



ARTICLE

Comparative Analysis of Genetic and Quantum-Inspired Optimization for Zero-Trust Microsegmentation in Brownfield Networks

Reen-Cheng Wang¹, Hong-Sheng Wang² and Kuo-Chun Tseng^{2,*}

¹Department of Computer Science and Information Engineering, National Taitung University, Taitung, Taiwan

²Department of Computer Science and Information Engineering, National Ilan University, Ilan, Taiwan

*Corresponding Author: Kuo-Chun Tseng. Email: kctseng@niu.edu.tw

Received: 29 March 2026; Accepted: 25 May 2026; Published: 23 July 2026

ABSTRACT: Network microsegmentation has become a key mechanism for enforcing zero-trust architecture in enterprise environments, yet its effectiveness remains closely tied to initialization quality. This study formulates network microsegmentation as a state-dependent combinatorial optimization problem in which optimization behavior depends on the availability of structural guidance. A comparative analysis is conducted across four representative optimization paradigms, including genetic algorithms (GA), differential evolution (DE), particle swarm optimization (PSO), and amplitude-ensemble quantum-inspired tabu search (AE-QTS), under both structured and unstructured conditions. Experiments are conducted on a representative brownfield enterprise network using 30 independent runs per configuration. In addition to cost-based evaluation, a fragmentation metric is used to assess the structural quality and manageability of segmentation outcomes. The results indicate that under structured conditions, GA and AE-QTS achieve the best overall performance, with AE-QTS obtaining the best average objective value of -61.29 and GA demonstrating rapid convergence under limited optimization time. Under unstructured conditions, AE-QTS consistently outperforms all other methods, reducing the average objective value from 344.28 (GA) and 1214.60 (DE) to 4.54 under uniform initialization. Moreover, PSO demonstrates comparatively stable and robust behavior, although its performance remains below that of AE-QTS. These findings suggest that microsegmentation can be more appropriately viewed as a condition-dependent optimization problem, in which different optimization methods exhibit different strengths across operational scenarios. The results provide empirical evidence and practical insights that may support the future development of adaptive or hybrid optimization strategies for real-world deployment environments.

KEYWORDS: Network microsegmentation; zero-trust architecture; quantum-inspired algorithm; genetic algorithm; state-dependent optimization

1 Introduction

Zero-Trust Architecture (ZTA) [1] serves as a cornerstone in network security, replacing implicit trust tied to network position with rigorous, continuous authentication of every device and user. This concept was practically demonstrated in large-scale systems like Google's BeyondCorp [2] and codified within industry standards such as NIST SP 800-207 [3] and the CISA Zero Trust Maturity Model [4]. In all of these frameworks, microsegmentation is a core mechanism of ZTA [4,5] as it restricts access to the minimum required to accomplish a given task. However, despite its conceptual maturity, the implementation of microsegmentation remains challenging, particularly in brownfield Ethernet environments where legacy infrastructure, budget constraints, and operational risks limit the adoption of software-defined networking solutions [5].

In such environments, IEEE 802.1Q [6] VLANs remain the most feasible mechanism for implementing logical segmentation without hardware replacement [7]. However, traditional VLAN planning relies on manual configuration or static rule sets, which do not scale with increasing network complexity and evolving security requirements [5]. In this way, microsegmentation can be treated as a combinatorial optimization problem. Given the increasing size and heterogeneity of modern enterprise networks, the difficulty of solving this problem exactly grows exponentially, suggesting that metaheuristic optimization techniques provide a practical approach for obtaining high-quality solutions efficiently.

Genetic algorithms (GAs) have been used in topology design, routing, and resource allocation [8–10], especially when heuristics can assist in initialization [11], while differential evolution (DE) [12] and particle swarm optimization (PSO) [13] provide alternative population-based search mechanisms for large-scale optimization. However, these methods may behave differently when structural information is unavailable. In high-cardinality and discrete assignment problems such as VLAN-based microsegmentation in this study, optimization behavior can be highly sensitive to initialization quality and search-space structure.

To address these limitations, quantum-inspired evolutionary algorithms, an alternative class of optimization approaches, have been explored for combinatorial optimization problems [14]. These algorithms treat each solution as a set of probabilistic states that evolve via quantum-inspired operations, which may improve population diversity and global exploration capability. In addition, variants such as quantum-inspired tabu search (QTS) [15] and amplitude-ensemble quantum-inspired tabu search (AE-QTS) [16] extend this capability by incorporating population-level information. While these approaches have shown strong performance in solving classical combinatorial optimization, their effectiveness for high-cardinality assignment problems such as VLAN-based microsegmentation has not been systematically investigated.

A key observation in practical network environments is that segmentation optimization does not operate under a single static condition. Instead, it exhibits two fundamentally different regimes. In structured scenarios, domain knowledge, organizational policies, or historical configurations provide informative initialization that guides the search toward desirable regions of the solution space. In contrast, in unstructured scenarios, like the initial segmentation of an unknown network or recovering from a failure, the system has to find a segmentation from scratch. This is analogous to the cold-start problem [17,18] in data-driven systems, where the lack of prior knowledge substantially affects the effectiveness of the algorithms. To the best of our knowledge, the existence of this dual-regime structure has not been explicitly considered before. Furthermore, fragmentation is a significant but frequently ignored consideration in the network segmentation domain. While fine-grained segmentation is beneficial for network security, excessive segmentation can lead to overly dispersed VLAN assignments, increasing policy complexity and operational overhead. Existing approaches [19] focus on either minimizing the security risk or maximizing isolation without explicitly considering this trade-off.

Motivated by these observations, this study revisits network microsegmentation from a state-dependent perspective. Rather than proposing a new system architecture, the focus is on understanding the behavior of different optimization strategies under varying initialization conditions in high-dimensional VLAN assignment problems. To this end, a comparative analysis is conducted across four representative optimization approaches, including GA, DE, PSO, and AE-QTS. The study evaluates convergence behavior, solution quality, scalability, and fragmentation characteristics under both structured and unstructured conditions, supported through statistical analysis.

Specifically, this work does not aim to introduce a fundamentally new metaheuristic algorithm. Instead, the contribution lies in formulating network microsegmentation as a state-dependent optimization problem and systematically analyzing optimization behavior across multiple methods under varying initialization conditions. The major contributions of this work are as follows. First, a state-dependent formulation of

network microsegmentation optimization is introduced to examine how much initialization can affect optimization performance. Second, a comparative analysis across the four aforementioned optimization approaches under both structured and unstructured conditions is conducted. Third, fragmentation is identified and quantified as a critical factor affecting the practical usability of segmentation results. Finally, the complementary advantages of different optimization approaches are discussed to inform the design of adaptive optimization strategies for real-world network microsegmentation problems.

2 Problem Formulation and State-Dependent Optimization

2.1 Problem Definition

In brownfield network environments, microsegmentation aims to partition network assets into logical segments such that lateral attack risk is minimized while maintaining required communication and operational manageability. This task can be formulated as a multi-objective combinatorial optimization problem over VLAN assignments.

Let $A = \{a_1, a_2, \dots, a_N\}$ denote the set of assets in network. A segmentation configuration is defined by a mapping $f: A \rightarrow \{1, 2, \dots, K\}$, where K is the number of VLANs and $f(a_i)$ denotes the assignment of asset a_i . A pairwise risk function $R(i, j)$ quantifies the security risk between assets. The total intra-segment risk is $R_{total} = \sum_{v \in V} \sum_{(i,j) \in v} R(i, j)$, where V denotes the set of VLANs.

The overall objective is defined as

$$\min_f C(f) = W_r \cdot R_{total} + W_b \cdot P_{business} + W_h \cdot P_{hard} - W_p \cdot B_{practice} - W_m \cdot B_{mgmt},$$

where $P_{business}$ penalizes violations of communication requirements, P_{hard} enforces strict constraints, and $B_{practice}$ and B_{mgmt} reward security best practices and manageability, respectively. The weights (W_r, W_h, W_b, W_p, W_m) control the relative importance of each objective. The weight values are not intended to provide a universally optimal formulation, but rather to establish a deployment-oriented prioritization strategy appropriate for practical network environments. For example, higher weights are used to emphasize those functions that represent good security practices or hard constraints. This ensures that poor or insecure solutions are strongly penalized in the final result.

This formulation defines a high-dimensional optimization problem where the search space size grows exponentially as the number of network assets increases. Due to label permutation and structural redundancy, many configurations are equivalent, resulting in a degenerate and poorly structured search space that challenges most optimization methods.

2.2 Structured and Unstructured States and Initialization

In this study, optimization states are characterized by initialization conditions. Structured states employ knowledge-guided initialization derived from organizational roles, security policies, or existing configurations, enabling faster convergence from more informative starting points. Unstructured states instead rely on uninformed uniform initialization, such as during new deployments or post-incident recovery, where global exploration becomes increasingly important.

To quantify this distinction, an initialization quality metric Q_{init} , which measures the relative quality of the initial population P_0 is defined. Let $C(x)$ denotes the cost of a candidate solution x , and let C^* and C_{max} denote the best and worst achievable costs, respectively. The initialization quality is defined as

$$Q_{init} = 1 - \frac{E_{x \in P_0} [C(x)] - C^*}{C_{max} - C^*}$$

where $E_{x \in P_0} [C(x)]$ represents the average cost across all initial solutions, capturing the overall quality of the starting population. The metric is introduced primarily as a normalized analytical indicator for controlled experimental comparison rather than a directly deployable operational metric. The formulation is intended to quantify relative initialization quality across experimental conditions and may additionally serve as a threshold-based indicator for selecting appropriate optimization methods under different initialization states. A higher value of Q_{init} indicates that the initial population is closer to optimal solutions (structured conditions) while lower values indicate unstructured conditions. This definition sets the problem of microsegmentation as a state-dependent optimization problem and provides a unified way to explore the performance of different methods in relation to initialization for a heterogeneous set of network conditions.

2.3 Fragmentation Metric

Aside from the concern for risk reduction, excessive segmentation may introduce unnecessary operational complexity in terms of the management. To account for this, we define the segmentation fragmentation matrix.

Let p_k denote the proportion of assets assigned to VLAN k . The segmentation entropy is defined as

$$H_s = - \sum_{k=1}^K p_k \log p_k$$

which measures the distributional dispersion of assets across VLANs.

A normalized VLAN count is defined as

$$F_v = \frac{K}{N}$$

where K is the number of VLANs and N is the total number of assets.

The overall fragmentation metric is then defined as

$$F_{frag} = \lambda \cdot H_s + (1 - \lambda) \cdot F_v$$

where $\lambda \in [0, 1]$ is a weighting parameter that balances distributional dispersion and segmentation granularity.

2.4 Optimization Methods

The selected methods represent different optimization paradigms, specifically, GA (exploitation-based evolution), DE (differential perturbation), PSO (swarm), and AE-QTS (probabilistic distributional search). Our motivation is not simply to benchmark these optimization methods, but rather to investigate how they behave in high-cardinality VLAN assignment optimization problems.

2.4.1 Genetic Algorithm (GA)

GA is a well-known optimization algorithm and is therefore adopted as a baseline and as an algorithm which represents the exploitation-based evolution optimization. Each solution is encoded as an integer vector representing VLAN assignments. The algorithm applies selection, crossover, and mutation to iteratively refine solutions based on the objective function. While effective in structured search spaces, GA performance depends strongly on initialization quality and may suffer from premature convergence under unstructured conditions. The GA implementation adopts tournament selection, one-point crossover, and mutation-based diversification to iteratively refine VLAN assignment structures.

2.4.2 Differential Evolution (DE)

DE is included as a representative differential perturbation-based optimization approach. Unlike GA, new candidate solutions are generated by taking the scaled difference of pairs of population members, and is thus emphasizing direction-based diversity of the population. For this algorithm the VLAN allocation is encoded with integers, and the perturbations are translated back to VLAN numbers by constrained discretization. While differential evolution is known to perform well for continuous search spaces, it remains unclear whether differential evolution is effective for high cardinality problems. Additionally, since VLAN labels are permutation-equivalent no matter how the labels are assigned (i.e., the label 1 and label 2 could be swapped without changing the semantics), it is not clear if the differential perturbation-based search mechanism is appropriate for the VLAN allocation problem.

2.4.3 Particle Swarm Optimization (PSO)

PSO is adopted as a representative swarm-based optimization method emphasizing collective exploration behavior. Each particle maintains an individual best solution and updates its search trajectory according to both personal and global best positions.

For this algorithm, the particle is transformed from the continuous number representation to VLAN discrete values by probabilistic discretization. Compared with GA, PSO is less dependent on crossover-based inheritance and has good global search capabilities. However, due to its reliance on swarm attraction dynamics, the search efficiency may also be affected by high-cardinality problems.

2.4.4 Amplitude-Ensemble Quantum-Inspired Tabu Search (AE-QTS)

To address the limitations of population-based methods under unstructured conditions, this study adopts AE-QTS as a probabilistic exploration-driven optimization approach. Although AE-QTS adopts quantum-inspired representations, the optimization process remains fully classical and operates through probabilistic distribution updates rather than quantum computation. In AE-QTS, the search space is represented with a probability matrix $P \in \mathbb{R}^{N \times K}$, where each element $p_{i,k}$ denotes the probability that asset i is assigned to VLAN k , subject to the normalization constraint $\sum_{k=1}^K p_{i,k} = 1, \forall i$. This probabilistic representation functions as a generative model from which candidate solutions are sampled at each iteration. This compression mechanism increases the probability of assignments observed in higher-quality solutions while suppressing alternatives associated with lower-quality solutions.

While AE-QTS itself is not a contribution of this paper, some problem-specific adaptations to the VLAN-based microsegmentation have been made. These include the permutation-equivalence relabeling of solutions and probability update mechanism that incorporates problem-specific knowledge of the state space. A summary of the adopted AE-QTS procedure is provided in Algorithm 1. The probability matrix is initialized either uniformly or using domain knowledge, reflecting unstructured and structured conditions, respectively. At each iteration, candidate solutions are generated via categorical sampling, evaluated using the objective function, and ranked according to their costs. The probability matrix is then updated using a population-informed compression mechanism.

Algorithm 1: AE-QTS for VLAN optimization

Input: Probability matrix $\mathbf{P} \in \mathbb{R}^{N \times K}$, risk weight matrix $\mathbf{RW} \in \mathbb{R}^{N \times N}$,
assets $\mathbf{A}$, population size P , max iterations T , base update Δ_0

Output: Optimal VLAN assignment $\mathbf{s}^*$

(Continued)

Algorithm 1 (continued)

```

1:   $s^* \leftarrow \text{null}; f^* \leftarrow +\infty$ 
2:  for  $t = 1$  to  $T$  do
3:      for  $j = 1$  to  $P$  do                                     // Step 1: Categorical Sampling
4:          for  $i = 1$  to  $N$  do
5:               $S_j[i] \leftarrow \text{sample from Categorical}(\Pi_{i,0}, \Pi_{i,1}, \dots, \Pi_{i,K-1})$ 
6:          end for
7:      end for
8:      for  $j = 1$  to  $P$  do                                     // Step 2: Cost Evaluation
9:           $F_j \leftarrow \text{Cost}(S_j, \text{RW}, A)$ 
10:     end for
11:      $(S, F) \leftarrow \text{SortAscending}(S, F) \triangleright S_1 = \text{Best}, S_P = \text{Worst}$  // Step 3: Sort by Cost (ascending)
12:      $\Pi \leftarrow \text{UpdateQubit}(\Pi, S, \Delta_0, P) \triangleright \text{Algorithm 2}$  // Step 4: Update Qubit
13:     if  $F_1 < f^*$  then                                     // Step 5: Update Global Best
14:          $s^* \leftarrow S_1; f^* \leftarrow F_1$ 
15:     end if
16: end for
17: return  $s^*$ 

```

The update mechanism in AE-QTS is based on a global probability compression strategy. As illustrated in Algorithm 2, the population is first ranked by objective value, and pairs of high-quality and low-quality solutions are formed. For each paired solution, if asset i is assigned to different VLANs, let b denotes the VLAN assignment in the better solution. The probability update rule is defined as

$$p'_{i,k} = \begin{cases} p_{i,k} + \Delta_p & \text{if } k = b \\ p_{i,k} - \frac{\Delta_p}{K-1} & \text{otherwise} \end{cases}$$

where $\Delta_p = \Delta_0/p$ with p representing the ranking index of the solution pair. This design ensures that higher-ranked solutions exert stronger influence on the update process. Consequently, the probability of selecting assignments present in high-quality solutions is progressively increased, while that of inferior assignments is suppressed.

Algorithm 2: UpdateQubit—global probability compression

Input: Probability matrix $\Pi \in \mathbb{R}^{N \times K}$, sorted population $S = \{S_1, \dots, S_P\}$,
base update Δ_0 , population size P

Output: Updated probability matrix Π

```

1:   $H \leftarrow \lfloor P/2 \rfloor \triangleright \text{Number of pairs}$ 
2:  for  $p = 1$  to  $H$  do
3:       $s_{\text{best}} \leftarrow S_p \triangleright \text{Rank } p$ 
4:       $s_{\text{worst}} \leftarrow S_{P+1-p} \triangleright \text{Rank } P+1-p$ 
5:       $\Delta \leftarrow \Delta_0/p$ 
6:      for  $i = 1$  to  $N$  do
7:          if  $s_{\text{best}}[i] \neq s_{\text{worst}}[i]$  then
8:               $b \leftarrow s_{\text{best}}[i] \triangleright \text{Target VLAN from better solution}$ 
9:              for  $k = 0$  to  $K-1$  do

```

(Continued)

Algorithm 2 (continued)

```

10:           if  $k = b$  then
11:                $\Pi_{i,k} \leftarrow \Pi_{i,k} + \Delta$ 
12:           else
13:                $\Pi_{i,k} \leftarrow \Pi_{i,k} - \Delta/(K - 1)$ 
14:           end if
15:       end for
16:   end if
17: end for
18: end for
19: return  $\Pi$ 

```

Furthermore, the VLAN assignment problem exhibits permutation equivalence, where different label indices may correspond to identical grouping structures. To resolve this redundancy, a relabeling procedure is performed after each sampling step to ensure consistent representation of equivalent solutions. This process improves both search efficiency and stability. At each iteration, candidate solutions are sampled from the probability matrix, evaluated using the objective function, and used to update the probabilistic model.

3 Experimental Design

3.1 Network Scenario and Experimental Setup

As shown in Fig. 1, we utilize a small- and medium-size enterprise (SME) brownfield network as our simulation environment. The network includes a range of heterogeneous assets: user workstations (WS), servers, networking devices, voice systems, and Internet of Things (IoT) nodes forming a multi-role and multi-trust-level network. The original network follows a conventional functional subnet design, resulting in coarse-grained broadcast domains.

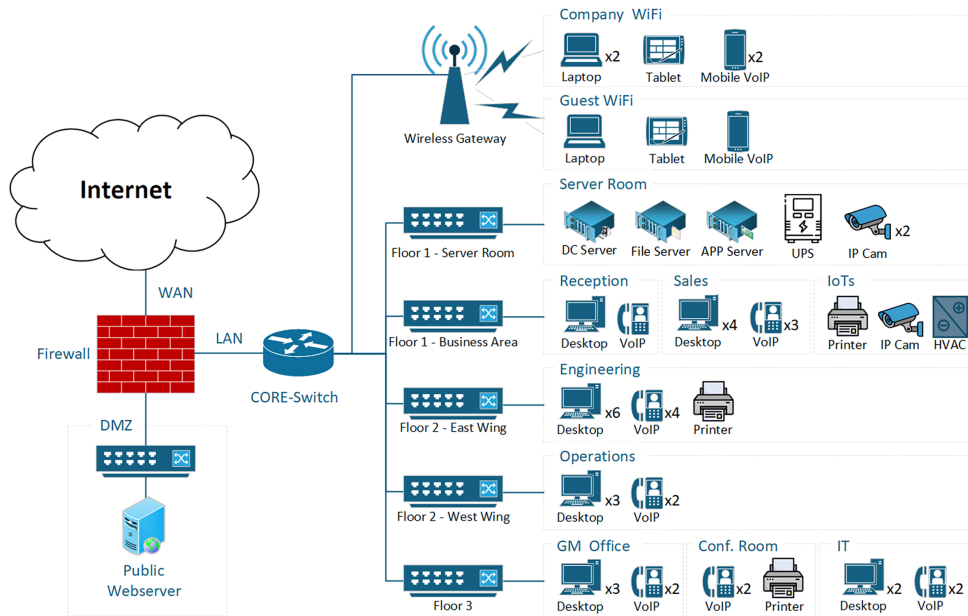


Figure 1: A real-world SME test environment.

For microsegmentation optimization, the network is modeled as an asset graph in which the nodes are devices and the edges are potential connections between them. All experiments were performed using the same topology and the same set of assets.

3.2 Experimental Configurations and Initialization Control

We designed eight experimental configurations to study the optimization performance under two initialization strategies: GA-DK, GA-Uniform, DE-DK, DE-Uniform, PSO-DK, PSO-Uniform, AE-QTS-DK, and AE-QTS-Uniform. Configurations ending with suffix “DK” represent structured initializations using domain-knowledge-based initialization, and configurations ending with “Uniform” denote uninformed initialization using uniform random variables. To quantify initialization quality, the metric Q_{init} , defined in Section 2.2, is employed, allowing initialization conditions to be treated as a continuous variable rather than a categorical distinction.

To mitigate stochastic variability, each configuration is executed over multiple independent runs with different random seeds. All simulations used the same problem formulation, cost functions, termination criteria, and parameter settings, with the AE-QTS probability update step size fixed at $\Delta_0 = 0.01$ across all configurations, ensuring that observed differences can be attributed to intrinsic algorithmic characteristics. Because operational priorities, security policies, and deployment requirements vary across organizations, no universally standardized weighting distribution currently exists for enterprise VLAN microsegmentation objectives. Accordingly, all objective-function weights are uniformly set to 1 in this study to establish a neutral and controlled comparative baseline. GA used a tournament selection with a crossover rate of 0.8 and mutation rate of 0.15. DE employs differential weight $F = 0.5$, and crossover rate $CR = 0.9$. PSO uses inertia weight $w = 0.7$, cognitive coefficient $c_1 = 1.5$ and social coefficient $c_2 = 1.5$.

3.3 Evaluation Metrics

Several metrics are used to evaluate convergence behavior, solution quality, and structural properties of segmentation results. Convergence behavior is assessed by tracking the evolution of the objective value over iterations or time, enabling comparison of convergence speed and stability.

Solution quality is evaluated by the objective values, reflecting the balance among risk minimization, business constraints, and operational considerations. In order to evaluate the solutions found, F_{frag} , defined in Section 2.3, is adopted. This metric accounts for both the dispersion of assets across VLANs and the number of segments, providing a quantitative measure of segmentation complexity. Additional indicators, including VLAN count and distribution balance, are also considered.

3.4 Statistical Analysis and Design Rationale

To determine whether observed differences between configurations are significant, non-parametric tests are employed. Given that the distributions of objective values and fragmentation metrics may violate normality assumptions, the Mann-Whitney U test is used for pairwise comparisons. Effect size is further evaluated using Cliff’s delta to quantify the magnitude of differences and distinguish practically meaningful effects from negligible ones.

The design compares the optimization methods with different initialization schemes while all problem formulations, objective functions, termination criteria and other algorithm parameters remain the same across different configurations. This isolates the impact of the structured and unstructured initial conditions from other parameters in the experiment. The selected methods represent distinct optimization methods,

including GA, DE, PSO and AE-QTS. This allows us to compare optimization behavior across different methods and initialization strategies.

4 Results and Analysis

4.1 Performance Comparison under Structured and Unstructured Conditions

The performance of GA, DE, PSO, and AE-QTS is evaluated under both structured and unstructured initialization conditions using iteration-based (population = 150, 5000 iterations) and time-based (population = 50, 1 min limit) settings. All experiments are conducted on a standard workstation equipped with an Intel Core i5-13500 processor, 24 GB of DDR4 memory, and a Python 3.11+ environment with NumPy $\geq 2.3.1$. Each configuration is executed over 30 independent runs. The statistical results are summarized in Table 1, and the convergence curves are illustrated in Fig. 2.

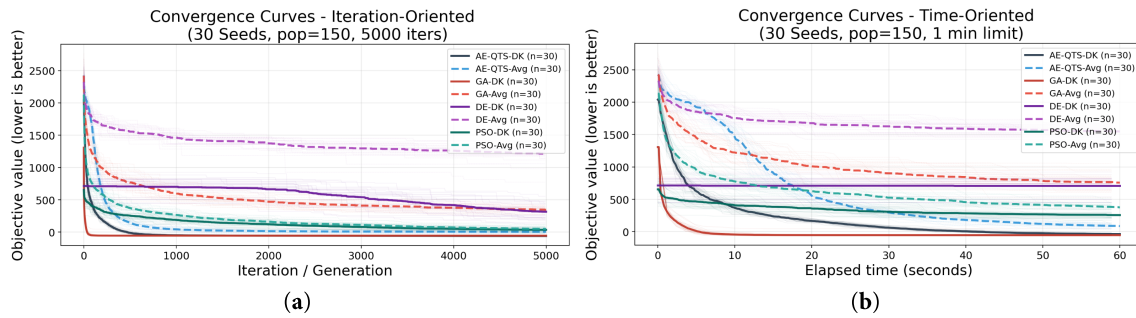


Figure 2: Convergence curves. (a) Iteration-based; (b) time-based.

Table 1: Statistical results (30 independent runs, population = 150).

Configuration	Iteration-Based (5000 iter)		Time-Based (1 min)	
	Mean	Std	Mean	Std
AE-QTS-DK	−61.29	3.53	−37.08	9.42
AE-QTS-Uniform	4.54	19.65	86.66	38.33
GA-DK	−55.48	19.59	−54.54	19.43
GA-Uniform	344.28	47.90	756.77	129.66
DE-DK	315.03	114.10	706.66	135.71
DE-Uniform	1214.60	120.86	1547.42	84.96
PSO-DK	34.68	29.11	256.04	58.93
PSO-Uniform	50.50	28.50	374.79	62.66

Under structured conditions, AE-QTS-DK and GA-DK achieved the strongest overall results, both converging to negative objective values. AE-QTS-DK obtained the lowest iteration-based mean objective value (−61.29), while GA-DK achieved comparable performance (−55.48) and faster early convergence. PSO-DK remained positive, and DE-DK performed substantially worse despite informed initialization. These results suggest that structured initialization mainly benefits methods capable of effectively exploiting prior structural information.

We observe a more distinct difference under unstructured conditions. AE-QTS-Uniform achieved the best iteration-based mean objective value (4.54), outperforming PSO-Uniform (50.50), GA-Uniform

(344.28), and DE-Uniform (1214.60). Statistical analysis confirmed a large practical effect between AE-QTS-Uniform and GA-Uniform (Mann–Whitney $U = 0$, $p < 0.0001$, Cliff’s $\delta = -1.000$). As shown in Fig. 2, AE-QTS-Uniform converged steadily toward near-zero objective values, whereas GA and DE stagnated at much higher values.

Similar trends are observed in the time-based cases. GA-DK still performs well under time-limited cases, indicating a good exploitation capability when the initialization is informed. AE-QTS suffers somewhat in the unstructured cases under the strict time limitation, but it still performs better than the others. In general, DE shows the worst performance regardless of whether the initialization is structured or not, suggesting that the differential mutation operations are not well suited to the VLAN assignment problem which has a large search space and permutation-equivalent structure.

The experimental results support our state-dependent perspective in Section 2. Exploitation-driven methods such as GA perform well with informed initialization. On the other hand, probabilistic exploration-driven optimization (represented here by AE-QTS) demonstrates better robustness under unstructured cases where segmentation structures cannot be directly obtained and must be inferred during the search process. The very large effect sizes observed over multiple configurations also indicate that the difference between these methods is statistically significant and practically meaningful.

4.2 Structural Quality and Fragmentation Analysis

To examine whether objective values correspond to meaningful segmentation structures, Table 2 presents structural metrics such as VLAN count, functional purity, departmental purity, critical isolation effectiveness, and WS-IoT mixing ratio under iteration-based conditions.

Table 2: VLAN segmentation quality metrics under iteration-based conditions.

Configuration	VLANs	Functional Purity	Departmental Purity	Critical Isolation	WS-IoT Mixing
AE-QTS-DK	22.7 (± 0.5)	100.0%	71.2%	30/30	0%
AE-QTS-Uniform	26.1 (± 1.3)	100.0%	67.8%	30/30	0%
GA-DK	23.0 (± 0.5)	100.0%	85.9%	30/30	0%
GA-Uniform	34.4 (± 1.1)	95.0%	69.9%	28/30	77%
DE-DK	33.8 (± 1.7)	96.8%	74.2%	24/30	57%
DE-Uniform	37.3 (± 1.6)	84.4%	68.5%	4/30	100%
PSO-DK	28.3 (± 1.3)	100.0%	70.2%	30/30	0%
PSO-Uniform	28.4 (± 1.2)	99.8%	69.7%	30/30	7%

Under structured conditions, AE-QTS-DK and GA-DK produced compact and high-quality segmentation structures, achieving complete critical isolation, zero WS-IoT mixing, and approximately 23 VLANs. GA-DK obtained higher departmental purity, whereas AE-QTS-DK provided comparable compactness and isolation with lower variance. PSO preserved isolation quality but required more VLANs, while DE produced more fragmented structures with weaker isolation consistency.

The distinction in structure grows even more noticeable under unstructured scenarios. AE-QTS-Uniform maintained complete critical isolation, zero WS-IoT mixing, and a moderate VLAN count, indicating that coherent segmentation structures can be derived from uniform initialization. PSO-Uniform retained acceptable isolation with limited WS-IoT mixing, whereas GA-Uniform and DE-Uniform produced highly fragmented structures with weaker isolation consistency.

These observations suggest that reductions in objective value do not necessarily imply deployable segmentation quality, particularly when fragmentation and isolation consistency are not jointly considered.

4.3 Sensitivity to Hyperparameters and Stability

Table 3 summarizes the sensitivity analysis under different population sizes using 1000 iteration-based runs. Under structured conditions, AE-QTS-DK and GA-DK both achieve competitive performance, although AE-QTS-DK exhibits more stable convergence as population size increases. PSO-DK demonstrates moderate improvement with larger populations, whereas DE-DK remains substantially less effective across all settings.

Table 3: Population size sensitivity analysis: mean $\pm$ std of objective value (30 runs, 1000 iterations each).

Algorithm	Init	Pop = 50	Pop = 100	Pop = 150
AE-QTS	DK	-12.00 ± 14.58	-41.89 ± 8.30	-49.41 ± 5.02
GA	DK	-51.22 ± 16.71	-40.17 ± 42.26	-55.28 ± 19.84
DE	DK	451.20 ± 173.21	696.41 ± 137.88	701.17 ± 137.48
PSO	DK	290.48 ± 66.68	230.80 ± 50.16	187.79 ± 53.43
AE-QTS	Uniform	147.22 ± 46.46	68.15 ± 34.72	41.97 ± 28.91
GA	Uniform	513.57 ± 88.05	597.26 ± 109.52	595.52 ± 100.98
DE	Uniform	1400.51 ± 187.91	1436.60 ± 95.54	1461.90 ± 82.61
PSO	Uniform	429.76 ± 78.60	337.09 ± 72.08	265.50 ± 55.01

Under unstructured conditions, the differences between optimization methods become more pronounced. AE-QTS-Uniform consistently improves as population size increases and achieves the best overall performance. PSO-Uniform also benefits from larger populations and maintains moderate robustness, while GA-Uniform rapidly stagnates with limited improvement across different population sizes. DE-Uniform exhibits the weakest scalability and stability among all evaluated methods.

We can again see that exploration-driven optimization methods are more robust to varying population sizes under unstructured conditions, whereas exploitation-driven approaches remain more dependent on initialization quality.

4.4 Convergence Limits under Unstructured Conditions

Long-horizon behavior under unstructured conditions is further evaluated using uniform initialization, as shown in Fig. 3. AE-QTS rapidly converges toward near-zero objective values and maintains complete critical isolation with zero WS-IoT mixing. The example segmentation structure in Fig. 4 further shows clear functional grouping, including server grouping, infrastructure consolidation, and guest-device isolation. PSO also improves over time but converges more slowly, whereas GA and DE stagnate at higher objective values and tend to produce more fragmented structures. These results further confirm that probabilistic exploration-driven optimization is more resilient under cold-start segmentation scenarios.

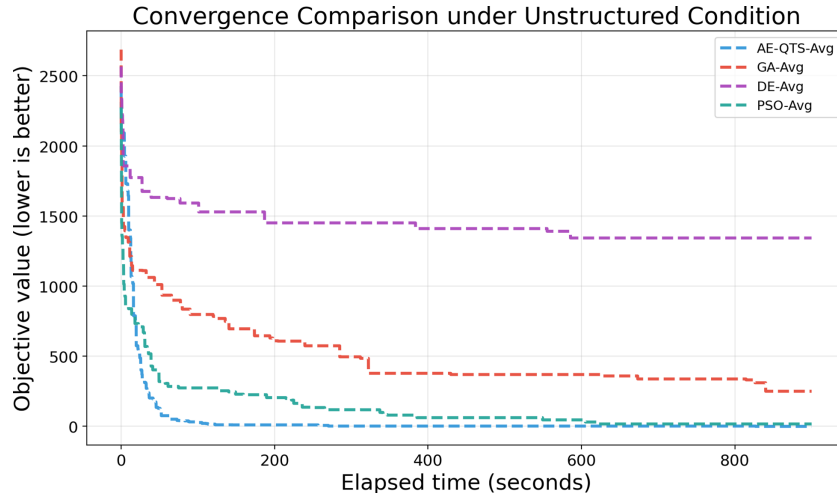


Figure 3: Long-horizon convergence under unstructured initialization (15 min).

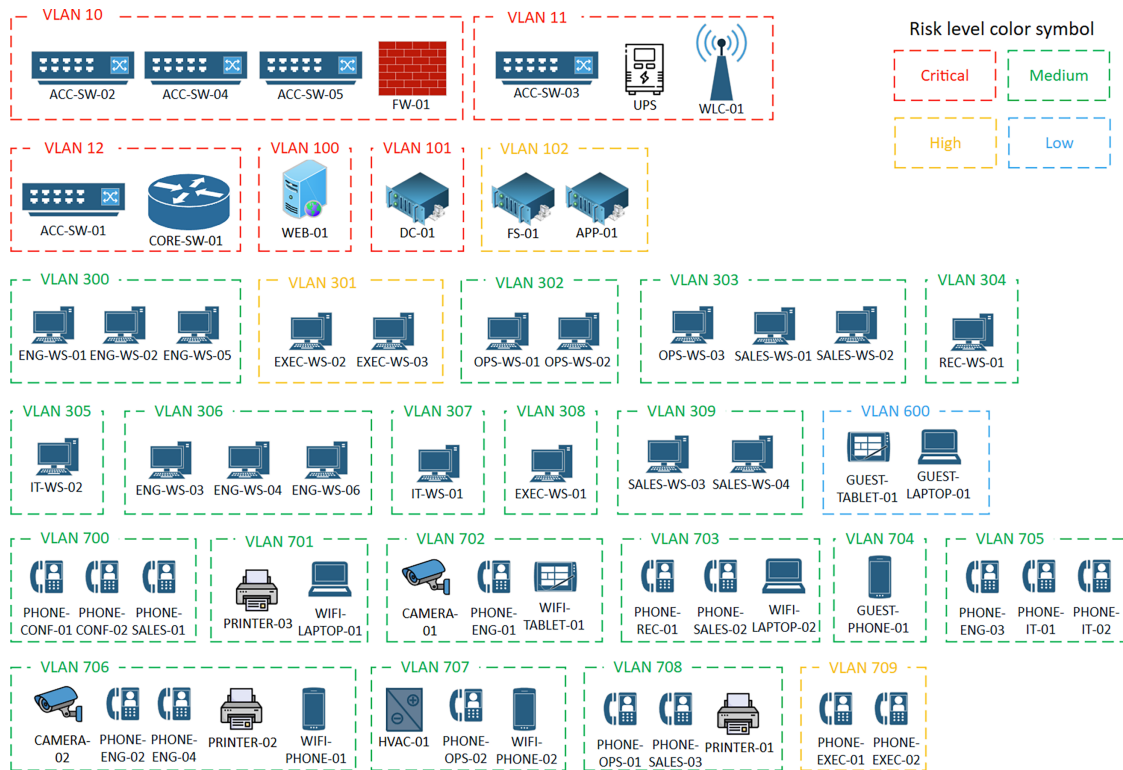


Figure 4: Example of VLAN segmentation structure under unstructured initialization (AE-QTS-uniform).

The result indicate that probabilistic exploration-driven optimization demonstrates stronger robustness and more stable convergence behavior under unstructured conditions than exploitation and differential-perturbation-based approaches, which are more susceptible to stagnation in highly fragmented local optima.

5 Discussion

The experimental results suggest that the effectiveness of optimization methods for network microsegmentation is strongly influenced by initialization conditions. Rather than exhibiting uniform behavior across all scenarios, the evaluated methods demonstrate distinct characteristics under knowledge-guided and cold-start environments.

Exploitation-oriented methods, particularly GA, benefit directly from informative initialization and are able to converge to high-quality solutions within limited optimization time. However, their performance degrades as initialization quality decreases. In contrast, exploration-oriented methods, especially AE-QTS, maintain more stable optimization behavior under unstructured conditions by preserving search diversity and progressively refining probabilistic search distributions. PSO also demonstrates moderate robustness under similar conditions, whereas DE exhibits comparatively weaker performance in high-cardinality VLAN assignment problems.

The fragmentation analysis further indicates that objective values alone are insufficient to characterize segmentation quality. Solutions with competitive objective values may still produce highly fragmented segmentation structures that increase operational complexity and reduce manageability. Under unstructured conditions, AE-QTS generally produces more coherent segmentation structures, whereas GA and DE are more susceptible to fragmentation in the absence of structural guidance. These observations suggest that fragmentation should be considered an important complementary evaluation dimension in practical microsegmentation optimization.

The results also indicate that no single optimization method is universally optimal across all scenarios. Exploitation-oriented approaches perform more effectively when prior knowledge is available, whereas exploration-oriented methods become more advantageous under weakly structured conditions. This observation suggests that adaptive optimization frameworks capable of dynamically selecting optimization strategies according to initialization conditions may represent a promising direction for future research.

Several limitations should also be acknowledged. The experiments are conducted on one SME-scale enterprise topology, and additional validation is still required for larger and more heterogeneous enterprise environments. Furthermore, although the proposed fragmentation metric captures important structural characteristics, additional operational considerations, such as policy deployment complexity, real-time adaptability, and network dynamics, remain outside the scope of this study.

6 Conclusion

In this paper, network microsegmentation is formulated as a state-dependent combinatorial optimization problem, and the influence of initialization conditions on optimization behavior is systematically analyzed. The comparative analysis of GA, DE, PSO, and AE-QTS under both structured and unstructured conditions shows that optimization performance depends strongly on the availability of structural guidance. The results show that exploitation-oriented methods perform effectively when informative initialization is available, whereas exploration-oriented methods provide greater stability under weakly structured conditions. This contrast confirms that algorithm selection should account for the availability of prior structural knowledge rather than relying on a single optimization method.

Beyond objective-based evaluation, the fragmentation analysis further indicates that segmentation quality cannot be characterized by objective values alone. High-quality microsegmentation should balance isolation effectiveness with structural coherence while avoiding excessive fragmentation. Under unstructured conditions, competitive objective values do not necessarily correspond to practically manageable

segmentation structures. These observations suggest that fragmentation provides an important complementary evaluation dimension for real-world deployment scenarios.

Future work includes extending the evaluation to larger-scale and more heterogeneous network environments and incorporating additional operational considerations, including dynamic policy adaptation and real-time network reconfiguration. Overall, the findings suggest that no single optimization paradigm is universally optimal across all initialization conditions. Instead, different optimization methods exhibit different strengths depending on the availability of prior structural knowledge. These observations may support future research on adaptive or hybrid optimization frameworks capable of dynamically selecting or combining optimization strategies according to deployment conditions.

Acknowledgement: We acknowledge the use of ChatGPT (<https://chat.openai.com/>) during manuscript preparation solely for language polishing, grammar correction, and wording refinement. All scientific content, experimental design, data analysis, interpretation of results, and conclusions were developed and verified entirely by the authors.

Funding Statement: This research was funded in part by the National Science and Technology Council (NSTC), Taiwan, R.O.C., under Grant NSTC 114-2221-E-197-010-.

Author Contributions: The authors confirm contribution to the paper as follows: Conceptualization, all authors; methodology, all authors; software, Hong-Sheng Wang; validation, Kuo-Chun Tseng; formal analysis, Reen-Cheng Wang; investigation, Kuo-Chun Tseng; resources, Kuo-Chun Tseng; data curation, Reen-Cheng Wang and Hong-Sheng Wang; writing—original draft preparation, Reen-Cheng Wang and Hong-Sheng Wang; writing—review and editing, Reen-Cheng Wang; visualization, Reen-Cheng Wang and Hong-Sheng Wang; supervision, Kuo-Chun Tseng; project administration, Kuo-Chun Tseng; funding acquisition, Kuo-Chun Tseng. All authors reviewed and approved the final version of the manuscript.

Availability of Data and Materials: https://github.com/hongyui/GA_AE-QTS_Microsegmentation.

Ethics Approval: Not applicable.

Conflicts of Interest: The authors declare no conflicts of interest.

References

1. Kindervag J. Build security into your network's DNA: the zero trust network architecture. Cambridge, MA, USA: Forrester Research Inc.; 2010.
2. Ward R, Beyer B. BeyondCorp: a new approach to enterprise security. *Login*. 2014;39(6):6–11.
3. Rose S, Borchert O, Mitchell S, Connelly S. Zero trust architecture. Gaithersburg, MD, USA: National Institute of Standards and Technology (NIST); 2020.
4. Cybersecurity and Infrastructure Security Agency (CISA). Zero trust maturity model, version 2.0. Washington, DC, USA: CISA; 2023.
5. Harshavardini S, Bertia A. A software-defined zero trust framework for secure access control and microsegmentation using SDN and SDP. In: *Proceedings of the 2025 Fourth International Conference on Smart Technologies, Communication and Robotics (STCR)*; 2025 May 9–10; Sathyamangalam, India. p. 1–7. doi:10.1109/STCR62650.2025.11020089.
6. IEEE 802.1Q-2018. IEEE standard for local and metropolitan area networks—bridges and bridged networks. New York, NY, USA: IEEE; 2018.
7. Shaik A. A survey of emerging techniques for large networks of virtual local area networks (VLANs) with benefits and limitations. *Int J Curr Eng Technol*. 2024;14(6):478–86. doi:10.14741/ijcet/v.14.6.11.
8. Parfenov D, Grishina L, Zhigalov A, Bolodurina I. Research of genetic optimization algorithms in the design of VLAN. In: *Proceedings of the 2021 29th Telecommunications Forum (TELFOR)*; 2021 Nov 23–24; Belgrade, Serbia. p. 1–4. doi:10.1109/TELFOR52709.2021.9653295.

9. Ai H, Fan Y, Zhang J, Ghafoor KZ. Topology optimization of computer communication network based on improved genetic algorithm. *J Intell Syst.* 2022;31(1):651–9. doi:10.1515/jisys-2022-0050.
10. Wang HS, Wang RC. Low-capital expenditure AI-assisted zero-trust control plane for brownfield ethernet environments. *Eng Proc.* 2025;120(1):54. doi:10.3390/engproc2025120054.
11. Vlašić I, Đurasević M, Jakobović D. Improving genetic algorithm performance by population initialisation with dispatching rules. *Comput Ind Eng.* 2019;137:106030. doi:10.1016/j.cie.2019.106030.
12. Storn R, Price K. Differential evolution—a simple and efficient heuristic for global optimization over continuous spaces. *J Glob Optim.* 1997;11(4):341–59. doi:10.1023/A:1008202821328.
13. Kennedy J, Eberhart R. Particle swarm optimization. In: *Proceedings of ICNN'95—International Conference on Neural Networks*; 1995 Nov 27–Dec 1; Perth, WA, Australia. p. 1942–8. doi:10.1109/ICNN.1995.488968.
14. Han KH, Kim JH. Quantum-inspired evolutionary algorithm for a class of combinatorial optimization. *IEEE Trans Evol Comput.* 2002;6(6):580–93. doi:10.1109/TEVC.2002.804320.
15. Chou YH, Yang YJ, Chiu CH. Classical and quantum-inspired tabu search for solving 0/1 knapsack problem. In: *Proceedings of the 2011 IEEE International Conference on Systems, Man, and Cybernetics*; 2011 Oct 9–12; Anchorage, AK, USA. p. 1364–9. doi:10.1109/icsmc.2011.6083848.
16. Tseng KC, Lai WC, Chen IC, Hsiao YH, Chiue JY, Huang WC. Amplitude-ensemble quantum-inspired tabu search algorithm for solving 0/1 knapsack problems. In: *Proceedings of the 2024 IEEE International Conference on Systems, Man, and Cybernetics (SMC)*; 2024 Oct 6–10; Kuching, Malaysia. p. 718–25. doi:10.1109/SMC54092.2024.10831250.
17. Schein AI, Popescul A, Ungar LH, Pennock DM. Methods and metrics for cold-start recommendations. In: *Proceedings of the 25th Annual International ACM SIGIR Conference on Research and Development in Information Retrieval*; 2002 Aug 11–15; Tampere, Finland. p. 253–60. doi:10.1145/564376.564421.
18. Vomberg A, Schauerte N, Krakowski S, Ingram Bogusz C, Gijsenberg MJ, Bleier A. The cold-start problem in nascent AI strategy: kickstarting data network effects. *J Bus Res.* 2023;168(1):114236. doi:10.1016/j.jbusres.2023.114236.
19. Basta N, Ikram M, Ali Kaafar M, Walker A. Towards a zero-trust micro-segmentation network security strategy: An evaluation framework. In: *Proceedings of the NOMS 2022—2022 IEEE/IFIP Network Operations and Management Symposium*; 2022 Apr 25–29; Budapest, Hungary. p. 1–7. doi:10.1109/NOMS54207.2022.9789888.