International Journal of Mathematics, Statistics and Operations Research
Vol. 2; Number 2; 2022, pp. 255-271
https://doi.org/10.47509/IJMSOR.2022.v02i02.10

ARF INDIA
Academic Open Access Publishing
www.arfjournals.com

A new particle swarm optimization algorithm for identifying the hydraulic conductivity and diffusivity parameters in the unsaturated soil water

Yiyang Wang[a], Leilei Pang[a], Peixin Sun[a], Zhongguo Zhou[a,*], Yun Wang[a,*]

[a]*School of Information Science and Engineering, Shandong Agricultural University, Taian, Shandong 271018, China*

Abstract

In this paper, we present the new particle swarm optimization algorithm by defining the inertia weight which depends on linearly the iteration step for solving the unsaturated soil water problem. By comparing with the exist particle swarm optimizations such as such as PSO algorithm, FPSO algorithm, P-PSO-SA algorithm and IDWPSO algorithm, numerical experiments verify that our proposed algorithm is more efficient and accurate to reach the optimal solution in the multimodal function extremum problem. Combining the characteristic difference method, our algorithm is applied to solve the unsaturated soil water problem. Numerical experiments show that our algorithm is quicker and more precision to approximate the exact solution.

Keywords: Particle swarm optimization; Unsaturated soil water; Inertia weight; Characteristic difference; Parameter identification.

1. Introduction

The unsaturated soil water flow (see [2, 3, 11, 17, 23]) is an important form of flow in porous media and is widely used in atmospheric science, soil science, agricultural engineering, environment engineering and groundwater hydrology, etc. Because of the nonlinearity of water content equations and the complexity of physical parameters and boundary conditions, it is very difficult and impossible to obtain its analytical solution. Some numerical schemes [11, 12, 10] are been developed for solving the unsaturated soil water flow problem. Paper [11] presented the general difference methods for one-dimensional unsaturated soil water flow problem. Paper [10] proposed the efficient reduced-order finite volume element formulation based on proper orthogonal decomposition method for solving two-dimensional unsaturated soil water flow problem. By the Crank-Nicolson extrapolation method, paper [21] presented the time second-order proper orthogonal decomposition method. Paper [9] considered conforming finite element discretizations based on a multiscale formulation along with recently developed, local postprocessing schemes. Paper [5] proposed the enriched Galerkin method,

[*] Corresponding author. Email: zhg_zhou@sdau.edu.cn (Zhongguo Zhou)

Article History
Received : 30 August 2022; Revised : 29 September 2022; Accepted : 16 November 2022; Published : 29 December 2022

To cite this paper

Yiyang Wang, Leilei Pang, Peixin Sun, Zhongguo Zhou & Yun Wang (2022). A new particle swarm optimization algorithm for identifying the hydraulic conductivity and diffusivity parameters in the unsaturated soil water. *International Journal of Mathematics, Statistics and Operations Research.* 2(2), 255-271.

which augments piecewise constant functions to the classical continuous Galerkin finite element method. By employing a coarse partition of the fine grids and multiscale basis function for mapping the fine-scale information to the coarse-scale unknowns, paper [13] proposed a Multiscale Locally Conservative Galerkin (MsLCG) method to accurately simulate multiphase flow in heterogeneous and fractured porous media. Paper [6] proposed an element based post-processing technique through which local conservation can be established. Using the property of local conservation at steady state conditions to define a numerical flux at element boundaries, Paper [14] proposed a locally conservative Galerkin (LCG) finite element method for two-phase flow simulations in heterogeneous porous media.

As is known that the above numerical schemes were solved the the unsaturated soil water flow problem depend on the assumption that the hydraulic conductivity and diffusivity is known. However, the hydraulic conductivity and diffusivity could not given in advance in the unsaturated soil water flow in the porous media which is only the empirical function. Thus, it is very necessary to propose an efficient algorithm to solve the hydraulic conductivity and diffusivity. Particle swarm optimization (PSO) algorithm which had the advantage of fast convergence, good robustness and strong versatility have bee widely used in scientific research and engineering.

With the in-depth application of particle swarm optimization, the limitations of traditional PSO algorithms have been discovered one after another, such as premature convergence or non-convergence, dimensionality disaster, and easy to fall into local extreme values. Paper [1] presented the particle swarm optimization algorithm by using chaotic mapping logic to initialize the population of particle swarm optimization algorithm. Paper [16] proposed an intelligent fuzzy level set method and an improved quantum particle swarm optimization algorithm with global search capability are proposed. Paper [25] developed a new algorithm-Improved quantum evolutionary PSO (IQEPSO) while the learning factor and inertia factor varied with the number of iterations.

Recently, by changing the learning factor synchronously and combining simulated annealing algorithm, paper [8] proposed a balance between particle inertia and optimal behavior by linearly adjusting the learning factor. Paper [20] developed the coupling OD with IPSO (ODIPSO) algorithm. Paper [19] presented an improved multi-objective particle swarm optimization algorithm to solve the workflow scheduling problem. Paper [7] proposed a particle swarm optimization algorithm with two evolutionary operators of multiple crossover operator and swarm operator combined with genetic algorithm. Paper [15] studied a novel multi-objective self-organizing particle swarm optimizer to solve multiple objective functions. Paper [24] presented a stochastic cognitive superiority leading particle swarm optimization (SCDLPSO) algorithm.

The existing particle swarm optimization algorithm for solving the parameters of unsaturated soil water flow is not efficient and accurate, and there exists some difficulty as well. As a result, it is important to propose the efficient particle swarm optimization algorithm. In this paper, we first develop a new inertia weight which depends linearly on the iterations. Our proposed PSO algorithm is rather simple in calculation. Five test functions are used to verify our algorithm by comparing with other four algorithm such as PSO, FPSO, P-PSO-SA and IDWPSO. Due to the advantage in convergence speed and accuracy, our algorithm is applied to solve the unsaturated soil water flow. The one-dimensional unsaturated soil water flow equations (nonlinear parabolic equation) is transformed into the advection diffusion equation by defining $v = \frac{dK(\theta)}{d\theta}$. Applying the characteristic difference method, we propose an efficient swarm optimization algorithm to identify the hydraulic conductivity and diffusivity parameters parameters. Numerical results show that our algorithm is nicer than the classic PSO.

The rest of this paper is organized as follows. In Section 2, a new particle swarm optimization algorithm is presented, where we test our algorithm by comparing with other four PSO algorithm. We apply our algorithm to identifying the hydraulic conductivity and diffusivity parameters in the unsaturated soil water in Section 3. the conclusion is given in Section 4.

2. Particle swarm optimization algorithm

The Particle swarm optimization (PSO) algorithm originally proposed by Kennedy and Eberhart was used to discuss the social behavior of organisms such as birds flocking and fish schooling, coupled with swarm theory. Just like many other evolutionary computation techniques, an initial population of random potential solutions were given at first while the optimal fitness solution was searched according to its optimization mechanism, particle improvement, and swarm evolution, through an iterative computation. The conventional particle swarm optimization algorithm is described as follows,

$$x_{i+1} = x_i + v_i, \tag{1}$$

$$v_{i+1} = w_i v_i + c_1 \text{ rand } \left(pbest_i - x_i\right) + c_2 \text{ rand } \left(gbest_i - x_i\right), \tag{2}$$

$$w_i = w_{\min} + \left(w_{\max} - w_{\min}\right) i/k. \tag{3}$$

where, x_i and v_i is the position and velocity of the particles in the $i-$th iteration, respectively. k is the number of iterations, c_1 and c_2 are learning factors and $pbest_i, gbest_i$ is the historical optimal position and the global optimal position. w_i is the inertia weight factor of the $i-$th iteration and w_{max} and w_{min} are the maximum and minimum inertia weight factors. k is the number of iterations.

As is known that the inertia weight which determines directly the iteration step and the robustness of solutions is an important parameter in particle swarm optimization. As is generally agreed, the larger the search step is and the stronger the global search capability is. On the contrary, the stronger the local search ability of the particle. Thus, it is very challenging and valuable to improve the inertia weight. In this paper, we present a new inertia weight which depends linearly i as follows,

$$w_i = \frac{w_{\min}}{1 - \frac{w_{\max} - w_{\min}}{w_{\min} k}} i, \tag{4}$$

where we take $w_{max} = 0.5$ and $w_{min} = 0.1$.

2.1. Test function

In order to verify the performance of our proposed inertia weight, five test functions are used to test it. Meanwhile, our simulation results are also compared with standard particle swarm optimization[18] (PSO), a particle swarm optimization algorithm based on compression factor[4] (FPSO), a hybrid particle swarm optimization algorithm combined with adaptive inertia weight [25](P-PSO-SA) and a hybrid particle swarm optimization algorithm that dynamically adjusts inertia weight[22] (IDWPSO). The optimal value of each test function was solved 50 times by using these five algorithms respectively. Define absolute error=$\| x^* - x \|_2$, where x^* is *Gbest* value point and x is the optimal value point.

Test function 1. The function with many steep peaks is consider as follow,

$$f\left(x_1, x_2\right) = \sin x_1 \sin x_2 \sin\left(x_1 + x_2\right), \quad x_1, x_2 \in [-4, 4]. \tag{5}$$

By calculating derivatives, we can solve the maximum point $\left(\frac{\pi}{3}, \frac{\pi}{3}\right)$, $\left(\frac{\pi}{3}, -\frac{2\pi}{3}\right)$ and $\left(-\frac{2\pi}{3}, \frac{\pi}{3}\right)$ and the maximum value is $\frac{3\sqrt{3}}{8}$. The 3-D surface of the function (5) is given in Figure 1.

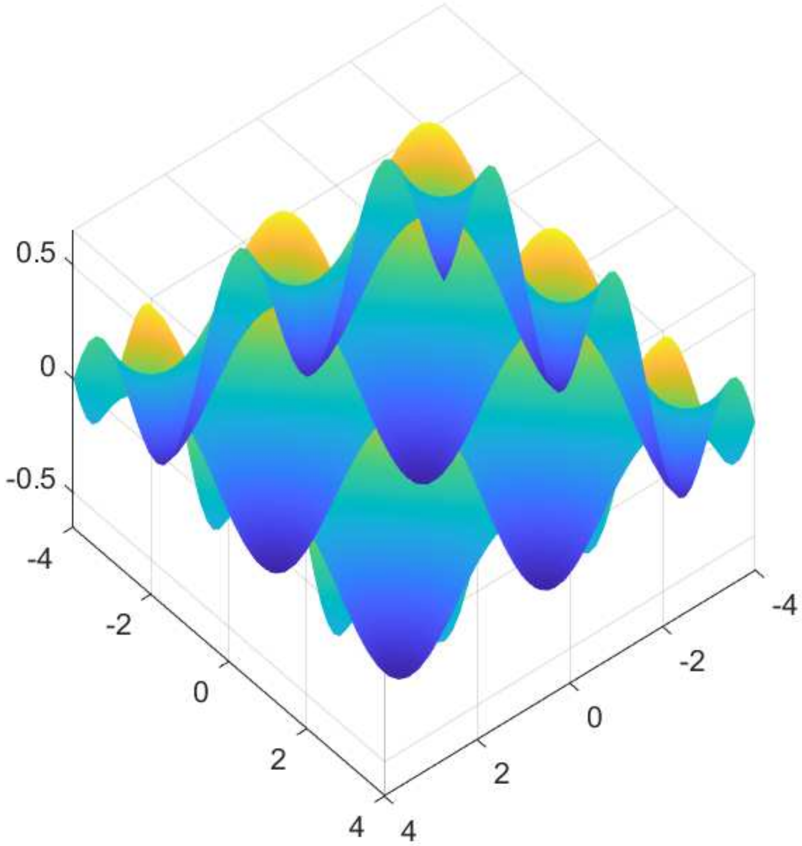

Figure 1: The surface of the function $f(x_1, x_2)$

We can find that there are many local optimum values (peak) by Figure 1. Thus, how to jump out of the local optimum and reach the maximum fast is very important. In the following, we will show our scheme (IIWPSO) is more advantage in convergence speed and accuracy by comparing with the above four particle swarm optimization algorithms in Figure 2 and Table 1.

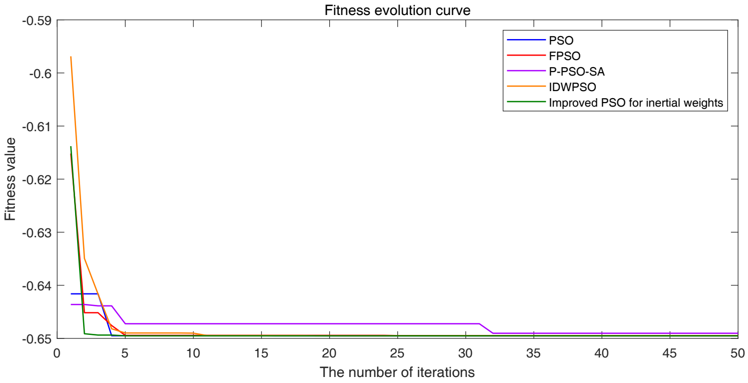

Figure 2: The fitness value of five particle swarm optimization algorithms.

Table 1: The results of five particle swarm optimization algorithms

Methods	Average number of iterations	Average time	Gbest value point	Absolute error
PSO	50	0.0511	(1.047199, -2.094397)	2.3875E-06
FPSO	48	0.0550	(1.043701, -2.093059)	3.7431E-03
P-PSO-SA	32	0.1369	(-2.102588, 1.053358)	1.0251E-02
IDWPSO	50	0.8646	(1.047148, 1.047273)	9.0265E-05
IIWPSO	23	0.0384	(1.047198, -2.094395)	4.6034E-07

We can see that the optimal solutions obtained by P-PSO-SA and IDWPSO are only the local optimal solutions in Table 1. Compared with PSO and FPSO, our IIWPSO algorithm is the fastest in convergence speed and the shortest in cost of time. Thus, our algorithm is superior to other four algorithms in accuracy and convergence.

Test function 2. The global optimum of Rosenbrock function (Figure 3) is located in a long and smooth valley which is difficult to find the optimal solution of the function.

$$f(x_1, x_2) = 100\left(x_2 - x_1^2\right)^2 + \left(1 - x_1\right)^2, \quad x_1, x_2 \in [-5.12, 5.12]. \tag{6}$$

By calculating derivatives, we can solve the minimum point $x = (1,1)$ and the minimum value 0.

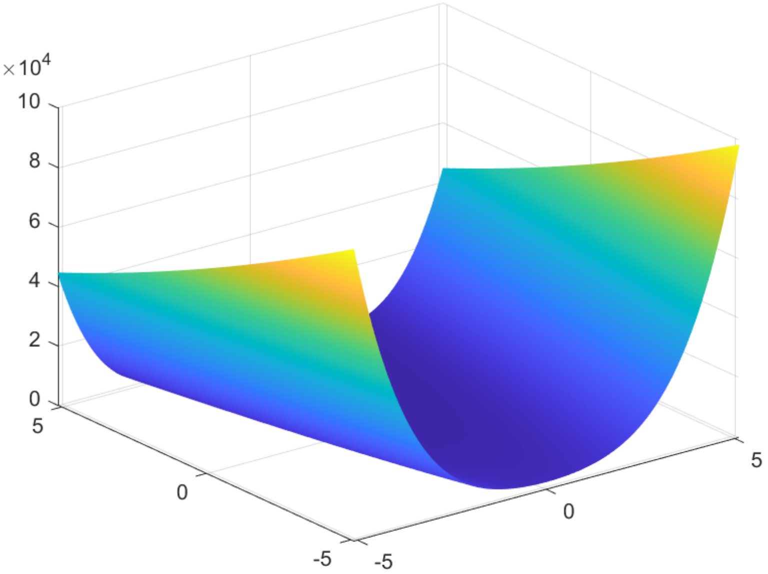

Figure 3: The surface of the function $f(x_1, x_2)$

We see clearly that the global minimum of the function is also located in a parabolic valley which is easier to find, but it is quite difficult to search the global minimum because that the values vary much slower in the valley.

Table 2: The results of five particle swarm optimization algorithms

Methods	Average number of iterations	Average time	Gbest value point	Absolute error
PSO	49	0.0243	(0.999998, 0.999998)	2.8284E-06
FPSO	32	0.0223	(1.008236, 1.017707)	1.9329E-02
P-PSO-SA	27	0.0674	(1.036690, 1.073482)	8.2133E-02
IDWPSO	44	0.0837	(1.000941, 1.002044)	2.2502E-03
IIWPSO	43	0.0256	(1.000000, 1.000000)	0

By Table 2, we can see that our IIWPSO is the fastest to reach the global minimum point.

Test function 3. The Rastrigrin function is a typical nonlinear multi-mode function, and its peak shape shows the appearance of high and low fluctuation and jump.

$$f(x_1, x_2) = 20 + x_1^2 - 10\cos(2\pi x_1) + x_2^2 - 10\cos(2\pi x_2), \quad x_1, x_2 \in [-5.12, 5.12]. \tag{7}$$

The minimum point is (0,0) and the minimum value is 0. The 3-D surface of the function (7) is given in Figure 4.

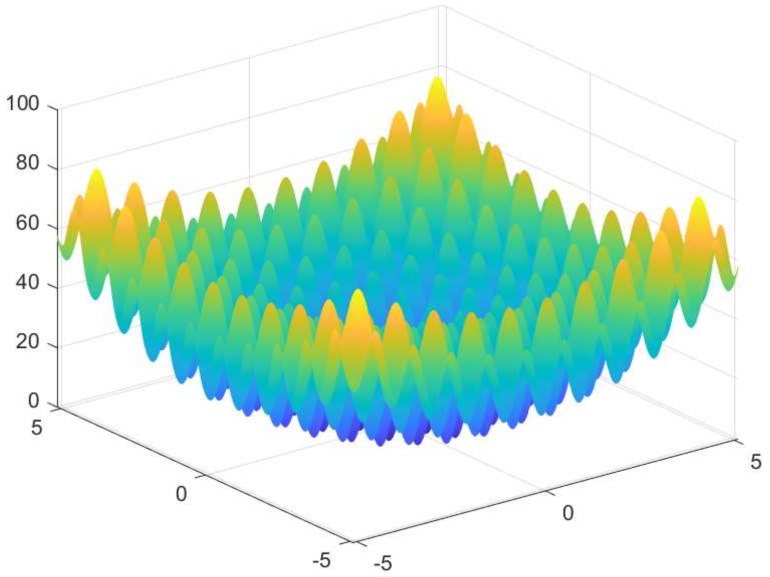

Figure 4: The surface of the function $f(x_1, x_2)$.

Table 3: The results of five particle swarm optimization algorithms

Methods	Average number of iterations	Average time	Gbest value point	Absolute error
PSO	50	0.0259	(0.000002, 0.000002)	2.8284E-06
FPSO	34	0.0211	(0.003568, -0.000075)	3.5688E-03
P-PSO-SA	35	0.0545	(0.004182, 0.000300)	4.1928E-03
IDWPSO	30	0.0480	(0.000037, -0.000170)	1.7398E-04
IIWPSO	23	0.0203	(0.000000, -0.000000)	0

By Table 3, we can see clearly that our IIWPSO reaches the minimum value fastest and the absolute error is 0 by compared with the other four algorithms.

Test function 4. The Griewank functions have many local minima, and the number of local minima is related to the dimension of the problem.

$$f\left(x_1, x_2\right) = \frac{\left(x_1^2+x_2^2\right)}{4000} - \cos x_1 \cos\left(\frac{x_2}{\sqrt{2}}\right) + 1, \quad x_1, x_2 \in [-10, 10]. \tag{8}$$

The 3-D surface of the function is shown in Figure 5 and the minimum point is (0,0) and the minimum value is 0.

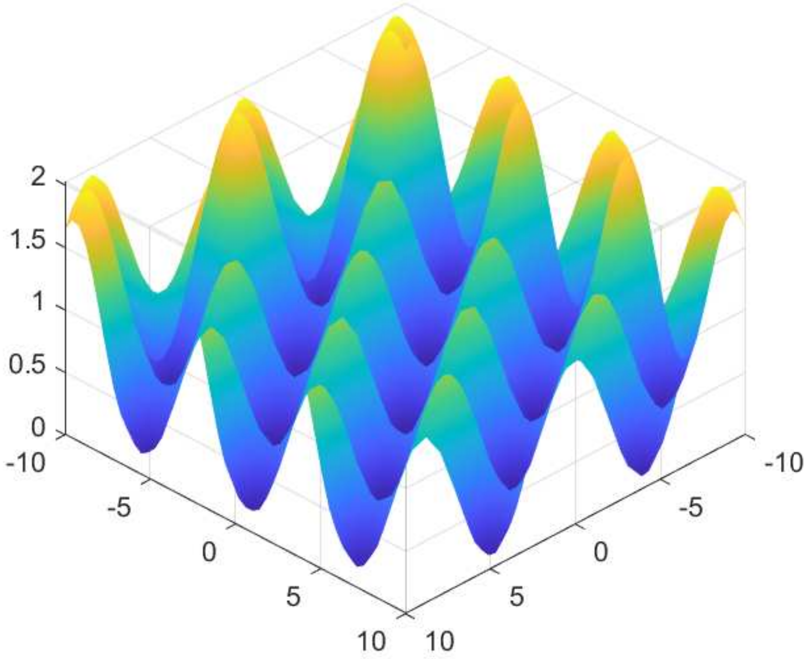

Figure 5: The surface of the function $f(x_1, x_2)$

The Griewank function has many widespread local minima, which are regularly distributed.

Table 4: The results of five particle swarm optimization algorithms

Methods	Average number of iterations	Average time	Gbest value point	Absolute error
PSO	50	0.0213	(0.000000, 0.000000)	0
FPSO	20	0.0225	(-0.000148, -0.003603)	3.6060E-03
P-PSO-SA	50	0.0592	(0.050740, -0.026887)	5.7424E-02
IDWPSO	47	0.0496	(0.000175, 0.000048)	1.8146E-04
IIWPSO	18	0.0228	(0.000000, 0.000000)	0

Table 4 shows that our IIWPSO algorithm changes smoothly in the whole optimization process and jumps out of the constraint of local extreme value quickly to reach the global minimum.

Test function 5. The Beale function is a multi-peak non-convex continuous function defined in two dimensions with sharp peaks in the corner of the input field.

$$f(x_1, x_2) = (1.5 - x_1 + x_1x_2)^2 + (2.25 - x_1 + x_1x_2^2)^2 + (2.625 - x_1 + x_1x_2^3)^2, \quad x_1, x_2 \in [-10, 10]. \tag{9}$$

The minimum point and the minimum value is solved at (3,0.5) and 0, respectively.

Table 5: The results of five particle swarm optimization algorithms

Methods	Average number of iterations	Average time	Gbest value point	Absolute error
PSO	50	0.0193	(2.999999, 0.500000)	1.0000E-06
FPSO	36	0.0192	(3.004467, 0.500143)	4.4693E-03
P-PSO-SA	50	0.0556	(3.012026, 0.508447)	1.4696E-02
IDWPSO	34	0.0472	(3.000248, 0.500009)	2.4816E-04
IIWPSO	39	0.0224	(3.000000, 0.500000)	0

We can find that FPSO, IDWPSO and our IIWPSO algorithms have the similar average speed in Table 5, but only our IIWPSO can reach the minimum point.

3. The unsaturated soil water flow problem

3.1. Equations

Soil water content is an important climate factor, and its seasonal change has an important influence on weather and climate at mid-high latitudes. Hydraulic processes at surface and subsurface, such as precipitation, evaporation, and evapotranspiration, seepage of surface water, and capillary elevation of deep-level water, absorption in root zone and liquid moisture flow of groundwater, all can be reduced to unsaturated flow problems. In fact, in all studies (see papers [2, 3, 17, 23, 11]) of the unsaturated zone, the fluid motion is assumed to obey the classical Richards equation. Based on horizontal resolution of a general circulation model, liquid moisture flow in soil along horizontal direction may be ignored. The one-dimensional unsaturated soil water flow equations with the absorption rate of root are considered as

$$\tfrac{\partial \theta}{\partial t} = \tfrac{\partial}{\partial z}\left(D(\theta)\tfrac{\partial \theta}{\partial z}\right) - \tfrac{\partial K(\theta)}{\partial z} + S_r, \tag{10}$$

where $\theta(z,t)$ is soil moisture, $D(\theta)$ is the soil water diffusivity, $K(\theta)$ is the unsaturated hydraulic conductivity, $-S_r$ is absorption rate of root zone.

Initial conditions are given as

$$\theta = \theta_a, \quad (0, z).$$

The first kind of boundary conditions are given as

$$\begin{cases} \theta = \theta_b, & (t, 0), \\ \theta = \theta_a, & (t, L), \end{cases} \tag{11}$$

The empirical formulas of unsaturated hydraulic conductivity and diffusivity are given as follows,

$$D(\theta) = D_0 \left(\frac{\theta - \theta_r}{\theta_b - \theta_r} \right)^b, \tag{12}$$

$$K(\theta) = K_s \left(\frac{\theta - \theta_r}{\theta_b - \theta_r} \right)^m. \tag{13}$$

In the fact engineering problem, the parameters b and m are empirical constants, which have effect on the quality of the solution. Thus, it is an important to develop the efficient method to predict the parameters b and m.

3.2. Characteristic difference method for solving the soil moisture

The equation (10) is rewritten as the following nonlinear advection-diffusion equations

$$\tfrac{\partial\theta}{\partial t} = \tfrac{\partial}{\partial z}\left(D(\theta)\tfrac{\partial\theta}{\partial z}\right) - \tfrac{dK(\theta)}{d\theta}\tfrac{\partial\theta}{\partial z} + S_r, \quad x \in [0,L],\, t \in [0,T]. \tag{14}$$

Let $\Delta z = \frac{L}{N}$ and $\Delta t = \frac{T}{M}$, M and N are positive integers. Define the staggered meshes as

$$x_i = i\Delta z, \quad i = 0, 1, \cdots, N, \quad x_{i+\frac{1}{2}} = (i + \tfrac{1}{2})\Delta z, \quad i = 0, \cdots, N-1,$$

Let $f_i = f(x_i)$, $f_{i+\frac{1}{2}} = f(x_{i+\frac{1}{2}})$ at the mesh points (x_i) and $(x_{i+\frac{1}{2}})$. Define the following difference operators as

$$\delta_z f_{i+\frac{1}{2}} = \tfrac{f_{i+1}-f_i}{\Delta z}, \quad \delta_z f_i = \tfrac{f_{i+\frac{1}{2}}-f_{i-\frac{1}{2}}}{\Delta z}.$$

Now, we introduce the following nodes,

$$\bar{z}_i = z_i - \tfrac{dK(\theta)_i}{d\theta}\Delta t,\ i = 1, \cdots, N-1.$$

In general, we assume that the point $\bar{z}_i$ falls between z_{i-1} and z_{i+1}. Define $\bar{\Theta}_i^n$ with the quadratic interpolation as follows,

$$\bar{\Theta}_i^n = \tfrac{\lambda_i^2}{2}(\Theta_{i+1}^n + \Theta_{i-1}^n) + (1-\lambda_i^2)\Theta_i^n + \tfrac{\lambda_i}{2}(\Theta_{i+1}^n - \Theta_{i-1}^n), \tag{15}$$

where $\lambda_i = -\frac{dK(\Theta)_i}{d\Theta}\frac{\Delta t}{\Delta z}$. Let $S_r = 0$. Now, we will present our characteristic difference method for solving the the soil water flow as

$$\tfrac{\Theta_i^{n+1}-\bar{\Theta}_i^n}{\Delta t} = \tfrac{D_{i+\frac{1}{2}}^{n+1}\left(\Theta_{i+1}^{n+1}-\Theta_i^{n+1}\right)-D_{i-\frac{1}{2}}^{n+1}\left(\Theta_i^{n+1}-\Theta_{i-1}^{n+1}\right)}{\Delta z^2}, \tag{16}$$

where $D_{i+\frac{1}{2}}^{n+1} = D(\Theta_{i+\frac{1}{2}}^{n+1})$ and $\Theta_{i+\frac{1}{2}}^{n+1} = \frac{\Theta_{i+1}^{n+1}+\Theta_i^{n+1}}{2}$.

The equation (16) can be written as

$$-rD_{i-\frac{1}{2}}^{n+1}\Theta_{i-1}^{n+1} + \left[1 + r\left(\ D_{i-\frac{1}{2}}^{n+1} + D_{i+\frac{1}{2}}^{n+1}\ \right)\right]\Theta_i^{n+1} - rD_{i+\frac{1}{2}}^{n+1}\theta_{i+1}^{n+1} = \bar{\Theta}_i^n, \tag{17}$$

where $r=\frac{\Delta t}{(\Delta z)^2}$. Let

$$\begin{cases} a_i=-rD^{k+1}_{i-\frac{1}{2}}, \quad i=1,2,\cdots,N, \\ b_i=1+r\left(D^{k+1}_{i-\frac{1}{2}}+D^{k+1}_{i+\frac{1}{2}}\right), \quad i=1,2,\cdots,N-1. \end{cases} \tag{18}$$

We can obtain the matrix form of (17) ($\mathbf{A}\ \Theta = \mathbf{H}$), i.e.,

$$\begin{bmatrix} b_1 & a_2 & 0 & 0 & 0 \\ a_2 & b_2 & a_3 & 0 & 0 \\ \ddots & \ddots & \ddots & 0 & 0 \\ 0 & \ddots & \ddots & \ddots & 0 \\ 0 & 0 & a_{N-2} & b_{N-2} & a_{N-1} \\ 0 & 0 & 0 & a_{N-1} & b_{N-1} \end{bmatrix} \begin{bmatrix} \Theta_1^{n+1} \\ \Theta_2^{n+1} \\ \vdots \\ \vdots \\ \Theta_{N-2}^{n+1} \\ \Theta_{N-1}^{n+1} \end{bmatrix} = \begin{bmatrix} h_1 \\ h_2 \\ \vdots \\ \vdots \\ h_{N-2} \\ h_{N-1} \end{bmatrix}, \tag{19}$$

where $h_1=\bar{\Theta}_1^n-a_1\theta_b$, $h_{N-1}=\bar{\Theta}_{N-1}^n-a_{N-1}\theta_a$ and $h_i=\bar{\Theta}_i^n$ $(i=2,3,\cdots,N-1)$.

Remark 1. *By (19), it is obviously that our coefficient matrix* $\mathbf{A}$ *is strictly diagonally dominant* ($\mid b_i \mid > \mid a_i \mid + \mid a_{i+1} \mid$) *and symmetry. Thus, we can prove that* $\mathbf{A}$ *is symmetric positive definite and the scheme (16) exists the only numerical solution.*

3.3. The parameters identification optimization method

In the section, we construct the algorithm for identifying the parameters such as b and m. In order to more convenient and efficient to predict, we assume that θ_i^j is the only exact solution by the characteristic difference. $\theta_i^j(b,m)$ is the numerical solution with the different choice of b and m in the particle swarm optimization which is solved by the characteristic difference. In order to more accuracy to predict the parameter b and m, the following objective function is chosen as

$$\min Z=\sqrt{\sum_{j=1}^{M}\sum_{i=1}^{N-1}\left(\frac{\theta_i^j-\theta_i^j(b,m)}{\theta_i^j(b,m)}\right)^2/M}, \tag{20}$$

and the constraints condition are given as

$$b_{\min}\le b\le b_{\max}, \quad m_{\min}\le m\le m_{\max}. \tag{21}$$

Next, we propose our algorithm as follow,

Step 1. Generating randomly $l-$ particles determined by position vector x_α and velocity vector v_α,

$$x_\alpha=x\left(b_\alpha,m_\alpha\right), \quad v_\alpha=v\left(b_\alpha,m_\alpha\right), \quad \alpha=1,2,\cdots,l. \tag{22}$$

Step 2. Computing the diffusivity and hydraulic conductivity of each particle as

$$D(\Theta)=D_0\left(\frac{\Theta-\theta_r}{\theta_b-\theta_r}\right)^{b_\alpha}, \quad K(\Theta)=K_s\left(\frac{\Theta-\theta_r}{\theta_b-\theta_r}\right)^{m_\alpha}. \tag{23}$$

Step 3. Computing the soil water content $\{\Theta_i^n\}$ by combining the characteristic difference method with the above diffusivity and hydraulic conductivity as

$$\frac{\Theta_i^{n+1}-\bar{\Theta}_i^n}{\Delta t}=\frac{D_{i+\frac{1}{2}}^{n+1}\left(\Theta_{i+1}^{n+1}-\Theta_i^{n+1}\right)-D_{i-\frac{1}{2}}^{n+1}\left(\Theta_i^{n+1}-\Theta_{i-1}^{n+1}\right)}{\Delta z^2}. \tag{24}$$

Step 4. Calculating the fitness value of the each particle Z_α as

$$Z_\alpha=\sqrt{\sum_{n=1}^{M}\sum_{i=1}^{N-1}\left(\frac{\theta_i^n-\Theta_i^n}{\Theta_i^n}\right)^2/M}, \tag{25}$$

where θ_i^n is the exact solution.

Step 5. Identifying the optimal value *pbest* of each particle and the optimal value *gbest* of all particles. By comparing the smaller between Z_α^k and Z_α^{k-1} of each particle in the current $k-$iteration, we determine the new fitness value Z_α^k and the local optimal $pbest_k$. By comparing Z_α^k with all the swarm particle, we determine the optimal value $gbest_k$.

Step 6. The position vector x_k, velocity vector v_k and the inertia weight factor w_k of the each particle in the $k-$iteration are updated as

$$\begin{aligned} x_{k+1}&=x_k+v_k,\\ v_{k+1}&=w_k v_k+c_1\text{ rand }\left(pbest_k-x_k\right)+c_2\text{ rand }\left(gbest_k-x_k\right),\\ w_k&=\frac{w_{\min}}{1-\frac{w_{\max}-w_{\min}}{w_{\min}K}}k. \end{aligned} \tag{26}$$

Step 7. If $k<K$ and $Z_\alpha^k>\varepsilon_0$, go to **Step 2-6.**

3.4. Numerical experiments

In the section, we will verify our algorithm efficiently by some experiments. Table 6 shows the fitness value and the approximation $\bar{b}$ and $\bar{m}$ with the different parameters b and m by comparing our algorithm IIWPSO with PSO when the particle number $l=30$.

Table 6: The PSO and IIWPSO algorithm under the different b and m

b	m	Algorithm	$\bar{b}$	$\bar{m}$	Z
2	10	PSO	2.0006	9.5949	3.5E-03
		IIWPSO	2.0000	9.9440	5.2E-05
4	6	PSO	4.0004	5.7636	2.0E-03
		IIWPSO	3.9999	6.1114	6.7E-04
8	3	PSO	8.0018	2.7622	7.3E-03
		IIWPSO	8.0004	2.9462	1.6E-03
12	7	PSO	11.9999	6.6986	1.4E-03
		IIWPSO	12.0000	7.1438	1.1E-04
18	8	PSO	18.0000	7.8264	8.5E-05
		IIWPSO	18.0000	8.0754	7.6E-05
21	2	PSO	20.6965	2.7163	1.1E-02
		IIWPSO	20.8786	2.1764	6.4E-03

It is seen clearly that the solutions obtained by our IIWPSO algorithm are more accurate than PSO under different measured conditions.

Next, we take $b = 8$ and $m = 3$ as an example to test the fitness value with the iteration k in Table 7.

Table 7: The PSO and IIWPSO algorithm with the different iteration.

k	Algorithm	$\bar{b}$	$\bar{m}$	Z
1	PSO	8.6686	3.2082	1.6
	IIWPSO	8.2340	4.6597	9.5E-01
2	PSO	8.2411	4.7366	9.7E-01
	IIWPSO	8.0228	2.6932	1.7E-01
4	PSO	8.0096	4.9367	1.2E-01
	IIWPSO	8.0030	5.9479	7.7E-02
8	PSO	7.9993	3.2518	3.2E-03
	IIWPSO	7.9996	3.0843	1.4E-03
16	PSO	7.9993	3.2518	3.2E-03
	IIWPSO	8.0003	2.9281	1.0E-03
32	PSO	7.9992	3.1544	3.1E-03
	IIWPSO	8.0000	3.0739	2.8E-03
50	PSO	7.9995	3.1098	1.6E-03
	IIWPSO	8.0002	2.9640	9.9E-04

We can find that the fitness value gets smaller remarkable when k becomes bigger from 1 to 16. Meanwhile the number of the iteration exceeds 16, the fitness value is almost invariable. It is clearly that our IIWPSO algorithm is superior to the PSO algorithm and quicker to approximate the exact solution.

By Figure 6 and Figure 7, we can see clearly that our IIWPSO algorithm is quicker to reach the exact solution. The fitness value reaches to 5.9E-03 when the iteration $k = 2$ while the iteration $k = 16$ by using the PSO algorithm.

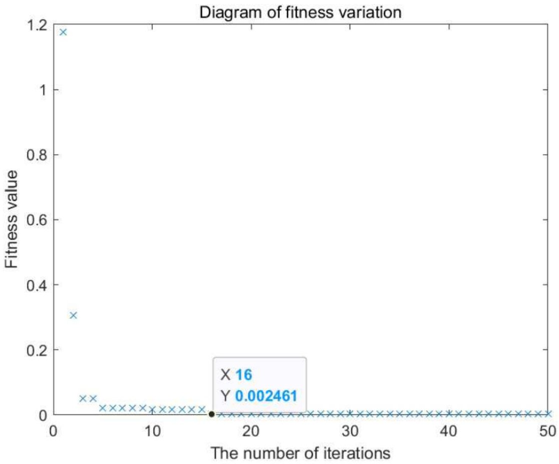

Figure 6: Fitness variation diagram of PSO

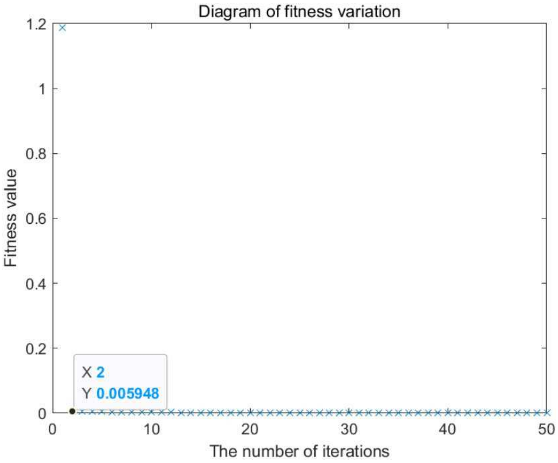

Figure 7: Fitness variation diagram of improved PSO.

Thus, our IIWPSO algorithm can identifying well the parameters such as b and m.

4. Conclusion

In this paper, we propose a new particle swarm algorithm (IIWPSO) for solving the unsaturated soil water problem by combining the characteristic difference method. Firstly, we test the efficiency of our algorithm compared with the other four algorithms such as PSO algorithm, FPSO algorithm, P-PSO-SA algorithm and IDWPSO algorithm. Numerical experiments verify that our IIWPSO algorithm has the faster convergence speed and the better performance in the optimization, and is easier to obtain a suitable solution. Secondly, by combining the characteristic difference method, we apply our IIWPSO to identify the hydraulic conductivity and diffusivity parameters. It is clear that our algorithm has more advantage in simulating the the unsaturated soil water problem and other engineering problem.

Funding

This work was supported partially by Natural Science Foundation of Shandong Government (Grant No. ZR2021MA002) and Shandong Agricultural University. Yiyang Wang, Leilei Pang and Peixin Sun are co-first author.

References

[1] Ahmed G., Sheltami T., Mahmoud A., and Yasar A., 2020. Iod swarms collision avoidance via improved particle swarm optimization. Transportation Research Part A Policy and Practice, 142:260-278.

[2] Bear J., 1972. Dynamics of Fluids in Porous Media, American Elsevier Publishing Company Inc.

[3] Celia M., Bouloutas E. and Zarba R., 1990. A general mass conservation numerical solution for the unsaturated flow equation, Water Resour. Res., 26:1483-1496.

[4] Clerc M., Kennedy J., 2002. The particle swarm-explosion, stability, and convergence in a multidimensional complex space, IEEE T Evolut. Comput., 6:58-73.

[5] Choo J. and Lee S., 2018. Enriched Galerkin finite elements for coupled poromechanics with local mass conservation. Comput. Methods Appl. Mech. Engrg., 341:311-332.

[6] Deng Q., Ginting V., McCaskill B., Torsu P., 2017. A locally conservative stabilized continuous Galerkin finite element method for two-phase flow in poroelastic subsurfaces, J. Comput. Phys.,347:78-98.

[7] Das P. and Jena P., 2020. Multi-robot path planning using improved particle swarm optimization algorithm through novel evolutionary operators, Appl. Soft Comput., 92:106312.

[8] Du Y., Peng Y., Shao S. and Liu B., 2020. Multi-UAV collaborative track planning based on improved particle swarm optimization, Science Technology and Engineering, 20:13258-13264.

[9] Kees C., Farthing M. and Dawson C., 2009. Locally conservative, stabilized finite element methods for variably saturated flow, Comput. Methods Appl. Mech. Engrg., 197:4610-4625.

[10] Luo Z., Li H. and Chen J., 2012. A reduced-order finite volume element formulation based on POD method and implementation of its extrapolation algorithm for unsaturated soil water flow equation, Scientia Sinica, 42:1263-1280.

[11] Li H., Luo Z., Xie Z. and Zhu J., 2006. Generalized difference methods and numerical simulation for the unsaturated soil water flow problems, Mathematcia Numerica Sinica, 28:331-336.

[12] Luo Z., 2006. Mixed Finite Element Methods and Applications, Chinese Science Press, Beijing.

[13] Na Z. , Wang Y. , Wang Y. , Yan B. , and Qian S, 2018. A locally conservative Multiscale Finite Element Method for multiphase flow simulation through heterogeneous and fractured porous media, J. Comput. Appl. Math, 343:501-519.

[14] Na Z. , Huang Z. , Yao J., 2013. Locally conservative galerkin and finite volume methods for two-phase flow in porous media, J. Comput. Phys., 254:39-51.

[15] Purushothaman K. and Nagarajan V., 2021. Multiobjective optimization based on self, rganizing particle swarm optimization algorithm for massive MIMO 5G wireless network, Int. J commun. syst, 34:4725

[16] Radha R. and Gopalakrishnan R., 2020. A medical analytical system using intelligent fuzzy level set brain image segmentation based on improved quantum particle swarm optimization, Microprocessors and Microsystems, 79:103283.

[17] Rathfelder K. and Linda M., 1994. Mass conservative numerical solutions of the bead-based Richards equation, Water Resour. Res., 30:2579-2586.

[18] Shi Y. and Eberhart R., 1998. A modified Particle swarm optimizer, Proc IEEE Int Conf on Evolutionary Computation, 69-73.

[19] Saeedi S., Khorsand R., Bidgoli S. and Ramezanpour M., 2020. Improved many-objective particle swarm optimization algorithm for scientific workflow scheduling in cloud computing, Computers & Industrial Engineering, 147:106649

[20] Tran-Ngoc H., He L., et al., 2020. An efficient approach to model updating for a multi-span railway bridge using orthogonal diagonalization combined with improved particle swarm optimization, Journal of Sound and Vibration, 476:115315

[21] Teng F. and Luo Z., 2014. A POD-based reduced-order CN finite element extrapolating model for unsaturated soil water flow equation, Applied Mathematics-A Journal of Chinese Universities, 29:45-54

[22] Hu T., Zhang X. and Cao X., 2020. A hybrid particle swarm algorithm for dynamically adjusting inertial weights, Electro-Optics and Control, 27:16-21.

[23] Xie Z., Dai Y. and Zeng Q., 1999. An unsaturated soil water flow problem and its numerical simulation, Adv. Atmos. Sci., 16:183-196.

[24] Yang Q., Hua L. and Gao X. et al,2022. Stochastic Cognitive Dominance Leading Particle Swarm Optimization for Multimodal Problems, Mathematics, 10:1-34.

[25] Yu L., Han Y. and Mu L,2020. Improved quantum evolutionary particle swarm optimization for band selection of hyperspectral image, Remote Sensing Letters , 11:866-875.

[26] Zhou Y., 1996. Difference schemes with nonuniform meshes for nonlinear parabolic system, J. Comput. Math., 14, 319-335.