Sensors and Materials, Vol. 38, No. 9 (2026) 5031–5050
MYU Tokyo

S & M 4609

Layout Optimization of Tunnel Foundation Pseudo-satellite Based on Zone-adaptive Precision Genetic Algorithm

Chengling Cai,[1] Zhouwang Yang,[1] Kaihui Lv,[2] Kun Xie,[3] and Li Shaohui[1*]

[1]College of Automation and Electronic Information, Xiangtan University, Xiangtan 411105, P. R. China
[2]Xiangtan University, College of Mathatics and Computational Science, Xiangtan 411105, P. R. China
[3]Xiangtan University, College of Material Science and Engineering, Xiangtan 411105, P. R. China

(Received October 14, 2025; accepted February 3, 2026)

Keywords: genetic algorithm, pseudo-satellite layout optimization, *PDOP*, tunnel layout

To address positioning failures caused by signal obstruction in tunnel environments for Global Navigation Satellite Systems, we propose a pseudo-satellite layout optimization method based on the Zone-adaptive Precision Genetic Algorithm (ZAP-GA). First, we constructed a tunnel simulation model and sampled the entire tunnel grid. Then, we evaluated the quality of pseudo-satellite layouts using the average position dilution of precision (*PDOP*) and coverage rate as key metrics at each sampling point. Second, the ZAP-GA partitioning method provided a uniform coverage of the pseudo-satellite and made adaptive adjustments for variable stride lengths and edge regions. At the same time, the fitness function was optimized precisely by balancing the weighted *PDOP* and the spacing of the pseudo-satellite with the configuration of the mounting surface. Optimization was finally achieved by integrating tournament selection, single-satellite gene segment crossing, and a robust elite retention mechanism. Experimental results demonstrate that the proposed method achieves an average *PDOP* of 2.7326 across all sampling points with only eight pseudo-satellites, accompanied by 95.82% overall regional coverage and full coverage in the core area. These performance metrics fully meet the requirements for precise positioning in tunnel scenarios.

1. Introduction

The Global Navigation Satellite System (GNSS) is widely used in real-world scenarios. However, terrain can significantly impact positioning accuracy in certain environments. For example, satellite signals are difficult to receive in obstructed settings such as tunnels, mountainous regions, and urban canyons. This can result in poor positioning accuracy and may even lead to a complete loss of positioning capability.(1,2) GNSS signals are completely unavailable in narrow, enclosed environments such as highway tunnels and mine shafts. However, vehicles and personnel within tunnels require stringent positioning accuracy and continuity. Pseudosatellite systems can achieve “GNSS-independent” high-precision positioning through their own networking technology. This makes them capable of meeting positioning

*Corresponding author: e-mail: 202421623293@smail.xtu.edu.cn
https://doi.org/10.18494/SAM5972

ISSN 0914-4935

requirements in scenarios such as tunnels.[3] The layout of a pseudo-satellite has a direct impact on positioning accuracy and signal coverage. Therefore, optimizing the deployment of pseudo-satellites in tunnel environments is critical to achieving high-precision positioning in long, narrow, and enclosed spaces.

The evaluation criteria for pseudo-satellite layouts are accuracy and signal coverage. These factors have been extensively studied by numerous scholars. Peng *et al.*[4] demonstrated that geometric dilution of precision (*GDOP*) is inversely proportional to the volume of the tetrahedron formed by the observed pseudolocators. However, the algorithm verification framework is basic, and its robustness has not been validated. Zhao *et al.*[5] proposed a genetic-algorithm-based indoor pseudo-satellite layout method that considers constellation integrity. However, the theoretical boundaries of geometric optimization remain unclear. Yang *et al.*[6] proposed a layout method for satellites in geometric configurations (*GDOP*), demonstrating that increasing the number of satellites significantly improves the configuration and positioning accuracy. However, the layout is not sufficiently robust to withstand extreme space environments. Liang *et al.*[7] optimized positioning accuracy through an indoor pseudo-satellite point layout and multi-objective coordination using the Multi-objective Particle Swarm Optimization Algorithm (MG-MOPSO). However, their algorithm exhibits poor adaptability to indoor environmental heterogeneity. Zhao *et al.*[8] proposed a synchronous pseudo-satellite layout scheme based on the weighted horizontal dilution of precision (*HDOP*) to improve layout performance by optimizing *HDOP*. However, the vertical positioning accuracy remains insufficiently optimized. Shangguan *et al.*[9] proposed a pseudo-satellite layout method based on genetic algorithms (GAs). They used the average position dilution of precision (*PDOP*) as the objective function, but did not consider the synergistic effects of coverage. Shao *et al.*[10] proposed a layout strategy for a pseudo-satellite system based on the particle swarm optimization algorithm. However, they did not incorporate core positioning accuracy metrics, such as *GDOP*, for further optimization. Ma *et al.*[11] proposed an improved specular reflection optimization algorithm that effectively mitigates blind spots in regional coverage. However, it did not incorporate energy consumption control for multi-objective optimization. Chen and Yang[12] proposed a counter-flow algorithm that incorporates invasive weed strategies and can achieve uniform distribution through node allocation, although its coverage adaptability has not been verified. Liu *et al.*[13] proposed a multi-objective site selection method based on the non-dominated sorting genetic algorithm-II (NSGA-II). However, as the NSGA-II algorithm was not modified to incorporate the characteristics of pseudo-satellite-based site selection, the initial population was of low quality. Li *et al.*[14] proposed a method for optimizing pseudo-satellite grid coverage that enhances uniformity through grid refinement. However, if the grid granularity does not match the radius of the pseudo-satellite signal coverage, this can result in local redundancy or blind spots in the coverage. Sun *et al.*[15] proposed a coverage optimization algorithm that combines genetic and whale algorithms. This effectively balances global and local optimization performance while accelerating convergence. However, the algorithm tends to converge prematurely and become trapped in local optima. Wang *et al.*[16] divided the tunnel into fixed sections and constrained the number of pseudolites in each section. Whilst equidistant deployment is uncomplicated to implement, it tends to overlook multidimensional constraints in tunnel scenarios, such as engineering installation feasibility, signal propagation characteristics, and *PDOP* balance. This approach, however, does not address issues such as uneven positioning accuracy and insufficient coverage caused by edge effects, obstacle obstructions, or structural variations. Existing methods generally have two shortcomings. First, optimization algorithms do not employ adaptive strategies, which makes balancing convergence speed and accuracy difficult. Second, they often focus solely on DOP extremes or coverage rates, neglecting the global equilibrium between the DOP extremes and the algorithm itself.

To address the aforementioned issues, we propose a pseudo-satellite optimization layout method based on the Zone-adaptive Precision Genetic Algorithm (ZAP-GA). The aim is to achieve a high-precision layout for tunnel pseudo-satellites. A barrier-free tunnel model with the following dimensions was constructed: 200 m (length) × 6 m (width) × 5 m (height). Meta-heuristic algorithms, represented by GAs, possess robust global optimization capabilities and constraint adaptation abilities, enabling the integrated optimization of "feasibility-accuracy-coverage" in pseudo-satellite layout. We proposed optimizing the partitioning of tunnels based on the number of pseudo-satellites to prevent clustering and enhance coverage uniformity. We designed an adaptive adjustment mechanism that balances global exploration and local optimization by varying the step size and adapting the regional weight. ZAP-GA achieves precise optimization control through constraints by establishing a precision optimization control mechanism and incorporating engineering constraint penalties and fitness function directionality. We conducted simulation experiments to validate the proposed method by comparing it with conventional GAs and cooperative whale optimization algorithms (CWOAs), thereby demonstrating its superiority. Finally, we generated a three-dimensional visualization of the layout plan to offer a novel solution for the practical installation of tunnel pseudo-satellites.

2. ZAP-GA for Tunnel-oriented Pseudo-satellite Layout

2.1 Partition optimization and installation of a surface-prioritized population initialization

Equal-interval deployment is a traditional layout method that offers operational simplicity, but it has significant limitations in tunnel scenarios. First, it only considers linear spacing, ignoring engineering constraints such as the availability of the installation surface, electromagnetic interference, and train clearance limits. This makes implementation challenging. Second, it fails to optimize the global distribution of *PDOP*, resulting in poor geometric configurations and weak signals at the edges of tunnels. Accuracy is much lower in these peripheral areas than in core zones, creating positioning blind spots. Finally, it lacks scalability in different scenarios. In complex environments such as curved or obstructed tunnels, it cannot adapt flexibly and loses positioning effectiveness immediately.

Genetic algorithms are often employed for optimization in traditional layouts. However, randomly generated initial populations tend to exceed the spatial constraints of the tunnel, resulting in poor convergence and a tendency to become trapped in local optima. ZAP-GA divides the tunnel to enable the population size to be initialized in a targeted manner, thereby avoiding the excessive number of invalid individuals caused by random population initialization. The area is divided into two zones: edge zone and remaining zone. The edge zone comprises the areas on either side of the tunnel, with one pseudo-satellite installed in each. The remaining zone is subdivided further on the basis of the number of pseudo-satellites. The edge zone is the primary optimization area and is set to 80% of the remaining zone's length to prevent the satellites from being very close to the sides, which can result in a poor satellite structure. The lengths of the other regions can normally be divided according to the number of satellites.

In the confined space of a tunnel, there are only three possible locations for installing satellites: the two side walls and the ceiling. As shown in Table 1, each satellite corresponds to three-dimensional parameters.

We deployed pseudo-satellites at the ends of the tunnel with a 60% probability on either side and a 40% probability on the ceiling. This mitigates the issue of satellite signal obstruction from the ceiling, which is prone to occurring in edge areas, while significantly improving connectivity between satellites and ground sampling points. Other zones are deployed randomly as usual to maintain diversity in the geometric layout.

2.2 Weighted *PDOP* and constraint fusion

2.2.1 Dilution of precision

In satellite navigation and pseudolocation systems, positioning errors are primarily dependent on noise, time synchronization errors, range effects, multipath effects, and satellite geometry configurations.(17) The dilution of precision (DOP) is a key metric for evaluating pseudolocation configurations, with a smaller DOP indicating a higher positioning accuracy.

The magnitude of DOP is derived from the covariance matrix of the positioning solution. Let the estimation error of the sampling position and time be $\Delta X = [\Delta x, \Delta y, \Delta z, \Delta t]^T$. The observation matrix $\boldsymbol{H}$ consists of the direction cosines and time terms of n pseudo-satellites:

$$\boldsymbol{H} = \begin{bmatrix} a_1^x & a_1^y & a_1^z & 1 \\ a_2^x & a_2^y & a_2^z & 1 \\ \vdots & \vdots & \vdots & \vdots \\ a_n^x & a_n^y & a_n^z & 1 \end{bmatrix}. \tag{1}$$

In this equation, $\left(a_i^x, a_i^y, a_i^z\right)$ denotes the direction cosine of the ith pseudo-satellite relative to the sampling point and 1 represents the time error term.

Table 1
Code rules for pseudo-satellites.

Gene index	Physical meaning	Value range	Decoding logic
3k-2	*X*-coordinate area ratio	[0,1]	The *X* range is mapped to the kth region.
3k-1	*Y*/*Z* coordinate scale	[0,1]	Based on the mounting surface, determine the actual *Y*/*Z* values.
3k	Mounting surface type	{1,2,3}	1 = top, 2 = left, 3 = right

The covariance matrix of positioning errors is

$$\operatorname{cov}\left(\Delta X\right) = (\boldsymbol{H}^T \boldsymbol{H})^{-1} \cdot \sigma^2 = \boldsymbol{D} \cdot \sigma^2. \tag{2}$$

In this equation, σ^2 denotes the unit measurement error, $\boldsymbol{D} = (\boldsymbol{H}^T\boldsymbol{H})^{-1}$ represents the geometric covariance matrix, and D_{11}, D_{22}, D_{33}, amd D_{44} correspond to the covariance components of positioning errors in the *x*-, *y*-, and *z*-directions, as well as temporal errors.

The common DOP includes

$$GDOP = \sqrt{D_{11} + D_{22} + D_{33} + D_{44}}, \tag{3}$$

$$PDOP = \sqrt{D_{11} + D_{22} + D_{33}}, \tag{4}$$

$$HDOP = \sqrt{D_{11} + D_{22}}. \tag{5}$$

The geometric DOP is represented by *GDOP*, the positioning DOP by *PDOP*, and the horizontal DOP by *HDOP*. In this study, we focus on tunnels, which are confined and elongated 3D spaces. Although sampling points are only deployed on the *x*-*y* plane (tunnel ground), pseudo-satellites can be positioned on the ceiling and both side walls. Therefore, positioning accuracy must be considered in all three directions. Concurrently, the time synchronization error in the pseudo-satellite experiment is negligible ($D_{44} \approx 0$): In the simulation model, it is assumed that the pseudo-satellite system possesses a unified clock source, i.e., all station clocks are fully synchronized. In practical engineering applications, this objective can be realized through the implementation of high-precision GNSS timing modules or the IEEE 1588v2 Precision Time Protocol (PTP), which is based on fiber optics. This ensures that clock differences between pseudo-satellites are controlled within the nanosecond range. At this level of synchronization, the equivalent distance error introduced by timing asynchrony ranges from centimeters to millimeters. This is far smaller than the positioning error dominated by satellite geometry configuration (i.e., *PDOP*). Consequently, within the context of this study, which is focused on optimizing the geometric layout, the impact of timing synchronization error can be regarded as a secondary factor. Consequently, *HDOP* and *GDOP* are not applicable, and *PDOP* is adopted as the experimental evaluation criterion.

2.2.2 Weighted *PDOP* and constraint fusion

When all sampling points are measured using a standard *PDOP*, each point carries equal weight. However, in the marginal areas of narrow, elongated spaces such as tunnels, satellite signals are weak and the geometric configuration is poor, which often results in reduced positioning accuracy. To prioritize the optimization of these peripheral areas, a method of calculating a weighted DOP is employed. This approach strengthens the optimization priority of edge regions by allocating different weights.

We introduce the weighting matrix as:

$$\boldsymbol{W} = \mathrm{diag}\left(w_1, w_2, \ldots, w_n\right). \tag{6}$$

In this equation, w_i denotes the weight of the ith sampling point. The weighted covariance matrix is modified as

$$Q = \left(\boldsymbol{H}^T \boldsymbol{W}^{-1} \boldsymbol{H}\right)^{-1}. \tag{7}$$

The weighted positioning DOP (*WPDOP*) is defined as

$$WPDOP = \sqrt{Q_{11} + Q_{22} + Q_{33}}. \tag{8}$$

The transition from standard *PDOP* to weighted *PDOP* adjusts the optimization priorities across different zones. This is achieved by assigning distinct weights to each area. For example, increasing the weighting of peripheral zones encourages the algorithm to optimize the geometric layout in those regions, while applying a lower weighting to core zones prevents excessive optimization, which can result in the waste of resources. The weighted precision factor calculation method preserves the original *PDOP*'s sensitivity to geometric configuration while assigning different weights to reflect the importance of each area in the optimization process. Ultimately, this approach yields a geometric layout that better aligns with actual engineering requirements.

The algorithm uses weighted *PDOP* and constrained fusion as its fitness function. This allows issues such as low accuracy in fringe areas and distribution conflicts between pseudo-satellites to be accurately reflected by weighted *PDOP* and pseudo-satellite distribution penalties. The fitness function is

$$f = \left(\frac{\sum_{i=1}^{N} w_i \cdot WPDOP_i}{\sum_{i=1}^{N} w_i}\right) \times \left(1 + \varnothing\right). \tag{9}$$

The weighting coefficients w_i in the formula are

$$w_i = \begin{cases} 1.6, & \text{Edge region sampling points} \\ 1.0. & \text{Sampling points in core zone} \left(10 - 190 \text{ m}\right) \end{cases} \tag{10}$$

To achieve positioning across the entire area, the algorithm places considerable emphasis on peripheral regions with lower *PDOP* accuracy during iteration, thereby balancing the distribution of precision throughout the tunnel. w_i is determined by combining a theoretical analysis of the characteristics of the tunnel signal with an experimental determination of the system control variables. Edge zones (0–10 m and 190–200 m) are prone to signal attenuation and have poor geometric configurations. Therefore, they are assigned a higher weight (w_i = 1.6) to prioritize optimization. The core region (10–190 m) exhibits stable signal conditions and has a weight of 1.0 assigned to it to achieve an overall balance in accuracy. Through multiple experimental sets with w_i in the range [1.2, 2.0], w_i = 1.6 was found to achieve the optimal balance between the reduced edge *PDOP* and stability of the core region.

To prevent signal interference caused by the proximity of the pseudo-satellite, a "penalty" term is introduced. The distribution penalty coefficient $(1+\varnothing)$ represents the engineering constraints for the layout of the pseudo-satellite, comprising the penalty for pseudo-satellite spacing ($\varnothing_1$) and the penalty for mounting surface uniformity ($\varnothing_2$).

If the distance between two pseudo-satellites falls below a predetermined threshold, a penalty is applied according to the rule that smaller distances incur greater penalties:

$$\varnothing_1 = \sum_{i<j} \frac{p - d_{ij}}{p} \times k. \tag{11}$$

In this formula, d_{ij} represents the distance between pseudo-satellites i and j; p denotes the threshold for separation between the two pseudo-satellites; and k is the penalty coefficient. The spacing threshold p and penalty coefficient k are optimized on the basis of engineering constraints and signal coverage characteristics. In the 6-m-wide edge zone, where space is limited, p = 2.5 m is adopted to prevent signal interference. In the core zone, where layout flexibility is greater, p = 3.5 m is selected to balance coverage and efficiency. The core zone is a critical positioning area, where excessive proximity has a significant impact on geometric configurations and signal interference. This warrants a strict penalty of k = 3.0. In the edge zone, however, limited spatial constraints mean that overly stringent penalties can compromise deployment flexibility; thus, k = 1.5 strikes a balance between constraints and feasibility. Orthogonal experiments ($k \in [1.0, 4.0]$) validate that this approach suppresses clustering while ensuring stable positioning accuracy.

Pseudo-satellites require relatively uniform deployment. If they are concentrated on a single surface, a uniformity penalty must be applied to the installation based on the degree of deviation from a uniform distribution:

$$\varnothing_2 = \sum_{f=1}^{3} \left(\frac{n_f}{N_S} - \frac{1}{3} \right)^2 \times 1.5. \tag{12}$$

In this formula, N_S denotes the total number of pseudo-satellite, and n_f represents the number of pseudo-satellites on the fth installation plane. Ideally, this number should equal the total number of satellites divided by three; 1.5 is the overlay penalty coefficient. The uniformity penalty coefficient of 1.5 for the mounting surface was validated through multiple comparative experiments. After testing various coefficients within the range of 0.8 to 2.0, it was found that a value of 1.5 effectively constrains the uneven distribution of pseudo-satellites across the top, left, and right surfaces. At the same time, it allows for reasonable deviations to maintain an optimal geometric configuration, thereby avoiding performance degradation caused by excessive penalties. Ultimately, $\varnothing$ is the sum of Eqs. (11) and (12):

$$\varnothing = \varnothing_1 + \varnothing_2. \tag{13}$$

Equation (9) addresses the issue of sacrificing accuracy in edge regions while preserving other areas. It does this by strengthening edge region weights and applying reasonable constraint penalties. This approach ensures that edge regions are not treated with the same weight as other areas within tunnels, which would otherwise compromise their precision. It also resolves the issue of pseudo-satellites failing to deploy owing to spacing limitations between them and the installation surface, and the fact that installation surface deployment is not considered. This ensures that the algorithm transcends theoretical constraints and aligns more closely with engineering realities, thereby becoming a practical and feasible solution.

2.3 Adaptive regulation mechanism and scene-adaptive genetic operations

Conventional GAs tend to converge prematurely following optimization through inheritance, crossover, and mutation, resulting in local optima. We first used tournament selection with $k = 4$ to replace roulette wheel selection. Four individuals were randomly selected from the initial population and the individual with the smallest *PDOP* was chosen to enter the parent generation. This approach maintains population diversity by preventing individuals with similar fitness values from being eliminated, thereby reducing the risk of premature convergence.

We used a single pseudo-satellite for gene segment crossover. This involved a positional exchange between the three genes of the kth satellite and the three genes of the $(k + 1)$th satellite, while keeping the genes of the other satellites unchanged. This crossover method employs the complete gene segment (three genes) of a single pseudo-satellite as its fundamental unit, thereby circumventing the issues associated with standard multi-point crossovers' random gene splitting. The latter approach may result in layout irregularities, such as mismatches between installation surfaces and coordinates, or violations of partition constraints. The exchange of complete gene segments is imperative to ensure that each pseudo-satellite's layout information complies with coding rules and tunnel scenario constraints. This, in turn, safeguards population validity.

We ensured global search capability during early iterations and localized fine-tuning in later stages by employing classification-based and adaptive mutation logic. The mutation step size is calculated using a linearly decreasing formula:

$$s = 0.12 - 0.10 \times \frac{c}{m}. \tag{14}$$

In this formula, c represents the current iteration count, m denotes the maximum iteration count, and 0.12 signifies the initial step size. Initially, a large step size is employed to identify optimal layouts in peripheral regions, whereas a small step size is adopted later to refine the core areas with precision. The initial mutation step size (0.12) and decay coefficient (0.10) were optimized for a tunnel length of 200 m and 100 iterations. Validation across the stride range of 0.08–0.16 confirms that a stride of 0.12 ensures global exploration capability during the early iterations, covering high-quality layouts in peripheral areas, while gradually transitioning to local fine-tuning in the later iterations (stride ≈ 0.02). This effectively prevents premature convergence.

To prevent invalid variations where random perturbations result in noninteger values and to avoid the special case where mounting surfaces become overly concentrated on a single face, we randomly varied the mounting surface to other surfaces. We introduced the Gaussian random number *randn* into the variation of pseudo-satellite position parameters:

$$\alpha_i^j = \alpha_i^j + s \times randn. \tag{15}$$

In this formula, α_i^j represents the jth gene of the ith pseudo-satellite. The Gaussian distribution restricts the magnitude of the perturbation within the stride range, thereby preventing extreme values that can disrupt the rationality of the layout.

2.4 Strong elite retention mechanism

We retained the superior genes from the parent generation. We calculated the number of genes as

$$e = \max\left(5, \left[r \cdot N + 0.5\right]\right), \tag{16}$$

where r represents the elite ratio ($0 < r < 1$), N denotes the population size, and the addition of 0.5 accounts for rounding $r \cdot N$ to the nearest integer. This ensures that at least five elites are retained throughout the entire iterative process of the algorithm, guaranteeing stable convergence and preventing inherent degradation and oscillation around the optimal solution.

2.5 ZAP-GA: overall process

Figure 1 shows the execution flow of ZAP-GA described in this paper. First, we construct the tunnel scenario and complete the partition optimization, while simultaneously initializing the pseudo-satellite layout population and generating a set of qualified initial pseudo-satellite layouts. Next, *WPDOP* is calculated using Eq. (8), and the constraint penalty term is computed using Eq. (13). Summing these two components derives the algorithm's fitness function, which quantifies the relative merits of each pseudo-satellite layout scheme. Then, genetic operations are performed on each pseudo-satellite layout scheme. The parent population is filtered through tournament selection, and crossover operations are applied to the single pseudo-satellite gene segment. Adaptive mutation is then conducted to retain individuals with strong genes. This ensures that high-quality layouts are not lost during the iterative process. Finally, we determine

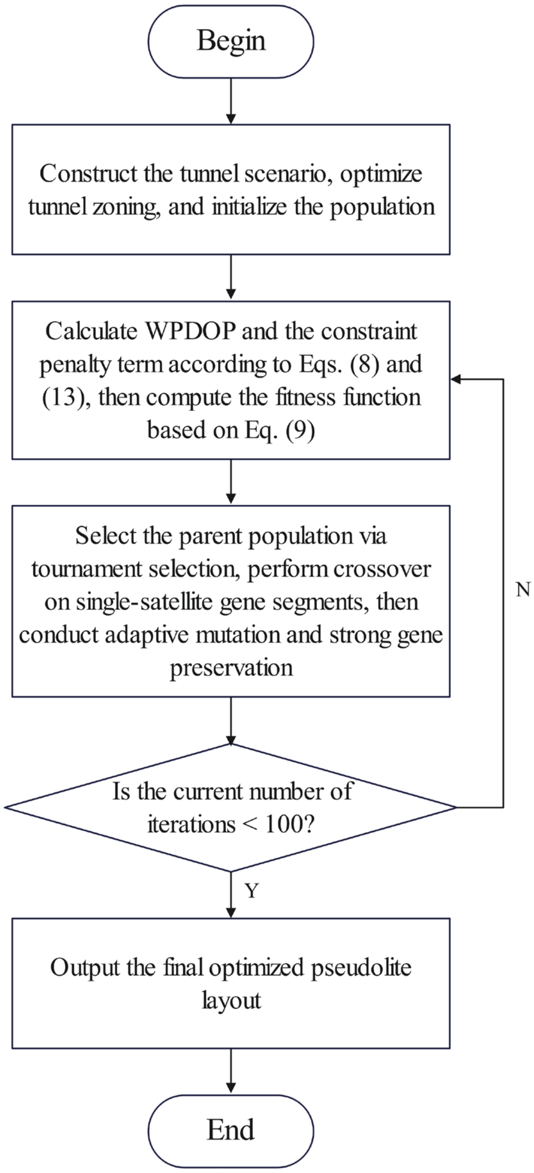

Fig. 1. ZAP-GA flowchart.

whether the current iteration count has reached the set threshold. If not, we return to Step 2 to continue the optimization. If the threshold has been reached, we output the final optimized pseudo-satellite layout.

Unlike conventional GAs, ZAP-GA performs scenario-constrained embedding on the initial population. This prevents randomly generated populations from easily breaking through regional constraints from the outset. It also addresses the issue of significant variations in *PDOP* across different regions of the tunnel, which is a common challenge for standard GAs, by employing a specific fitness function. Its precise genetic optimization overcomes the slow convergence and tendency to become trapped in local optima that plague conventional GAs. ZAP-GA outperforms conventional GAs in terms of constraint satisfaction capability, accuracy, and convergence efficiency, owing to targeted improvements in the pseudo-satellite layout .

3. Simulation Experiments and Validation

3.1 Evaluation criteria for the advantages and disadvantages of pseudo-satellite layouts

In simulations of ZAP-GA for the pseudo-satellite layout in narrow tunnels, the average *PDOP* obtained at each sampling point following the final iteration of the optimization process is a key metric for evaluating the layout of the pseudo-satellite. Coverage is another metric used to evaluate the quality of the satellite layout. In tunnel environments, coverage is defined as the proportion of sampling points that meet "reliable positioning conditions." The coverage defined herein is the ratio of sampling points that simultaneously satisfy the observed pseudo-satellite count ≥4 and *PDOP* ≤5 to all sampling points. The specific coverage calculation formula is

$$C = \frac{N_v}{N_t} \times 100\%. \tag{17}$$

In this formula, N_t denotes the total number of sampling points within the tunnel and N_v represents the number of sampling points that satisfy the coverage requirement. The number of pseudolocators observed at a sampling point is primarily determined by the distance between the sampling point and each pseudolocator. Differences in the products of pseudo-satellite transmitters result in varying transmission distances for pseudo-satellites. In this paper, the transmission distance threshold for a pseudo-satellite is set at 1.5 times the redundant distance: D_{max} = 200 × 1.5 = 300 m. This value represents a universal setting compatible with most commercial products, designed to counteract signal attenuation, multipath effects, and edge obstruction, ensuring stable signal reachability. The distance between the *i*-th sampling point and the *j*-th pseudo-satellite is given by

$$D(i,j) = \sqrt{(x_j - x_i)^2 + (y_j - y_i)^2 + (z_j - z_i)^2}\,. \tag{18}$$

In this equation, (x_i, y_i) are the coordinates of the sampling point and (x_j, y_j, z_j) are those of the pseudo-satellite. To meet one of the coverage requirements, each sampling point only needs to satisfy the condition that the distance to at least four pseudo-satellites is ≤300 m. However, if the pseudo-satellites cluster together, the excessive proximity between them may cause the azimuth and elevation angles between the sampling point and the pseudo-satellite to exceed the receiver's resolution threshold (azimuth angle ≥5°, elevation angle ≥3°). In this case, the number of satellites recognized by the sampling point may be less than four. Even if the average *PDOP* is low under this configuration, if coverage is insufficient (e.g., if there is signal loss in peripheral areas), it still fails to meet practical positioning requirements.

3.2 Simulation results

To quantify the performance advantages of the proposed ZAP-GA in tunnel scenarios, the following parameter settings and experimental plan have been adopted for the simulated tunnel:

(1) The tunnel is 200 m long, 6 m wide, and 5 m high. For practical engineering purposes, the height of the side pseudo-satellite must be at least 2 m. Sampling shall be conducted at 10 m intervals across a 200 m horizontal section of the tunnel and at 1 m intervals across a 6-m-wide section. This will ensure that sampling points cover the entire ground surface uniformly. The average *PDOP* and the overall coverage at each sampling point on the tunnel surface will be calculated according to the number of pseudo-satellites that have been configured. The following three strategies are employed to deploy stations for the experimental plan: the conventional GA, the circumventing weight of area (CWOA), and the proposed ZAP-GA.

(2) To evaluate the performance of various optimization algorithms with different numbers of pseudo-satellites, the population size was set to 150 and the iteration count to 100, striking a balance between computational efficiency and thorough optimization. Three satellite gradients — 4, 6, and 10 satellites — were used to analyze algorithmic differences.

(3) To ensure fairness in the comparative experiments, the parameters for the GA and the CWOA were determined through a systematic optimization process consistent with ZAP-GA. GA selected the crossover probability (Pc) and the mutation probability (Pm) via grid search. This combination yielded optimal GA performance. CWOA optimized the circumventing prey coefficient (b) and the spiral update weight (l) through orthogonal experiments. The same population size (150) and iteration count (100) as ZAP-GA were maintained to eliminate external factors that can interfere with the experimental results.

The average *PDOP* convergence curves for each strategy under different numbers of pseudo-satellites are shown in the experimental results.

Figure 2 shows that, when the number of satellites is small, the optimized space complexity is low, enabling all algorithms to rapidly traverse potential solutions. This hinders the “partition-adaptive” mechanism of ZAP-GA from demonstrating its advantages. Additionally, the low fault tolerance of the four-satellite geometry, combined with the narrow tunnel configuration, resulted in minimal differences among the three algorithms’ optimization outcomes. As shown in Figs. 3 and 4 and Table 2, the optimization space increases when the number of pseudo-satellites increases to six or eight, which highlights the advantages of ZAP-GA. On the one hand, the initial *PDOP* calculation is more accurate. During the initial iteration phase, the *PDOP* of ZAP-GA is 15% lower than that of CWOA and 25% lower than that of GA. Conversely, in scenarios involving six pseudo-satellites, ZAP-GA produced a final average *PDOP* that was 23.02% lower than that of GA and 24.50% lower than that of CWOA. ZAP-GA outperformed GA and CWOA by 29.53 and 32.82%, respectively, with eight pseudo-satellites. As the number of pseudo-satellites increases, it can be observed that the final *PDOP* values of the three algorithms show a decreasing trend. Moreover, ZAP-GA demonstrates a more significant improvement than the other two.

By combining Fig. 4 with the findings from further analysis in Table 3, the convergence characteristics of ZAP-GA can be determined. With regard to convergence rate, ZAP-GA

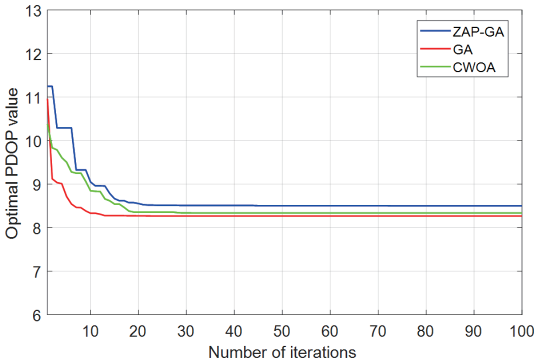

Fig. 2. (Color online) Average *PDOP* convergence curves for each strategy under four pseudorange conditions.

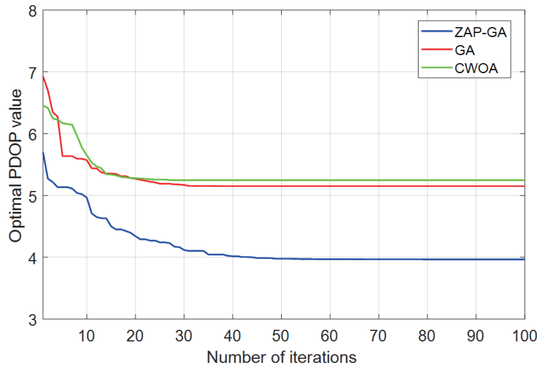

Fig. 3. (Color online) Average *PDOP* convergence curves for each strategy under six pseudorange conditions.

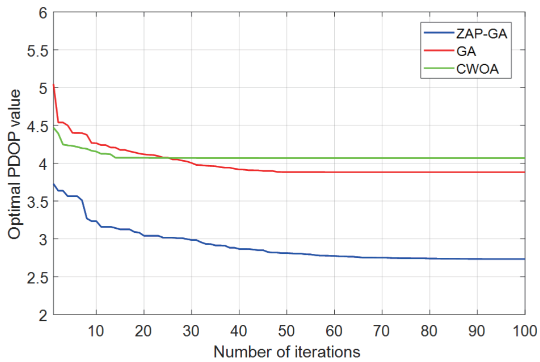

Fig. 4. (Color online) Average *PDOP* convergence curves for each strategy under eight pseudorange

Table 2
Overall average *PDOP* for each strategy under different pseudorange conditions.

Pseudosatellite count	ZAP-GA (*PDOP*)	GA (*PDOP*)	CWOA (*PDOP*)
4	8.4984	8.2655	8.3311
6	3.9637	5.1494	5.2471
8	2.7326	3.8812	4.0673

Table 3
Convergence characteristics of various strategies under pseudosatellite conditions.

	ZAP-GA	GA	CWOA
Convergence time	54.32 s	50.60 s	44.89 s
Average convergence rate	0.013851 PDoP/generation	0.024528 PDoP/generation	0.031968 PDoP/generation

employs a hybrid initialization strategy combining "scenario constraints + optimal configuration guidance." The first-generation average *PDOP* of 3.65 demonstrates a substantial improvement over the GA (5.07) and CWOA (4.47) models. Subsequent iterations require only the fine-tuning of the optimal solution, resulting in a low convergence rate (*PDOP* reduction per generation) compared with those obtained by the comparison algorithms. However, this is indicative of the efficacy of the initial optimization process rather than algorithmic inefficiency. In terms of temporal complexity, ZAP-GA requires a greater temporal investment than the comparison algorithms. The primary additional overhead is attributable to partition adaptation, weighted *PDOP* calculation, and multi-constraint penalty modules. These modules constitute the core of its tunnel adaptation and signal interference suppression capabilities, yet they provide essential support for high accuracy and robustness, representing a reasonable trade-off between performance and efficiency. In comparison with the suboptimal optimization patterns exhibited by GA and CWOA, which are distinguished by "low initial quality and high incremental gains per iteration," ZAP-GA's "high initial quality and low incremental gains" serves to reduce the number of iterations necessary in practical deployment contexts. In addition, it is notable for its enhanced robustness, a quality that renders it especially well suited to engineering scenarios involving subterranean infrastructure, such as tunnels.

To investigate the impact of tunnel dimensions on positioning performance, tests were conducted using eight pseudo-satellites across three scenarios involving tunnels of varying sizes: one measuring 200 m in length; one measuring 250 m in length; and one measuring 200 m in length, 10 m in width, and 8 m in height. The experimental results for the average *PDOP* convergence curve are as follows.

Figure 5 shows that P1 is 200 × 6 × 5 m^3, P2 is 250 × 6 × 5 m^3, and P3 is 200 × 10 × 8 m^3. The results demonstrate that ZAP-GA is highly robust in the face of changes in tunnel dimensions. Whether the length increases by 25%, or the width and height expand significantly (by 66.7 and 60%, respectively), its stable average *PDOP* consistently remains below 3.5, ensuring a high level of precision. The algorithm exhibits stable convergence efficiency, with iteration differences of ≤12.5% required to achieve stable convergence across three operational scenarios. No premature convergence or oscillation of the solution was observed. Owing to the synergistic effects of dynamic partitioning, weighted *PDOP*, and adaptive variable stride length, the algorithm can adapt to tunnels of different sizes without the need for major parameter adjustments. This pattern demonstrates the characteristic of "gradually expanding performance gains" as the number of pseudo-satellites increases, which effectively validates the adaptability of ZAP-GA to tunnel pseudo-satellite configurations.

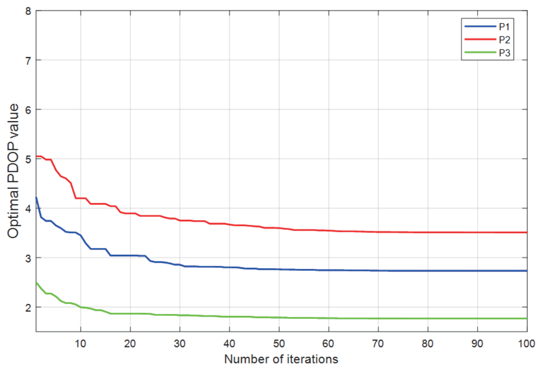

Fig. 5. (Color online) Average *PDOP* convergence curves for different sizes under eight pseudorange

The coverage rates for each sampling point under different numbers of pseudo-satellites for each strategy are shown in the experimental results. First, we consider the four pseudo-satellites. The analysis of Figs. 6 to 8 yields Table 4. This shows that, when the number of pseudolocations is low, the minimum *PDOP* values across all sampling points differ minimally among the three algorithms. However, the maximum *PDOP* of 23.2759 for ZAP-GA is higher than those for GA (19.2870) and CWOA (19.9304). This precisely reflects the distinct emphasis of their respective optimization strategies. ZAP-GA increases the weight of constraints in regions at the extreme edges to achieve more effective overall coverage. However, low-dimensional search space constraints and low tolerance for geometric configurations due to the insufficient number of satellites result in excessively high *PDOP* values at the edges. Nevertheless, the algorithm's

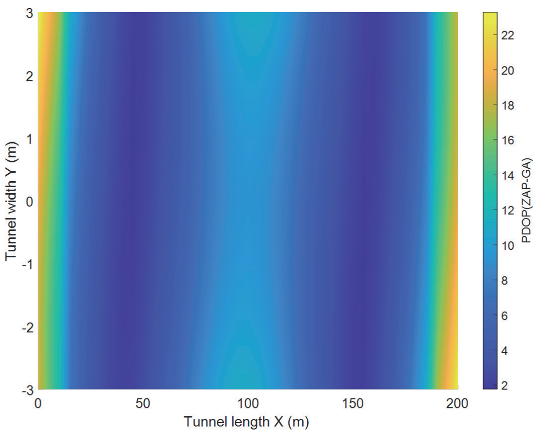

Fig. 6. (Color online) *PDOP* distribution at each sampling point of ZAP-GA under four pseudosatellite conditions.

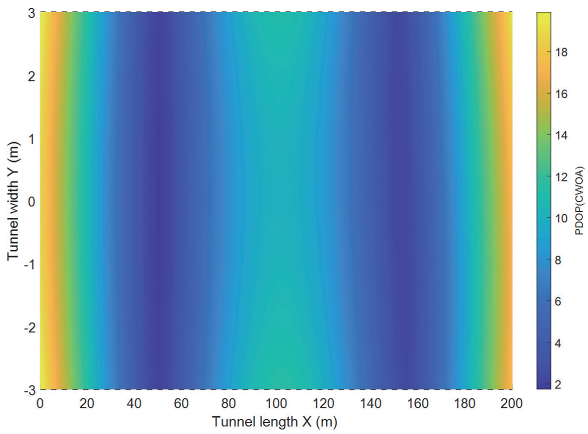

Fig. 7. (Color online) *PDOP* distribution at each sampling point of CWOA under four pseudosatellite conditions.

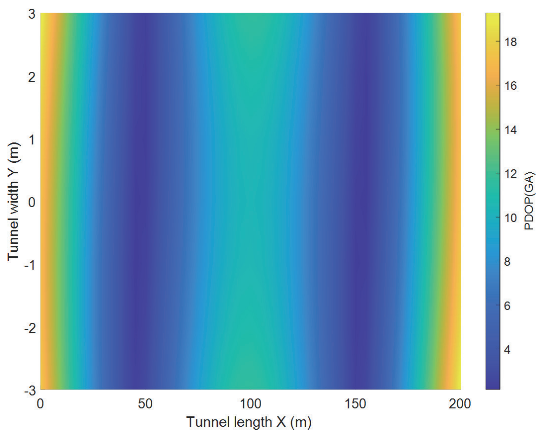

Fig. 8. (Color online) *PDOP* distribution at each sampling point of GA under four pseudosatellite

Table 4
Maximum and minimum *PDOP* values and overall coverage of each strategy under four pseudosatellite conditions.

Optimization algorithm	Maximum *PDOP*	Minimum *PDOP*	Coverage (%)
ZAP-GA	23.2759	1.7702	44.25
GA	19.2870	2.1611	27.87
CWOA	19.9304	1.7514	30.89

overall optimization effectiveness remains unaffected. Overall pseudo-satellite coverage is effectively enhanced, with increases of 16.38 and 13.36% compared with those obtained by GA and CWOA, respectively. This demonstrates that, when four pseudo-satellites are deployed, ZAP-GA provides superior coverage capabilities for tunnels.

Second, the conditions of the six pseudo-satellites are as follows. As shown in Table 5, from the analysis of Figs. 9 to 11, ZAP-GA begins to demonstrate its impact on improving positioning performance and its unique advantages as the number of pseudo-satellites increases to six. In terms of the maximum *PDOP*, ZAP-GA achieved 11.7643, comparable to the values of 11.3994 for GA and 11.6627 for CWOA. This suggests that the increased number of pseudo-satellites aligns well with the "partition optimization" design. We achieved uniform pseudo-satellite coverage across all areas while optimizing regions on both sides of tunnels by weighting pseudo-satellite partitioning and edge regions. This approach prevents pseudo-satellite clustering and eliminates random exploration blind spots. With a minimum *PDOP* of 1.2268, ZAP-GA achieved a significantly lower result than GA (1.6783) and CWOA (1.7246), fully demonstrating its ability to optimize positioning within tunnels. In terms of overall coverage, ZAP-GA achieved 85.02%, which is a significant improvement from 46.69% achieved by GA and 52.90% achieved by CWOA. This clearly demonstrates that, in the presence of six pseudo-satellites, ZAP-GA provides a superior global coverage of tunnels.

Finally, the results were obtained under conditions involving eight pseudo-satellites. As the number of pseudo-satellites increases to eight, the positioning performance of ZAP-GA becomes even more pronounced. Figures 12 to 14 demonstrate this, showing that the number of sampling points with $PDOP \geq 5$ is significantly lower than those for the other two algorithms. This clearly illustrates the superior control capability of ZAP-GA in low-precision areas. The analysis of Table 6 shows that the ZAP-GA maximum *PDOP* of 7.1911 is lower than those achieved by GA and CWOA (8.1665 and 8.4863, respectively), indicating that ZAP-GA produces more accurate positioning results. ZAP-GA achieved a minimum *PDOP* of 1.0230, which is as close as possible to the theoretical optimum of 1. By this stage, ZAP-GA had achieved a coverage rate of 95.82%, which far exceeded the rates of 79.09 and 73.87% achieved by GA and CWOA, respectively, and was approaching full-area coverage. Core area coverage, excluding peripheral regions, had reached 100% (in the 10–190 m zone), with a maximum *PDOP* of 3.7737 and a fluctuation of only 2.7507 in the core area. These results fully validate the comprehensive positioning advantages of ZAP-GA in tunnel scenarios.

Improvements in *PDOP* and coverage were achieved through three algorithms, increasing the number of pseudo-satellites from four to six and then eight. This effectively validates the core logic that "optimizing the geometric configuration of multiple pseudo-satellite enhances positioning performance." ZAP-GA achieves ultralow *PDOP* values and near-complete coverage

Table 5
Maximum and minimum *PDOP* values and overall coverage of each strategy under six pseudosatellite conditions.

Optimization algorithm	Maximum *PDOP*	Minimum *PDOP*	Coverage (%)
ZAP-GA	11.7643	1.2268	85.02
GA	11.3994	1.6783	46.69
CWOA	11.6627	1.7246	52.90

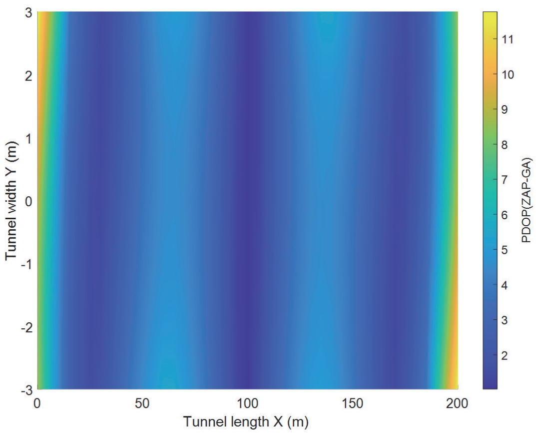

Fig. 9. (Color online) *PDOP* distribution at each sampling point of ZAP-GA under six pseudosatellite conditions.

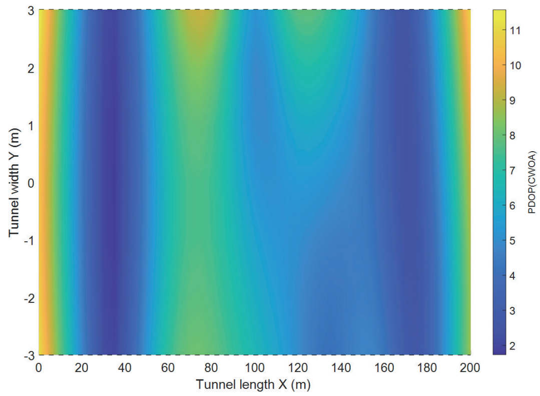

Fig. 10. (Color online) *PDOP* distribution at each sampling point of CWOA under six pseudosatellite conditions.

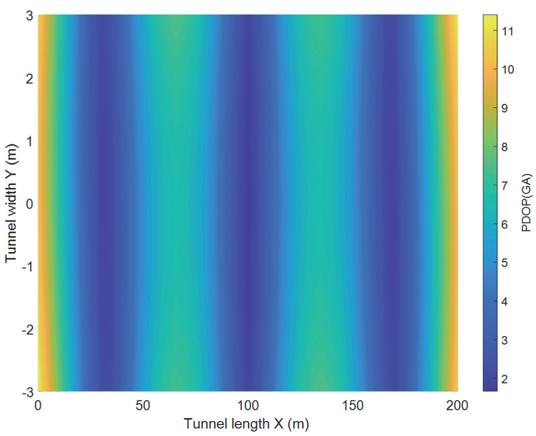

Fig. 11. (Color online) *PDOP* distribution at each sampling point of GA under six pseudosatellite conditions.

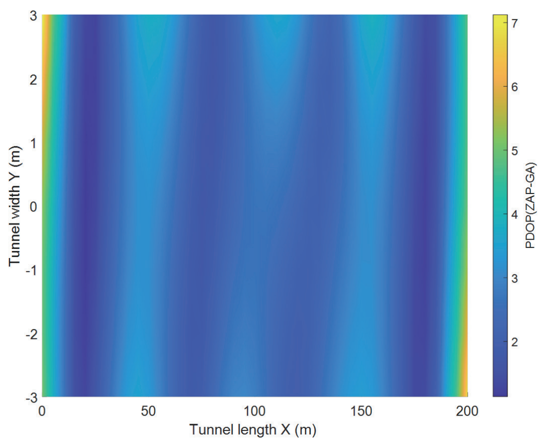

Fig. 12. (Color online) *PDOP* distribution at each sampling point of ZAP-GA under eight pseudosatellite conditions.

across the entire region when a sufficient number of pseudo-satellites are available, owing to its partitioned, weighted, and adaptive optimization design. Full coverage is achieved in core areas.

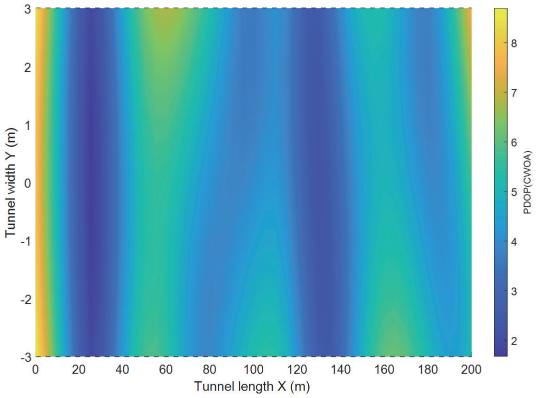

Fig. 13. (Color online) *PDOP* distribution at each sampling point of CWOA under eight pseudosatellite conditions.

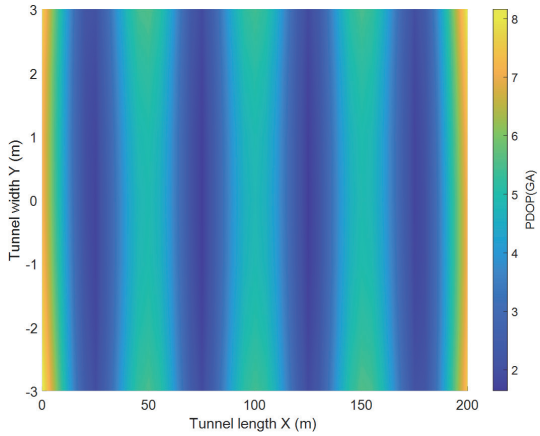

Fig. 14. (Color online) *PDOP* distribution at each sampling point of GA under eight pseudosatellite conditions.

Table 6
Maximum and minimum *PDOP* values and overall coverage of each strategy under eight pseudosatellite conditions.

Optimization algorithm	Maximum *PDOP*	Minimum *PDOP*	Coverage (%)
ZAP-GA	7.1911	1.0230	95.82
GA	8.1665	1.6377	79.09
CWOA	8.4863	1.6498	73.87

4. Conclusion

In this paper, we addressed the issue of positioning failure in GNSSs caused by signal obstruction in confined, elongated spaces such as tunnels. With a focus on the core requirement of pseudo-satellite layout optimization, we proposed a scheme based on ZAP-GA. From the theoretical analysis, algorithm design, and experimental simulation, the primary research results are as follows. Traditional optimization algorithms suffer from issues such as random clustering, blind spots in exploration, and insufficient optimization accuracy. This renders them incapable of meeting the high-precision positioning demands of tunnel environments, which are narrow, elongated spaces with unique characteristics. Therefore, feasible optimization methods must be designed with a focus on scenario adaptability and optimization efficiency. First, we created a simulation model of the narrow, elongated tunnel environment. Next, we designed an optimization algorithm based on clear objectives and constraints. We partitioned the entire tunnel into zones and placed one pseudo-satellite in each zone. We employed an adaptive adjustment mechanism to control the mutation step size (decreasing from 0.12 in early iterations to 0.02 in late iterations) and zone weight (sampling point weights of 1.6 for edge zones, 0.7 for core zones, and 1.0 for core regions). The precision *PDOP* optimization mechanism designs the algorithm's fitness function through weighted *PDOP* and satellite distribution penalties. This ensures positioning accuracy and mitigates practical engineering issues, such as excessive satellite concentration on the mounting surface and overly close satellite positions. Within the MATLAB simulation environment, the effectiveness of the algorithm cannot be fully demonstrated in a four-pseudosatellite scenario owing to the limitations of the low-dimensional search space and the poor geometric configuration. Nevertheless, the algorithm's ability to ensure coverage with a small number of pseudo-satellites was validated. As the number of pseudo-satellites increased from 6 to 8, there were significant improvements in both accuracy and coverage compared with those achieved by the other two algorithms. With eight pseudo-satellites, the average *PDOP* improved by around 30% compared with those achieved by the other two algorithm, and the coverage approached full coverage. The optimization method proposed in this paper effectively addresses the positioning accuracy and coverage challenges posed by the unique environment of tunnels, meeting the requirement for high-precision positioning. It provides an efficient engineering deployment solution for tunnel pseudo-satellite systems and demonstrates significant potential for practical applications. Subsequent extensions can be made to tunnel scenarios containing obstacles. Note that signal blockage, caused by the presence of obstacles, will result in the occurrence of blind spots within partitions due to non-line-of-sight propagation. This, in turn, has the capacity to affect the pseudo-satellite geometry configuration and signal accessibility. The adaptive dynamic characteristics of existing partitioning strategies provide a foundation for adaptation by reducing partition granularity in densely obstructed areas and avoiding deployment along obstructed paths, thereby achieving the initial mitigation of obstruction effects. Subsequent research endeavours will concentrate on the integration of NLOS error compensation algorithms with partition optimization to achieve enhanced positioning accuracy and coverage integrity in obstructed environments.

References

1 W. Tang and J. Chen: World Sci. Technol. Res. Dev. **45** (2023) 276.
2 L. Zhang and M. Bi: J. Eng. Res. Rep. **26** (2024) 30.
3 X. Liu, W. Liu, Q. Wu, and Y. Zhu: China Transp. Inf. (2025) 124.
4 R. Peng, L. Xu, and F. Wang: Military Equip. Autom. **29** (2010) 53.
5 Y. Zhao, J. Zou, J. Guo, and P. Zhang: Surv. Bull. **S1** (2020) 288.
6 X. Yang, F. Wang, W. Liu, W. Xiao, and X. Ye: Chin. J. Electron. **32** (2023) 1050.
7 X. Liang, S. Pan, S. Du, B. Yu, S. Li, and S. Li: Remote Sens. **17** (2025) 1909.
8 X. Zhao, Q. Shuai, G. Li, F. Lu, and B. Zhu: ISPRS Int. J. Geo-Inf. **10** (2021) 601.
9 X. Shangguan, X. Yu, and D. Wen: Proc. 2019 Academic Annual Conf. Jiangsu Society of Surveying, Mapping and Geographic Information (Jiangsu Society of Surveying, Mapping and Geographic Information, 2019) 4.
10 K. Shao, K. Li, and J. Wang: Commun. Technol. **50** (2017) 2454.
11 B. Ma, Y. Liu, and H. Han: Transducer Microsyst. Technol. **42** (2023) 156–160, 164.
12 W. Chen and P. Yang: J. Shenzhen University (Science and Engineering Edition) **41** (2024) 241.
13 Liu, L. Wang, and Z. Yao: Comput. Appl. Res. **37** (2020) 1839.
14 H. Li, G. Chen, and Y. Liu: J. Navigation **74** (2021) 654.
15 S. M. Sun, Y. J. Chen, and L. G. Dong: Math. Biosci. Eng. **21** (2024) 2787.
16 Wang, H. Zhang, and D. Li: IEEE Trans. Intelligent Transp. Syst. **24** (2023) 2134.
17 H. Chang, Z. Huang, Y. Zhao, and R. Li : Telemetry and Telecontrol (2007) 16–21.

About the Authors

Chenglin Cai is currently a professor, PhD supervisor, subject leader, and Vice Dean of the School of Automation and Electronic Information, Xiangtan University, Hunan, China. His research interests include BeiDou real-time precision positioning, RTK, inertial navigation, indoor positioning. and multisource fusion navigation.

Yang Zhouwang is a graduate student in the Electronic Information program at Xiangtan University. His primary research interest is in pseudo-satellite positioning.

Kaihui Lv received his M.S. degree in electronic information from Xiangtan University, Hunan, China, in 2023. He is currently pursuing his Ph.D. degree in mathematics at the School of Mathematics and Computational Science, Xiangtan University, Hunan, China. His research interests include precise single-point localization and RTK.

Kun Xie graduated with a master's degree from the School of Computer and Communication Engineering, Lanzhou University of Technology. He is currently pursuing his studies at the School of Automation and Electronic Information, Xiangtan University. His main research interests include communication positioning.

Li Shaohui is a graduate student in the Electronic Information program at Xiangtan University. His primary research interest is in pseudo-satellite positioning.