Carbon Emission Reduction of Tunnel Construction Machinery System Based on Self-Organizing Map-Global Particle Swarm Optimization With Multiple Weight Varying Models

ZHANPING SONG¹,², ZHENZHAO XIA¹,²,³
¹School of Civil Engineering, Xi’an University of Architecture and Technology, Xi’an, 710055, China
²Shaanxi Key Laboratory of Geotechnical and Underground Space Engineering, Xi’an, 710055, China
³XAUAT UniSA An De College, Xi’an University of Architecture and Technology, Xi’an, 710000, China
Corresponding author: ZHENZHAO XIA (e-mail: kashimyou@163.com).

ABSTRACT The impact of greenhouse gas emissions from construction activities on the environment is becoming more and more obvious. It is imperative to adopt corresponding techniques and management measures to restrain carbon emissions from construction industry, especially construction of infrastructures such as transportation tunnels. Through the Life Cycle Assessment theory and carbon emission factor method, we established a carbon emission model which can be applied to quantify the carbon emission of the mechanized construction system of tunnels constructed by drill-and-blast method. This model exhibits considerable non-convexity after incorporating various constraint conditions, and thus its task of finding the optimums becomes an NP-hard problem. Then, we propose a self-organizing map-global particle swarm optimization algorithm which incorporates multiple weight varying strategies and self-organizing mapping networks for adaptive adjustment of particle trajectories to improve the ability of searching optimums. The feasibility and advantages of the proposed algorithm were verified through a series of experiments, and finally it is combined with a quantitative model and applied to the optimization of the construction machinery unit combination for the F4 section of the Wushaoling Tunnel, and quite good optimization results are achieved, optimized configuration reduced the cycle duration from 660 minutes to 432 minutes while the total emission of carbon equivalent was reduced by 88.4% compared to original machinery system configuration, which offers reference and inspiration to other researches of carbon emission reduction in the construction of tunnels or other infrastructures.

INDEX TERMS Carbon emission reduction, configuration optimization, particle swarm optimization (PSO), self-organizing map network, tunnel construction, uniform design.

I. INTRODUCTION
Engineering construction emits a large amount of greenhouse gases. In 2009, production activities in the construction sector contributed up to 23% (5.7 billion tons) of global carbon emissions [1], especially in the tunnel construction process. Moreover, by quantifying the carbon emissions of the tunnel construction process through Life Cycle Assessment (LCA) method, researchers have found significant differences in the contribution parts of the construction process to the total GHG (greenhouse gas) emissions. For example, Li et al. [2] found that 93% of the equivalent carbon emissions per meter of the shield tunnel construction comes from building materials by using the carbon coefficient method. Besides, Xu et al. [3] studied the construction process of tunnels using the drill-and-blast method and simulated the tunnel carbon emissions hierarchically according to the excavation process, which ultimately found that the construction process of weak rock enclosures requires higher levels of rock support because lining have higher carbon emission levels. Most studies generally agree that, within the framework of the LCA theory, the carbon emissions of tunnels to the environment are more concentrated on the upstream of the supply chain, especially for the manufacture of concrete and steel, and that the impact of the operation of construction machinery is relatively small. However, this does not mean that absolute share of the construction process in the international and domestic economic construction activities is low. Previous study [4] pointed that the twin-bore tunnels for the Channel Tunnel Rail Link (CTRL) in the United Kingdom (UK) accounted for 2.1% of all emissions associated with the UK construction industry in 1999, which is only 7.5 km long in total. For such a highly mechanized project, even if the percentage of greenhouse gas emissions from fossil fuel and electricity consumption of the construction machinery system is small (compared with upstream emissions), the converted absolute emission is still very impressive because its contribution is immediately after the upstream emissions. Meanwhile, the progress of carbon emission monitoring technology represented by carbon emission remote sensing satellite technology has made it easier to monitor regional and short-term carbon emission behavior, all of these objective realities place considerable demands on the control of overall GHG emissions from mechanical systems at tunnel construction sites.

Besides the above-mentioned, various studies appeared recently which focus on the carbon emission of construction industry and construction activities, they can be roughly categorized as: 1. Monitoring [5-7] and tracking [8] of emission comes from constructions; 2. Evaluation for carbon emission and effect [9-12]; 3. Models [13-17], measures [18-21] or techniques [22-24] of analysis and optimization used in carbon and other pollution matters reduction. However, fewer similar studies have been executed on mechanical system of tunnel construction, it is necessary to supplement this research gap.

Given such reality and need, this paper focuses on the configuration optimization of machine systems in tunnel construction. For this combinatorial optimization problems, the Pareto solution of the model needs to be searched in the feasible domain space with the help of certain search strategies. Heuristic algorithms are commonly used to search for the optimal feasible solution of combinatorial optimization problems. The main heuristic algorithms which are widely used in this problem are: Genetic Algorithm (GA) [25], Simulated Annealing Algorithm (SAA) [26], Ant Colony Algorithm (ACA) [27], Artificial Neural Network (ANN) [28] and Particle Swarm Optimization (PSO) [29-33]. In practical applications, GPOSO is preferred for searching the ideal solution of the combinatorial optimization represented by the economical dispatch (ED) problems. However, multiple factors make this method easy to fall into local minima or perform slow convergence rate of iterations in the actual solution process.

The integration of machine learning techniques with particle swarm algorithms is also more commonly used by scientific researches and in engineering practices, and here a machine learning algorithm called the self-organizing map (SOM) network [29, 34, 35] is focused on. The SOM is an unsupervised artificial neural network based on a competitive learning strategy, which generates a low-dimensional discrete map by learning data in a high-dimensional input space [36].

In this paper, a GPOSO algorithm using self-organizing map networks for particle motion trajectory adjustment is proposed, which is named Self-Organizing Map-Global Particle Swarm Optimization (SOMGPSO), to overcome the defects of traditional GPOSOs. The SOM network generates a visualization of the node-particle structure by unsupervised clustering of the global optimal particle values after a certain number of iterations. Network’s nodes automatically identify and correspond to particles with different degrees of aggregation and generates weights based on each particle assigned to the node. When the SOM network reaches a certain number of iterations, this weight would be theoretically close to each particle falling in the same node with the smallest variance or even exactly equal to the whole particles, which would be used as the projection origin. When the node becomes an anomalous node with a large number of particles falling in, the new particles with equal number of old particles are projected to the original feasible domain space with the reference of the uniform design schedule and replace the particles in the anomalous node. Finally, the reorganized particle population is put into the iterative process of the particle population in the new weight varying model, the iteration would be stopped adaptively according to the multiple termination criteria.

The SOMGPSO is compared horizontally with the linear decreasing inertia weight global particle swarm algorithm (LDIWGPSO) and the global particle swarm algorithm with adaptive inertia weight (AWGPSO), and it is found that the proposed SOMGPSO has advantages in terms of robustness, stability and accuracy. Finally, the coding and decoding technical measures are integrated with the carbon emission calculation model of the tunnel construction mechanization.
system to optimize the mechanical system of the F4 section of the Wushaoling Mountain Tunnel excavation by the bench method.

II. METHODS

A. PROBLEM FORMULATION

It is more straightforward and easier to optimize the configurations of construction machinery system to maximize the reduction of greenhouse gas emissions than to improve the performance of a single type of machinery in the system. Through the carbon emission factor method, the emission type, emission process and volume of each machine unit in the system can be clarified and quantified to generate the linear sum of equivalent carbon emission of each machine unit. On this basis, the quantifiable engineering requirement such as duration is introduced into the multi-objective combinatorial optimization model.

In this paper, tunnels constructed by drill-and-blast method are selected as the research objects, which is the mainstream construction technology nowadays[37]. Drill-and-blast method is a construction process which maintains the stability of the excavation surface and excavation process as engineering purposes. Advancing in circular processes is one of the features of drill-and-blast method. According to the researches [38-40] in recent years the conventional drilling and blasting method of tunnel boring within single cycle of the process mainly includes: drilling and excavation, controlled blasting, initial support construction, and secondary lining construction.

In accordance with the idea of energy flow path tracing and previous tunnel construction engineering and modeling practice[41], the tunnel construction machinery system can be divided into four subsystems, i.e., the drilling and blasting system, anchor spraying support system, excavation and transportation system, and energy supply and drive system. The four subsystems in the construction process of energy flow relationship are shown in Fig.1, and it should be noted that this energy source flow relationship diagram does not take into account too many other systems in addition to the excavation and transportation system with their own capacity mechanical equipment caused by the power and fuel consumption.

1) OBJECT FUNCTION

The modeling is guided by the existing tunnel construction machinery scheduling configuration model, and the number of all possible machinery and equipment components is denoted as a d-dimensional vector \( A = \{A_1, A_2, A_3, \ldots, A_d\} \). In the mechanized construction process of tunnels, if no heavy oil-consuming large machinery and equipment (such as TBM,) is used, the main types of energy required are electricity, diesel and gasoline, which can be converted into carbon dioxide equivalent values for direct assessment of carbon emission levels, i.e., carbon emission calculation using the carbon emission factor method. And the total amount of time consumed and carbon emission in the construction process can be regarded as the functions of \( A \), which are denoted as \( T(A) \) and \( E(A) \). The total carbon emissions have the following expressions:

\[
E = \sum_{i=1}^{u} \sum_{j=1}^{d} E_{ij} T_{ij}(A)
\]

where, \( T_{ij}(A) \) is the direct working time of the corresponding \( i \)-th machinery group on the specified workplace \( j \) of the current process, while \( E_{ij} \) is the \( CO_2 \) equivalent value converted from one of the \( i \)-th kind of machinery’s total unit emissions, and the expression is:

\[
E_i = C_d D_1 + C_g D_2 + C_e D_3
\]

where, \( C_d \), \( C_g \) and \( C_e \) are the carbon emission factors of diesel, gasoline and electricity, respectively, while \( D_1 \), \( D_2 \) and \( D_3 \) are the total amount of diesel, gasoline, and electricity consumed per unit shift of the current machinery, respectively. The optimization model of the tunnel construction machinery system configuration (normalized) has considered the duration as well as the carbon footprint which can be written as:

\[
\text{Min } Y = k_1 \frac{E(A)}{E_{\text{origin}}} + k_2 \frac{T(A)}{T_{\text{origin}}}
\]

where, the \( k_1 \) and \( k_2 \) are the preferred weights for emission and duration, respectively; While \( E_{\text{origin}} \) and \( T_{\text{origin}} \) are the original configuration’s emission and duration, respectively. It should be noted that carbon emission factor refers to the unit carbon emission of various energy sources from production, to transportation and full energy conversion. Considering the regional and time-domain nature of carbon emission factor, in this paper the Chinese life-cycle base database (CLCD) and China Emission Accounts and Datasets (CEAD) are chosen to calculate the carbon emission factor of three energy sources in an integrated manner.

2) OBJECTIVE CONSTRAINTS

Inflexible Constrains: In tunnel boring, any workplace in a certain period of time cannot be unlimited to accommodate
a type of mechanical equipment, thus there is an inflexible constraint expressed as follows:

\[ 0 \leq A_{ij} \leq A_{ij}^{\text{max}} \]  \hspace{1cm} (4)

where, \( A_{ij}^{\text{max}} \) is the upper limit of accommodation of the \( j \)-th workplace for the \( i \)-th machinery.

**Flexible Constraints:** For dynamic mechanical units, the capacity of the workplace where they operate is reflected in a pattern with no clear boundaries, and the working time is additionally accumulated by a penalty factor named interference time, while for different workplaces, there are flexible accommodation thresholds for each mechanical unit, and the interference effect only appears when the number of such machines is greater than the threshold.

This threshold is defined as \( A_{ij}^{\text{dist}} \), and the then disturbance duration can be expressed as:

\[ T_{ij}^{\text{dist}} = e_{ij} \times (A_{ij} - A_{ij}^{\text{dist}}) \]  \hspace{1cm} (5)

where, the \( T_{ij}^{\text{dist}} \) is the extra duration which should be calculated in the objective function, meaning the disturbance effect for the \( i \)-th machinery in \( j \)-th workplace; \( e_{ij} \) is the disturbance factor for the \( i \)-th machinery in \( j \)-th workplace.

**Hierarchical constraints:** The number of mechanical units which can be accommodated in each workplace is constrained by its own dimensional factors, while the size of the mechanical units available on site also has an upper limit. If the mechanical scheduling of the tunnel construction process is considered as a dynamic time-phased scheduling, it must be considered that this scale constraint has the hierarchical effect:

\[ \sum_{j=1}^{m} A_{ij} \leq A_{i}^{\text{d}} \]  \hspace{1cm} (6)

where for one type of machinal units, the total number of such mechanical units distributed on all workplaces at the same time cannot exceed the maximum scheduling size of such mechanical units.

**Balance constraints:** Supply and demand balance constraints are present throughout the entire process of tunnel mechanized construction, strictly speaking, from the stream balance of various types of energy to the supply of water, compressed air and even lubricants. Here the relationship between the supply and demand balance of energy and compressed air with the specific form of the model are focused on according to the energy stream framework shown in Fig.1.

For one type of energy \( e \), the balance constraints relationship is below:

\[ \sum_{j=1}^{n} \sum_{i=1}^{d} P_{e}(A_{ij}) \leq P_{e}^{\text{max}} \]  \hspace{1cm} (7)

For the compressed air, the balance constraints relationship is below (assumed that the driven energy of compressed air comes from energy type \( e \)):

\[ \sum_{j=1}^{n} \sum_{i=1}^{d} P_{a}(A_{ij}) \leq P_{a}^{\text{max}} \leq P_{e}^{\text{max}} \]  \hspace{1cm} (8)

### 3) ENCODING AND DECODING DESIGN

When applied to solve combinatorial optimization problems, particle swarm optimization algorithms need to incorporate the appropriate encoding and decoding mechanisms, which need fit with the need for realistic problems[33, 42, 43]. The characteristics of the drill-and-blast method of tunnel construction are analyzed, and it is roughly concluded that there is a distinct and one-to-one subjective and objective correspondence between various machine units and workplaces, and then a workplace-machine unit particle coding and decoding method applicable to this model is proposed by analogy with the machine-PPR coding technique used in the parallel machine scheduling model (PMS) in the field of scheduling problems, which is combined with several practical cases[42]. Define a two-dimensional particle as follows, the first dimension in the two-dimensional particle with natural numbers 1, 2, …, \( n \) denotes the \( n \)-th machinery under the current machinery type, the second dimension denotes the position of the particle, and the length of the particle is the total number of similar machineries dispatchable.

**TABLE 1.** Particle coding based on workplace-machine unit.

| Machine | 1  | 2  | …  | m |
|---------|----|----|-----|---|
| Position | \( x_{1j} \) | \( x_{2j} \) | ... | \( x_{mj} \) |

The generation process of configuration scheme is the decoding process of above codes, and the specific operation is the downward rounding of \( x_{ij} \), so that the rounded particle position \( \text{INT}[x_{ij}] \) is the workplace corresponding to the current mechanical unit \( i \) or no workplace is assigned (i.e., 0) to this unit. The configuration scheme of current kind of machinery is generated by assigning workplaces to all mechanical units in this way, and by splitting the whole process of the work steps into several configuration problems, the remaining mechanical configuration schemes are generated and aggregated in parallel independently, resulting in the overall mechanical configuration scheme. Since the above-mentioned particle coding and configuration scheme generation method is oriented to the workplaces, this particle representation is called the workplaces-based particle coding in this paper. It is worth noticing that this coding approach focuses on the workplace response of each scheduled machine unit, which causes the disadvantage of high computational expenditure. But it is very helpful for the subsequent accurate flow tracking of machine units and maintains the usability of the computational model when there is a significant difference in the working capacity of different machine units.
The configuration of energy supply and drive system performs a different calculation method: after the particles of mechanical configuration size of other subsystems are generated and decoded into the configuration scheme, the minimum size of corresponding energy supply or drive machine is calculated directly according to its required drive energy and necessary redundancy. The calculation is rounded upward to obtain the minimum number of corresponding machines, while the cycle duration is determined by recalculating the main process duration corresponding to each workplace which adopts the optimized configuration and add it with the remaining main process duration. In order to minimize the probability of particle generation for infeasible solutions, the operation of randomly generating particle positions in the specified domain should be randomly re-valued in the domain when it crosses the specified domain boundary instead of assigning upper or lower limits to the particle positions [43].

B. GPSO ALGORITHM

1) GPSO PROCESS
The core mechanism of GPSO algorithm is the effective and benign information exchange of particles distributed in the feasible domain space: by arranging the particle swarm randomly distributed in the feasible domain space, each particle in the population is guided by its own best fitness scale and the best fitness scale of the whole population. The spatial position of each particle is constantly modified until the ideal fitness value of all particles converges, the effective control of the information exchange rate and particle motion intensity of the particle swarm is achieved with fewer user parameters. Considering the swarm population with \( m \) particles searching for an optimum solution in \( d \)-dimensional search space, where \( m > 1 \). Then, the following shows the process of GPSO:

| Step | Description |
|------|-------------|
| 1    | Randomly initialize the velocity and position vectors of particle \( i \) \( (i = 1, 2, \ldots, m) \) and the vectors can be presented as: |

\[
V_i(0) = [v_{i1}, v_{i2}, \ldots, v_{id}]
\]

\[
X_i(0) = [x_{i1}, x_{i2}, \ldots, x_{id}]
\]

2    | Calculate the objective function \( f(x) \) by the position vector \( X_i(0) \) for each particle \( i \). |

3    | Initialize the position vector: |

\[
P_{best,i}(0) = [g_{p1,1}, g_{p2,2}, \ldots, g_{p1,d}]
\]

4    | Search the global best position vector \( G_{best}(0) \) in \( P_{best}(0) \). |

5    | In iteration \( t \), the velocity vectors of particles are updated as follows: |

\[
V_i(t) = w(t)V_i(t - 1) + c_1r_1(t)[P_{best,i}(t - 1) - X_i(t - 1)] + c_2r_2(t)[G_{best}(t - 1) - X_i(t - 1)]
\]

And the position vectors are updated as follows:

\[
X_i(t) = X_i(t - 1) + V_i(t)
\]

6    | For each particle \( i \), the renewed objective function \( f(x) \) is calculated by the position vector \( X_i(t) \). |

7    | The \( P_{best,i} \) and \( G_{best} \) are updated as follows: |

\[
P_{best,i}(t) = \begin{cases} P_{best,i}(t - 1), & \text{if } f(P_{best,i}(t - 1)) \leq f(X_i(t)) \\ X_i(t), & \text{otherwise} \end{cases}
\]

\[
G_{best}(t) = \arg \min \{ f(P_{best}(t)) \}
\]

Stop  | \( t = N_{iter} \) |

2) INHERENT DEFECT OF GPSO
The GPSO algorithm has significant advantages of requiring fewer set parameters, fast convergence rate in early stages, and depend not on the gradient mechanism. But it also has the drawbacks of losing some information in the solution process, relatively slow convergence in later stage, and easily falling into local optimal solutions, which are more significant when applied to solve computational tasks in discrete spaces, such as combinatorial optimization or further economic scheduling optimization[30].

The special particle motion principle and iteration mechanism are the root causes of the above unfavorable performance of GPSO in the calculation of complex functions (e.g., multi-peak functions). Each moving particle has two tendencies which are its own optimal position and velocity, and the collective optimal position and velocity. The results of linear superposition of two velocities and positions brings the risk of particles being pulled into local minima or causing particle trajectory oscillations[32]. In high-dimensional space, when the search process enters the final stage, the trajectory oscillation will eventually manifest itself as a chaotic state of particles, and the majority
of particles cannot converge through the benign two-track system, which makes the search process inefficient or even invalid, thus it is necessary to propose an improved GPSO process to address its main drawbacks.

C. SOMGPSO ALGORITHM

Here, the details and full process of the proposed SOMGPSO algorithm and explanation of the recognition and reorganization for adverse convergence of particle swarm are provided. Before explanation, it is necessary to give a brief description of the knowledges and technologies required to constitute SOMGPSO process.

1) SELF-ORGANIZING MAP NETWORK

Self-organizing map (SOM) network was first proposed by Kohonen[36], which can automatically find out the similarity between the input data and configure similar inputs in close proximity to each other on the network, or even place them to the same neuron. It is called self-organizing map network because the topology of neurons is also forced to be recombined during the activation of neurons by input data, and this recombination is also fully reflected during the activation of neuron nodes with high-dimensional input data on discrete low-dimensional structures (Shown in Fig.3), which also provides a convenient way to observe and describe the dispersion of continuous high-dimensional spatial data.

![FIGURE 3. Map relationship between input space and output space in SOM.](image)

If the input space is D-dimensional (e.g., there are D input units), the input pattern can be written as \( x = \{x_i; i = 1, \ldots, D\} \), and the connection weights between input unit \( i \) and neuron \( j \) at the computational layer can be written as \( w_{ij} = \{w_{ij}; j = 1, \ldots, N; i = 1, \ldots, D\} \), where \( N \) in here is the total number of neurons. The squared Euclidean distance between the input vector \( x \) and the weight vector \( w_j \) of each neuron \( j \) is evaluated as follows:

\[
d_j(x) = \sum_{i=1}^{D} (x_i - w_{ij})^2	ag{9}
\]

The neuron with the weight vector closest to the input vector (i.e., most similar to it) is declared the winner-takes-all neuron in one certain time step.

Next, the neighboring neurons are adjusted by the quantified topological neighborhood definition, and the expression is as follows:

\[
T_{ij}(x) = \exp\left(-\frac{s_{ij}^2}{2\sigma^2}\right)	ag{10}
\]

where, \( S_{ij} \) is the lateral distance between neurons \( i \) and \( j \) on the neuron grid, \( I(x) \) is the index of the winning neuron; and \( \sigma \) is a variable which decays with time, and a common time-dependent relationship is exponential decay.

With the help of topological neighborhoods, the weights of each neuron are adaptively adjusted:

\[
\Delta w_{ij} = \eta(t) \cdot T_{ij}(x) \cdot (x_i - w_{ij})	ag{11}
\]

where the learning rate:

\[
\eta(t) = \eta_0 \cdot \exp(-t/\tau_\eta)	ag{12}
\]

The effect of each learning weight update is to move the weight vector \( w_i \) of the winning neuron and its neighbors toward the input vector \( x \). Over this process, the iteration results in a topologically ordered network.

If the parameters \((\sigma, \tau_\sigma, \tau_0, \tau_\eta)\) are chosen correctly, a completely unordered initial state can be started from and the SOM algorithm will gradually make the activation pattern representation obtained from the input space ordered.

2) UNIFORM DESIGN

Uniform design, also known as uniform design test method, was proposed by Fang[44], initially to solve the practical problem in which the trial number of orthogonal tests increases substantially with the number of factors and levels, which can also generate more dispersed combinatorial forms in the feasible domain space with the same number of factors and levels.

The practical application is realized by consulting the uniform design schedule, and there are mainly two methods, namely, the good lattice point method and the integer congruent power method to generate the uniform design schedule. The former is used in this paper, and its specific operation rules are as follows:

1. Determine the trial number \( n \), which is the initial population particle number in the initial population generation problem, find all integers \( h \) which are mutually prime with \( n \) in the integer domain of \([1, n]\), and compose the set \( h = (h_1, h_2, \ldots, h_n) \).
2. Determine the recursive values of each column of the uniform design schedule with each subsequent column of the row by congruence calculation with the following formula:

\[
u_{ij} = i \cdot h_i \cdot \lfloor \text{mod } n \rfloor \tag{13}
\]

\[
u_{1j} = h_j \tag{14}
\]

\[
u_{i+1,j} = \begin{cases} u_{ij} + h_i & \text{if } u_{ij} + h_i \leq n \\ u_{ij} + h_i - n & \text{if } \text{otherwise } i = 1, 2, \ldots, n - 1 \end{cases} \tag{15}
\]

The generated uniform design schedule can be noted as \( U_n(n)^l \). where the \( l \) is the number of levels.

3. It should be noted that according to the Euler function \( E(n) \) deduction in the field of number theory, it is known that the maximum number of factors under a certain number
of trials or initial population particles is a fixed value, which means that particles of finite size cannot be completely uniformly distributed in the high-dimensional solution space. But compared with orthogonal design, the uniform design has achieved a more efficient distribution form, and the maximum number of factors allowed under a certain number of trials or initial population particles can be further determined by the Euler function \( E(n) \), i.e., they can be uniformly scattered in the highest dimension of the corresponding solution vector space, as follows:

\[
E(n) = n \cdot (1 - \frac{1}{p_1}) \cdot (1 - \frac{1}{p_2}) \cdots (1 - \frac{1}{p_k})
\]

(16)

for \( n \) is the power of a prime number or the product of square powers of different prime numbers, \( p_1, \ldots, p_k \) are prime numbers less than \( n \).

\[
E(n) = n
\]

(17)

for \( n \) is a prime number.

4. If the maximum number of factors of the uniform design schedule which can be generated by a specified number of dimensions is less than the number of target dimensions, a compliant uniform design schedule can be generated by intercepting a uniform design schedule of higher dimensionality, and to distinguish it from conventionally generated uniform design schedule, which is noted as \( U_{n,n}^d(n) \).

3) SOMGPSO PROCESS

As shown in the explanation of SOM network, this method can reduce the dimensionality of the original data and fully preserve the topological relationships and distances between the original data, which provides the possibility to analyze and visualize the closeness between particles of long length in the particle population and the distribution pattern of all particles, and to determine whether the particle population is aggregated with increasing number of iterations for the adoption of a timely adjustment strategy.

In this paper, a SOMGPSO algorithm is proposed to improve the performance of the GPSO algorithm. The purpose of the SOMGPSO algorithm is to solve the global minimum of fitness functions in a \( d \)-dimensional feasible domain. The SOMGPSO algorithm can be decomposed into three main phases: the linear decreasing inertia weight (LDIW) phase, the SOM recognition and reorganization phase, and the adaptive inertia weight and acceleration coefficients phase. First, let the \( d \)-dimensional particle swarm with population size \( m \) iterate a certain number of times in the process of LDIW particle swarm algorithm, mark and record the trajectories of \( G_{best} \) in the order of iterations, import the trajectories into the SOM neural network with specific parameters to automatically cluster and generate the low dimensional patterns (2 dimensions), determine whether abnormal aggregation of particle trajectories occurs in previous phases based on the difference in excitability of neuron nodes, and use the generated pattern weights as the basis, combine with the uniform design schedule of the same size as abnormal particles, reorganize the particle trajectories to generate the adjusted particle swarm, and submit to the adaptive particle swarm process to complete the remaining iterations. The steps involved in SOMGPSO algorithm are given below. Assume a \( d \)-dimensional objective function \( f(x) \) needs to be optimized:

| Algorithm | SOMGPSO: LDIW Phase |
|-----------|---------------------|
| **Input** | Necessary hyperparameters. |
| **Output** | \( G_{best} \) trajectory (raw data sample of \( L \times d \) size.) |
| 1 | Randomly initialize the velocity and position vectors of particle \( i \) (\( i = 1, 2, \ldots, m \)) and the vectors can be presented as: \( V_i(0) = [v_{i1}, v_{i2}, \ldots, v_{id}] \) \( X_i(0) = [x_{i1}, x_{i2}, \ldots, x_{id}] \) |
| 2 | Calculate the objective function \( f(x) \) by the position vector \( X_i(0) \) for each particle \( i \). |
| 3 | Initialize the position vector:
| \( P_{best,i}(0) = [p_{i1,1}, p_{i2,2}, \ldots, p_{ipl,dl}] \) = \( X_i(0) \) |
| 4 | Search the global best position vector \( G_{best}(0) \) in \( P_{best}(0) \). |
| **Update** | Renew the iteration, \( t = 1, 2, \ldots, N_{iter} \) |
| 5 | In iteration \( t \), the velocity vectors of particles are updated as follows:
| \( V_i(t) = w(t)V_i(t - 1) \)
| \(+ c_1r_1(t)[P_{best,i}(t - 1) - X_i(t - 1)] \)
| \(+ c_2r_2(t)[G_{best}(t - 1) - X_i(t - 1)] \)
| And the position vectors are updated as follows:
| \( X_i(t) = X_i(t - 1) + V_i(t) \)
| where the \( w(t) \) follows the varying rule:
| \( w(t) = w_{start} - (w_{start} - w_{end})d/L_{iter} \)
| Generally, \( w_{start} = 0.9, w_{end} = 0.4 \). |
| 6 | For each particle \( i \), the renewed objective function \( f(x) \) is calculated by the position vector \( X_i(t) \). |
| 7 | The \( P_{best,i} \) and \( G_{best} \) are updated as follows:
| \( P_{best,i}(t) = \)
| \( \{ P_{best,i}(t - 1), \text{ if } f(P_{best,i}(t - 1)) \leq f(X_i(t)) \), \)
| \( X_i(t), \text{ otherwise} \) \)
| \( G_{best}(t) = \arg \min \{ f(P_{best}(t)) \} \) |
| 8 | Record \( G_{best}(i) \) with \( f(G_{best}(i)) \) by iteration number. |
| **Stop** | \( t = L_{iter} \) |
Input
- \( G_{\text{best}} \) trajectory,
- Necessary hyperparameters.

Output
- Adjusted population \( \tilde{X} \), Hit frequency map,
- Nodes’ neighborhood distance.

9. Put \( G_{\text{best}} \) trajectory samples into SOM network with preset parameters.

10. Output the sample hit frequency of each neuron nodes, the weight of the neuron nodes, and the neighborhood distance of the neuron nodes. According to the output, the nodes corresponding to the abnormal particle aggregation can be identified by determining whether the sample hit reaches a preset threshold and whether the node is an abnormal node.

11. Once a node called \( a \) is determined to be an anomaly, all particles (assume \( b \) in all) corresponding to the node \( a \) are replaced according to the following results: \( w_a \cdot U_a(n) / l \), where \( w_a \) is the abnormal node’s weight and new swarm is a matrix with \( b \times d \) size.

12. The reconstituted trajectory data samples are combined with the remaining part of data samples, and \( m \times d \) data samples are randomly selected from the reorganized \( L \times d \) samples as the initial population for the next round of GPSO process according to the preset population size while retaining the final global optimal particles of the previous phase. It can be noted the selected populations as \( \tilde{X} \).

**FIGURE 4.** Correspondence between particles and nodes, with red node representing anomalous node which gathers more particles, whose surrounding nodes (dark blue) tend to counterpart to certain particles as well, and both node weights and particles have high closeness.
FIGURE 5. Process of SOMGPSO, including LDIW phase (left side), SOM recognition and reorganization phase (upper left side), adaptive weight and acceleration coefficients phase (bottom left side), the latter two phases have the multiple strategies to induce the particles to avoid the local optimums when they are flying.

| Algorithm | SOMGPSO: Adaptive Phase |
|-----------|--------------------------|
| Input     | Adjusted population $X$, Necessary hyperparameters $g_{\text{best}}(t)$, $f(g_{\text{best}}(t))$ |
| Output    | $G_{\text{best}}(t)$, $f(G_{\text{best}}(t))$ |
| 13 Randomly initialize the velocity vectors of particle $i$ ($i = 1, 2, \ldots, m$) and the vectors can be presented as: |
|           | $V_i(L_{\text{iter}} + 1) = [v_{i1}, v_{i2}, \ldots, v_{id}]$ |
| 14 Calculate the objective function $f(x)$ by the position vector $\bar{X}_i(L_{\text{iter}} + 1)$ for each particle $i$. |
| 15 Initialize the position vector $P_{\text{best},i}(L_{\text{iter}} + 1)$ as step 3. |
| 16 Same as step 4. |
| Update    | Renew the iteration, $t = L_{\text{iter}} + 2, \ldots, N_{\text{iter}}$ |
| 17 In iteration $t$, the velocity vectors of particles are updated as step 5. While the $w(t)$ and $c(t)$ follow the varying rule, if the $P_{\text{best},i}(t)$ has been modified in this iteration: |
|           | $c(t) = \max[0, c(t - 1) - 1]$, |
|           | $w(t) = \begin{cases} 2 \cdot w(t - 1), & \text{if } c(t) < 2 \\ \frac{w(t-1)}{2}, & \text{if } c(t) > 5 \end{cases}$, |
|           | Otherwise: $c(t) = c(t) + 1$ |
| 18 Same as step 6. |
| 19 Same as step 7. |
| Stop      | $t = N_{\text{iter}}$ where $N_{\text{iter}}$ is the total number of iterations, which is guided by users or automatic stop mechanism such as tolerance value. |

The explanation figure of SOMGPSO algorithm can be shown as Fig. 5.

The number of competing layer SOM neuron nodes determines the granularity and size of the model, which has a significant influence on the accuracy and generalization ability of the model. To ensure that the finest possible granularity is obtained with adapted input samples, and to
construct a square output layer for uniform observation and analysis, the competing layer nodes’ number is taken as $\sqrt{\frac{m}{2}}$.

The threshold for judging whether the node is abnormal should also be defined and customed for different situations: around the main excited node, there are a certain number of accompanying activated nodes due to neighborhood regulation as shown in Fig.4, and these accompanying nodes tend to gather particles of considerable quantity, the weights of which are also very close to that of the central excited node. When the certain size of SOM network and the input population of sample is obtained, the judgement threshold can be defined by the ratio of the above two. And according to the observation, $0.1m$ can be chosen as the threshold when the competing layer nodes obey the size above.

It is worth noting that the mechanism of adaptive phase in SOMGPSO is different from that for integrate weight in [45], which is also named as Adaptive Particle Swarm Optimization. For distinguishing, the latter is called as Adaptive Weight Global Particle Swarm Optimization (AWGPSO) in the below and is chosen as one of the control groups in some trail projects.

It is easy to find that the additional adjustment mechanism and the multiple weight varying modes are the features which distinguish the proposed GPSO from other improved GPSOs. The reason for choosing two inertial weight change mechanisms is to eliminate the "mode memory" in the previous motion mode as much as possible, which aims at avoiding the particle trajectory to overlap with the original trajectory, and avoiding repeated searches for particles or falling back into local minimums.

Further observing the specific mechanism of SOM network in the process, after being activated by the inputs, each node will adjust its own position with reference to the spatial position of the particle, and its topological distribution will contain the information of the distribution of the particle after reaching the terminational iterations. It is feasible to analysis the aggregate form of particles via the SOM neighbor weight distances figures.

As shown in Fig.6, it can be found that the nodes have been divided into several parts (at least 3 parts), of which the same parts would gather more similar particles, and the particles would hold different while they are on the nodes with long weight distances. Therefore, the weight distances figures would give the direct distribution performance of particle swarm.

It is worth noting that the incorporation of coding and decoding measures makes certain difference between the pure SOMGPSO flow and which used in application of carbon reduction, the encoding and decoding of mechanical configurations cut the original flow and insert themselves into the intermediate process of SOMGPSO.
Fig. 8 shows that the flow of SOMGPSO which used in the final optimization of selected case, while there are some necessary discussions about feasibility and superiority of pure SOMGPSO before it is incorporated and adopted to execute the applied calculation.

The design of computational flow of the SOMGPSO algorithm proposed in this paper is an iterative validated process. A major and a relatively isolated works in the design of computational flow are introduced below, and the improvement of SOMGPSO over the traditional improved GPSO algorithm in different dimensional computational tasks is initially demonstrated. A full analysis and comparison are presented in the next chapter in the benchmark function trial section.

The SOMGPSO algorithm programs of all the trials in this paper is implemented by MATLAB R2021a in the personal computer with the following specifications: AMD Ryzen 7 4800H with Radeon Graphics 2.90 GHz. RAM of 16 GB and Windows 10 H.20 64bit.

4) NECESSITY OF RETAINING THE GLOBAL BEST PARTICLE OF LDIW PHASE

Regarding the retention of the global best particle of LDIW phase, it can be first noted that according to the neighborhood weight update mechanism of the SOM network, the difference in the weights of neighboring nodes around the anomalous neural node is very small, and further direct observation shows that when the iteration number reaches a certain value, the weights of neuron nodes are very close to those of the trajectory particles. If the trajectory particles aggregate in the middle and latter stages, the global best particle will have enough "trace particles" to map itself, and if the trajectory particles aggregate in the early stages or do not aggregate throughout the process, the possibility of the existence of trace particles for the global best particle decreases dramatically, even the global best particle is isolated from other particles because of the lack of trace particles, and there is a risk that it will be completely erased in the reorganizing operation. In order to continue the calculation results of LDIW phase, it is necessary to keep the global best particle of LDIW phase exclusively when reorganizing the generation of modified populations. It is worth mentioning that this is not contrary to the desired goal of avoiding the final result of GPSO operation to fall into a local optimum.

In order to fully recognize the effect of the complete erasure of particles on convergence form of the full process and final computational results, an extreme control approach by adopting this operation for the control group is taken: no corresponding recognition and reorganization operations are performed, and the corrected populations distributed in the space of feasible domains are generated directly by a uniform design schedule.

Four test functions are selected for the control test, $d = 30$, and every test function would be tested $N_{run} = 20$ independently under two models.

Fig. 9 shows a result comparison of the Rastrigin benchmark function, and it is easy to see that the control group (shown by the blue line) processed more iterations and also showed a deterioration in convergence. The unusual elevation of the convergence curve in the middle indicates that the previous premium fitness value solution computed in the LDIW stage is completely discarded and latter fitness value is never converged to the previous level in the subsequent iterations.

Twenty independent control trials are conducted on each of the four benchmark functions in both modes, resulting in the following statistical results listed in Table 2.

Where, WFV (Worst Fitness Value), BFV (Best Fitness Value) and MFV (Mean Fitness Value) are the worst fitness value, best fitness value and the average of the fitness values
among the 20 independent tests respectively, and MFEV (Minimum Function Error Value) is the absolute value of the difference between the theoretical minimum value of the benchmark function and the MFV, as follows:

$$MFEV = |\min f(x) - MFV|$$  \hspace{1cm} (18)

where, WI (Worst Iteration), BI (Best Iteration) and MI (Mean Iteration) represent the largest iteration, the smallest iteration and the mean of the number of iterations among the 20 independent trials.

From the two tables, the adverse effects brought by the erasure of the global best particles in the LDIW phase to the final calculation results can be more comprehensively perceived: firstly, the degradation of average performance level, in the calculation of the Griewank benchmark function, the MFV of the two differs by an order of magnitude, and there are significant differences in the MFV calculation results of the remaining three benchmark functions. In addition, the stability deterioration is also an adverse effect brought by the erasure of the global best particles in the LDIW phase, and this effect can be confirmed by the difference in MSE.

The effect of erasure on the iterations is not as pronounced as that on the fitness values, but the adverse effect of the removal of global best particles in the LDIW phase on the stability of the iterations number is observable in the work on the computation of several benchmark functions, indicating that the retention of the global best particles in LDIW phase is indispensable for maintaining the robustness of the algorithm throughout the process.

From the perspective of maintaining the coherence of search process, the LDIW phase's global best particles can be regarded as the control point linking the two phases of the swarm search process before and after reorganization operation. If the valid search information of the previous phase is completely eliminated, the search processes of the two modes are mutually isolated swarm search optimization processes, rather than a complete continuous. The above control experimental results and process analysis conclude that the protective retention of LDIW phase's global best particles is an extremely important and indispensable part of the SOMGPSO algorithm.

### TABLE 2. Statistical results of fitness values for two modes in 4 benchmark functions.

| Function   | Retain | WFV | BFV | MFV | MFEV | Mean Square Error (MSE) |
|------------|--------|-----|-----|-----|------|------------------------|
| Schwef     | Retain | 3597.200 | 2353.600 | 2969.450 | 615.850 | 8955201.023 |
|            | No     | 4739.100 | 3088.600 | 4003.050 | 914.450 | 16345478.625 |
| Rosenbrock | Retain | 29.424 | 22.278 | 27.279 | 5.001 | 746.708 |
|            | No     | 156.257 | 22.705 | 69.820 | 47.116 | 6737.181 |
| Rastrigin  | Retain | 111.535 | 27.862 | 60.403 | 32.541 | 4149.454 |
|            | No     | 137.253 | 88.576 | 121.878 | 33.302 | 15073.948 |
| Griewank   | Retain | 0.013 | 9.042e-05 | 0.003 | 0.003 | 2.750e-05 |
|            | No     | 0.040 | 5.197e-04 | 0.019 | 0.019 | 5.125e-04 |

### TABLE 3. Statistical results of iterations for two modes in 4 benchmark functions.

| Function   | Retain | WI | BI | MI | MSE |
|------------|--------|----|----|----|-----|
| Schwef     | Retain | 441 | 355 | 404.200 | 164033.900 |
|            | No     | 447 | 314 | 383.750 | 4450858.500 |
| Rosenbrock | Retain | 336 | 141 | 198.850 | 1242904.500 |
|            | No     | 331 | 125 | 188.900 | 1195791 |
| Rastrigin  | Retain | 250 | 139 | 180.850 | 996031.500 |
|            | No     | 270 | 163 | 203.800 | 1271865 |
| Griewank   | Retain | 300 | 260 | 278.800 | 2335527 |
|            | No     | 315 | 279 | 293.600 | 2589093 |

5) **ITERATION’S SELECT OF SOM**

The closeness of the weight of activated node to corresponding particle vectors on the node is directly linked to the SOM network's iteration. If the iteration is too small, it is difficult for the weight to converge to the true value of particle vectors, and if the iteration is too large, the computational overhead will be significantly increased. The iteration of SOM is a predefined parameter which is completely independent of other operations and parameters in the whole process, and it only plays a direct role in the clustering analysis and adjustment efficiency of the SOM network, thus it is necessary to evaluate and determine the optimal value.

The mean Eulerian distance $\mu_{RD}$ between the node weights and the particle vector values is calculated within the nodes under different iterations, and the mean iteration times $\bar{t}$ is recorded as the basis for the evaluation of the optimal number of iterations, as follows:
\[ \mu_{E|a} = \frac{1}{I_a} \sum_{i=1}^{I_a} \sum_{j=1}^{d} (w_{a,i,j}^2 - p_{a,i,j}^2) \]  
\[ \mu_{ED} = \frac{1}{A} \sum_{a=1}^{A} \mu_{E|a} \]  

where, \( w_{a,i,j} \) represents the \( j \)-th component of the weight vector of the activated node \( a \); \( p_{a,i,j} \) represents the \( i \)-th particle vector on the activated node \( a \) of the \( j \)-th component; \( I_a \) is the total number of particles on activated node \( a \); \( A \) is the total number of activated nodes.

The Schwef function \((d = 10,20,30)\) is selected as the benchmark function for the experiment, and the values between 10 and 140 at intervals of 10 are examined. Each value would be tested five times independently in different dimensions, and its \( \mu_{E|a} \) value is recorded along with the MSE of the \( \mu_{ED} \). The results of the trials are listed in Table 4.

### Table 4. Statistical results of fit level for different iterations in 10, 20, 30 dimensions.

| D   | 10      | 20      | 30      |
|-----|---------|---------|---------|
|     | Iterations | \( \mu_{ED} \) | MSE(\( \mu_{ED} \)) | \( \mu_{ED} \) | MSE(\( \mu_{ED} \)) | \( \mu_{ED} \) | MSE(\( \mu_{ED} \)) |
| 10  | 19.3121 | 770.9371 | 10.8893 | 329.5190 | 19.4730 | 559.0028 |
| 20  | 2.1010  | 6.9064  | 11.9381 | 549.6299 | 13.9070 | 356.8363 |
| 30  | 0.7311  | 0.7699  | 1.9236  | 6.8578   | 11.9016 | 167.2845 |
| 40  | 2.5748  | 21.5853 | 0.6425  | 0.5577   | 10.4312 | 215.6799 |
| 50  | 0.9353  | 1.3780  | 0.4280  | 0.3570   | 5.4760  | 43.2522  |
| 60  | 0.5311  | 0.3818  | 0.4615  | 0.2598   | 5.6429  | 96.2115  |
| 70  | 0.3035  | 0.1456  | 0.5937  | 0.4267   | 2.7699  | 13.6279  |
| 80  | 0.3330  | 0.1365  | 0.5409  | 0.3457   | 1.6285  | 4.2797   |
| 90  | 0.5563  | 0.3999  | 0.4291  | 0.2448   | 0.8869  | 1.1954   |
| 100 | 0.3634  | 0.1729  | 0.3265  | 0.1548   | 0.4797  | 0.4154   |
| 110 | 0.3708  | 0.2143  | 0.2775  | 0.1377   | 0.9361  | 0.9797   |
| 120 | 0.5004  | 0.2749  | 0.2612  | 0.1180   | 0.6733  | 0.6463   |
| 130 | 0.3677  | 0.2275  | 0.3913  | 0.1824   | 0.6429  | 0.6382   |
| 140 | 0.2380  | 0.0939  | 0.4454  | 0.2305   | 0.6429  | 0.6392   |

The iteration time of the SOM network with its corresponding MSE are recorded in the same way, and the results are listed in Table 5.

### Table 5. Statistical results of convergence time for different iterations in 10, 20, 30 dimensions.

| D   | 10      | 20      | 30      |
|-----|---------|---------|---------|
|     | Iterations | \( t \) (s) | MSE(\( t \)) | \( t \) | MSE(\( t \)) | \( t \) | MSE(\( t \)) |
| 10  | 0.5523  | 0.3085  | 0.5914  | 0.3503  | 0.6352  | 0.4046  |
| 20  | 0.6828  | 0.4704  | 0.7188  | 0.5184  | 0.8578  | 0.7401  |
| 30  | 0.8680  | 0.7586  | 0.9086  | 0.8262  | 1.1617  | 1.3534  |
| 40  | 0.9242  | 0.8617  | 1.0617  | 1.1283  | 1.4047  | 1.9756  |
| 50  | 1.2375  | 1.5343  | 1.2266  | 1.5707  | 1.6476  | 2.7289  |
| 60  | 1.4266  | 2.0391  | 1.3852  | 1.9294  | 1.9258  | 3.7222  |
| 70  | 1.5609  | 2.4482  | 1.5320  | 2.3597  | 2.0250  | 4.1091  |
| 80  | 1.6266  | 2.6251  | 1.7211  | 2.9669  | 2.1828  | 4.7826  |
| 90  | 1.7797  | 3.1750  | 1.8703  | 3.5103  | 2.3117  | 5.3546  |
| 100 | 1.9719  | 3.8965  | 1.8930  | 3.5915  | 2.4305  | 5.9398  |
| 110 | 1.9680  | 3.8775  | 1.8891  | 3.5747  | 2.5172  | 6.3422  |
| 120 | 2.0805  | 4.3420  | 2.1359  | 4.5742  | 2.6492  | 7.0254  |
| 130 | 2.1789  | 4.7595  | 2.2773  | 5.1995  | 2.8437  | 8.1044  |
| 140 | 2.1711  | 4.7216  | 2.3320  | 5.4447  | 2.8977  | 8.4826  |

From the above two tables, it is easy to see that the accuracy and stability of the SOM network weights’ closeness on the particle vectors within the fitted nodes are significantly improved at the beginning with the growth of iteration. However, this improvement becomes less significant after a certain number of iterations, and even performance degradation occurs due to overfitting. While abnormal fluctuations in computation time develop in an approximately linear trend, which is fully reflected in all three dimensions. Considered the results together,
iteration=100 is sufficient to achieve the required fitting accuracy in a short time and with high stability for the computation tasks under d ≤ 30

6) COMPARISON OF PERFORMANCE IMPROVEMENT IN DIFFERENT DIMENSIONS

In order to get a preliminary understanding of the advantages of the proposed SOMGPSO algorithm over the traditional improved GPSOs, three non-convex functions are first selected for benchmark trials with dimensions of 10, 20, 30, and each algorithm is tested 20 times independently with a fixed number of dimensions. The iterations of the whole process obey the following rules: firstly, the number of iterations of the SOMGPSO algorithm is discussed, the fixed number of iterations $L_{iter}$ of the LDIW phase is taken as $\text{Max} \{m, 2.5d\}$ as required above guidelines and recommendations, and the iteration termination mechanism of the AW phase is controlled jointly by the adaptation value tolerance and the maximum number of iterations. The fitness value tolerance is taken as $10e-6$ (10e-3 for Rosenbrock), and the maximum iteration is 20d (only for the AW phase). Acting as the control group, the number of iterations of AWGPSO and LDIWGPSO is $\text{Max} \{10d, NSOMGPSO\}$, while $NSOMGPSO$ is the total iteration of SOMGPSO’s processes, and its performance results are counted in Table 6.

| Function | D | Type | WFV | BFV | MFV | MEV | MSE |
|----------|---|------|-----|-----|-----|-----|-----|
|          | 10 | SOMGPSO | 1.7383e-05 | 1.7823e-07 | 2.0827e-06 | 1.9045e-06 | 1.9131e-11 |
|          | 20 | AWGPSO | 0.1370 | 4.4203e-10 | 0.0362 | 0.0362 | 0.0041 |
|          | 30 | LDIWGPSO | 0.1369 | 6.3619e-12 | 0.0182 | 0.0182 | 0.0023 |
|          | 10 | AWGPSO | 0.0895 | 3.4386e-06 | 0.0134 | 0.0134 | 0.0012 |
|          | 20 | LDIWGPSO | 1.3019 | 0.0184 | 0.4336 | 0.4152 | 0.3226 |
|          | 30 | SOMGPSO | 1.6048 | 0.0013 | 0.5408 | 0.5395 | 0.6892 |
|          | 10 | AWGPSO | 0.1791 | 1.2493e-05 | 0.0672 | 0.0672 | 0.0084 |
|          | 20 | LDIWGPSO | 1.9863 | 0.4789 | 1.1874 | 0.7085 | 1.5679 |
|          | 30 | SOMGPSO | 2.7232 | 0.1715 | 1.1428 | 0.9712 | 1.8421 |
|          | 10 | AWGPSO | 0.2285 | 0.0148 | 0.0065 | 0.0051 | 0.0052 |
|          | 20 | LDIWGPSO | 0.9770 | 0.0887 | 0.3432 | 0.2545 | 0.1646 |
|          | 30 | SOMGPSO | 0.6590 | 0.0739 | 0.2622 | 0.1883 | 0.0896 |
|          | 10 | AWGPSO | 0.3366 | 0.0387 | 0.1862 | 0.1475 | 0.0418 |
|          | 20 | LDIWGPSO | 0.2525 | 0.0394 | 0.1395 | 0.1001 | 0.0233 |
|          | 30 | SOMGPSO | 0.3874 | 1.093q-04 | 0.0085 | 0.0083 | 0.1759e-04 |
|          | 10 | AWGPSO | 0.6906 | 0.3654 | 0.5526 | 0.1872 | 0.3123 |
|          | 20 | LDIWGPSO | 0.8299 | 0.3286 | 0.5729 | 0.2443 | 0.3464 |
|          | 30 | SOMGPSO | 0.4873 | 0.0396 | 0.1057 | 0.0661 | 0.0208 |
|          | 10 | AWGPSO | 1.2463e-06 | 0.2582 | 12.4266 | 12.1684 | 825.1668 |
|          | 20 | LDIWGPSO | 0.1988 | 1.8323e-10 | 7.3467 | 7.3467 | 735.077 |
|          | 30 | SOMGPSO | 0.3906 | 0.0453 | 1.2091 | 1.1638 | 19.3976 |
|          | 10 | AWGPSO | 0.4078 | 18.0311 | 20.5252 | 2.4941 | 443.6084 |
|          | 20 | LDIWGPSO | 0.4648 | 0.0413 | 0.5408 | 0.5395 | 0.6892 |
|          | 30 | SOMGPSO | 0.1986 | 0.4789 | 1.1874 | 0.7085 | 1.5679 |
|          | 10 | AWGPSO | 0.0322 | 4.2975e-05 | 0.0123 | 0.0123 | 2.0731e-04 |
|          | 20 | LDIWGPSO | 0.3366 | 0.0387 | 0.1862 | 0.1475 | 0.0418 |
|          | 30 | SOMGPSO | 0.2525 | 0.0394 | 0.1395 | 0.1001 | 0.0233 |
|          | 10 | AWGPSO | 0.0174 | 1.093q-04 | 0.0085 | 0.0083 | 0.1759e-04 |
|          | 20 | LDIWGPSO | 0.6906 | 0.3654 | 0.5526 | 0.1872 | 0.3123 |
|          | 30 | SOMGPSO | 0.8299 | 0.3286 | 0.5729 | 0.2443 | 0.3464 |
|          | 10 | AWGPSO | 0.4873 | 0.0396 | 0.1057 | 0.0661 | 0.0208 |
|          | 20 | LDIWGPSO | 1.2463e-06 | 0.2582 | 12.4266 | 12.1684 | 825.1668 |
|          | 30 | SOMGPSO | 0.1986 | 0.4789 | 1.1874 | 0.7085 | 1.5679 |
|          | 10 | AWGPSO | 0.4078 | 18.0311 | 20.5252 | 2.4941 | 443.6084 |
|          | 20 | LDIWGPSO | 0.4648 | 0.0413 | 0.5408 | 0.5395 | 0.6892 |
|          | 30 | SOMGPSO | 0.1986 | 0.4789 | 1.1874 | 0.7085 | 1.5679 |

Overall, the search performance of SOMGPSO is much higher than that of AWGPSO and LDIWGPSO, the two improved GPSO algorithms. But when $d=10$, the search performance advantage of SOMGPSO over LDIWGPSO is not significant, and even the optimal value searched by SOMGPSO is much larger than that by LDIWGPSO in separate tests. But with the increase of dimensionality, this performance difference is significantly magnified, and the difference of MFV is often reflected in the order of magnitude.

Further to observe the change of stability with dimensionality. Although all the three algorithms show an elevated MSE in the high-dimensional computation task, SOMGPSO presents the smallest increment, especially in the process of the optimization-seeking calculation for Rosenbrock function, while the MSE of AWGPSO and LDIWGPSO always maintain a high level, which represents the optimization-seeking fluctuations of the two. The MSE of the proposed SOMGPSO remains within 10.0000 in all cases, except for the MSE=19.3976 at $d=20$, and thus has relatively high stability. In summary, the proposed SOMGPSO will have higher viability and superiority when the dimensionality of computational model is higher.

D. APPLICATION OF SOMGPSO TO BENCHMARK FUNCTIONS

The carbon emission objective function of the tunnel construction machinery system under various constraints becomes non-convex and possesses multiple local optima. Meanwhile, in order to provide a comprehensive and equal comparison to demonstrate the validity and usability of the proposed SOMGPSO, ten benchmark functions were selected from the CEC benchmark function set. Here, a brief description of these benchmark functions is given and more comprehensively the performance of SOMGPSO compared
with AWGPSO and LDIWGPSO are investigated.

1) PERFORMANCE MEASURE
In order to verify the advantages of proposed algorithm in terms of computational accuracy, stability, and convergence characteristics, the following performance metrics are considered, some of which have been mentioned previously:

1. For a more reasonable evaluation, the number of function evaluations (NFE) are chosen in the next experiments to measure the computational complexity of the algorithm, which is the number of times the fitness function \( f(x) \) is evaluated in a single run of the algorithm, as follows:

\[
NFE = m \times N
\]  
(21)

where, \( m \) is the population of particle swarm and \( N \) is the full process iteration of the algorithm.
2. Best fitness value (BFV): definition as above.
3. Worst fitness value (WFV): definition as above.
4. Mean fitness value (MFV): definition as above.
5. Mean error fitness value (MEFV): definition as above.
6. Success rate (SR): the algorithm is defined successful if the MFEV of each function falls below the “threshold error”. SR is used to measure the stability of the algorithm, as follows:

\[
SR = \frac{Number \ of \ success}{N_{run}} \times 100\% \quad (22)
\]

7. Reliability rate (RR): the average of SR in all benchmark functions for one algorithm.

2) BENCHMARK FUNCTIONS
In this study, the 10 benchmark functions listed in Table 7 are used for performance comparison of single-objective global optimization algorithms and are all for minimization tasks. Except for \( f_1, f_2 \) and \( f_3 \), all benchmark functions are non-convex, and the typical representatives are: Rosenbrock function (\( f_4, f_5 \)), Griewank function (\( f_6 \)), Ackley function (\( f_7 \)), Levy function (\( f_8 \)), Rastrigin function (\( f_9 \)), and Schwefel function (\( f_{10} \)), which are also named as multimodal functions. The “threshold error” value of each function is available in the table too, the solution of each function is judged successful, when the algorithm reaches to a value smaller than “threshold error”, in other words, the algorithm can be seen that it passes the trail.

### TABLE 7. Benchmark functions chosen in the trail and evaluation.

| \( f \) | Name          | Function                                                                 | Threshold Error | Optimum | Minimum |
|-------|----------------|--------------------------------------------------------------------------|-----------------|---------|---------|
| \( f_1 \) | Sphere        | \( \sum_{i=1}^{d} x_i^2 \)                                              | 0.1             | \( x_i = 0 \) | 0       |
| \( f_2 \) | Shifted Sphere | \( \sum_{i=1}^{d} x_i^2 + 390 \)                                       | 0.1             | \( x_i = 0 \) | 390     |
| \( f_3 \) | Sum Squares   | \( \sum_{i=1}^{d} x_i^2 \)                                              | 0.1             | \( x_i = 0 \) | 0       |
| \( f_4 \) | Rosenbrock    | \( \sum_{i=1}^{d-1} 100(x_{i+1}^2 - x_i^2)^2 + (1 - x_i)^2 \)            | 1               | \( x_i = 1 \) | 0       |
| \( f_5 \) | Shifted Rosenbrock | \( \sum_{i=1}^{d-1} 100(x_{i+1}^2 - x_i^2)^2 + (1 - x_i)^2 + 390 \)    | 1               | \( x_i = 1 \) | 390     |
| \( f_6 \) | Griewank      | \( \sum_{i=1}^{d} x_i^2 - \prod_{i=1}^{d} \cos \left( \frac{x_i}{\sqrt{i}} \right) + 1 \) | 1               | \( x_i = 0 \) | 0       |
| \( f_7 \) | Ackley        | \( -20 \exp \left( -0.2 \sqrt{\frac{1}{d} \sum_{i=1}^{d} x_i^2} \right) - \exp \left( \frac{1}{d} \sum_{i=1}^{d} \cos(2\pi x_i) \right) + 20 + e \) | 1               | \( x_i = 0 \) | 0       |
3) PROJECTS OF PERFORMANCE MEASURES AND EXPERIMENTAL SETUP

For a reasonable and fair comparison of the proposed SOMGPSO with the traditional improved GPSO, an explicit arrangement of the particle swarm optimization process parameters is made: for all the algorithms, the dimension \( d = 30 \), the population of swarm \( m = 50 \), \( c_1 = c_2 = 2.00 \) when \( c \) under the inflexible strategy or is set as the initial values. For LDIWPSO and the LDI phase of SOMGPSO, \( w_{\text{start}} = 0.9 \) while \( w_{\text{end}} = 0.4 \), for the AWGPSO, \( w_{\text{max}} = 0.9 \), \( w_{\text{min}} = 0.4 \) and \( v_{\text{max}} = 6.0 \) are considered to be the recommended configurations of hyperparameters.

The stop iterations also obey the above rules, while the iteration of LDIW phase of SOMGPSO \( L_{\text{iter}} \) obeys the rule of Max \( [2d, m] \) to ensure the sufficiency of executed particles.

**TABLE 8.** Statistical results comparison of fitness values in 3 GPSOs.

| Function | Type   | WFV  | BFV   | MFV   | MFEV  | MSE     |
|----------|--------|------|-------|-------|-------|---------|
| F1       | SOMGPSO | 0.3027 | 1.64e-07 | 0.0338 | 0.0338 | 0.0085  |
|          | AWGPSO  | 8.4975 | 1.3694 | 4.4700 | 3.1006 | 24.0951 |
|          | LDIWGPSO | 2.7402 | 0.2281 | 0.7469 | 0.5188 | 0.8657  |
| F2       | SOMGPSO | 390.2497 | 390.0001 | 390.0319 | 0.0318 | 152124.9000 |
|          | AWGPSO  | 399.1276 | 392.2276 | 394.7588 | 2.5312 | 155838.3000 |
|          | LDIWGPSO | 392.0286 | 392.2508 | 391.0311 | 0.7803 | 152905.5000 |
| F3       | SOMGPSO | 1.2578 | 0.0011 | 0.0858 | 0.0847 | 0.0805  |
|          | AWGPSO  | 135.5069 | 24.5244 | 73.4356 | 48.9112 | 6311.9950 |
|          | LDIWGPSO | 50.9948 | 6.5675 | 21.6954 | 15.1279 | 587.3996 |
| F4       | SOMGPSO | 4.1444 | 0.0087 | 0.7836 | 0.7749 | 1.4305  |
|          | AWGPSO  | 218.3274 | 27.8817 | 52.2932 | 24.4115 | 5467.9440 |
|          | LDIWGPSO | 104.4255 | 25.7072 | 35.2315 | 9.5243 | 1705.2240 |
| F5       | SOMGPSO | 392.6430 | 390.0002 | 390.5971 | 0.5969 | 152566.5000 |
|          | AWGPSO  | 482.0848 | 417.7218 | 424.1593 | 6.4375 | 180092  |
|          | LDIWGPSO | 485.2711 | 413.0423 | 424.0975 | 11.0552 | 180233  |
| F6       | SOMGPSO | 0.0863 | 0.0002 | 0.0098 | 0.0097 | 0.0005  |
|          | AWGPSO  | 0.6513 | 0.2505 | 0.4761 | 0.2256 | 0.2384  |
|          | LDIWGPSO | 0.7825 | 0.2718 | 0.5407 | 0.2689 | 0.3120  |
| F7       | SOMGPSO | 2.0920 | 0.0001 | 0.2120 | 0.2118 | 0.2765  |
|          | AWGPSO  | 3.1285 | 0.0114 | 0.9806 | 0.9692 | 1.6493  |
|          | LDIWGPSO | 4.0894 | 0.0358 | 1.2172 | 1.1814 | 3.2930  |
| F8       | SOMGPSO | 0.5446 | 0.0031 | 0.1430 | 0.1399 | 0.0530  |
|          | AWGPSO  | 15.3606 | 1.7879 | 8.4443 | 6.6564 | 83.4194 |
|          | LDIWGPSO | 16.7443 | 1.2313 | 7.5736 | 4.5063 | 51.4239 |
| F9       | SOMGPSO | 181.5861 | 6.9647 | 48.4547 | 41.4900 | 2673.1590 |
|          | AWGPSO  | 173.5034 | 47.4453 | 101.2604 | 53.8151 | 11585.5800 |
|          | LDIWGPSO | 160.7612 | 33.9076 | 90.3496 | 56.4420 | 9195.5210 |
| F10      | SOMGPSO | 4367.9000 | 2182.8000 | 3149.4800 | 3149.4800 | 10248225 |
|          | AWGPSO  | 9440.9000 | 5026.8000 | 6891.5550 | 6891.5550 | 48672957 |
|          | LDIWGPSO | 9089.4000 | 4288.5000 | 6193.3600 | 6193.3600 | 39555246 |

**TABLE 9.** Statistical results comparison of numbers of function evaluations in 3 GPSOs.

| Function | Type   | WNFE  | BNFE  | MNFE  | MSE    |
|----------|--------|-------|-------|-------|--------|

3
| Function | Algorithm | F1 | F2 | F3 | F4 | F5 | F6 | F7 | F8 | F9 | F10 |
|----------|-----------|----|----|----|----|----|----|----|----|----|-----|
|          | SOMGPSO   | 37250 | 9750 | 21850 | 37250 | 21300 | 21300 | 303800 | 18300 | 61350 | 44450 |
|          | AWGPSO    | 37250 | 9500 | 21850 | 15000 | 21300 | 15000 | 129850 | 15000 | 10900 | 15000 |
|          | LDIWGPSO  | 37250 | 9500 | 21850 | 15000 | 21300 | 15000 | 129850 | 15000 | 10900 | 15000 |
|          | SOMGPSO   | 21300 | 16957.5 | 24637.5 | 7700 | 16957.5 | 18757.5 | 299725 | 299725 | 27865 | 27865 |
|          | AWGPSO    | 21300 | 18032.5 | 24637.5 | 15000 | 18032.5 | 18757.5 | 299725 | 299725 | 27865 | 27865 |
|          | LDIWGPSO  | 21300 | 18032.5 | 24637.5 | 15000 | 18032.5 | 18757.5 | 299725 | 299725 | 27865 | 27865 |
|          | SOMGPSO   | 21850 | 18495 | 24637.5 | 7700 | 18495 | 18757.5 | 299725 | 299725 | 27865 | 27865 |
|          | AWGPSO    | 21850 | 18757.5 | 24637.5 | 15000 | 18757.5 | 18757.5 | 299725 | 299725 | 27865 | 27865 |
|          | LDIWGPSO  | 21850 | 18757.5 | 24637.5 | 15000 | 18757.5 | 18757.5 | 299725 | 299725 | 27865 | 27865 |
|          | SOMGPSO   | 21300 | 9750 | 21850 | 37250 | 21300 | 21300 | 303800 | 18300 | 61350 | 44450 |
|          | AWGPSO    | 21300 | 9500 | 21850 | 15000 | 21300 | 15000 | 129850 | 15000 | 10900 | 15000 |
|          | LDIWGPSO  | 21300 | 9500 | 21850 | 15000 | 21300 | 15000 | 129850 | 15000 | 10900 | 15000 |

**f1-Sphere:**
- SOMGPSO
- AWGPSO
- LDIWGPSO

**f2-ShiftedSphere:**
- SOMGPSO
- AWGPSO
- LDIWGPSO
4) COMPARISON IN TERMS OF FITNESS VALUES
From Table 8, it can be seen that in the AWGPSO and LDIWGPSO algorithms, BFV, and MFV have quite differences from the theoretical minim of the ten benchmark functions. Whereas in the proposed SOMGPSO, the final results fit well with the theoretical minims. In AWGPSO and LDIWGPSO, most of the MFEV of both fail to be smaller than the threshold error of the benchmark functions, and only 3 MFEV values fit with the threshold error for AWGPSO and 2 for LDIWGPSO. While 9 MFEV values reach to the threshold error of benchmark functions, which represents higher reliability in performance. However, it should be noted that all algorithms, including SOMGPSO, do not perform well on f9 (relatively) and f10, which deserves further analysis.

5) SUCCESS RATE & RELIABILITY RATE
Based on the two metrics of SR and RR, the stability differences of the three algorithms are studied and discussed further. The SR and RR of three GPSO algorithms can be seen in Table 10 (20 independent runs).

| Function | SOMGPSO | AWGPSO | LDIWGPSO |
|----------|---------|--------|-----------|
| f1       | 0.900   | 0      | 0         |
| f2       | 0.900   | 0      | 0         |
| f3       | 0.950   | 0      | 0         |
| f4       | 0.800   | 0      | 0         |
| f5       | 0.850   | 0      | 0         |
| f6       | 1       | 1      | 1         |
| f7       | 0.950   | 0.650 | 0.500     |
| f8       | 1       | 0      | 0         |
| f9       | 1       | 0.550 | 0.650     |
| f10      | 0       | 0      | 0         |
| RR       | 0.835   | 0.220 | 0.215     |

From Table 10, it can be seen that except f10, SOMGPSO achieves considerable success in these benchmark functions, while AWGPSO and LDIWGPSO only have non-zero SR in part of these functions. In terms of RR values, SOMGPSO is significantly higher than the latter two methods. There is now further evidence of the higher performance stability of SOMGPSO relative to conventional modified GPSO.

6) CONVERGENCE CHARACTERISTICS
Since the mechanisms for terminating iterations of the three algorithms are highly linked, the WFV levels of the three algorithms are the same on all of these benchmark functions. To verify whether SOMGPSO has a faster convergence rate, the f1-f3 benchmark functions are chosen such that on the relatively simple convex optimization solution task, the algorithms can complete the solution with a higher probability within the maximum iteration of the separate processes, which in turn magnifies the gap for observational evaluation. From the solution results of f1-f3, it is not difficult to find that SOMGPSO can have less NFE in reaching the optimal solutions. When it comes to the non-convex tasks, such as f6-f8 and f10 (although it has not reached the goal but shows better solutions relatively), this advantage has been maintained even the stop iterations setting may be little unreasonable for this control group test.

In summary, the proposed SOMGPSO has higher accuracy, stability and faster convergence rate compared with the traditional improved GPSO algorithm.

E. ENGINEERING APPLICATION

1) CASE OVERVIEW
The F4 fault section of Wushaoling Tunnel is located in the south of Mountain Wushaoling, its main fault zone is 200 meters long, and both sides of the main zone are the fault influence zones, with the total length of the whole section of 450 meters. The influence zone is dominated by fractured rock, the stability is poor, and the palm face is prone to collapse during excavation, but there is no obvious groundwater. The main fault zone is dominated by faulted mud gravel, breccia, which is judged as VI-V level surrounding rock, and the stability is poor. The main influence zone of the fault is dominated by faulted mud gravel and angular gravel, judged as grade VI-V surrounding rock, poor stability, which is easy to collapse during excavation, and there is strong high ground stress extrusion, but no obvious groundwater existed. After the long bench method used for the construction of No. 5 inclined shaft and led to a series of engineering problems such as abnormal convergence deformation and support damage, the main cavern was changed to the horseshoe-shaped section with the elevation arch applied as the three-benches method for construction[46]. In this paper, the configuration of the machineries in this section of the circular construction process is optimized in order to obtain a balanced configuration which achieves the schedule, and carbon emission.

2) CASE ANALYSIS
According to the excavation and support process flow, the main processes which control the single cycle time can be further divided into over-support works, drilling and blasting works, excavation and ballast works, loading works and support works.

After the completion of the upper section overrun support, drilling and jacking operations, the drilling and explosives loading operations of the upper, middle and lower sections are carried out at the same time, and then the light surface blasting and smoke removal operations are carried out, which directly require rock drills and air compressors respectively.

The first digging and loading operation are in a single part of cycle, digging ballast for the upper and middle sections are at the same time, the second digging and loading operation in the lower section are after the completion of digging and ballast work is in the upper and middle sections, the machineries required for this phase are excavator (digging and loading machine), and dump truck.

The spray concrete operations for upper section start at the end of the cycle, and the machineries required at this time are concrete spraying machine, concrete transfer pump truck, and automatic metering mixing machine.

The over-support for upper section parallels to the concrete spraying operation of lower section, which needs to use the concrete sprayers, and concrete transfer pump truck. The digging ballast and loading operation is divided into two successive parts, which are parallel to the installation of the steel frames, anchor rods and reinforcing mesh, and need to mobilize the drilling machine and grouting pump. The upper section for the concrete spraying operation parallels to the lower section for the steel frame and anchor rod installation operation, which needs to mobilize the drilling machine and grouting pump. To keep safe, after light surface blasting, the ventilation work also needs to use the axial flow fan.

3) COLLECTION OF NECESSARY DATA
After deconstructing all the steps in the cycle and confirming the corresponding machines and equipment to be put into, the machinery and equipment available on site are counted, and their quantitative consumption performance is obtained based on consultation with the engineering staff involved in the project or upstream suppliers involved in equipment service support, as well as by reviewing organizational process assets (OPA) and relevant literature [46], as shown in Table 11.

### Table 11. Machinery systems and units available in studied case.

| System                       | Machinery name          | Specification    | Work efficiency | Available quantity |
|------------------------------|-------------------------|------------------|-----------------|--------------------|
| Drilling blasting            | Rock drill              | YT-28            | 3-4 piles/h     | 20                 |
|                             | Axial flow fan          | SDF(C)N0-12.5    | 220 kW          | 1                  |
| Anchor spraying support      | Wet spraying machine    | TK961            | 1.6 m³/h        | 5                  |
|                             | Metering mixing machine | HZD25            | 25 m³/h         | 3                  |
|                             | Concrete transfer pump  | HBT60.7.75.ZA    | 67 m³/h         | 1                  |
|                             | Concrete transfer pump  | HBT60            | 60 m³/h         | 1                  |
|                             | Double-liquid grouting  | KBY50/70         | 50 m³/h         | 3                  |
| Excavation& transportation   | Backhoe loader          | ITC312           | 300 m³/h        | 4                  |
|                             | Dump truck              | VOLVO-FM7        | 19.5 t/cycle    | 7                  |
|                             | Bucket excavator        | CAT320           | 320 m³/h        | 2                  |
| Energy supply& drive         | Electric air compressor | 4L-20/8          | 12 m³/min       | 15                 |
|                             | Internal combustion generator | HC-C450 | 400 kW         | 5                  |

The carbon emissions of mechanical energy consumption should be considered from three perspectives: energy

This work is licensed under a Creative Commons Attribution 4.0 License. For more information, see https://creativecommons.org/licenses/by/4.0/
production, transportation and use. Since the transportation and production of energy in different regions causes differences in carbon emissions, in this paper the Chinese localized life cycle basis database (CLCD) is mainly considered which is embedded in the software. Since the geographical differences of carbon emissions from energy use are not significant enough, the default values from CEAD are considered. The results of carbon emission factors for gasoline, diesel, and electricity can be obtained by integrating and aggregating the three main GHGs, as shown in Table 12.

### Table 12: Carbon emission factor of energy in studied case

| GHG type          | Carbon dioxide (kg) | Methane (kg) | Nitrogen oxides (kg) | CO2 equivalent (kg) |
|-------------------|---------------------|--------------|----------------------|--------------------|
| Process           | Production & Use    | Production & Use | Production & Use |
| Gasoline          | 0.481               | 2.910        | 0.022                | 4.31e-5            |
| Diesel            | 0.375               | 3.100        | 0.021                | 4.27e-5            |
| Electricity       | 0.940               | 0            | 0.003                | 1.43e-5            |

Combining the two tables above, the carbon emissions per unit shift can be calculated for all the above machines which need to be used, it should be noted that the time interval is per shift.

### Table 13: Shift consumptions and CO2 equivalents of machineries in studied case

| Machinery name       | Specification | Air (m³/min) | Diesel (L) | Electricity (kW-h) | CO2 equivalent (kg/unit shift) |
|----------------------|---------------|--------------|------------|--------------------|-------------------------------|
| Rock drill           | YT-28         | 4.896        | -          | -                  | -                             |
| Axial flow fan       | SDF(C)N0-12.5 | -            | -          | 18.3               | 18.75                         |
| Wet spraying machine | TK961         | -            | -          | 24.64              | 24.87                         |
| Metering mixing machine | HZD25      | -            | -          | 158.9              | 164.5                         |
| Concrete transfer pump | HBT60.7.75.ZA | -         | 54.4       | -                  | 217.75                        |
| Concrete transfer pump | HBT60       | -            | 47.7       | -                  | 190.1                         |
| Double-liquid grouting pump | KBY50/70 | -            | 103.6      | -                  | 413.3                         |
| Backhoe loader       | ITC312        | -            | 60.9       | 338.7              | 582.1                         |
| Dump truck           | VOLVO-FM7     | -            | 77.9       | -                  | 310.8                         |
| Dump truck           | BAIC Benz     | -            | 76.2       | -                  | 304.05                        |
| Bucket excavator     | CAT320        | -            | 133.3      | -                  | 531.87                        |
| Electric air compressor | 4L-20/8  | -            | -          | -                  | 684.173                       |
| Internal combustion generator | HC-C450 | -            | 915.2      | 1117.2             | 3651.6                        |

### III. Results

With the incorporation of coding and decoding measures, the SOMGPSO algorithm process becomes more complex, with an increase in the number of loop iterations nested within it, which in turn increases its computational complexity.

It is necessary to select and implement adapted parameter setting rules: for the whole process, the swarm population $m=50$, which is sufficient for the search task when the particle length (dimension) is below 40. For LDIW and adaptive weight (AW) phases, iterations $L_{iter}$ and $N_{iter}$ are both set as 50, in which $c_1$ and $c_2$ are also set as 2.05 when they are under the inflexible control, and the weights varying can be still under the rule as above.

For this case, 35 objects of 5 types of machineries were selected as adjustable variables directly, and the supply and drive machines followed the other adjust rule as mentioned, which include 2 types, thus the particle length is 35.

Considering the optimization model mentioned, the preferred weights can be determined by trial-and-error method, and the final selection was $k_1=0.15$ while $k_2=0.85$. 20 independent calculations were performed and the configuration which corresponds to the best performance was selected as the final optimal configuration.
TABLE 14. Calculation results of 20 independent runs trials.

| Run | Fitness value |
|-----|---------------|
| 1   | 0.344110      |
| 2   | 0.343569      |
| 3   | 0.528867      |
| 4   | 0.311586      |
| 5   | 0.387332      |
| 6   | 0.390066      |
| 7   | 0.311116      |
| 8   | 0.388209      |
| 9   | 0.391642      |
| 10  | 0.456096      |
| 11  | 0.391642      |
| 12  | 0.406853      |
| 13  | 0.387332      |
| 14  | 0.534413      |
| 15  | 0.391104      |
| 16  | 0.456096      |
| 17  | 0.393006      |
| 18  | 0.339701      |
| 19  | 0.401170      |
| 20  | 0.462715      |

It is worth noting that under selected condition, the fitness values of objective function in 20 independent trials maintain continuous fluctuation rather than convergence to a fixed or a relatively fixed state. Actually, when the preferred weights configuration was selected as (0.1,0.9) or (0.2, 0.8) and other configurations, there would be several similar values and even all the fitness value keeping similar to each other, which indicates that the local optimum has been avoid as far as possible with current preferred weights configuration of (0.15,0.85).

TABLE 15. Statistical results of 20 independent runs trials.

| WFV | BFV | MFV | MSE  |
|-----|-----|-----|------|
| 0.5344 | 0.3111 | 0.4010 | 0.1644 |

Table 15 shows the statistical results of independent trials, which furtherly indicate the fluctuation effect existing in this condition, however, it still cannot prove that the best fitness value and idealist configuration in theory were certainly obtained via this computing process, which is beyond the ability of all heuristic computing algorithms, while the final target in this paper is to reduce the carbon emission as far as possible, so the calculation results reached the expectation.

The optimized configuration combination is as follows:

TABLE 16. Optimized configuration combination of best fitness value in dependent trials.

| Machinery name | Specification | Original configuration scheme | Optimized configuration scheme | Note |
|----------------|---------------|------------------------------|-------------------------------|------|
| Rock drill     | YT-28         | 7                            | 7                            | Upper bench |
| Axial flow fan | SDF(C)N0-12.5 | 1                            | 1                            |      |
| Wet spraying machine | TK961 | 4                            | 4                            |      |
| Metering mixing machine | HZD25 | 2                            | 2                            |      |
| Concrete transfer pump | HBT60.7.75.ZA | 0                           | 1                            |      |
| Concrete transfer pump | HBT60 | 1                            | 0                            |      |
| Double-liquid grouting pump | KBY50/70 | 2                        | 2                            |      |
| Backhoe loader | ITC312        | 1                            | 1                            |      |
| Dump truck     | VOLVO-FM7     | 6                            | 3                            |      |
| Dump truck     | BAIC Benz     | 6                            | 4                            |      |
| Bucket excavator | CAT320   | 1                            | 1                            |      |
| Electric air compressor | 4L-20/8 | 10                           | 4                            |      |
| Internal combustion generator | HC-C450 | 5                        | 2                            |      |

IV. DISCUSSION

According to the proposed configuration scheme, the theoretical cycle duration is 432 minutes, while the original cycle duration is 660 minutes, and the carbon emission decreases by 88.4% in per cycle compared with the original cycle, which is a significant improvement. The highly compact energy distribution pattern is the main reason for the effective reduction of carbon emissions from the recommended scheme, and by strictly controlling the direction and flow of energy stream and timely shutting down the extra internal combustion engine drives in the system, the direct carbon emissions in the process can be weakened. Besides, the optimization of consumption time is also an important method to control carbon emissions, which appropriately reduces the transport load in the tunnel to eliminate the impact of the blockage effect, and improves the efficiency while avoiding the accumulation of carbon emissions in both ways: the quantity of units and durations.

Actually, for the most tunnel constructions used drill-and-blast methods in China, the environmental effectiveness of mechanical systems for excavation and construction are always ignored, this optimization case can be selected as
reference for subsequent similar studies, while only one case (excavated by three benches methods) was considered in this study, which is insufficient to establish a universal framework for all the reduction of carbon emission based on drill-and-blast methods.

When it comes to the program, SOMGPSO’s code performs slowly under the situation that combines encoder and decoder modules, and causes extra consumption on computing duration due to the nested loops. What is more, the procedure of SOMGPSO is a little tedious compared with other mainstream improved GPSOs, more adaptive strategies and less parameters should be considered in subsequent improvements. It is worth considering that let SOM network recognize and change the particles’ trajectories dynamically and the recognize process can parallel to the GPSO process without interruption and re-engagement. The proposed SOMGPSO requires users to determine the first stage’s iterations by the priori experience, which is extreme inconvenient in the application, the above-mentioned improvement idea would eliminate this defect, while the improvement idea has certain challenges in the design of algorithm’s procedure and programming.

V. CONCLUSION

Further and comprehensive discussion of the main work in each section and findings of this paper is presented:

By setting the control trail of whether retaining the global best particle of LDIW phase or not in SOMGPSO, the necessity of retaining is more significant in terms of final fitness values.

For the parameters’ section of full process of SOMGPSO, except some can be guided by the previous work, the remaining part should be defined by trial-and-error method and SOM’s iteration is selected to be presented in this paper, combining aspects of accuracy and time consumption, 100 is selected as the default value to solve the search task for dimensionality is below 30 with an ideal accuracy performance (0.3634, 0.3265, 0.4797 respectively for $d = 10, 20$ and 30 in the mean of $\mu$) and time consumption (1.9719s, 1.8930s and 2.4305s for $d = 10, 20$ and 30 in the mean of CPU time for calculating in SOM process).

Comprehensive comparisons have been executed in aspects of robustness, accuracy and reliability for proposed SOMGPSO and other improved GPSOs, which finally shows that SOMGPSO holds obvious superiority in above aspects especially in higher dimensional tasks.

When it comes to the case and the final result of optimization for case machinery system, the feasibility of results has been ensured and validated with compact supplement of energy and accurate allocations of machinery system, the proposed configuration is fully achievable according to the analysis, because of the fineness of modeling. However, the more universal framework to quantify and optimize the carbon emission for drill-and-blast tunnels should be proposed in further study.

REFERENCES

[1] L. Huang, G. Krigsvoll, F. Johansen, Y. Liu, and X. Zhang, “Carbon emission of global construction sector,” Renewable and Sustainable Energy Reviews, vol. 81, pp. 1906-1916, January 2018, doi: https://doi.org/10.1016/j.rser.2017.06.001.

[2] Q. Li, Y. Bai, and L. Li, “Study of influential factors and measures for low carbonization during the construction of shield tunnels,” Modern Tunneling Technology, vol. 52, pp. 3-17, March 2015.

[3] J. Xu et al., “Emission transition of greenhouse gases with the surrounding rock weakened–A case study of tunnel construction,” Journal of Cleaner Production, vol. 209, pp. 169-179, February 2019, doi: https://doi.org/10.1016/j.jclepro.2018.09.010.

[4] J. A. Pritchard and J. Preston, “Understanding the contribution of tunnels to the overall energy consumption of and carbon emissions from a railway,” Transportation Research Part D: Transport and Environment, vol. 65, pp. 551-563, December 2018, doi: https://doi.org/10.1016/j.trd.2018.04.029.

[5] L. Pirjola et al., “Exhaust emissions of non-road mobile machine: Real-world and laboratory studies with diesel and HVO fuels,” Fuel, vol. 202, pp. 154-164, 2017, doi: https://doi.org/10.1016/j.fuel.2016.10.048.

[6] X. Liu et al., “Integration of air pollution data collected by mobile measurement to derive a preliminary spatiotemporal air pollution profile from two neighboring German-Czech border villages,” Science of The Total Environment, vol. 722, p. 137632, 2020, doi: https://doi.org/10.1016/j.scitotenv.2019.137632.

[7] S. Abolhasani, H. C. Frey, K. Kim, W. Rasdorf, P. Lewis, and S.-h. Pang, “Real-world in-use activity, fuel use, and emissions for nonroad construction vehicles: a case study for excavators,” Journal of the Air & Waste Management Association, vol. 58, no. 8, pp. 1033-1046, 2008, doi: https://doi.org/10.3155/1047-3299.58.8.1033.

[8] T. Cao et al., “Characterization of the emissions impacts of hybrid excavators with a portable emissions measurement system (PEMS)-based methodology,” Science of The Total Environment, vol. 635, pp. 112-119, 2018, doi: https://doi.org/10.1016/j.scitotenv.2018.04.011.

[9] M. Giunta, D. Lo Bosco, G. Leonardi, and F. Scopelliti, “Estimation of gas and dust emissions in construction sites of a motorway project,” Sustainability, vol. 11, no. 24, p. 7218, 2019, doi: https://doi.org/10.3390/su11247218.

[10] M. Giunta, “Assessment of the impact of co, nox and pm10 on air quality during road construction and operation phases,” Sustainability, vol. 12, no. 24, pp. 10549, 2020, doi: https://doi.org/10.3390/su122410549.

[11] P. Faber, F. Drewnick, and S. Borrmann, “Aerosol particle and trace gas emissions from earthworks, road construction, and asphalt paving in Germany: Emission factors and influence on local air quality,” Atmospheric Environment, vol. 122, pp. 662-671, 2015, doi: https://doi.org/10.1016/j.atmosenv.2015.10.036.

[12] J. Chen, F. Zhao, Z. Liu, X. Ou, and H. Hao, “Greenhouse gas emissions from road construction in China: A province-level analysis,” Journal of cleaner production, vol. 168, pp. 1039-1047, 2017, doi: https://doi.org/10.1016/j.jclepro.2017.08.243.

[13] J. P. M. Miguel, C. S. de Blas, and A. E. G. Sipols, “A forecast air pollution model applied to a hypothetical urban road pricing scheme: An empirical study in Madrid,” Transportation Research Part D: Transport and Environment, vol. 55, pp. 21-38, 2017, doi: https://doi.org/10.1016/j.trd.2017.06.007.

[14] M. Marzouk, E. Abdelkader, and K. Al-Gahtani, “Building information modeling-based model for calculating direct and indirect emissions in construction projects.” Journal of cleaner production, vol. 152, pp. 351-363, 2017, doi: https://doi.org/10.1016/j.jclepro.2017.03.138.

[15] M. Marzouk and E. Abdelkader, “A hybrid fuzzy-optimization method for modeling construction emissions,” Decision science letters, vol. 9, no. 1, pp. 1-20, 2020, doi: https://doi.org/10.5267/j.dsl.2019.9.002.
