A New Highly Efficient Differential Evolution Scheme and Its Application to Waveform Inversion

Zhaoqi Gao, Zhibin Pan, and Jinghuai Gao

***Abstract*—In this letter, a new differential evolution (DE) algorithm is proposed and applied to waveform inversion. The traditional evolution strategy of this algorithm is not efficient because it treats the individuals in a population equally and evolves all of them in each generation. In order to overcome this shortcoming, we propose a new population evolution strategy (PES) to decrease the population size based on the differences among individuals during an evolution process. We embed the new strategy into the cooperative coevolutionary DE (CCDE) and obtain a new highly efficient DE (HEDE). We apply this new algorithm to waveform inversion experiments of both synthetic and real seismic data to test its performance and demonstrate its validity. The results have clearly shown that, under the same inversion precision, the HEDE can reduce the runtime by about 50% compared with the CCDE.**

***Index Terms*—Cooperative coevolutionary differential evolution (CCDE), differential evolution (DE), population evolution strategy (PES), waveform inversion.**

I. Introduction

WAVEFORM inversion becomes increasingly popular in the exploration geophysics industry because it can develop an accurate earth model. In waveform inversion, gradient-based algorithms are widely used [1], [2]. However, the successful application of these algorithms remains a challenge because of the nonlinearity of the mathematical problems and the existence of its local minima, which requires an accurate initial model to avoid. Global optimization methods, on the other hand, do not need the computation of the gradient nor an accurate initial model. These methods are more and more attractive to geophysicists [3], [4].

Evolutionary algorithms (EAs) are a set of global optimization methods. They use a mechanism inspired by biological evolution to seek optimal solutions. In the past two decades, different kinds of EAs, such as evolutionary programming [5], genetic algorithm [6], [7], and differential evolution (DE) [8], [9], have been developed and applied in geophysical problems. However, a conventional EA has two defects: first, its performance deteriorates quickly as the dimension of search space increases beyond 500 [10]; second, it is typically more time consuming.

Recently, Wang and Gao have proposed a new algorithm called cooperative coevolutionary DE (CCDE) [11]. Compared with some other methods, the CCDE exhibits superior performance when it is applied to the waveform inversion problems with large model dimension. However, the CCDE still suffers from the second defect of the EA mentioned earlier.

In this letter, we propose a new highly efficient DE (HEDE) algorithm. The letter is organized as follows: In Section II, we first briefly review the theory of waveform inversion and CCDE. In Section III, a detailed description to the population evolution strategy (PES) and HEDE is provided. Finally, numerical experiments are given to test the effectiveness of HEDE in Section IV followed by conclusions in Section V.

II. Waveform Inversion and CCDE

Waveform inversion is a challenging data-fitting algorithm based on wavefield simulation to extract quantitative information of the earth from seismograms [12]. Its objective function quantitatively measures the difference between synthetic and recorded data. It is often treated as an optimization problem to find the best model that can minimize the objective function. This problem is usually highly nonlinear and has many local minima. This fact makes global optimization methods attractive to solve such problems.

As a global optimization method, the CCDE has been proven to be effective to solve a waveform inversion problem with large model dimension [11]. This method can be briefly described as follows: suppose our goal is to minimize an objective function $f(\mathbf{x})$ with respect to the D-dimensional vector of parameters $\mathbf{x} = \{x_1, x_2, \ldots, x_D\}$. The CCDE is based on a population with a size of NP, where each candidate solution is a D-dimensional vector

$$\mathbf{x}_i^G = \left\{x_{i,1}^G, x_{i,2}^G, \ldots, x_{i,D}^G\right\}, \qquad i = 1, 2, \ldots, NP \tag{1}$$

where G denotes the serial number of generation. Each candidate solution is updated iteratively by searching the global minimum of $f(\mathbf{x})$.

Manuscript received July 27, 2013; revised December 7, 2013 and January 20, 2014; accepted February 1, 2014. Date of publication March 5, 2014; date of current version May 12, 2014. This work was supported in part by the Specialized Research Fund for the Doctoral Program of Higher Education under Grant 20120201110016, by the National Science and Technology Major Project of China under Grant 2011ZX05023-005-009 and Grant 2011ZX05044, and by the Open Project Program of the State Key Laboratory of Computer-Aided Design and Computer Graphics, Zhejiang University under Grant A1115.

Z. Gao and J. Gao are with the School of Electronic and Information Engineering and the National Engineering Laboratory for Offshore Oil Exploration, Xi'an Jiaotong University, Xi'an 710049, China (e-mail: adom7.la@stu.xjtu.edu.cn; jhgao@mail.xjtu.edu.cn).

Z. Pan is with the School of Electronic and Information Engineering, Xi'an Jiaotong University, Xi'an 710049, China (e-mail: zbpan@mail.xjtu.edu.cn).

Color versions of one or more of the figures in this paper are available online at http://ieeexplore.ieee.org.

Digital Object Identifier 10.1109/LGRS.2014.2306263

The initial population $(G = 0)$ of the CCDE covers the entire searching space by uniformly randomizing individuals within the searching space constrained by prescribed minimum and maximum bounds. After generating the initial population, individuals are refined by applying mutation, crossover, and selection, iteratively.

During mutation, for each candidate solution $\mathbf{x}_i^G$, three distinct solutions $\mathbf{x}_{r1}^G$, $\mathbf{x}_{r2}^G$, and $\mathbf{x}_{r3}^G$, which are mutually different and different from $\mathbf{x}_i^G$, are randomly selected to generate a mutant individual $\mathbf{v}_i^G$. Interested readers are referred to [11] for details of this operation.

After mutation, the crossover operation creates a trial individual $\mathbf{u}_i^G$ by getting its variables from $\mathbf{x}_i^G$ and $\mathbf{v}_i^G$ based on the following scheme:

$$u_{i,n}^G = \begin{cases} v_{i,n}^G, & \text{if } h_n \le CR \\ x_{i,n}^G, & \text{if } h_n > CR \end{cases} \tag{2}$$

where $n = 1, 2, \ldots, D$; h_n is a random number within [0,1]; and $CR \in (0, 1)$ is a predefined crossover probability.

Finally, during the selection operation, the offspring $\mathbf{x}_i^{G+1}$ is generated by comparing the candidate solution $\mathbf{x}_i^G$ with the trial individual $\mathbf{u}_i^G$ that gives the smaller value of the objective function as follows:

$$\mathbf{x}_i^{G+1} = \begin{cases} \mathbf{u}_i^G, & \text{if } f\left(\mathbf{u}_i^G\right) \le f\left(\mathbf{x}_i^G\right) \\ \mathbf{x}_i^G, & \text{otherwise.} \end{cases} \tag{3}$$

III. HEDE

The CCDE can solve a waveform inversion problem with large model dimension effectively. However, it requires calculating forward modeling for each individual in the population in every generation. If the stopping criterion is set to $G > G_{\max}$ and the population size is NP, the entire evolution process will need $G_{\max} \times NP$ forward modeling, which can be very time consuming. If the number of the forward modeling can be reduced, the efficiency will be significantly improved. Now, the question is: can we effectively reduce the number of forward modeling?

By carefully evaluating the evolution process of the CCDE, we find that the evolution strategy of equally treating individuals in a population is not optimal, since the contribution of individuals in a population is obviously different. For a particular population, only some dominant individuals are significant to the evolution, others only make small contribution and may be eliminated. Inspired by this fact, in this letter, we propose a more reasonable evolution strategy called the PES.

A. PES

Different from the conventional strategy, the PES divides the whole evolution process into two phases: complete evolution and selective evolution.

Complete Evolution Phase: In this phase, each individual in a population will be evolved equally and kept to ensure an optimal result at the end. This is because individuals are initially uniformly distributed in the parameter space; in addition, at the beginning of an evolution, the diversity of the population is crucial for finding the optimal solution, keeping all individuals in the population can guarantee the diversity.

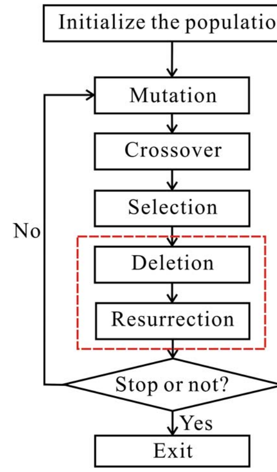

Fig. 1. Flow diagram of HEDE. Two new operations defined in HEDE are in the dashed box.

Selective Evolution Phase: In this phase, individuals are divided into two groups based on their influence on the evolution. Those whose influences are strong would be preserved, whereas those having weak influence will be left out.

Through the selective evolution phase, the PES can considerably reduce the number of individuals in the evolution, which leads to a significant improvement of the efficiency in solving waveform inversion problems.

B. HEDE

Here, we propose a new HEDE algorithm by embedding the PES discussed earlier into CCDE, and its workflow is shown in Fig. 1.

In HEDE, an important step is to define a reasonable criterion about the transformation between "Complete evolution phase" and "Selective evolution phase."

By analyzing an evolution process, we find that, when an evolution is in its complete evolution phase, each individual has equal opportunity to be improved toward the optimal solution. As the evolution continues, the average difference of the best and worst individual in each dimension will reduce. Based on this phenomenon, the phase transformation of HEDE will take place when the following inequality is true:

$$\frac{\left| f\left(\mathbf{x}_{\text{worst}}^G\right) - f\left(\mathbf{x}_{\text{best}}^G\right) \right|}{D} \le \alpha \tag{4}$$

where D is the dimension of the individuals, $f(\mathbf{x}_{\text{worst}}^G)$ and $f(\mathbf{x}_{\text{best}}^G)$ are the worst and best individuals in generation G, respectively. α controls when the phase transformation takes place.

HEDE shares the same mutation, crossover, and selection operations with CCDE. However, this new algorithm has added two new operations, i.e., deletion and resurrection, where individuals would be judged every three generations since these generations have a strong correlation.

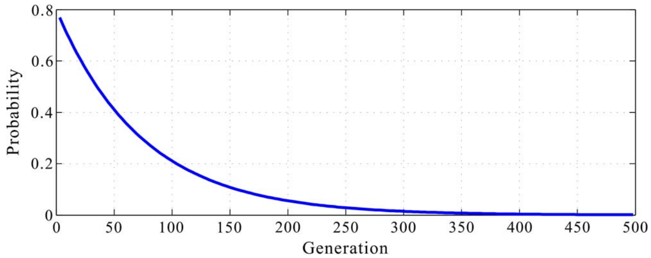

Fig. 2. Probability function used in the resurrection operation. The x-axis is the generation, and the y-axis is the probability value.

In the HEDE, the deletion operation determines which individuals should be ruled out in the selective evolution phase, which will be explained in detail in the following section.

Deletion Operation: For generation G, if it can be expressed as $G = 3 \times k,\ k = 1, 2, 3, \ldots$ and all the following four conditions are true at the same time, this individual will not be evolved in the following three generations.

- Condition 1: The evolution process is in the selective evolution phase.
- Condition 2: The population size in generation G is not less than a given minimum, which is defined as a positive real number $\gamma \in (0, 1)$ plus the initialized population size.
- Condition 3: The test individual participates evolution in generations G, $G - 1$, and $G - 2$.
- Condition 4: The values of the global fitness function for the test individual are all ranked at the bottom $(1 - \beta) \times 100\%$ in generations G, $G - 1$ and $G - 2$; here, β is a parameter that controls the proportion of individuals that will not be deleted in the population.

Considering the fact that the importance of an individual to a population is changing during the evolution process, i.e., a deleted individual in generation G may make significant contributions in later evolutions. Based on this, we introduce resurrection operation into the HEDE to resurrect some deleted individuals to guarantee the diversity of the population. At the beginning of an evolution, more individuals may be resurrected than the later evolution to ensure the diversity of the population is enough to find the optimal result. The resurrection operation is explained in detail in the following section.

Resurrection Operation: If generation G can be expressed as $G = 3 \times k,\ k = 1, 2, 3, \ldots$ and there exist N individuals that do not participate in evolution, we then randomly resurrect $\lfloor N \times p \rfloor$ of them into the evolution in the next three generations, i.e., $G + 1$, $G + 2$, and $G + 3$, where p is the probability defined by an exponential function shown in Fig. 2.

By introducing deletion and resurrection operations, the number of individuals involved in the evolution will be dynamically reduced during the evolution process in HEDE. In the following tests, we show that the HEDE considerably improves the efficiency of CCDE without degrading the result of waveform inversion.

IV. Application of HEDE to Waveform Inversion

Here, we apply HEDE to waveform inversion problems. For simplicity, we study an N-layer 1-D model $\mathbf{m} = [v_1, v_2, \ldots, v_N]$, where v_i denotes the ith layer velocity. Let $d_{\mathrm{obs}}(t)$ be the observed data and $d_{\mathrm{cal}}(t)$ be the calculated data using the model. The objective function (global fitness function) is defined as follows:

$$f(\mathbf{m}) = \sum_{r=1}^{Nr} \int |d_{\mathrm{obs},r}(t) - d_{\mathrm{cal},r}(t, \mathbf{m})|\, dt + \lambda \|\mathbf{m}\|_1 \quad (5)$$

where Nr is the number of receivers, and t denotes the travel time. We also include a regularization, i.e., the ℓ_1 norm of the model parameters multiplied by a regularization parameter λ, in our objective function to stabilize the inversion. We choose the ℓ_1 norm in the regularization scheme because, as compared with the ℓ_2 norm, ℓ_1 norm can preserve more local information of the model. The local fitness function of HEDE is defined as follows:

$$Lf_j(\mathbf{m}) = \sum_{r=1}^{Nr} \int |[d_{\mathrm{obs},r}(t) - d_{\mathrm{cal},r}(t, \mathbf{m})] \cdot window_j(t)|\, dt \quad (6)$$

where $window_j(t)$ denotes a window function, which is defined according to a specific problem. More information about the local fitness function can be refereed to [11].

Here, we take $G > G_{\max}$ as the stopping criterion, where $G_{\max}$ is the maximum number of generations. When the evolution process is completed, the vector yielding the smallest objective function in generation $G_{\max}$, which is noted as $\mathbf{x}_{\mathrm{best}}^{G_{\max}}$, is taken as the solution of the inversion problem.

When the HEDE is applied for waveform inversion, the key is to determine the values of parameters α and β. In order to determine the best parameters, we conduct the following tests.

In this letter, all experiments use the same objective function and local fitness function, as defined in (5) and (6); the value of γ is set to 0.2 empirically; and each subcomponent only contains one layer. The length of $window_j(t)$ used in (6) is chosen to be three times of the length of the subcomponent plus the width of the wavelet.

HEDE has been compared with CCDE for a velocity model taken from a real well log, as shown in Fig. 3. The model is parameterized as a function of travel time. There are 200 velocities to be estimated. A low-frequency trend of the real velocity is used to define the upper and lower bounds by superimposing ± 0.8 km/s. Two experiments have been conducted to find how the different values of α and β can influence the performance of HEDE. In both experiments, we choose the regularization parameter $\lambda = 0$ because the synthesized data

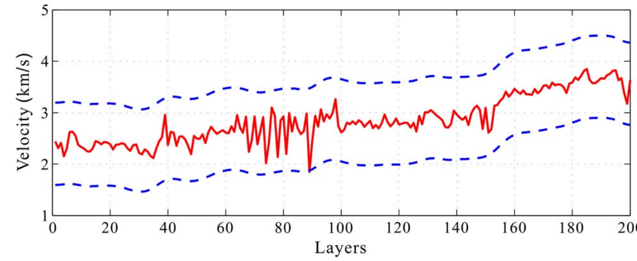

Fig. 3. Real velocity model taken from a real well log (red) and the search space defined by upper and lower bounds (blue).

TABLE I
PERFORMANCE COMPARISON OF HEDE WITH DIFFERENT α AND CCDE WHEN $D = 200$. (RT: RUNTIME; IR: IMPROVEMENT RATIO (COMPARED TO CCDE); NFM: NUMBER OF FORWARD MODELING)

	CCDE	$\alpha = 0.3$	$\alpha = 0.03$	$\alpha = 0.003$
RT(s)	135.76	66.92	68.51	77.62
IR (%)	0	50.71	49.54	42.82
NFM	50000	23204	23934	27975

TABLE II
PERFORMANCE COMPARISON OF HEDE WITH DIFFERENT β AND CCDE WHEN $D = 200$. (RT: RUNTIME; IR: IMPROVEMENT RATIO (COMPARED TO CCDE); NFM: NUMBER OF FORWARD MODELING)

	$\beta = 0.4$	$\beta = 0.5$	$\beta = 0.6$	$\beta = 0.7$	$\beta = 0.8$	$\beta = 0.9$
RT(s)	41.92	45.72	46.19	50.07	55.92	68.51
IR (%)	69.12	66.33	65.98	63.12	58.81	49.54
NFM	14020	15055	16035	17667	19779	23934

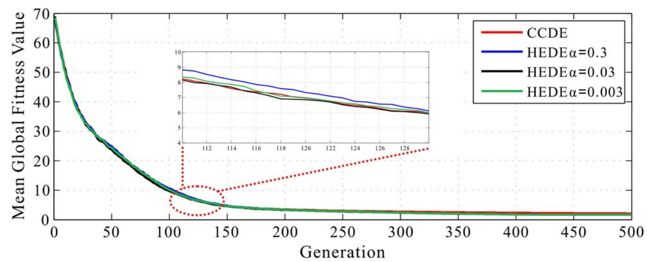

Fig. 4. Best global fitness values recorded in the evolution process. Inset shows detailed information in the dashed ellipse.

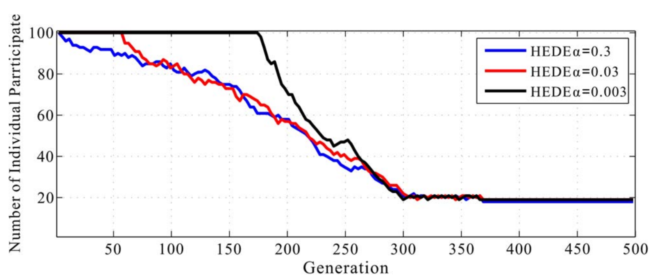

Fig. 5. Number of individuals that participate in the evolution.

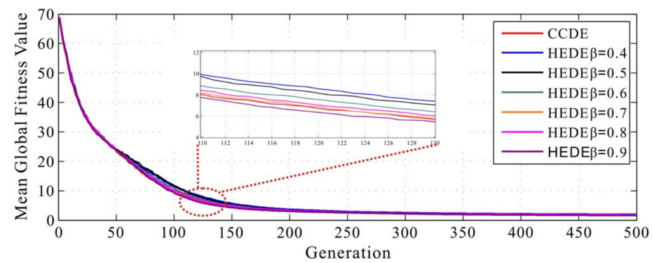

Fig. 6. Best global fitness values recorded in the evolution process. Inset shows detailed information in the dashed ellipse.

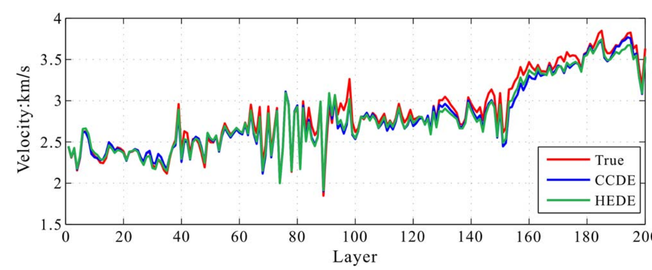

Fig. 7. Comparison of the inverted velocity by CCDE and HEDE with the real velocity. The results are averaged over 35 independent runs.

are noise free. We use $G_{\max} = 500$ as the stopping criterion. The initial population size is set as $NP = 100$.

1) Sensitivity of HEDE to Parameter α: To test the influence of parameter α to HEDE, we fix $\beta = 0.9$. Three different values of α with different orders of magnitude are tested. The convergence, the runtime, and the number of forward modeling are tested. From Table I and Figs. 4 and 5, we can conclude the following.

a) HEDE is quite insensitive to the value of α, as shown in Fig. 4, where the convergence behaviors are very close to each other, although the convergence curve for $\alpha = 0.3$ is slightly worse than the others. As mentioned earlier, parameter α determines which generation is the boundary of the complete evolution and the selective evolution phase. Fig. 5 shows the number of individuals that participate in the evolution in each generation; the starting generation of the selective evolution phase is 1, 48, and 165, for $\alpha = 0.3$, 0.03, and 0.003, respectively.

b) Compared with CCDE, the CPU time of HEDE with different α is all significantly reduced. The highest improvement ratio is 50.71%, while the lowest is 42.82%. The best choice for HEDE is $\alpha = 0.03$ because, as compared with CCDE, the accuracy of the result is kept unchanged and the runtime can be reduced by 49.54%.

2) Sensitivity of HEDE to Parameter β: Here, we fix $\alpha = 0.03$ in the following experiments, where six different β are tested. From Table II and Fig. 6, we can conclude as follows:

a) HEDE is strongly dependent on β. In Fig. 6, we find that the convergence of HEDE is much worse than CCDE when β is less than 0.7. This is because, when β is too small, too many individuals will be deleted, the diversity of the population is reduced, and its performance will then be degraded. The result in the experiment confirms the theoretical analysis.

b) When β is larger than 0.7, HEDE performs almost the same as CCDE does. In the following experiments, we use $\beta = 0.9$ since this choice gives the optimal convergence.

In general, we find that HEDE with parameters $\alpha = 0.03$, $\beta = 0.9$ produce the best result for our experiments. The best global fitness values in generation 500 are 1.6274 and 2.1206 for HEDE and CCDE, respectively. Therefore, compared with CCDE, the HEDE can produce a better solution for waveform inversion with less than half of the runtime. In Fig. 7, we show the comparison of the true velocity model with the inverted velocity models using CCDE and HEDE, respectively. The inverted velocities match very well, and they both approximate to the real data nicely.

In addition, HEDE with parameters $\alpha = 0.03$ and $\beta = 0.9$ is also applied to real post-stack seismic data, which contain 410 traces and 201 ms long. In dealing with real seismic data, regularization is extremely necessary in order to obtain stable inverted model parameters. We choose the regularization

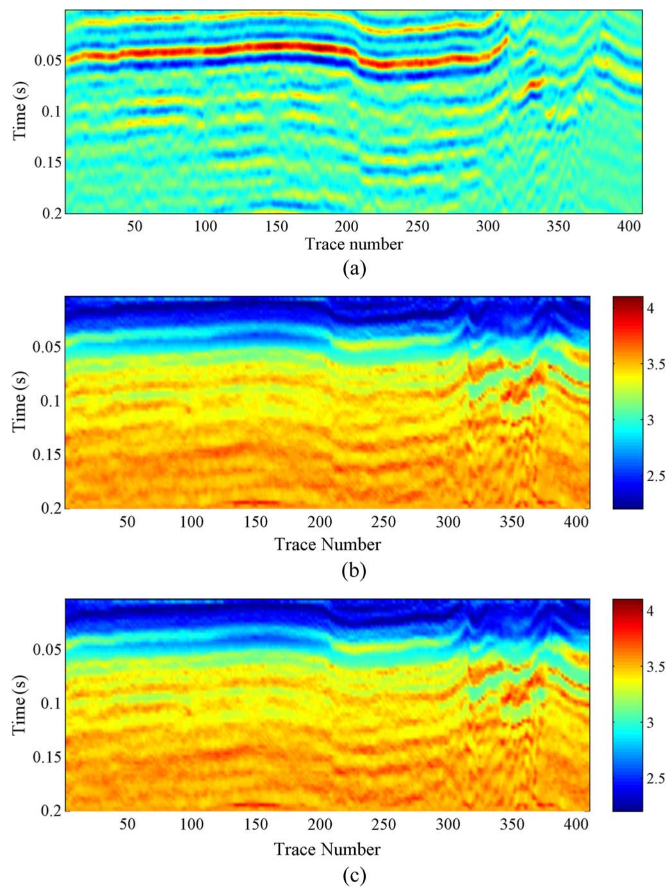

Fig. 8. (a) Real seismic data. (b) Inverted velocity model by CCDE. (c) Inverted velocity model by HEDE.

parameter $\lambda = 0.3$ in this experiment. Although this parameter is chosen empirical here, the theory and computer algorithms for selection of an optimal regularization parameter can be found in many publications [13], [14]. We divided the model into 67 layers with 3 ms of travel time for each layer. The density is assumed to be constant. We set the initial population size $NP = 50$, and the computation stops at generations 500. For comparison, CCDE is also applied to these data. Both results are shown in Fig. 8, from which we can observe that the inverted velocities of HEDE and CCDE are quite comparable with each other and they both resolve many geological features. However, their efficiencies are quite different. HEDE uses 72 s for each trace, whereas CCDE uses 142 s. HEDE is almost doubled the efficiency.

V. Conclusion

In this letter, a better evolution strategy called the PES has been proposed, which divides the whole evolution process into two phases, i.e., complete evolution and selective evolution. By integrating the PES into CCDE, we proposed a new global optimization method called the HEDE and applied it to waveform inversion. Different from the CCDE, individuals in the HEDE are treated adaptively based on their contributions in a generation. In the selective evolution phase, after every three generations, the HEDE finds some individuals by the criterion defined in the deletion operation and keeps them away from evolving in the next three generations. At the same time, in order to preserve the diversity of the whole population, the HEDE also revives some deleted individuals and continues to evolve them in the next three generations.

Experiments have shown that HEDE is very efficient for solving waveform inversion problems with large model dimension. Compared with CCDE, the HEDE can obtain inversion results of the same precision with nearly double efficiency.

References

[1] G. A. Meles, J. V. der Kruk, S. A. Greenhalgh, J. R. Ernst, H. Maurer, and A. G. Green, “A new vector waveform inversion algorithm for simultaneous updating of conductivity and permittivity parameters from combination crosshole/borehole-to-surface GPR data,” *IEEE Trans. Geosci. Remote Sens.*, vol. 48, no. 9, pp. 3391–3407, Sep. 2010.

[2] F. A. Belina, J. R. Ernst, and K. Holliger, “Inversion of crosshole seismic data in heterogeneous environments: Comparison of waveform and ray-based approaches,” *J. Appl. Geophys.*, vol. 68, no. 1, pp. 85–94, May 2009.

[3] S. Horen and C. Macbeth, “A comparison of global optimization methods for near-offset VSP inversion,” *Comput. Geosci.*, vol. 24, no. 6, pp. 563–572, Jul. 1998.

[4] R. Shaw and S. Srivastava, “Particle swarm optimization: A new tool to invert geophysical data,” *Geophysics*, vol. 72, no. 2, pp. 75–83, Mar. 2007.

[5] C. D. Groot-Hedlin and F. L. Vernon, “An evolutionary programming method for estimating layered velocity structure,” *Bull. Seismol. Soc. Amer.*, vol. 88, no. 4, pp. 1023–1035, Aug. 1988.

[6] R. Govindan, R. Kumar, S. Basu, and A. Sarkar, “Altimeter-derived ocean wave period using genetic algorithm,” *IEEE Geosci. Remote Sens. Lett.*, vol. 8, no. 2, pp. 354–358, Mar. 2011.

[7] P. Soupios, I. Akca, P. Mpogiatzis, A. T. Basokur, and C. Papazachos, “Application of hybrid genetic algorithms in seismic tomography,” *J. Appl. Geophys.*, vol. 75, no. 3, pp. 479–489, Nov. 2011.

[8] A. Semnani, M. Kamyab, and I. T. Rekanos, “Reconstruction of one-dimensional dielectric scatterers using differential evolution and particle swarm optimization,” *IEEE Geosci. Remote Sens. Lett.*, vol. 6, no. 4, pp. 671–675, Oct. 2009.

[9] M. Dehmollaian, “Through-wall shape reconstruction and wall parameters estimation using differential evolution,” *IEEE Geosci. Remote Sens. Lett.*, vol. 8, no. 2, pp. 201–205, Mar. 2011.

[10] S. Das and P. N. Suganthan, “Differential evolution: A survey of the state-of-the-art,” *IEEE Trans. Evol. Comput.*, vol. 15, no. 1, pp. 4–31, Feb. 2011.

[11] C. Wang and J. H. Gao, “High-dimensional waveform inversion with cooperative coevolutionary differential evolution algorithm,” *IEEE Geosci. Remote Sens. Lett.*, vol. 9, no. 2, pp. 297–301, Mar. 2012.

[12] J. Virieux and S. Operto, “An overview of full-waveform inversion in exploration geophysics,” *Geophysics*, vol. 74, no. 6, pp. 127–152, Nov. 2009.

[13] M. S. Zhdanov, “Geophysical inverse theory and regularization problems,” in *Methods in Geochemistry and Geophysics*. Amsterdam, The Netherlands: Elsevier, 2002.

[14] P. C. Hansen, *Discrete Inverse Problems: Insight and Algorithms*. Philadelphia, PA, USA: SIAM, 2010.