

Article

Structural Model Identification Using a Modified Electromagnetism-Like Mechanism Algorithm

Zhouquan Feng ^{1,2,3,*} , Zhengtao Ye ¹, Wenzan Wang ¹, Yang Lin ¹, Zhengqing Chen ^{1,2,3} and Xugang Hua ^{1,2,3} 

¹ Key Laboratory of Wind and Bridge Engineering of Hunan Province, College of Civil Engineering, Hunan University, Changsha 410082, China; det@hnu.edu.cn (Z.Y.); wenzanwang@hnu.edu.cn (W.W.); linyang@hnu.edu.cn (Y.L.); zqchen@hnu.edu.cn (Z.C.); cexghua@hnu.edu.cn (X.H.)

² Vibration and Shock Technology Research Center, College of Civil Engineering, Hunan University, Changsha 410082, China

³ State Key Laboratory of Advanced Design and Manufacturing for Vehicle Body, Hunan University, Changsha 410082, China

* Correspondence: zqfeng@hnu.edu.cn

Received: 8 August 2020; Accepted: 23 August 2020; Published: 25 August 2020



Abstract: A modified electromagnetism-like mechanism (EM) algorithm is proposed to identify structural model parameters using modal data. EM is a heuristic algorithm, which utilizes an attraction–repulsion mechanism to move the sample points towards the optimal solution. In order to improve the performance of original algorithm, a new local search strategy, new charge and force calculation formulas, new particle movement and updating rules are proposed. The test results of benchmark functions show that the modified EM algorithm has better accuracy and faster convergence rate than the original EM algorithm and the particle swarm optimization (PSO) algorithm. In order to investigate the applicability of this approach in parameter identification of structural models, one numerical truss model and one experimental shear-building model are presented as illustrative examples. The identification results show that this approach can achieve remarkable parameter identification even in the case of large noise contamination and few measurements. The modified EM algorithm can also be used to solve other optimization problems.

Keywords: structural model identification; electromagnetism-like mechanism; modal data; optimization

1. Introduction

Modeling and identification of structural systems is a fundamental subject in structural engineering. In the past decades, structural system identification has been widely used in structural health monitoring, damage detection, response prediction, vibration control and other fields [1–7]. Structural system identification can be regarded as an optimization problem from the point of view of mathematics. Its goal is to find an optimal model and make its predictive response as close as possible to the measured response of the actual system [3]. According to the literature, the structural model can be divided into parametric physical model and nonparametric data model. The nonparametric model is completely based on data and sometimes it is called data-driven model, which is generally only used when the structural system is very complicated and the physical model is difficult to build [8]. Data-driven simulation models have been widely used in other fields, such as logistics facilities management [9] and multicriteria transportation problem [10]. Parametric structural models are generally defined by a set of physical parameters such as mass and stiffness, and they are usually modeled by the finite element method [11]. The process of solving the optimal parameters of the model

is called parameter identification. The parameter identification of structural models is the foundation of all subsequent structural calculation and analysis.

Structural system identification can be divided into two types: frequency domain methods and time domain methods. The time-domain methods are directly based on the measured time-history data and represent the objective function as the residual between the predicted response and the measured data. The optimal model parameters can be obtained by solving this minimization optimization problem [3]. However, the response of the structural system is not only related to the structure itself but also related to the input of the system, so the input should also be measured or some assumptions should be made, which brings some inconvenience to the application. The frequency methods are based on the modal information of the structure, such as frequency response function and modal parameters [12]. In the identification of structural models based on modal parameters, the natural frequencies and mode shapes of the structure are first obtained through experimental modal analysis, and then the difference between the predicted modal parameters and the identified modal parameters is minimized by optimization method to estimate the model parameters.

Various techniques have been developed based on frequencies and mode shapes for structural identification. Brownjohn and Xia presented sensitivity-based model updating of a curved cable-stayed bridge based on measured modal data [13]. An iterative sensitivity-based FE model updating method was proposed by Teughels and De Roeck, in which the discrepancies in both the frequencies and mode shapes obtained from ambient tests are minimized [14]. Model updating based on frequency, mode shape, modal flexibility, and their combinations was studied by Jaishi and Ren [15]. Bakir et al. proposed a trust region algorithm for model updating to solve a nonlinear constrained optimization problem [16]. Adaptive regularization parameter optimization was proposed by Hua et al. for output-error-based FE model updating, in which an adaptive strategy was employed to change the value of the regularization parameter at different iteration steps [17]. Sarmadi et al. proposed a new iterative method named as least squares minimal residual for finite element model updating [18]. A tutorial was given by Mottershead et al. for the sensitivity-based finite element model updating with modal data [11].

However, most of the above methods generally need to obtain the gradient of the objective function and make a good initial guess of the solution. In addition, when measured data is incomplete, it may be difficult to use these methods in identifying large systems. Therefore, in recent years, heuristic optimization algorithms have been used to solve the optimization problem of structural model identification and achieved good results. Genetic algorithm was employed by Perera et al. for simultaneous optimizations of several objectives with the purpose of improving the performance of model updating [19]. Sun et al. proposed a modified artificial bee colony algorithm for structural identification [3]. A methodology based on particle swarm optimization with sequential niche technique was proposed by Shabbir and Omenzetter for dynamic finite element model updating [20]. Seyedpoor et al. proposed the differential evolution algorithm to identify multiple damage cases of structural systems [21]. Structural damage identification based on cuckoo search algorithm was presented by Xu et al. [22]. Performance studies of 10 metaheuristic techniques in structural model updating and damage detection were conducted by Mishra et al. [23].

In addition to the above heuristic algorithms, Birbil and Fang proposed a new heuristic algorithm called electromagnetism-like mechanism algorithm (EM) in 2003 to solve global optimization problems [24]. EM algorithm first produces a set of initial solutions (called a group of initial particles here) randomly from feasible domains, and then determines the attraction domain according to the objective function value of each particle, in order to produce a new generation of particles in some mechanism. The EM algorithm simulates the attraction and repulsion mechanism in the electromagnetic field, compares each solution to a charged particle, and then makes the particles move towards the optimal solution according to certain criteria. This idea comes from the analogy with the mechanism of attraction and repulsion in electromagnetic theory, which is called electromagnetism-like mechanism due to some differences between the two. More importantly, the convergence of EM algorithm has

been proved, and the results show that at least one particle in the population moves near the global optimum with probability of one as the number of iterations is large enough [25]. However, the original version of EM algorithm has some defects to be improved, such as the charge and force formulas, the local search method and the particle movement rule. Several researchers have proposed some improvements to the original EM algorithm to enhance its performance [26–29]. Rocha and Fernandes proposed a modified EM algorithm based on a new local search method and a new movement force vector, in which a pattern search method replaced the simplest linear search, and a new vector is defined to consider information in both the current iteration and the previous iteration [26]. Zhang et al. proposed an improved electromagnetism-like mechanism algorithm for constrained optimization, in which four modifications are made to enhance its accuracy and efficiency [27]. Tan et al. improved the original EM algorithm by using a new local search scheme with Split, Probe and Compare feature (SPC-EM) [28]. Later, Tan et al. introduced an experience-learning feature into the EM particle, in which two new components including the memory concept and the experience analyzing and decision-making mechanism were employed [29]. The EM algorithm and its variants have been widely used in many application fields, such as project scheduling [30], vehicle routing [31], course timetabling [32], manufacturing system design [33], diabetes mellitus prediction [34], power flow control [35] and truss structure optimization [36,37]. However, there are very few studies related to structural engineering using EM algorithm [36,37]. These researches were limited to the optimal design of truss structures, and parameter identification of structural models was not involved. This study aims to improve the accuracy and computing efficiency by modifying the original EM algorithm. Then, the modified EM algorithm is applied for parameter identification of structural models.

The paper is organized as follows. In Section 2, structural system identification is described as an optimization problem. In Section 3, the original EM algorithm is reviewed and a modified EM algorithm is proposed for structural parameter identification. Section 4 tests the performance of the modified EM algorithm with benchmark functions and applies it in parameter identification of a numerical truss model and an experimental shear-building model. Some discussions are given in Section 5 for further study. In Section 6, we draw conclusions of this work.

2. Optimization Formulation of Structural Model Identification

Structural model identification can be regarded as the optimization problem of objective function. For example, the discrepancy between measured structural response and model prediction response is defined as the objective function, and parameter identification of the model is treated as optimization (usually minimization) of the objective function. If the data consist of several modal data sets, structural model parameter identification is to solve a set of optimal model parameters (e.g., denoted as $\theta \in \mathbb{R}^{N_\theta \times 1}$, N_θ represents the number of parameters), so as to minimize the error between the predicted and measured modal parameters.

A linear structure model with N degrees-of-freedom (DOFs) is considered. The mass matrix M is assumed to be known and the stiffness matrix K is parameterized by θ , namely, $K = K(\theta)$. The global stiffness matrix $K(\theta)$ can be represented as a cumulative sum of element matrices by using the following equation

$$K(\theta) = \sum_{i=1}^{N_e} (1 + \theta_i) k e_i \quad (1)$$

where N_e is the total number of finite elements and $k e_i$ denotes the stiffness matrix of the i^{th} element. As we known that the predicted modal frequency $\hat{f}_j(\theta)$ and mode shape $\hat{\phi}_j(\theta)$ of j^{th} mode can be obtained through solving the following eigenvalue (characteristic) value problem.

$$[K(\theta) - \omega^2 M] \phi = 0 \quad (2)$$

where the quantities ω^2 are the eigenvalues, which are the square of the circular modal frequencies ($\omega = 2\pi f$ and f is the modal frequency in Hz), while the corresponding eigenvectors ϕ are the mode shapes.

The measured incomplete modal data are consisted of N_s sets of modal data (each data set contains modal frequencies and mode shapes with N_m modes). For a structural system with N DOFs, it should theoretically have N modes, and the mode shape of each mode contains N components. However, not all modes can be excited and some DOFs are difficult to be measured, therefore, the measured modal parameters are incomplete in practice. The data incompleteness is reflected in two aspects: one is measurement incompleteness, i.e., the number of measured DOFs N_o is less than N and the other is mode incompleteness, i.e., the number of identified modes N_m is less than N . Based on the least square principle, the sum of relative errors of modal data (modal frequencies and mode shapes) from prediction and measurement is used to establish the objective function. In this study, the objective function for model identification (updating) is defined as follows

$$J(\theta) = \sum_{j=1}^{N_m} \left[\sum_{i=1}^{N_s} \left(\frac{f_{j,i} - \hat{f}_j(\theta)}{f_{j,i}} \right)^2 + \sum_{i=1}^{N_s} \frac{\|\phi_{j,i} - a_{j,i} \hat{\phi}_j(\theta)\|^2}{\|\phi_{j,i}\|^2} \right] \quad (3)$$

where $a_{j,i} = (\phi_{j,i})^T \hat{\phi}_j(\theta) / \|\hat{\phi}_j(\theta)\|^2$ is a scaling factor that ensures that the predicted mode shape $\hat{\phi}_j(\theta)$ is closest to the measured mode shape $\phi_{j,i}$ at the measured DOFs [38], $f_{j,i}$ is the measured modal frequency of j^{th} mode in the i^{th} data set and $\hat{f}_j(\theta)$ is the predicted modal frequency of j^{th} mode, $\phi_{j,i}$ is the measured partial mode shape of j^{th} mode in the i^{th} data set and $\hat{\phi}_j(\theta)$ is the predicted partial mode shape of j^{th} mode.

3. Electromagnetism-Like Mechanism Algorithm

3.1. The Original EM Algorithm

According to the attraction–repulsion mechanism in electromagnetic theory, EM algorithm imagines each solution as a charged particle in space, and the charge of each particle is determined by the value of the objective function to be optimized. This charge also determines how strongly the particle attracts or repels other particles: the better the objective function value, the stronger the attraction. The direction of each particle’s next move is determined by calculating the resultant force applied by other particles to the current particle. As with electromagnetic forces, the resultant force is obtained by superimposing vector quantities of forces from other particles. As shown in Figure 1, we select three particles to briefly illustrate how one particle searches according to the attraction–repulsion mechanism. Suppose that the solution represented by particle 2 is better than the solution represented by particle 1 and the solution represented by particle 3 is worse than the solution represented by particle 1, then particle 2 will have an attractive force F_{21} to particle 1 and particle 3 will have a repulsive force F_{31} to particle 1. The resultant force F , which is the superposition of two forces, will determine the direction in which particle 1 will move, forcing the particle to move towards a better region. Furthermore, similar to some other population-based hybrid algorithm, EM algorithm can also use a local search to improve the objective function value of the current population.

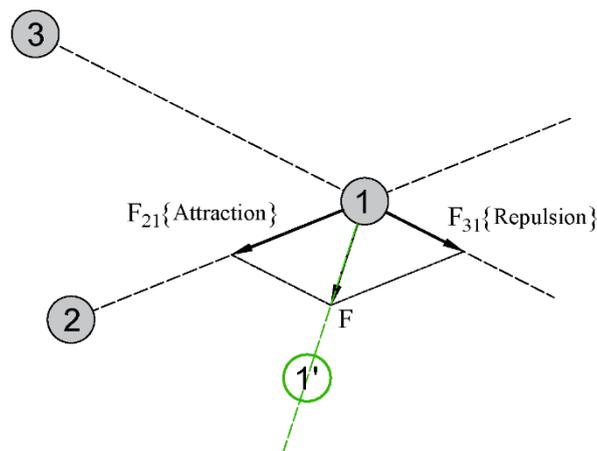


Figure 1. Schematic diagram of the mechanism of attraction and repulsion.

The EM algorithm consists of four steps, namely, initialization, local search, calculation of resultant force and movement of particles. The flow chart of the original EM algorithm is shown in Figure 2.

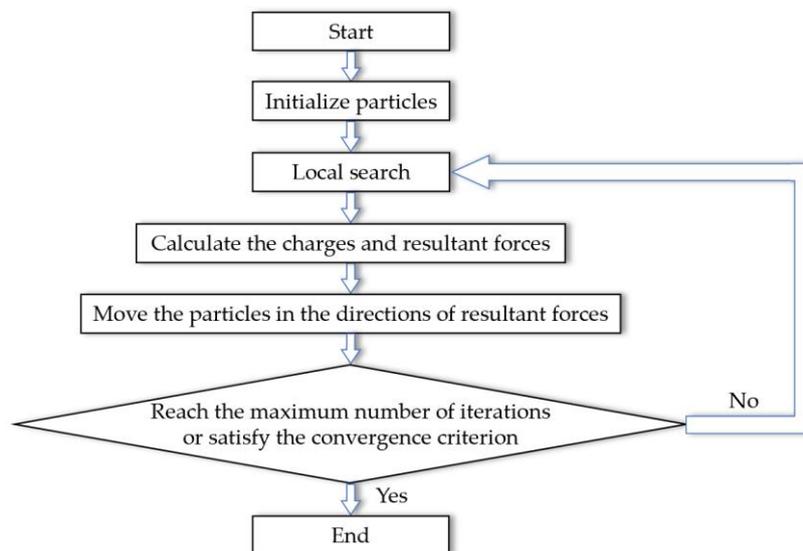


Figure 2. Flowchart of the original electromagnetism-like mechanism (EM) algorithm.

- Initialization

Initialization is to take a number of random points from a known feasible region. Here, the initial particles are randomly and uniformly distributed in the feasible region, then the objective function value of each particle is calculated and the particle with the optimal objective function value is denoted as x_{best} .

- Local search

Local search is performed on individual particles to improve the solution that the population has searched for. For EM algorithm, local search plays a very important role, because it provides effective local information for global search of population, so that the algorithm has both global search ability and fine search ability in local area. The local search used in the original EM algorithm is the simplest linear search, searching each dimension of each particle by a certain step size δ and stopping as soon as a better solution is found.

- Calculation of resultant force

The calculation of resultant force is the most important step of EM algorithm, which combines the local information obtained by particles with the global information. Simulating the superposition

principle in electromagnetic theory, EM algorithm provides information for the next search by calculating the resultant force.

The charge quantity q_i of particle i determines the magnitude of attractive force or repulsive force on particle i . The charge quantity q_i is calculated as follows:

$$q_i = \exp\left(-n \frac{f(x_i) - f(x_{best})}{\sum_{k=1}^m (f(x_k) - f(x_{best}))}\right) \quad (4)$$

where $f(x)$ is the objective function to be minimized, n is the dimension of the problem and m is the number of samples (particles) in each iteration.

In this way, particles with better objective function value will have larger charge value and stronger attraction. After comparing the objective function value of two particles, the direction of the interaction force between two particles will be determined. The resultant force F_i acting on particle i is calculated as follows:

$$F_i = \sum_{j \neq i}^m \left\{ \begin{array}{l} (x_j - x_i) \frac{q_i q_j}{\|x_j - x_i\|^2} \quad \text{if } f(x_j) < f(x_i) \\ (x_i - x_j) \frac{q_i q_j}{\|x_j - x_i\|^2} \quad \text{if } f(x_j) \geq f(x_i) \end{array} \right\} \quad (5)$$

According to Equation (5), between every two particles, the particle with a better objective function value (i.e., the smaller one) will attract another particle. Conversely, a particle with a worse value of the objective function (i.e., a larger value) will repel another particle. Since the objective function value of the current optimal particle x_{best} is the smallest, it acts as an absolutely attractive particle, attracting the whole population.

- Movement of particles

After calculating the resultant force vector F_i , particle i will move along the direction of the resultant force with a random step length (as given in Equation (6)). The step λ is evenly distributed over $[0, 1]$. In Equation (6), RNG is a vector whose k^{th} component represents the feasible step size towards the upper bound u_k or lower bound l_k . In addition, the forces acting on each particle are “normalized,” thus ensuring the possibility of movement. The movement formula of the particle is

$$x_i = x_i + \lambda \frac{F_i}{\|F_i\|} (RNG) \quad i = 1, 2, \dots, m \quad (6)$$

In such a way, the position of the particles is updated and one iteration of the EM algorithm completes. The readers can see the reference for details of the original EM algorithm [24].

3.2. The Modified EM Algorithm

The EM algorithm is simple in structure, and it is not necessary to obtain derivatives of the function like many traditional methods to solve the global optimization problem. Moreover, the EM algorithm has no requirement for the type of function to be optimized. Therefore, this method can be used to solve global optimization of not only general functions but also of complex functions that are difficult to obtain information such as derivatives. However, in the original EM algorithm, the local search is the simplest random linear search. When the algorithm falls into a local minimum too early, it cannot jump out, which affects the optimization performance of the algorithm. Then, the calculation formulas for charge and force are not as good as possible, which affects the computation efficiency and precision. Next, the particle updating rules are fixed, and it is difficult to search the optimal value more accurately in the later stages of evolution, which affects the speed and accuracy of the algorithm. In order to solve the shortcomings of the original EM algorithm, we made improvements in the following five aspects to achieve better performance. These improvements are detailed below, and the pseudocodes for each modification are presented in the Appendix A.

- New local search method

The original local search is the simplest linear search, which is not efficient. Here, we intensify local search by branching current best particles with a certain radius. For example, uniformly distributed points inside the ball centered at x_{best} with radius of $\|x_{best} - x_{2nd_{best}}\|$ or $\|x_{best} - x_{3rd_{best}}\|$ are generated. Herein, x_{best} , $x_{2nd_{best}}$ and $x_{3rd_{best}}$ are the particle with best, second best and third best function value, respectively. Afterwards for each new point (main branches), what is done to x_{best} to generate subpoints (leaves) is redone. Finally, all newly generated points to find if there is a point with better function value to replace current best particle are compared. The graphical representation of the new local search method is shown in Figure 3. Particle 1 is the best particle, and particle 2 is the second best particle. The red circles are the moving trail of particle 1 and particle 2, the brown circles are the main branches generated from the current best point and the green circles are the leaves generated from the main branches. In this rule, if particle 1 moves to a new point, particle 2 will move to the original position of particle 1. If the new point generated is not better than particle 1 but better than particle 2, particle 2 move to the new point without following particle 1. (Appendix A).

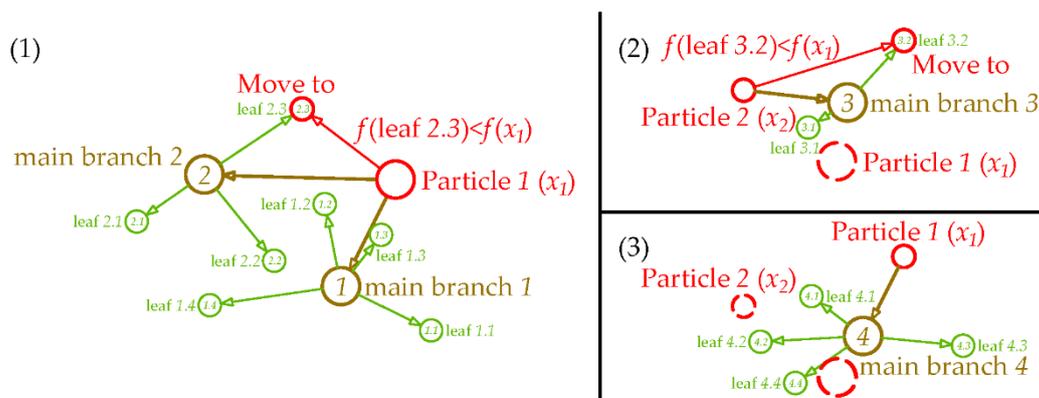


Figure 3. Graphical representation of the new local search method.

- Simplify charge calculation

The original charge formula with exponential calculations is complicated, and the function will become nearly linear in the interval [0,1]. In the later iterations of searching when many points crowd together, the exponents in the original charge formula is close to zero. Based on this observation, we propose to use a linear function to simplify the charge calculation. Some other researchers used linear functions for charge calculation as well, but in different forms [30,31]. The charge of particle used in the proposed modified EM algorithm is calculated as follows. (Appendix A)

$$q_i = \frac{f(x_i) - \min_j \{f(x_j)\}}{\max_j \{f(x_j)\} - \min_j \{f(x_j)\}} \quad (7)$$

- Sort particles and move particles with a new rule

In the original algorithm, movement of particles does not care about how strong the total force is, and only the information of direction is made use of. If the magnitude of force is not necessary, one may come up with why the force formula needs to be in the form of Coulomb's law. Therefore, it is perceived that the kernel of this optimizing algorithm is attraction of the better particles and repulsion of the worse. Inspired by a hybrid method combining electromagnetism-like mechanism and firefly algorithms [36], the proposed algorithm first have particles sorted then move them with only the attraction of one better particle and the repulsion of the next worse particle. In each iteration, the particles are sorted in ascending order according to their function values (Appendix A). Then, for particle i , it is attracted merely by particle $i-1$ and repelled only by particle $i+1$. The information of direction and charge decide where the particle will go, together with an adaptive rate of repulsion

α_r that is uniformly distributed in $[0, \alpha_u]$. In the early iterations, the upper bound α_u is close to 1, emphasizing the repulsion to disperse particles in order of exploration. Diminish α_u as the number of iterations grows, shifting focus from exploration to exploitation. The graphical representation of the new movement rule is shown in Figure 4.

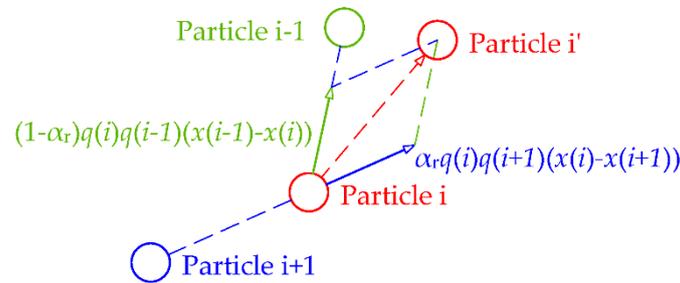


Figure 4. Graphical representation of new movement rule.

The attraction and repulsion forces of a sorted particle are defined as follows:

$$F_a = (x_{i-1} - x_i)q_{i-1}q_i \quad (8)$$

$$F_r = (x_i - x_{i+1})q_iq_{i+1} \quad (9)$$

The movement is defined as follows:

$$x'_i = x_i + v_{EM} \quad (10)$$

$$v_{EM} = F_a(1 - \alpha_r) + F_r \times \alpha_r \quad (11)$$

where F_a and F_r are, respectively, attraction and repulsion forces acting on the particle i and α_r is a random control parameter that defines the electromagnetic behavior, and it is uniformly distributed in a range of $[0, \alpha_u]$. α_u is the upper bound of the α_r , which changes its value with the number of iterations as follows. (Appendix A)

$$\alpha_u = e^{\left(\frac{\ln(\alpha_{u_min}) \times iter}{max_iter}\right)} \quad (12)$$

where max_iter is the number of maximum iterations, $iter$ is the number of current iteration and α_{u_min} is the minimum upper bound of the α_r in the final iteration. In this study, $\alpha_{u_min} = 0.3$.

- Find a new location if the current movement cannot lead particle to a better point

If the particle moves to a new place but get a worse function value, why should it move in that direction? It should find another new location. In order to fully utilize the information given by the particles $i-1$, $i+1$ and i and i' , it is recommended that the new location is the center of these four particles (Figure 5). The new location for particle i is written as follows. (Appendix A, Algorithm A5, line 13)

$$x''_i = \frac{1}{4}(x_{i-1} + x_{i+1} + x_i + x'_i) \quad (13)$$

- Do more random operations in early iterations to improve exploration

The original EM algorithm is somehow weak in exploration, and randomly generating of samples could help a lot in jumping out from local minimum. Therefore, in the early iterations, instead of doing the movement operation to the number of n_{random} worst particles, they are randomly regenerated. After a certain iteration (called the transition iteration), only the worst particle is regenerated. Other particles do the normal attracting and repelling movement operations with their adjacent sorted partners. In this study, n_{random} is 0.3 times the number of particles ($n_{particles}$), and the transition iteration is 0.1 times the number of maximum iterations (max_iter). In other words, $n_{random} = 0.3 \times n_{particles}$ from 1 to $(0.1 \times max_iter)$ iterations, while $n_{random} = 1$ from $(0.1 \times max_iter)$ to max_iter iterations. (Appendix A)

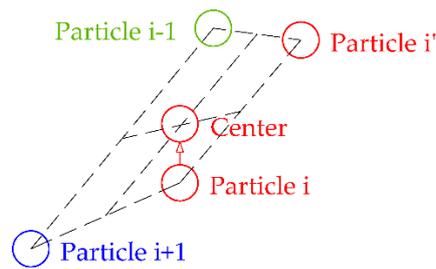


Figure 5. Graphical representation of location updating.

The flow chart of the modified EM algorithm is shown in Figure 6. The complete pseudocode of the modified EM algorithm is shown in Appendix B.

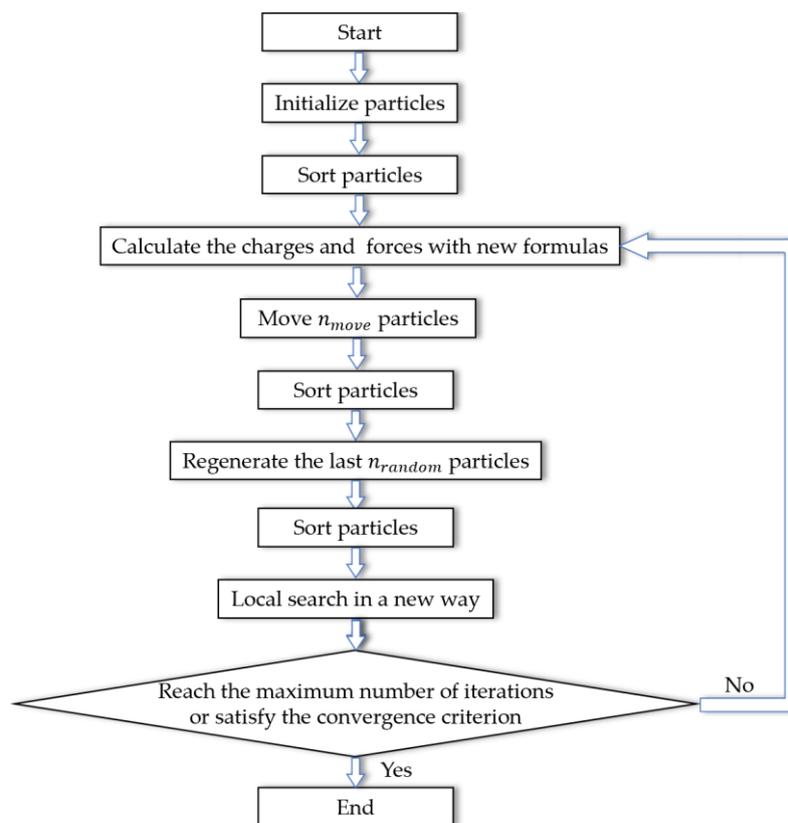


Figure 6. Flow chart of the modified EM algorithm.

4. Illustrative Examples

In this section, the modified EM algorithm is first tested in several benchmark functions, as compared with the original EM method [24] and particle swarm optimization (PSO) algorithm [39]. Then, the modified EM algorithm is applied and validated in a numerical truss model for model parameter identification, under different conditions of noise contamination and information incompleteness. Finally, an experimental shear-building model further validates the modified EM algorithm for structural model identification.

4.1. Benchmark Functions

In order to test the performance of the proposed modified EM method in numerical function optimization, numerical experiments are conducted on six well-known benchmark functions. Table 1 provides detailed information about these benchmark functions. Here, we compare the performance

of three algorithms: the modified EM algorithm, the original EM algorithm and the PSO algorithm. In the numerical experiments, each algorithm is independently run 50 times for each benchmark function. The termination criterion of these three algorithms, i.e., maximum number of iterations, is set to 1000. The number of particles is set to 16 for the modified EM algorithm and the original EM algorithm. In the original EM algorithm, the local search parameter δ is set to 0.001. There is a built-in function for PSO algorithm implementation in MATLAB, and the control parameters are set as default values. The computing hardware of the numerical experiment was a laptop with Intel Core i7-8650U @ 1.90 GHz and 16 GB RAM, and the software environment was MATLAB R2019b on Windows 10 platform.

Table 1. Test suite with six benchmark functions.

Name	Function	Search Range	Accepted Accuracy
Sphere	$f_1(X) = \sum_{i=1}^n x_i^2$	$[-100, 100]^n$	1×10^{-8}
SumSquare	$f_2(X) = \sum_{i=1}^n i \cdot x_i^2$	$[-100, 100]^n$	1×10^{-8}
Step	$f_3(X) = \sum_{i=1}^n (x_i + 0.5)^2$	$[-100, 100]^n$	1×10^{-8}
Exponential	$f_4(X) = \exp(0.5 \sum_{i=1}^n x_i^2) - 1$	$[-1.28, 1.28]^n$	1×10^{-8}
Rosenbrock	$f_5(X) = \sum_{i=1}^{n-1} [100(x_{i+1} - x_i^2)^2 + (x_i - 1)^2]$	$[-5, 10]^n$	5×10^0
Alpine	$f_6(X) = \sum_{i=1}^n x_i \cdot \sin x_i + 0.1 \cdot x_i $	$[-10, 10]^n$	1×10^{-8}

The mean, maximum and minimum values of the best solution and average computation time of six benchmark functions ($n = 30$) is tabulated in Table 2. It can be seen that the modified EM algorithm outperforms the original EM algorithm and the PSO algorithm in accuracy. For average computation time, the modified EM algorithm is much less than the original EM algorithm and is approximately the same level as the PSO algorithm. The convergence curves of six benchmark functions when using these three algorithms are shown in Figure 7. It can be seen that the modified EM algorithm has faster convergence rate than the original EM algorithm and the PSO algorithm. It is worth noting that for functions step and exponential, the convergence curves in the figure are suddenly cut off because their values drop to zero at very early iteration.

Table 2. Accuracy comparison on the six benchmark functions with dimension $n = 30$.

Function		Sphere	SumSquare	Step	Exponential	Rosenbrock	Alpine
Original EM	mean	1.20×10^{-6}	0.0089	13.64	1.03×10^{-10}	21.28	0.0031
	max	1.74×10^{-6}	0.12	22	1.45×10^{-10}	140.65	0.015
	min	6.89×10^{-7}	2.93×10^{-5}	6	6.66×10^{-11}	7.33×10^{-4}	5.32×10^{-5}
	avg time	9.24 s	9.60 s	9.51 s	9.38 s	9.03 s	9.46 s
Modified EM	mean	0	7.10×10^{-239}	0	0	0.0038	5.04×10^{-107}
	max	0	3.55×10^{-237}	0	0	0.053	2.52×10^{-105}
	min	0	5.08×10^{-241}	0	0	1.15×10^{-9}	2.79×10^{-141}
	avg time	2.78 s	2.88 s	4.89 s	2.65 s	3.66 s	3.01 s
PSO	mean	2.93×10^{-34}	2.25×10^{-29}	2.68	4.22×10^{-16}	30.19	1.07×10^{-14}
	max	7.04×10^{-33}	1.12×10^{-27}	19	1.78×10^{-15}	138.04	3.86×10^{-14}
	min	2.02×10^{-37}	6.76×10^{-37}	0	2.22×10^{-16}	0.21	1.33×10^{-15}
	avg time	3.10 s	3.24 s	2.51 s	2.71 s	2.96 s	2.97 s

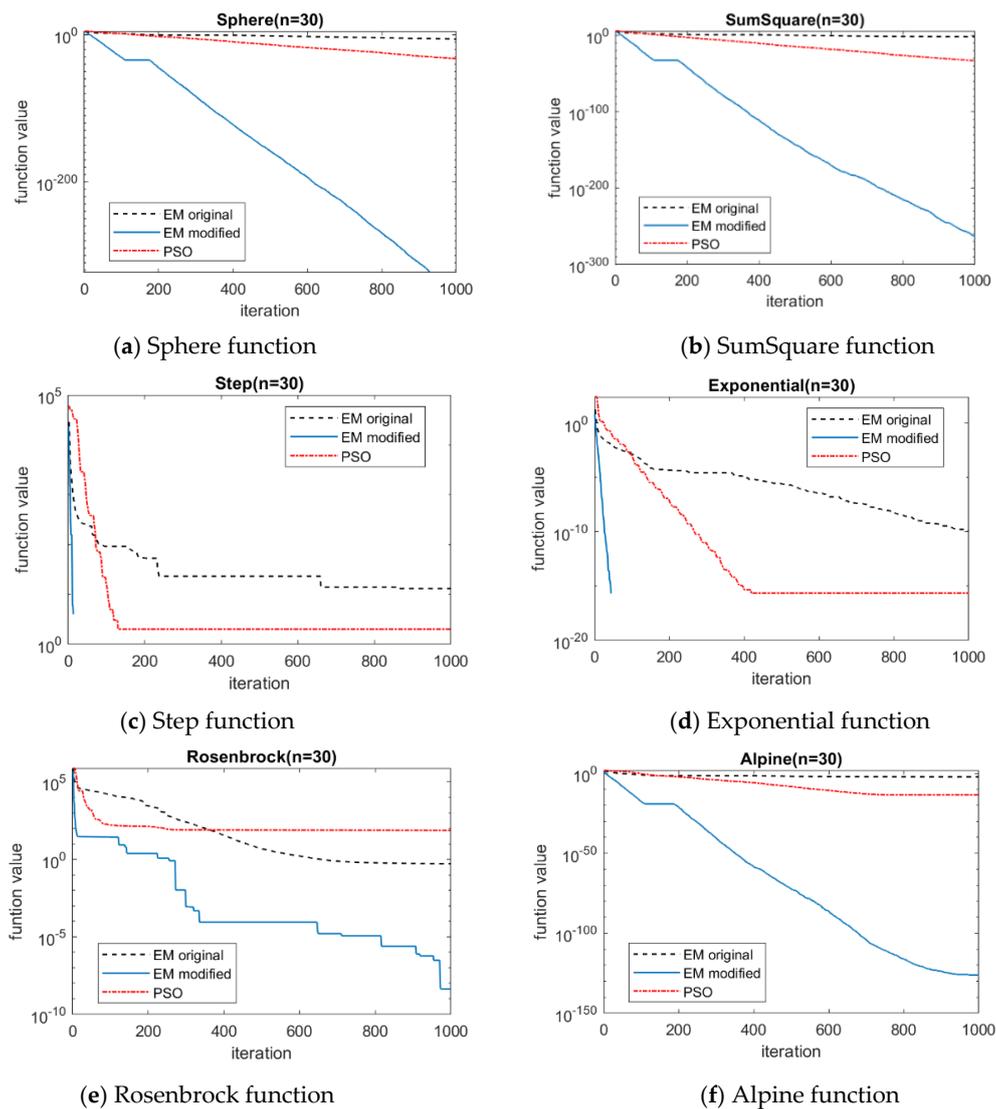


Figure 7. Convergence lines of six benchmark functions with 30 dimensions. (a): Sphere; (b): SumSquare; (c): Step; (d) Exponential; (e) Rosenbrock; (f) Alpine.

4.2. Numerical Truss Model

A simply supported truss model with 21 bars is considered in this example. The configuration and dimension of the truss model is shown in Figure 8. The truss has a span of 10 m and a height of 3.3 m. The elastic module of material is 1.8 GPa, the mass density of material is 2200 kg/m³ and the sectional area of all bars is 0.0025 m². For finite element modeling of the truss, 12 nodes and 21 bar elements are used. For one planar bar element, there are two nodes, with two degrees of freedom (DOFs) for each node, i.e., one in horizontal direction and the other in vertical direction. Minus the boundary constraints (2 DOFs in node 1 and 1 DOF in node 9), there are 21 DOFs ($2 \times 12 - 3 = 21$). It is assumed that there are some damages in elements 19 and 20, and the damage severity is -15% in element 19 and -20% in element 20. Overall, 21 stiffness parameters θ_n ($n = 1, 2, \dots, 21$) were used to adjust the values of element stiffness ke_n , so the updated ones would have the values of $ke_n(1+\theta_n)$ ($n = 1, 2, \dots, 21$), respectively.

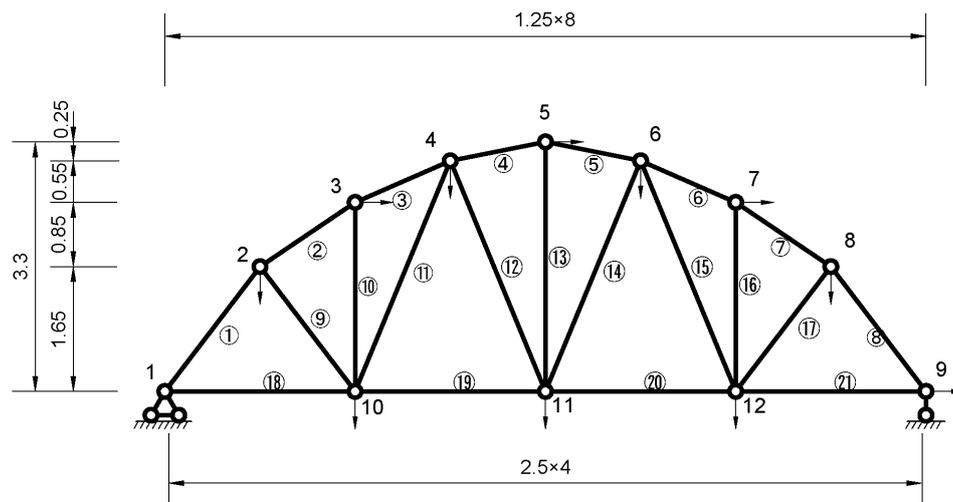


Figure 8. A 21-bar planar truss model (unit: m).

In order to consider the noise contamination levels and information incompleteness of the modal data, four cases are assumed. For each case, it is assumed that 20 sets of modal data are available for each simulation, and the simulation are run independently 100 times. For complete measurement, the truss should have 21 DOFs to be measured and 21 modes to be available. In reality, the complete information cannot be achieved. For case 1, 2 and 4, it is assumed that there are 11 measurements (shown in Figure 8 with arrows on the nodes), while there are 7 measurements for case 3 (remove the measurements on nodes 2, 4, 6 and 8). Here, it is assumed that there are only first 8 modes for case 1, 2 and 3, while there are only first 5 modes for case 4. These four cases are listed as follows.

- Case 1: 1% noise for frequency, 5% noise for mode shape, 11 measurements and 8 modes.
- Case 2: 2% noise for frequency, 10% noise for mode shape, 11 measurements and 8 modes.
- Case 3: 1% noise for frequency, 5% noise for mode shape, 7 measurements and 8 modes.
- Case 4: 1% noise for frequency, 5% noise for mode shape, 11 measurements and 5 modes.

The statistics of identified stiffness parameters for elements 19 and 20 are shown in Table 3 after 100 simulations. It can be seen that the identified stiffness parameters agree well with the preset values for both four cases. The standard deviations are larger in case 2, 3 and 4 compared with case 1, especially for case 3. This phenomenon indicates that noise contamination and information incompleteness will cause more uncertainty in the identification results, especially for the measurement incompleteness.

Table 3. The identified results of the model parameters.

	Case 1		Case 2		Case 3		Case 4	
	Mean	Standard Deviation						
θ_{19}	-0.1502	2.401×10^{-4}	-0.1510	4.082×10^{-4}	-0.1515	5.380×10^{-3}	-0.1501	7.174×10^{-4}
θ_{20}	-0.2006	6.237×10^{-4}	-0.2021	6.321×10^{-4}	-0.2013	4.450×10^{-3}	-2.005	7.855×10^{-4}

The statistics of identified stiffness parameters for all elements are shown in Figure 9 after 100 simulations, where the bars represent the mean values and I-shaped marks represent plus or minus one standard deviations with mean values. It can be seen that the mean values agree well with the preset values. Except for elements 19 and 20, the mean values of θ_n are almost zero. However, for case 3 and case 4, there are large standard deviations in some element stiffness parameters. This phenomenon indicates that information incompleteness has a great influence on the uncertainty of identification results. One may mitigate this effect by placing more sensors to collect measurement data.

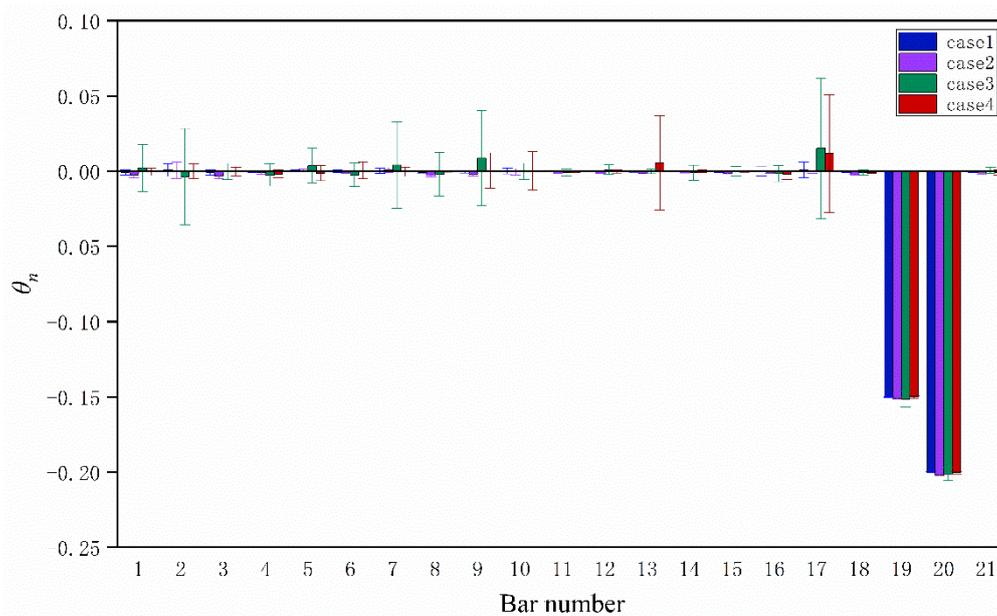
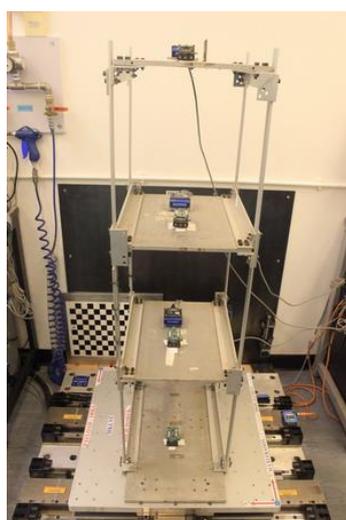


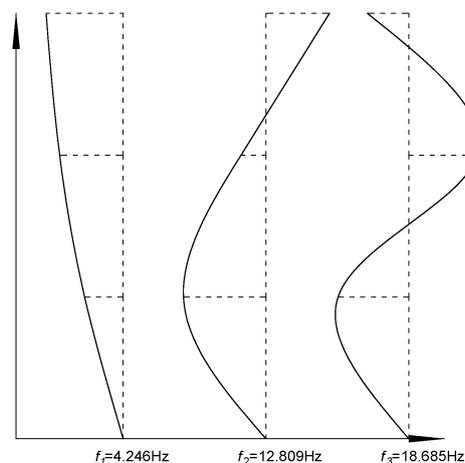
Figure 9. Updated stiffness parameters for truss model.

4.3. Experimental Shear-Building Model

A three-storey shear-building model [40–42] (Figure 10) was used to demonstrate the proposed technique. The dimensions of the model were 401 mm \times 314 mm \times 1158 mm (width \times depth \times height). The nominal values of storey mass were $m_1 = 5.63$ kg, $m_2 = 6.03$ kg and $m_3 = 4.66$ kg. The nominal values of interstorey stiffness were $k_1 = 20.88$ kN/m, $k_2 = 22.37$ kN/m and $k_3 = 24.21$ kN/m. The natural frequencies were then calculated as 4.5450, 13.023 and 18.210 Hz for the nominal model. In order to update the structural model, three stiffness parameters θ_1 , θ_2 and θ_3 were used to adjust the values of interstorey stiffness, so the updated ones would have the values of $k_1(1+\theta_1)$, $k_2(1+\theta_2)$ and $k_3(1+\theta_3)$, respectively. The structural model was fixed in a shake table and subjected to base excitations. The acceleration responses of all the three stories and the base were measured with a sampling frequency of 100 Hz. Specifically, 10 data sets with 60,000 points (10 min) for each data set were obtained.



(a) Structural model



(b) Identified mode shapes

Figure 10. The tested three-storey shear-type structure model and its identified mode shapes. (a): Structural model; (b): Identified mode shapes.

The frequencies and mode shapes were identified using the frequency domain decomposition method [43]. The mean values for modal frequency are shown in Table 4, which are very close to the results identified by some other methods [41,42]. The identified mode shapes are shown in Figure 10. By using the proposed structural identification method, the identified stiffness parameters for θ_1 , θ_2 and θ_3 are -0.221 , 0.099 and 0.032 , respectively. The stiffness of the first storey has a large reduction, which may be due to the loosening of bolts used to fix the structural model. Table 4 shows the comparison of modal frequencies and MAC (Modal Assurance Criterion) values for experimental model, nominal model and updated model. It can be seen that the modal frequencies of the updated model are more close to those of the experimental model, and the same goes for the MAC values (the larger of MAC value, the similarity of the two mode shapes). Although there are still some differences in modal parameters, these are smaller than those between the nominal model and the experimental model.

Table 4. Comparison of modal parameters of nominal and updated models with those of experimental model.

Mode	Identified Frequency	Nominal Frequency	MAC _{n-e}	Updated Frequency	MAC _{u-e}
1	4.246	4.545 (7%)	0.9975	4.258 (0.3%)	0.9997
2	12.809	13.023 (1.6%)	0.9842	12.783 (0.2%)	0.9949
3	18.685	18.21 (-2.5%)	0.9589	18.607 (0.4%)	0.9698

Notes: MAC_{n-e} and MAC_{u-e} are the MAC values of mode shapes between nominal model and experimental model and between updated model and experimental model, respectively.

5. Discussions

In this study, a modified EM algorithm is proposed to identify structural model parameters. The modified EM algorithm is more accurate than the original one, i.e., the convergence is faster and the calculation time is reduced. Compared with the PSO algorithm, it also has advantage in accuracy. Two application examples both show that the identification accuracy of the approach is very high, even in the case of large noise level and incomplete information. Nevertheless, the following points are worth for further study and discussion:

(1) For heuristic algorithms, due to its swarm and stochastic features, the solution results are often random in a small range. In order to eliminate the randomness, hybrid optimization algorithms are often used to combine the global searching ability of heuristic algorithm and the local precise searching ability of traditional algorithm, so as to achieve better solving performance.

(2) Due to the unavoidable measurement noise, the existence of model errors and incomplete information, the identified model parameters have certain uncertainties. How to quantify these uncertainties deserves further study.

(3) This study is based on the least squares principle for model parameter identification. If there is little information, the model may become unidentifiable, that is, there are multiple solutions. This situation may need to be treated by other methods such as Bayesian inference [44].

(4) The number and arrangement of sensors will have a certain impact on the identification accuracy. In general, the more the number of sensors, the more sufficient the information, and thus will improve the accuracy of identification. It is also necessary to optimize the arrangement of sensors. Different layouts of sensors will produce different identification accuracy for a certain parameter.

6. Conclusions

A modified EM optimization algorithm is proposed to identify structural parameters using modal data. The most important contribution of this study is the improvements of the original EM algorithm. Several modifications are made to the original EM algorithm to improve the optimization performance. A new local search strategy, new charge and force calculation formulas, new particle movement and updating rules are proposed to enhance the convergence and accuracy of the algorithm. The test results

of six well-known benchmark functions show that the modified EM algorithm outperforms the original EM algorithm in both accuracy and efficiency and outperforms the popular PSO algorithm in accuracy.

The second contribution of this study is the successful introduction of the modified EM algorithm into the parameter identification of structural models, which has achieved good results. Two illustrative examples are presented to demonstrate the applicability in structural model identification. The identification results of the numerical truss model agree well with the preset values of model parameters, even under the condition of a high noise level of 10% and incomplete measurement of few DOFs and modes. The simulation results also show that the information incompleteness has a great influence on the uncertainty of identification results, and one may mitigate this effect by densely placement of sensors. The identification results of the experimental shear-building model further validate the effectiveness and reliability of the proposed algorithm for structural model identification. The modal parameters of the updated structural model are more close to those of the experimental model than before updating.

Furthermore, the modified EM algorithm is not only useful for the structural model identification but also helpful when EM algorithm is applied for other optimization problems. In addition, the proposed modifications can be inspiring for improving other heuristic algorithms. Still and all, expanding application field of proposed algorithm, hybrid optimization algorithms development and structural parameters uncertainty studies would be the directions for future research.

Author Contributions: All the authors contributed to this paper: Z.F. proposed the conception of this research, derived some formulas, calculated the second and third examples and wrote the manuscript; Z.Y. prepared the codes of the optimization algorithms and made the calculations of the first numerical example; W.W. and Y.L. prepared the second example; Z.C. and X.H. revised the manuscript. All authors have read and agreed to the published version of the manuscript.

Funding: This work is supported by the following funds: the National Natural Science Foundation of China (Grant No. 51708203), Fundamental Research Funds for the Central Universities (Grant No. 531118010047), Open Research Fund Program for innovation platforms of universities in Hunan province (Grant No. 17K022), Independent Research Project of State Key Laboratory of Advanced Design and Manufacturing for Vehicle Body in Hunan University (Grant No. 71865006) and College Students' Innovation and Entrepreneurship Training Program (Grant No. S202010532285).

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

Nomenclature

The Nomenclature is listed as follows.

$a_{j,i}$	A scaling factor of the predicted mode shape $\hat{\phi}_j(\theta)$ with respect to $\phi_{j,i}$
DOFs	Degrees of freedom
$f(\cdot)$	Objective function to be minimized
F_a	Attraction force
F_r	Repulsion force
F_i	The resultant force on particle i of the original EM algorithm
$f_{j,i}$	The measured modal frequency of j^{th} mode in the i^{th} data set
\hat{f}_j	The predicted modal frequency of j^{th} mode
\mathbf{M}	Global mass matrix
MAC	Modal Assurance Criterion
m	The number of particles in the original EM algorithm
max_iter	The number of maximum iterations
N	The number of DOFs
n	The dimension of the problem
$n_{particles}$	The number of particles
n_{random}	The number of particles regenerated randomly in each iteration
N_{θ}	The number of parameters
N_e	The total number of finite elements
N_m	The number of modes
N_s	The number of data sets of measured modal data
q_i	The charge quantity of particle i

\mathbf{K}	Global stiffness matrix
ke_i	The stiffness matrix of the i^{th} element
RNG	The feasible step size vector
v_{EM}	Movement vector
x_i	Particle i
x_{best}	The particle with the optimal objective function value
$x_{2^{nd}best}$	The particle with the second best objective function value
$x_{3^{rd}best}$	The particle with the third best objective function value
α_r	Rate of repulsion uniformly distributed in $[0, \alpha_u]$
α_u	Upper bound of α_r in each iteration
δ	Step size in the local search of the original EM algorithm
θ	Parameters of the stiffness matrix \mathbf{K}
λ	The random step of movement used in the original EM algorithm
$\phi_{j,i}$	The measured mode shape of j^{th} mode in the i^{th} data set
$\hat{\phi}_j$	The predicted mode shape of j^{th} mode

Appendix A

The pseudocodes of the improvements presented in Section 3.2 are given below.

A1. Initialize

Algorithm A1. $[Particles, dim, n_{move}] = Initialize(n_{particles}, n_{random}, L, U)$

```

1:  $dim \leftarrow length(L)$  %  $U$  and  $L$  are row vectors
2:  $n_{move} \leftarrow n_{particles} - n_{random}$ 
3:  $Particles \leftarrow ones(n_{particles}, 1) * L + rand(n_{particles}, dim) .* (U - L)$ 
4: for  $i = 1$  to  $n_{particles}$  do
5:    $f\_vals(i) = f(Particles(i,:))$  %  $f\_vals$  is a column vector
6: end for
7:  $Particles \leftarrow [Particles, f\_vals]$  %  $Particles(i,:) = [x_i, f(x_i)]$ 
8:  $Particles \leftarrow Sort\_particles(Particles, dim)$ 

```

A2. Sort particles

Algorithm A2. $Particles = Sort_particles(Particles, dim)$

```

1:  $Particles \leftarrow sortrows(Particles, dim + 1)$  %  $Particles(i,:) = [x_i, f(x_i)]$ 

```

A3. Check boundary

Algorithm A3. $P_{temp} = Check_boundary(P_{temp}, L, U)$

```

1:  $P_{temp}(P_{temp} > U) \leftarrow U(P_{temp} > U)$  %  $P_{temp}(i) \leftarrow U(i)$  if  $P_{temp}(i) > U(i)$ 
2:  $P_{temp}(P_{temp} < L) \leftarrow L(P_{temp} < L)$  %  $P_{temp}(i) \leftarrow L(i)$  if  $P_{temp}(i) < L(i)$ 

```

A4. Charge calculation

Algorithm A4. $q = Cal_q(Particles, n_{move})$

```

1: for  $i = 1$  to  $n_{move} + 1$  do
2:    $q(i) \leftarrow (f(x_i) - f(x_1)) / (f(x_{n_{move}+1}) - f(x_1))$  %  $Particles(i,:) = [x_i, f(x_i)]$ 
3: end for

```

A5. Move n_{move} particles

AlgorithmA5. $Particles = Move(f, \alpha_u, Particles, q, n_{move}, dim, L, U)$

```

1:  $F_r \leftarrow q(2)(x_1 - x_2)$ 
2:  $\alpha_r \leftarrow rand * \alpha_u$ 
3:  $P_{temp} \leftarrow x_1 + \alpha_r F_r$ 
4:  $P_{temp} \leftarrow Check\_boundary(P_{temp}, L, U)$ 
5:  $Particles(1,:) \leftarrow [P_{temp}, f(P_{temp})]$  %  $x_1 \leftarrow P_{temp}$ 
6: for  $i = 2$  to  $n_{move}$  do
7:    $\alpha_r \leftarrow rand * \alpha_u$ 
8:    $F_a \leftarrow q(i-1)q(i)(x_{i-1} - x_i)$ 
9:    $F_r \leftarrow q(i+1)q(i)(x_i - x_{i+1})$ 
10:   $P_{temp} \leftarrow x_i + (1 - \alpha_r)F_a + \alpha_r F_r$ 
11:   $P_{temp} \leftarrow Check\_boundary(P_{temp}, L, U)$ 
12:  if  $Particles(i, end) < f(P_{temp})$ 
13:     $P_{temp} \leftarrow (P_{temp} + \sum_{j=i-1}^{j=i+1} Particles(j, 1 : dim))/4$  % Center of four particles
14:  end if
15:   $Particles(i,:) \leftarrow [P_{temp}, f(P_{temp})]$  %  $x_i \leftarrow P_{temp}$ 
16: end for

```

A6. Randomly regenerate the last n_{random} particles

AlgorithmA6. $Particles = Random_regenerarte(f, Particles, n_{move}, L, U)$

```

1: for  $i = n_{move} + 1$  to  $n_{particles}$  do
2:    $x_i \leftarrow rand(1, n) * (U - L) + L$ 
3:    $Particles(i,:) = [x_i, f(x_i)]$ 
4: end for

```

A7. Local search

AlgorithmA7. $Particles = Local_search(f, Particles, n_{branch}, n_{leaf}, dim, L, U)$

```

1:  $LS\_radius \leftarrow x_1 - x_2$  % or  $LS\_radius \leftarrow \|x_1 - x_{random\_integer(2,3)}\|$ 
2: for  $i = 1$  to  $n_{branch}$  do
3:   $P\_LS\_branch_i \leftarrow x_1 + LS\_radius * (rand(1, dim) - 0.5)$ 
4:   $P\_LS\_branch_i \leftarrow Check\_boundary(P\_LS\_branch_i, L, U)$ 
5:  if  $f(P\_LS\_branch_i) < Particles(i, end)$  %  $Particles(i, end)$  is  $f(x_i)$ 
6:     $Particles(2,:) \leftarrow Particles(1,:)$  %  $x_2 \leftarrow x_1$ 
7:     $Particles(1,:) \leftarrow [P\_LS\_branch_i, f(P\_LS\_branch_i)]$  %  $x_1 \leftarrow P\_LS\_branch_i$ 
8:  else if  $f(P\_LS\_branch_i) < f(x_2)$ 
9:     $Particles(2,:) \leftarrow [P\_LS\_branch_i, f(P\_LS\_branch_i)]$  %  $x_2 \leftarrow P\_LS\_branch_i$ 
10: end if
11: for  $k = 1$  to  $n_{leaf}$  do
12:   $P\_LS\_leaf_k \leftarrow P\_LS\_branch_i + LS\_radius * (rand(1, dim) - 0.5)$ 
13:   $P\_LS\_leaf_k \leftarrow Check\_boundary(P\_LS\_leaf_k, L, U)$ 
14:  if  $f(P\_LS\_leaf_k) < f(x_1)$ 
15:     $Particles(2,:) \leftarrow Particles(1,:)$  %  $x_2 \leftarrow x_1$ 
16:     $Particles(1,:) \leftarrow [P\_LS\_leaf_k, f(P\_LS\_leaf_k)]$  %  $x_1 \leftarrow P\_LS\_leaf_k$ 
17:    break
18:  elseif  $f(P\_LS\_leaf_k) < f(x_2)$ 
19:     $Particles(2,:) \leftarrow [P\_LS\_leaf_k, f(P\_LS\_leaf_k)]$  %  $x_2 \leftarrow P\_LS\_leaf_k$ 
20:  end if
21: end for
22: end for

```

Appendix B

The complete pseudocode of the modified EM algorithm is presented below.

AlgorithmA8. Complete pseudocode $[x_{best}, f(x_{best})] = EM_m(f, n_{particles}, L, U, max_iter)$

```

1: set_parameters( $\alpha_u, \alpha_{u\_min}, n_{random}, rand\_iter, n_{branch}, n_{leaf}$ )
2:  $[Particles, dim, n_{move}] \leftarrow Initialize(n_{particles}, n_{random}, L, U)$ 
3: for  $iter = 1$  to  $MAX\_iter$  do
4:    $q \leftarrow Cal\_q(Particles, n_{move})$ 
5:    $Particles \leftarrow Move(f, \alpha_u, Particles, n_{move}, dim, L, U)$ 
6:    $Particles \leftarrow Sort\_particles(Particles, dim)$ 
7:    $Particles \leftarrow Random\_regenerate(f, Particles, n_{move}, L, U)$ 
8:    $Particles \leftarrow Sort\_particles(Particles, dim)$ 
9:    $Particles \leftarrow Local\_search(f, Particles, n_{branch}, n_{leaf}, dim, L, U)$ 
10:   $\alpha_u \leftarrow \exp(\ln(\alpha_{u\_min}) * iter / max\_iter)$ 
11:  if  $iter == rand\_iter$ 
12:     $n_{move} \leftarrow n_{particles} - 1$ 
13:     $n_{random} \leftarrow 1$ 
14:  end if
15: end for
16:  $[x_{best}, f(x_{best})] \leftarrow [Particles(i, 1 : dim), Particles(1, end)]$ 

```

References

- Alvin, K.F.; Robertson, A.N.; Reich, G.W.; Park, K.C. Structural system identification: From reality to models. *Comput. Struct.* **2003**, *81*, 1149–1176. [\[CrossRef\]](#)
- Tang, H.; Xue, S.; Fan, C. Differential evolution strategy for structural system identification. *Comput. Struct.* **2008**, *86*, 2004–2012. [\[CrossRef\]](#)
- Sun, H.; Luş, H.; Betti, R. Identification of structural models using a modified Artificial Bee Colony algorithm. *Comput. Struct.* **2013**, *116*, 59–74. [\[CrossRef\]](#)
- Xu, B.; Chen, G.; Wu, Z.S. Parametric Identification for a Truss Structure Using Axial Strain. *Comput. Civ. Infrastruct. Eng.* **2007**, *22*, 210–222. [\[CrossRef\]](#)
- Feng, Z.; Lin, Y.; Wang, W.; Hua, X.; Chen, Z. Probabilistic Updating of Structural Models for Damage Assessment Using Approximate Bayesian Computation. *Sensors* **2020**, *20*, 3197. [\[CrossRef\]](#)
- Mao, J.-X.; Wang, H.; Feng, D.-M.; Tao, T.-Y.; Zheng, W.-Z. Investigation of dynamic properties of long-span cable-stayed bridges based on one-year monitoring data under normal operating condition. *Struct. Control Health Monit.* **2018**, *25*, e2146. [\[CrossRef\]](#)
- Chudzikiewicz, A.; Bogacz, R.; Kostrzewski, M.; Konowrocki, R. Condition monitoring of railway track systems by using acceleration signals on wheelset axle-boxes. *Transport* **2017**, *33*, 555–566. [\[CrossRef\]](#)
- Bogoevska, S.; Spiridonakos, M.; Chatzi, E.; Dumova-Jovanoska, E.; Höffer, R. A Data-Driven Diagnostic Framework for Wind Turbine Structures: A Holistic Approach. *Sensors* **2017**, *17*, 720. [\[CrossRef\]](#)
- Kostrzewski, M. Sensitivity Analysis of Selected Parameters in the Order Picking Process Simulation Model, with Randomly Generated Orders. *Entropy* **2020**, *22*, 423. [\[CrossRef\]](#)
- Ambroziak, T.; Malesa, A.; Kostrzewski, M. Analysis of multicriteria transportation problem connected to minimization of means of transport number applied in a selected example. *Pr. Nauk. Politech. Warsz. Transp.* **2018**, *123*, 5–20.
- Mottershead, J.E.; Link, M.; Friswell, M.I. The sensitivity method in finite element model updating: A tutorial. *Mech. Syst. Signal Process.* **2011**, *25*, 2275–2296. [\[CrossRef\]](#)
- Yan, W.-J.; Katafygiotis, L.S. A novel Bayesian approach for structural model updating utilizing statistical modal information from multiple setups. *Struct. Saf.* **2015**, *52*, 260–271. [\[CrossRef\]](#)
- Brownjohn, J.M.W.; Xia, P.-Q. Dynamic Assessment of Curved Cable-Stayed Bridge by Model Updating. *J. Struct. Eng.* **2000**, *126*, 252–260. [\[CrossRef\]](#)
- Teughels, A.; De Roeck, G. Structural damage identification of the highway bridge Z24 by FE model updating. *J. Sound Vib.* **2004**, *278*, 589–610. [\[CrossRef\]](#)

15. Jaishi, B.; Ren, W.-X. Structural Finite Element Model Updating Using Ambient Vibration Test Results. *J. Struct. Eng.* **2005**, *131*, 617–628. [[CrossRef](#)]
16. Bakir, P.G.; Reynders, E.; De Roeck, G. Sensitivity-based finite element model updating using constrained optimization with a trust region algorithm. *J. Sound Vib.* **2007**, *305*, 211–225. [[CrossRef](#)]
17. Hua, X.G.; Ni, Y.Q.; Ko, J.M. Adaptive regularization parameter optimization in output-error-based finite element model updating. *Mech. Syst. Signal Process.* **2009**, *23*, 563–579. [[CrossRef](#)]
18. Sarmadi, H.; Karamodin, A.; Entezami, A. A new iterative model updating technique based on least squares minimal residual method using measured modal data. *Appl. Math. Model.* **2016**, *40*, 10323–10341. [[CrossRef](#)]
19. Perera, R.; Ruiz, A.; Manzano, C. Performance assessment of multicriteria damage identification genetic algorithms. *Comput. Struct.* **2009**, *87*, 120–127. [[CrossRef](#)]
20. Shabbir, F.; Omenzetter, P. Particle Swarm Optimization with Sequential Niche Technique for Dynamic Finite Element Model Updating. *Comput. Civ. Infrastruct. Eng.* **2015**, *30*, 359–375. [[CrossRef](#)]
21. Seyedpoor, S.M.; Shahbandeh, S.; Yazdanpanah, O. An efficient method for structural damage detection using a differential evolution algorithm-based optimisation approach. *Civ. Eng. Environ. Syst.* **2015**, *32*, 230–250. [[CrossRef](#)]
22. Xu, H.; Liu, J.; Lu, Z. Structural damage identification based on cuckoo search algorithm. *Adv. Struct. Eng.* **2016**, *19*, 849–859. [[CrossRef](#)]
23. Mishra, M.; Barman, S.K.; Maity, D.; Maiti, D.K. Performance Studies of 10 Metaheuristic Techniques in Determination of Damages for Large-Scale Spatial Trusses from Changes in Vibration Responses. *J. Comput. Civ. Eng.* **2020**, *34*, 04019052. [[CrossRef](#)]
24. Birbil, Ş.I.; Fang, S.-C. An electromagnetism-like mechanism for global optimization. *J. Glob. Optim.* **2003**, *25*, 263–282. [[CrossRef](#)]
25. Birbil, Ş.I.; Fang, S.-C.; Sheu, R.-L. On the Convergence of a Population-Based Global Optimization Algorithm. *J. Glob. Optim.* **2004**, *30*, 301–318. [[CrossRef](#)]
26. Rocha, A.M.A.C.; Fernandes, E.M.G.P. Modified movement force vector in an electromagnetism-like mechanism for global optimization. *Optim. Methods Softw.* **2009**, *24*, 253–270. [[CrossRef](#)]
27. Zhang, C.; Li, X.; Gao, L.; Wu, Q. An improved electromagnetism-like mechanism algorithm for constrained optimization. *Expert Syst. Appl.* **2013**, *40*, 5621–5634. [[CrossRef](#)]
28. Tan, J.-D.; Dahari, M.; Koh, S.-P.; Koay, Y.-Y.; Abed, I.-A. An improved electromagnetism-like algorithm for numerical optimization. *Theor. Comput. Sci.* **2016**, *641*, 75–84. [[CrossRef](#)]
29. Tan, J.D.; Dahari, M.; Koh, S.P.; Koay, Y.Y.; Abed, I.A. A new experiential learning electromagnetism-like mechanism for numerical optimization. *Expert Syst. Appl.* **2017**, *86*, 321–333. [[CrossRef](#)]
30. Debels, D.; De Reyck, B.; Leus, R.; Vanhoucke, M. A hybrid scatter search/electromagnetism meta-heuristic for project scheduling. *Eur. J. Oper. Res.* **2006**, *169*, 638–653. [[CrossRef](#)]
31. Yurtkuran, A.; Emel, E. A new Hybrid Electromagnetism-like Algorithm for capacitated vehicle routing problems. *Expert Syst. Appl.* **2010**, *37*, 3427–3433. [[CrossRef](#)]
32. Abdullah, S.; Turabieh, H.; McCollum, B.; McMullan, P. A hybrid metaheuristic approach to the university course timetabling problem. *J. Heuristics* **2012**, *18*, 1–23. [[CrossRef](#)]
33. Guan, X.; Dai, X.; Qiu, B.; Li, J. A revised electromagnetism-like mechanism for layout design of reconfigurable manufacturing system. *Comput. Ind. Eng.* **2012**, *63*, 98–108. [[CrossRef](#)]
34. Wang, K.-J.; Adrian, A.M.; Chen, K.-H.; Wang, K.-M. An improved electromagnetism-like mechanism algorithm and its application to the prediction of diabetes mellitus. *J. Biomed. Inform.* **2015**, *54*, 220–229. [[CrossRef](#)] [[PubMed](#)]
35. El-Hana Bouchekara, H.R.; Abido, M.A.; Chaib, A.E. Optimal Power Flow Using an Improved Electromagnetism-like Mechanism Method. *Electr. Power Compon. Syst.* **2016**, *44*, 434–449. [[CrossRef](#)]
36. Le, D.T.; Bui, D.-K.; Ngo, T.D.; Nguyen, Q.-H.; Nguyen-Xuan, H. A novel hybrid method combining electromagnetism-like mechanism and firefly algorithms for constrained design optimization of discrete truss structures. *Comput. Struct.* **2019**, *212*, 20–42. [[CrossRef](#)]
37. Hosseinzadeh, Y.; Taghizadieh, N.; Jalili, S. Hybridizing electromagnetism-like mechanism algorithm with migration strategy for layout and size optimization of truss structures with frequency constraints. *Neural Comput. Appl.* **2016**, *27*, 953–971. [[CrossRef](#)]
38. Vanik, M.W.; Beck, J.L.; Au, S.K. Bayesian Probabilistic Approach to Structural Health Monitoring. *J. Eng. Mech.* **2000**, *126*, 738–745. [[CrossRef](#)]

39. Kennedy, J.; Eberhart, R. Particle swarm optimization. In Proceedings of the IEEE International Conference on Neural Networks—Conference Proceedings, Perth, Australia, 27 November–1 December 1995; Volume 4, pp. 1942–1948.
40. Chang, C.C.; Poon, C.W. Nonlinear Identification of Lumped-Mass Buildings Using Empirical Mode Decomposition and Incomplete Measurement. *J. Eng. Mech.* **2010**, *136*, 273–281. [[CrossRef](#)]
41. Feng, Z.Q.; Zhao, B.; Hua, X.G.; Chen, Z.Q. Enhanced EMD-RDT Method for Output-Only Ambient Modal Identification of Structures. *J. Aerosp. Eng.* **2019**, *32*, 04019046. [[CrossRef](#)]
42. Feng, Z.; Shen, W.; Chen, Z. Consistent Multilevel RDT-ERA for Output-Only Ambient Modal Identification of Structures. *Int. J. Struct. Stab. Dyn.* **2017**, *17*, 1750106. [[CrossRef](#)]
43. Brincker, R.; Zhang, L.; Andersen, P. Modal identification of output-only systems using frequency domain decomposition. *Smart Mater. Struct.* **2001**, *10*, 441–445. [[CrossRef](#)]
44. Katafygiotis, L.S.; Beck, J.L. Updating Models and Their Uncertainties. II: Model Identifiability. *J. Eng. Mech.* **1998**, *124*, 463–467. [[CrossRef](#)]



© 2020 by the authors. Licensee MDPI, Basel, Switzerland. This article is an open access article distributed under the terms and conditions of the Creative Commons Attribution (CC BY) license (<http://creativecommons.org/licenses/by/4.0/>).