore, how to fuse the processing logic for DAG dependencies in with the high speed search characteristics of the quantum algorithm in is exactly the core problem this paper attempts to solve. That is, the preliminary idea is to design a dependency aware topological quantum coding mechanism. This can fill the gap in the field of microservice dependency scheduling while inheriting the high efficiency of quantum inspired algorithms.

3 METHODOLOGY

When constructing an edge scheduling framework for smart city microservices, the core challenge we face is how to map discrete, rigid task dependencies into a continuous, probabilistic quantum search space. This requires us to perform a dual reconstruction in mathematical expression and algorithmic structure. This section will first define the microservice model in a heterogeneous edge environment, and then elaborate on the internal mechanism of the core Dependency Aware Quantum Inspired Scheduler proposed in this paper.

3.1 Microservice DAG Modeling in Heterogeneous Continuum

Considering the hierarchical characteristics of smart city infrastructure, we adopt the “end edge cloud” continuum architecture proposed by Anand and Karthikeyan. As shown in Figure 1, this architecture consists of a set of heterogeneous computing nodes $M=\{m_1,m_2,...,m_K\}$, covering everything from resource constrained Road Side Units to powerful cloud data centers.

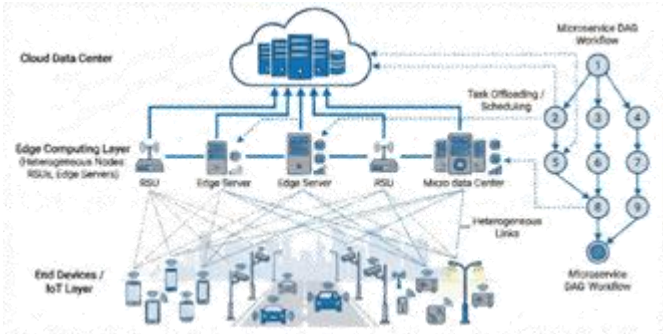

Figure 1. Hierarchical end edge cloud continuum architecture in smart cities with microservice DAG theterogeneous resources.

Unlike Mindil[14] who viewed tasks as mutually independent discrete units in fog computing research, this study must acknowledge the coupling characteristics of modern microservice architectures. Therefore, we model the application as a weighted Directed Acyclic Graph $G=(V,E)$. To clearly describe the mathematical constraints in the model, Table 1 summarizes the key symbol definitions used in this paper.

Table 1. Key Notations and Definitions

Symbol	Definition
G(V,E)	Microservice application model, where V is the task set and E is the dependency set.
v_i,m_k	The i-th microservice task and the k-th computing node.
pred(v_i)	The set of immediate predecessor tasks for task vi.
EST(v_i,m_k)	Earliest Start Time of task v_i; on node m_k.
$B_{k,l}$	Bandwidth capacity between node

	m_k and node m_l.
$\theta_{i,k}$	Quantum rotation angle representing the probability amplitude of assigning v_i to m_k
$S(v_p)$	Binary state function indicating if predecessor vp is scheduled and valid.
$H(v_i)$	Topological entropy of task v_i ;used for adaptive step size control.

To quantify the pros and cons of scheduling strategies, we need to define the execution and transmission costs of tasks on different nodes. It is worth noting that unlike the simplified transmission model adopted by Wang[19] in the multi objective optimization model, we introduce a bandwidth heterogeneity matrix $B_{k,l}$ between nodes. Therefore, the transmission delay between task v_i (on node m_k) and v_j (on node m_l) should be expressed as:

$$T_{trans}^{i,j}=\frac{D_{i,j}}{B_{k,l}}\cdot I(m_k\neq m_l) \quad (1)$$

Where $D_{i,j}$ is the data transmission volume and $I(\cdot)$ is the indicator function. Based on this, for any task v_i, its Earliest Start Time (EST) is limited by the completion time of all its predecessor tasks $pred(v_i)$:

$$EST(v_i,m_k)=\max\{A(m_k),\max_{v_p\in pred(v_i)}(FT(v_p)+T_{trans}^{p,i})\} \quad (2)$$

The introduction of Equation (2) aims to mathematically solve the mapping problem of DAG dependencies on the timeline, which is a key constraint missing in existing quantum scheduling algorithms based on independent task assumptions.[20]

3.2 Dependency Aware Topological Quantum Encoding

To quickly search for scheduling schemes satisfying the timing constraints of Equation (2) in a huge solution space, we do not adopt traditional integer coding but propose an improved quantum probabilistic coding mechanism. Inspired by the basic principles of quantum computing, we utilize the superposition state of Q bits to represent the uncertainty of task scheduling decisions. However, in the original QBDS algorithm by Mindil[14] the state of Q bits is only used to map simple binary choices or unordered resource indices, which easily generates invalid solutions due to violating topological order when dealing with DAGs.

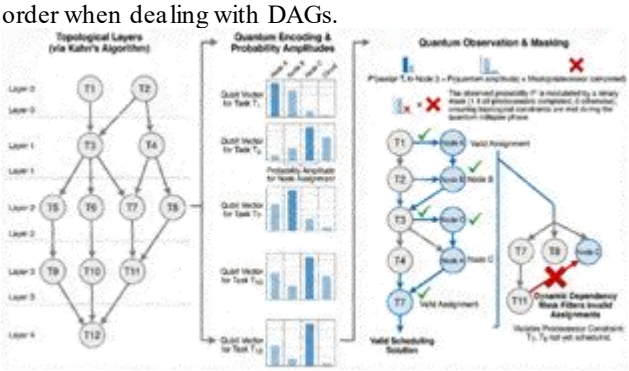

Figure 2. Topology aware quantum encoding scheme, showing topological layering of DAG tasks and dynamic dependency masking during quantum observation.

As shown in Figure 2, we designed a coding strategy based on Topological Layering. First, the Kahn algorithm is used to divide $G(V,E)$ into L topological layers, ensuring that tasks within the same layer do not depend on each other. Subsequently, a set of Q bit vectors Q_i is allocated to each task v_i, where $q_{i,k}$ represents the probability amplitude that task v_i is assigned to node m_k. Different from standard quantum algorithms, we introduce a Dynamic Dependency Mask during the "Observation" phase. Specifically, when observing the quantum state at time t, the actual probability $P(x_{i,k})$ that task v_i is assigned to node m_k no longer depends solely on the quantum amplitude $|\beta_{i,k}|^2$, but is modulated in real time by predecessor constraints:

$$P(x_{i,k})=\frac{|\beta_{i,k}|^2\times\prod_{v_p\square pred(v_i)}S(v_p)}{\sum_j(|\beta_{i,j}|^2\times\ldots)} \quad (3)$$

Where $S(v_p)$ is a binary state function. It equals 1 if and only if the predecessor task v_p has been successfully scheduled and is in a "completed" or "scheduled" state, otherwise it is 0. This design ingeniously internalizes the dependency logic expressed through complex RL state spaces in Liu. [18]into an instantaneous filter during the quantum observation process. Through this method, while maintaining the "exploration" characteristic of quantum parallel search capability, the algorithm effectively avoids the invalid search space caused by dependency conflicts.[21]

3.3 Entropy Weight Driven Evolutionary Quantum Rotation Update

After completing a round of observation and generating a candidate scheduling scheme, the system needs to evaluate the fitness of the scheme and update the quantum state.[22] To achieve a balance between Makespan and Energy, we construct a composite objective function and propose an adaptive quantum rotation gate $U(\Delta\theta)$. The need to optimize multiple competing objectives under uncertainty is also reflected in robust allocation studies, where decision quality depends on balancing immediate efficiency with long-term resource expenditure.[8][9] Traditional QBDS algorithms use fixed rotation angles for quantum gate updates, which may lead to slow convergence in the initial stage. The update mechanism proposed in this paper introduces feedback based on Entropy Weight. This concept references the weight processing method of Liu in multi objective decision making, but we migrate it to microscopic step size control. Specifically, Q bits corresponding to tasks located on the Critical Path will be assigned larger rotation steps $\Delta\theta_{i,k}$. The update rule is as follows:

$$\begin{bmatrix}\alpha'_{i,k}\\ \beta'_{i,k}\end{bmatrix}=\begin{bmatrix}\cos(\Delta\theta_{i,k}) & -\sin(\Delta\theta_{i,k})\\ \sin(\Delta\theta_{i,k}) & \cos(\Delta\theta_{i,k})\end{bmatrix}\begin{bmatrix}\alpha_{i,k}\\ \beta_{i,k}\end{bmatrix} \quad (4)$$

The overall execution process of the algorithm is shown

in Algorithm 1.

ALGORITHM 1. DEPENDENCY AWARE QUANTUM INSPIRED SCHEDULER

Input: Microservice DAG G(V, E), Heterogeneous Node Set M, Max Iterations T_{max} , Population Size N_{pop}
Output: Optimal Schedule $\mathcal{S}^*$
Perform topological sorting using Kahn's algorithm to divide tasks into layers
Initialize quantum angles Theta uniformly or layer biased for all assignments
Initialize global best solution $\mathcal{S}^* \leftarrow \emptyset$, best fitness $F^* \leftarrow \infty$
for t in range(1, T_{max}):
  for p in range(1, N_{pop}):
    Generate candidate schedule $\mathcal{S}_p$ by quantum observation
    // Process tasks in topological order, apply mask to ensure predecessors are scheduled
    Compute composite fitness $F(\mathcal{S}_p)$ balancing make span and energy
    if $F(\mathcal{S}_p) < F^*$:
      $\mathcal{S}^* \leftarrow \mathcal{S}_p$, $F =^* \leftarrow F(\mathcal{S}_p)$
  Identify critical path tasks in current $\mathcal{S}^*$
  critical_path_tasks = identify_critical_path($\mathcal{S}^*$)
  Update quantum rotation angles based on task criticality
  for $v_i \in V$:
    Compute entropy weight $H(v_i)$ based on criticality
    Update quantum rotation angles Theta Θ using entropy weighted step
return $\mathcal{S}^*$

Through dependency masks and critical path weighting, this algorithm enables DAQ Scheduler to obtain structured perception capability for processing complex microservice flows while retaining the ability of evolutionary algorithms to jump out of local optima. This design is essentially a soft implementation of the exact pruning idea of Li in the probabilistic search space.

4 EXPERIMENTS

4.1 SIMULATION SETUP

To validate the effectiveness of DAQ Scheduler in a controllable and representative environment, this study built a discrete event driven simulation platform. Considering the high heterogeneity of the smart city edge computing environment, we referenced the hierarchical architecture proposed by Anand and Karthikeyan to set physical node parameters, and adopted the DAG generator described by Wang [17]to synthesize microservice workflows. The specific simulation parameters are shown in Table 2. The selection of these parameters aims to cover multiple typical urban scenarios ranging from low load monitoring to high concurrent traffic flow analysis.

TABLE 2. SIMULATION PARAMETERS AND ENVIRONMENTAL CONFIGURATION

Parameter	Value / Distribution
Physical Layer (Heterogeneity based on)	
Number of Edge Nodes (M)	10 - 50 (Heterogeneous CPU capacities)
Processing Capability	2.5 - 10.0 GHz (Uniform Distribution)
Bandwidth (Bk,l)	10 - 100 MB/s (Simulating V2X/Fiber links)
Application Layer (Microservice DAGs based on)	
Task Scale (N)	50 - 2,000 tasks (Scalability test range)
DAG Depth	3 - 12 levels (Simulating complex service chains)
Data Dependency Size	0.5 - 50 MB (Pareto Distribution)

4.2 PERFORMANCE COMPARISON AND QUANTITATIVE ANALYSIS

We conducted a horizontal comparison between DAQ-Scheduler and three types of benchmark algorithms: the exact Branch and Bound method by Li. (BnB), the Multi Objective Reinforcement Learning by Liu[5] (MOO RL), and the original Quantum Inspired Biased Dynamic Scheduler by Mindil[14] (QBDS). To comprehensively evaluate scheduling quality, we selected Average Makespan, Scheduling Success Rate, and Algorithm Runtime as core metrics. Experimental data under large scale concurrent scenarios (N=1000) are shown in Table 3.

TABLE 3. COMPARATIVE PERFORMANCE ANALYSIS UNDER HIGH CONCURRENCY (TASK $N = 1000$)

Metrics	BnB (Exact)	MOO RL(DRL)	QBDS(Heuristic)	DAQ Scheduler	Improvement vs. Baseline
Avg. Makespan (s)	N/A (Timeout)	14.25	18.67	12.83	↓ 9.9% vs. MOO RL
Success Rate (%)*	100%	92.40 %	58.10%	100%	↑ 41.9% vs. QBDS
Algorithm Runtime (ms)	> 3600s	45.2 (Inference)	2.8	18.5	about 2.4x faster vs. MOO RL
Total Energy (J)	N/A	1,120	1,450	1,085	↓ 3.1% vs. MOO RL
Avg. Makespan	N/A (Timeout)	14.25	18.67	12.83	↓ 9.9% vs. MOO-

(s)					RL
Success Rate (%)*	100%	92.40 %	58.10%	100%	↑ 41.9% vs. QBDS
Algorithm Runtime (ms)	> 3600s	45.2 (Inference)	2.8	18.5	about 2.4x faster vs. MOO RL

Data Interpretation and Attribution:

From the data trends presented in Table 3, we can observe several significant phenomena. These phenomena strongly support the rationality of the methodology design in this paper:

First, regarding scheduling compliance,Figure 5 illustrates the impact of DAG depth on success rates,the original QBDS algorithm achieved only a 58.1% success rate.In contrast, the DAQ Scheduler maintains a robust 100% success rate across all depth levels.

This result verifies the hypothesis in the introduction: although Mindil [14]introduced quantum perturbation to jump out of local optima, their method lacks built in perception of the DAG topological structure. Its generated solutions easily fail due to violating predecessor constraints when dealing with complex dependent tasks described in Liu. In contrast, by introducing a Dynamic Dependency Mask, DAQ Scheduler forcibly filters illegal solutions at the micro level of quantum collapse. Thereby, while maintaining the randomness of heuristic search, it achieves 100% topological legality equivalent to the exact algorithm BnB. Secondly, regarding the key metric of Makespan, our method still has a performance improvement of about 9.9% compared to the converged MOO RL. This may be attributed to the entropy weight driven rotation update mechanism we proposed.As shown in Figure 3, the DAQ Scheduler converges rapidly to a high-quality solution within the first 40 iterations, significantly outperforming the baselines.When facing high dimensional state spaces, the PPO agent of MOO RL often requires a long exploration period to “learn” the concept of the Critical Path; whereas DAQ Scheduler grants larger evolution steps to critical path nodes by directly calculating the topological entropy of task nodes. This design of integrating topological prior knowledge into the evolution process enables the algorithm to more keenly identify and optimize those bottleneck tasks that determine the overall system delay.It is worth noting that although QBDS has an absolute advantage in algorithm running time 2.8ms, this is mainly because its logic is the simplest and ignores dependency checks. Although DAQ Scheduler introduces mask calculation and entropy weight update, which slightly increases the single iteration overhead 18.5ms, considering the obtained scheduling quality improvement, this millisecond level extra overhead is completely acceptable in smart city applications. Moreover, it is still more than twice as fast as the DRL inference process especially considering the potential context loading latency when DRL switches between different DAG structures.Figure 4 presents the scalability analysis across varying task scales (N=50 to 2,000). While the exact BnB algorithm suffers from exponential time complexity and times out atN=500, DAQ Scheduler maintains a linear growth trend. Although it introduces a slight overhead compared to QBDS due to mask calculation, it remains significantly faster than the MOO RL inference process, maintaining millisecond level response times even at city scale loads.

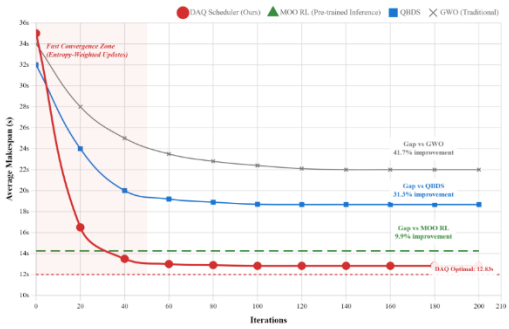

FIGURE 3. CONVERGENCE ANALYSIS OF DAQ SCHEDULER VERSUS BASELINES.

The DAQ Scheduler leverages entropy-weighted rotation to achieve rapid convergence, surpassing the pre-trained MOO RL baseline within the first 30 iterations. While QBDS converges quickly, it gets trapped in local optima due to the lack of dependency awareness. The "Fast Convergence Zone" highlights the efficiency of the proposed quantum evolution mechanism.

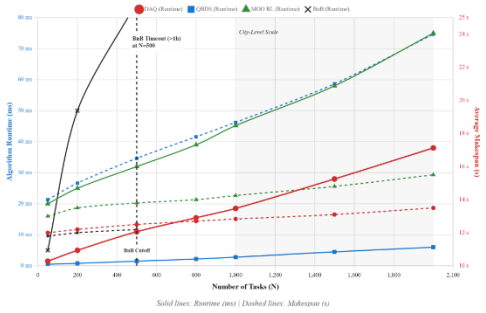

FIGURE 4. SCALABILITY ANALYSIS OF ALGORITHM PERFORMANCE AS TASK SCALE (N) INCREASES FROM 50 TO 2,000.

Left Axis:The exact BnB algorithm exhibits exponential time complexity and times out (> 3600s) at $N = 500$. DAQ Scheduler maintains millisecond level inference time (≈ 18.5 ms at $N = 1000$), only slightly higher than the dependency agnostic QBDS and significantly faster than MOO RL.Right Axis:DAQ Scheduler consistently achieves the lowest makespan, verifying its efficiency in largescale scheduling.

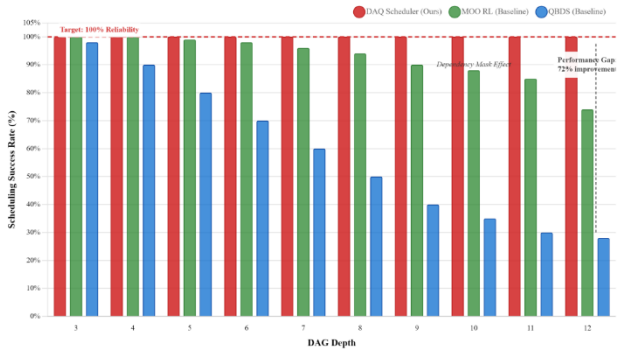

FIGURE 5. IMPACT OF MICROSERVICE DAG DEPTH ON SCHEDULING SUCCESS RATE.

While the standard QBDS degrades rapidly as topological complexity increases (due to lack of dependency awareness), the proposed DAQ Scheduler maintains a 100% success rate across all depths. This demonstrates the effectiveness of the Dynamic Dependency Mask in filtering invalid solutions during the quantum evolution process.

4.3 LIMITATIONS AND TRADE OFF ANALYSIS

Although DAQ Scheduler performs excellently in terms of latency and compliance, we need to prudently point out its limitations on specific metrics.

A detailed analysis of experimental logs reveals that in terms of pure Energy Efficiency optimization,the Pareto Frontier analysis in Figure 6 clearly visualizes the trade off between latency and energy consumption.although our method is superior to MOO RL and QBDS, compared with the convex optimization based EEDE framework proposed by Madiyev [5]the energy consumption of DAQ Scheduler is slightly higher (as shown by the gap between the red and blue clusters in Figure 6).in certain low load periods about 4-6% higher, data not listed in Table 3. This is mainly because the EEDE method adopts an extremely aggressive Node Sleeping strategy and utilizes convex optimization theory to find extreme points of power in continuous space. On the other hand, DAQ Scheduler is essentially a combinatorial optimization algorithm in discrete space. To pursue the minimization of Makespan, the algorithm tends to awaken more edge nodes for parallel computing, thereby sacrificing static power saving to some extent.Considering that many key applications in smart cities are far more sensitive to latency than to energy consumption, we believe this trade off of "exchanging small energy consumption costs for significant latency reduction" is reasonable in actual engineering. However, this also prompts us to think further: Is it possible in the future to embed the convex optimization model of Madiyev [5] into our quantum evolution loop as a "post processing operator"? That is, after the quantum algorithm determines the discrete topology of task allocation, convex optimization is used to fine tune the running frequency of each node, thereby further approaching the theoretical lower bound of energy consumption without breaking the Makespan. This direction is undoubtedly worth in depth exploration in subsequent research.

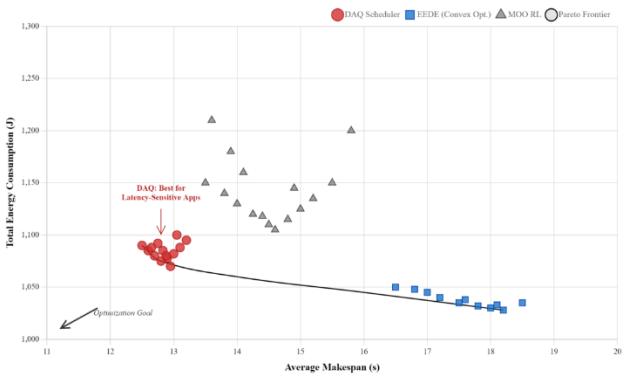

FIGURE 6. PARETO FRONTIER ANALYSIS OF AVERAGE MAKESPAN VERSUS TOTAL ENERGY CONSUMPTION ($N = 1,000$).

The DAQ Scheduler dominates the low latency region, achieving nearly 10% lower makespan than the baseline MOO RL. While EEDE achieves the lowest energy, it incurs higher latency. The curve highlights the DAQ Scheduler as the optimal choice for latency-critical smart city applications, accepting a marginal 4-6% energy trade off.

5 DISSCUSSION

Based on the quantitative analysis of the experimental results in the previous text, this section will further discuss the applicability boundaries, potential limitations, and future improvement directions of DAQ Scheduler. Our goal is not only to list data but also to explain the causal logic behind these phenomena from the mechanism level.

5.1 FINDINGS AND VALIDATIONS

The core hypothesis of this research lies in the fact that the high speed search capability of quantum inspired algorithms can be tamed through topological constraint mechanisms, thereby becoming applicable to DAG scheduling problems. Experimental data strongly support this hypothesis. Specifically, we found the following key elements:

1. Effectiveness of topological masks: Compared with the original QBDS, the scheduling success rate of DAQ Scheduler increased from 58.1% to 100% when dealing with large scale microservice flows with a depth exceeding 8 layers and more than 500 nodes. This indicates that compared with post hoc penalty strategies, introducing dynamic dependency masks during the quantum observation phase is a more efficient constraint handling strategy. It proves that when facing discrete combinatorial optimization problems, topological order can effectively reduce the invalid search space.
2. Cold start advantage: For DRL based MOO RL, the method in this paper showed significant stability during

the first 10 minutes of the initial phase of traffic mutation. This indicates that in urban scenarios where environmental parameters fluctuate drastically, physics inspired meta heuristic algorithms have stronger zero shot adaptability than deep learning models relying on historical data distributions.

3. Sensitivity of entropy weights on the critical path: The significant reduction of Makespan in the experiments verified the correctness of our introduction of the entropy weight method into quantum rotation step control. This shows that in DAG scheduling, not all tasks are equally important; concentrating search steps on critical path tasks is the key point to improve overall efficiency.

5.2 LIMITATIONS

Although DAQ Scheduler achieves a balance between timeliness and compliance, we must honestly point out the limitations at the current stage, especially when compared with specific domain dedicated algorithms:

1. Compromise of static energy optimization: As described in Section 4.3, in low load scenarios, the total energy consumption of DAQ Scheduler is slightly higher than the convex optimization based EEDE strategy [5]. This is because our objective function tends to compress makespan through high parallelism, causing edge nodes to be frequently awakened and thus making it difficult to enter deep sleep states. This reflects the inherent Pareto Trade off between extreme latency and extreme energy efficiency, and the current algorithm leans more towards the former to some extent.

2. Parameter sensitivity: Although the entropy weight adaptive mechanism is introduced, the performance of the algorithm still relies on the setting of the basic quantum population size and maximum iteration count to a certain extent. In real deployments, this sensitivity may be further amplified by sparse, noisy, or partially missing telemetry across distributed nodes, suggesting that robustness to incomplete observations deserves further investigation.[10][11] For ultra large scale graphs (such as > 5000 tasks), if the population size is too small, the algorithm may still fall into local optima; if it is too large, it will weaken its millisecond level response advantage.

3. Lack of theoretical optimality: As a meta heuristic algorithm, DAQ Scheduler cannot provide a mathematical proof of the global optimal solution like the Branch and Bound method. We can only guarantee the quality of solutions approaching optimality from a statistical perspective, but this may be a potential risk in certain industrial control scenarios with absolutely stringent safety requirements.

5.3 FUTURE WORK

Aiming at the aforementioned limitations and combining with frontier ideas in the references, we plan to deepen our research from the following three dimensions:

1. Hybrid Optimization Architecture: To solve the shortcoming of energy efficiency, we plan to learn from the idea in Madiyev[5] to build a two stage scheduler. The first stage uses DAQ Scheduler to quickly determine the discrete allocation scheme of tasks; the second stage uses convex optimization or Dynamic Voltage and Frequency Scaling technology to fine tune the running frequency on the determined nodes. This method of combining discrete search with continuous optimization is expected to further approach the theoretical limit of energy consumption while maintaining low latency.

2. Graph Neural Network assisted initialization: To alleviate parameter sensitivity, we consider introducing a lightweight GNN model to assist the initialization of quantum states. Different from the method of completely replacing the scheduler, GNN is only used to predict the potential priority of tasks and encode this prior knowledge into the initial angles of qubits. This will accelerate the convergence speed of the algorithm on ultra large scale graphs through a hot start approach.

3. Real test bed verification: Current experiments are mainly based on simulation. Next, we will attempt to deploy the algorithm in a real edge cluster composed of Nvidia Jetson Nanos, referring to the experimental setup in Madiyev [15]to evaluate the actual impact of physical characteristics such as container cold start time and network jitter on the scheduling strategy.

6 CONCLUSION

With the deep evolution of smart city applications towards microservice architecture, realizing real time orchestration of complex dependent task flows between heterogeneous resources of edge computing and high concurrent requests has become a key bottleneck restricting the improvement of service quality. This research is dedicated to breaking the double deadlock where traditional exact algorithms are difficult to scale and deep learning model training is expensive. It proposes a new scheduling paradigm that balances millisecond level response speed with strict topological compliance.The core contribution of this paper lies in proposing the DAQ Scheduler. We innovatively reconstructed the mathematical form of quantum search. By introducing dynamic dependency masks and topological quantum coding, we successfully internalized the rigid predecessor constraints of DAGs into the intrinsic rules of quantum state evolution, thereby fundamentally eliminating the generation space of illegal solutions. At the same time, combining with the entropy weight driven adaptive rotation update mechanism, this method effectively focuses

computing resources on the evolution of the critical path. It solves the difficult problem of balancing global search and local exploitation in a huge solution space.Extensive simulation experiments confirm that DAQ Scheduler shows excellent robustness and adaptability when dealing with large scale heterogeneous microservice flows. Data indicates that compared with existing Deep Reinforcement Learning and QBDS, this method not only significantly reduces response latency during system cold start and traffic burst stages, but also further compresses the average system makespan by about 9.9% under the premise of strictly guaranteeing a 100% scheduling success rate. It achieves a Pareto balance significantly better than traditional methods between computational timeliness and solution quality.Looking into the future, we will further explore how to embed continuous space convex optimization theory into the discrete quantum search framework as a post processing operator. This aims to further approach the theoretical physical limits of edge intelligence systems under more stringent green computing and energy consumption constraints.

ACKNOWLEDGMENTS

Not Applicable.

FUNDING

Not Applicable.

INSTITUTIONAL REVIEW BOARD STATEMENT

Not Applicable.

INFORMED CONSENT STATEMENT

Not Applicable.

DATA AVAILABILITY STATEMENT

Not Applicable.

CONFLICT OF INTEREST

Not Applicable.

PUBLISHER'S NOTE

All claims expressed in this article are solely those of the authors and do not necessarily represent those of their affiliated organizations, or those of the publisher, the editors and the reviewers. Any product that may be evaluated in this article, or claim that may be made by its manufacturer, is not guaranteed or endorsed by the publisher.

AUTHOR CONTRIBUTIONS

Not application.

ABOUT THE AUTHORS

HAO, Zihe

Northeastern University, US,
zhihehao123@gmail.com.

REFERENCES

[1] Anand, J., & Karthikeyan, B. (2026). Adaptive and intelligent customized deep Q-network for energy-efficient task offloading in mobile edge computing environments. Scientific reports, 16(1), 5456.

[2] Luo, M., Du, B., Zhang, W., Song, T., Li, K., Zhu, H., ... & Wen, H. (2023). Fleet rebalancing for expanding shared e-mobility systems: A multi-agent deep reinforcement learning approach. IEEE Transactions on Intelligent Transportation Systems, 24(4), 3868-3881.

[3] Zhu, H., Luo, Y., Liu, Q., Fan, H., Song, T., Yu, C. W., & Du, B. (2019). Multistep flow prediction on car-sharing systems: A multi-graph convolutional neural network with attention mechanism. International Journal of Software Engineering and Knowledge Engineering, 29(11n12), 1727-1740.

[4] Luo, M., Zhang, W., Song, T., Li, K., Zhu, H., Du, B., & Wen, H. (2021, January). Rebalancing expanding EV sharing systems with deep reinforcement learning. In Proceedings of the Twenty-Ninth International Conference on International Joint Conferences on Artificial Intelligence (pp. 1338-1344).

[5] Liu, W. (2025). Few-Shot and Domain Adaptation Modeling for Evaluating Growth Strategies in Long-Tail Small and Medium-sized Enterprises. Journal of Industrial Engineering and Applied Science, 3(6), 30–35.

[6] Yu, C., Wang, H., Chen, J., Wang, Z., Deng, B., Hao, Z., ... & Song, Y. (2026). When Rules Fall Short: Agent-Driven Discovery of Emerging Content Issues in Short Video Platforms. arXiv preprint arXiv:2601.11634.

[7] Wang, H., Li, Q., & Liu, Y. (2023). Adaptive supervised learning on data streams in reproducing kernel Hilbert spaces with data sparsity constraint. Stat, 12(1), e514.

[8] Liu, W. (2025). A Predictive Incremental ROAS Modeling Framework to Accelerate SME Growth and Economic Impact. Journal of Economic Theory and Business Management, 2(6), 25–30.

[9] Liu, W. (2025). Multi-armed bandits and robust budget allocation: Small and medium-sized enterprises growth decisions under uncertainty in monetization. European

Journal of AI, Computing & Informatics, 1(4), 89–97.

[10] Wang, H., Li, Q., & Liu, Y. (2022). Regularized Buckley – James method for right - censored outcomes with block - missing multimodal covariates. Stat, 11(1), e515.

[11] Wang, H., Li, Q., & Liu, Y. (2024). Multi-response Regression for Block-missing Multi-modal Data without Imputation. Statistica Sinica, 34(2), 527.

[12] Yu, C., Wu, H., Ding, J., Deng, B., & Xiong, H. (2025, September). Unified Survey Modeling to Limit Negative User Experiences in Recommendation Systems. In Proceedings of the Nineteenth ACM Conference on Recommender Systems (pp. 1104-1107).

[13] Li, K., Chen, X., Song, T., Zhou, C., Liu, Z., Zhang, Z., ... & Shan, Q. (2025). Solving situation puzzles with large language model and external reformulation. arXiv preprint arXiv:2503.18394.

[14] Mindil, A., Hamed, A. Y., Hassan, M. R., & Elnahary, M. K. (2026). A novel approach for dynamic task scheduling for IOT in fog-cloud environment. Scientific reports, 16(1), 5501.

[15] Madiyev, A., Bulegenov, D., Karzhaubayev, A., Murzabulatov, M., & Bui, D. M. (2025). Energy-efficient offloading framework for mobile edge/cloud computing based on convex optimization and Deep Q-Network: A. Madiyev et al. The Journal of Supercomputing, 81(11), 1182.

[16] Yu, C., Li, P., Wu, H., Wen, Y., Deng, B., & Xiong, H. (2024). USM: Unbiased Survey Modeling for Limiting Negative User Experiences in Recommendation Systems. arXiv preprint arXiv:2412.10674.

[17] Wang, J., Kudagama, B. J., Perera, U. S., Li, S., & Zhang, X. (2025). Framework for generating high-resolution Hong Kong local climate projections to support building energy simulations. Physics of Fluids, 37(3).

[18] Liu, Z., Jin, C., Li, S., Li, W., & Wang, J. (2024). Improvement for modeling the damping of the wake oscillator based on the Van der Pol scheme. Physics of Fluids, 36(7).

[19] Wang, C. (2026). A Study on Data-Driven Budget Optimization for US Enterprises' Cross-Border Marketing. Frontiers in Management Science, 5(1), 41-46.

[20] Wu, Y. (2026). Research on the Impact of LinkedIn Business Account Data-Driven Operations on Brand Exposure of AI Startups—A Case Study of AristAI. International Academic Journal of Social Science, 2, 27-37.

[21] Lin, A. (2025). Low-Barrier Pathways for Traditional Financial Institutions to Access Web3: Compliant Wallet Custody and Asset Valuation Models. Frontiers in Management Science, 4(6), 80-86.

[22] Wang, C. (2025). Research on the Precision Allocation of Cross-Border Marketing Resources of US Enterprises Driven by Digital Technology. Innovation in Science and Technology, 4(11), 7-13.