SCIENCE CHINA
Information Sciences

• RESEARCH PAPER •

October 2019, Vol. 62 202101:1–202101:16
https://doi.org/10.1007/s11432-018-9852-8

Wave models and dynamical analysis of evolutionary algorithms

Yuanxiang LI*, Zhenglong XIANG* & Daomin JI

School of Computer Science, Wuhan University, Wuhan 430072*, China*

Received 18 October 2018/Revised 2 January 2019/Accepted 29 March 2019/Published online 3 September 2019

Abstract By drawing an analogy between the population of an evolutionary algorithm and a gas system (which we call a particle system), we first build wave models of evolutionary algorithms based on aerodynamics theory. Then, we solve the models' linear and quasi-linear hyperbolic equations analytically, yielding wave solutions. These describe the propagation of the particle density wave, which is composed of leftward and rightward waves. We demonstrate the convergence of evolutionary algorithms by analyzing the mechanism underlying the leftward wave, and investigate population diversity by analyzing the rightward wave. To confirm these theoretical results, we conduct experiments that apply three typical evolutionary algorithms to common benchmark problems, showing that the experimental and theoretical results agree. These theoretical and experimental analyses also provide several new clues and ideas that may assist in the design and improvement of evolutionary algorithms.

Keywords evolutionary algorithm, wave model, convergence analysis, phase transition, dynamical analysis

Citation Li Y X, Xiang Z L, Ji D M . Wave models and dynamical analysis of evolutionary algorithms. Sci China Inf Sci, 2019, 62(10): 202101, https://doi.org/10.1007/s11432-018-9852-8

1 Introduction

Evolutionary algorithms are a class of heuristic swarm intelligence algorithms that have been widely studied and applied [1]. Recent decades have seen substantial in algorithm design [2, 3], theoretical analysis [4–6], and practical applications [7, 8].

From the 1990s to the early 2000s, the convergence of evolutionary algorithms was usually analyzed using finite Markov chain models from stochastic process theory. These Markov models were generally on the basis of reasonable hypotheses about the algorithms, and then their probabilistic convergence was proved using stochastic process theory. Some convergence criteria have also been derived via theoretical analysis. Summary and review articles have been written by the pioneers of this approach, such as Goldberg [9], Rudolph [10], among others. The primary approach to studying the time complexity is drift analysis, first introduced by Yao and He [11]. In this method, a distance function is defined and used to calculate the average distance between the current generation of solutions and the objective function's optimal solution, so as to estimate the average time when the optimal solution will be reached. Hence, the analysis is still based on Markov chain models. Later, they developed a general framework for analyzing the computation times of evolutionary algorithms based on absorbing Markov chains. Sudholt [12] proposed a method of analyzing the time complexity based on dividing the fitness into levels, and proved lower bounds on the expected running times of evolutionary algorithms by introducing an additional condition on the probabilities of transitions between fitness levels. Recently, Yu [13] and Bian [14] proposed

* Corresponding author (email: yxli@whu.edu.cn, zl_xiang@whu.edu.cn)

the switch analysis to estimate the running time of evolutionary algorithms and multi-objective evolutionary algorithms, based on studying alignment mappings between pairs of Markov chains. So far, the primary methods of analyzing evolutionary algorithms theoretically have come from stochastic process theory, particularly Markov chain models. These theoretical results have therefore been probabilistic in nature, and could hardly be used to guide algorithm design and improvement.

Another way of viewing, the population of an evolutionary algorithm is looked as a particle system, such as a gas. Given that, it is natural to apply dynamical systems theory and gas dynamics to model and analyze algorithms. In fact, methods taken from dynamical systems and statistical physics have already been used to model and design evolutionary algorithms. For instance, defining the gene entropy, Mori et al. [15] presented a thermodynamic selection rule for genetic algorithm based on defining the gene entropy. Cornforth and Lipson [16] designed a hybrid evolutionary algorithm for modeling multiple-time-scale dynamical systems symbolically, and proposed the idea of combining evolutionary algorithms with dynamical system modeling. By improving the way the entropy computed, we presented a general dynamical evolutionary algorithm based on the law of minimum free energy from statistical physics [17]. Then, we built dynamical system models of simulated annealing algorithm based on the theory of elastic mechanics [18], subsequently using them to analyze the convergence and time complexity [19]. Other results have also been derived within a dynamical systems theory framework that can be used in practice to design and improve the simulated annealing algorithm. Even though simulated annealing algorithm conduct iterative single point searches, we can still learn from and generalize these, because the underlying heuristic is interlinked with that of evolutionary algorithms. In addition, the nature of evolutionary algorithms is such that they are suited to being described by dynamical systems and aerodynamics theories.

By drawing an analogy between the population of an evolutionary algorithm and a gas system (which we call a particle system), we first build wave models of evolutionary algorithms based on aerodynamics theory. Next, we solve the model's hyperbolic equations analytically to obtain wave solutions that describe the particle density distribution's wavelike behavior. Then, we investigate these solutions, considering, for example, the propagation of the particle density waves, the diversity of the particle systems, and the convergence and convergence rate of evolutionary algorithms. After that, we carry out experiments to confirm these theoretical results. We also explain and extend the insights gained from the theoretical analysis and experiments so as to understand their meaning and consequence for the design and improvement of algorithms. Finally, we discuss the prospects for future research.

The remainder of this paper is organized as follows. In Section 2, we explain the analogy between the population of an evolutionary algorithm and a particle system. Next, we introduce a pressure formula and build a wave model aerodynamics theory, then discuss linearizing the model. In Section 3, we analytically solve the hyperbolic equations of linear and nonlinear models, which describe particle density waves. Then, we use these wave solutions to analyze the propagation of the particle density waves. After that, in Section 4, we investigate the convergence and population diversity of evolutionary algorithms in detail. In Section 5, we carry out experiments to verify the theoretical results and related explanations. Finally, in Section 6, we present our conclusion, summarizing the content of this paper and discussing future work.

2 Aerodynamic models for evolutionary algorithms

Consider an evolutionary algorithm for solving a minimization problem, based on definitions for genetic codes and evolutionary operators. The population consists of N individuals (which we refer to as particles), whose fitness values are given by the objective function and used to define the particle coordinates. We liken this population to a gas system of N particles, and the evolutionary process, whereby individual fitnesses change generation by generation, is analogous to the particles in the gas system moving over time. Our analysis ignores the way that individuals' genetic codes transform, instead focusing on the fitness changes due to coding and evolutionary operations, namely, the changes in particle coordinates.

Without loss of generality, let the minimum of the objective function be 0 and its values be bounded above by M, meaning that all the particles move in the interval $[0, M]$, which is called the function value space.

During the evolutionary process, all the particles are distributed within the function value space $[0, M]$. At a given time (evolution generation) t and coordinates x, there is a particle micelle (mass point), whose velocity, density, and pressure are denoted by u, ρ, and p. The particle population's evolution in the function value space in terms of the one-dimensional non-viscous motion of a particle or a gas system [20]. Hence, according to aerodynamics theory, the equations of motion of the particle system are given by

$$\frac{\partial \rho}{\partial t} + u\frac{\partial \rho}{\partial x} + \rho\frac{\partial u}{\partial x} = 0, \tag{1}$$

$$\rho\frac{\partial u}{\partial t} + \rho u\frac{\partial u}{\partial x} + \frac{\partial p}{\partial x} = 0, \tag{2}$$

where Eqs. (1) and (2) express the conservation of mass and momentum, respectively. We have not included the energy conservation equation here, since we are focusing on the motion of the particle system. Aerodynamics theory also provides us with a formula for the pressure in a low-speed micro-compressible gas system [21], namely, $p = \kappa\frac{(\rho-\rho_0)}{\rho_0}$, where κ denotes the coefficient of volume elasticity. In our particle system, this reflects the local system compression caused by evolutionary operations and the evolution of the particle population. Here, ρ_0 is a reference density value, such as the density at a certain point or the mean of the initial particle distribution. By substituting the pressure formula into (2), we can rewrite it as

$$\rho\frac{\partial u}{\partial t} + \rho u\frac{\partial u}{\partial x} + \frac{\kappa}{\rho_0}\frac{\partial \rho}{\partial x} = 0. \tag{3}$$

Eqs. (1) and (3) comprise a quasi-linear hyperbolic equation system, based only on the velocity u and density ρ. Setting definite conditions, such as initial conditions, then enables us to solve this equation system.

To clarify the nature of these wave models, we now linearize (1) and (3) to obtain a simplified and easily solved linear wave model, whose solution will assist us help in understanding wave propagation and solving the quasi-linear hyperbolic equation system. As mentioned above, the particle system can be seen as a low-speed, micro-compressible gas motion system. Here, we infer that "low-speed" means u and $\frac{\partial u}{\partial x}$ are small, while "micro-compressible" means that $\frac{\partial \rho}{\partial x}$ is a small, as are the changes in ρ relative to ρ_0. As a result, the second nonlinear terms in (1), (2), and (3) are second-order small quantities, and hence can be neglected. In addition, the variable coefficient ρ can be replaced by the constant coefficient ρ_0. Based on this analysis, Eqs. (1) and (3) can be reduced to the following system of linear hyperbolic equations, known as the acoustics equation system in aerodynamics:

$$\frac{\partial \rho}{\partial t} + \rho_0\frac{\partial u}{\partial x} = 0, \tag{4}$$

$$\rho_0\frac{\partial u}{\partial t} + a^2\frac{\partial \rho}{\partial x} = 0, \tag{5}$$

where $a = \sqrt{\kappa/\rho_0}$, called the acoustic speed, is the propagation speed of the gas density wave.

By taking partial derivatives of (4) with respect to t and (5) with respect to x, and then subtracting, we can eliminate the partial derivative product terms in these two equations. This results in a second-order linear hyperbolic equation in terms of ρ alone, namely,

$$\frac{\partial^2 \rho}{\partial t^2} - a^2\frac{\partial^2 \rho}{\partial x^2} = 0. \tag{6}$$

Similarly, we can also obtain a second-order linear hyperbolic equation in terms of the velocity u, with the same form as (6) expect with ρ replaced by u. However, as we mainly focus on the changes in the particle density ρ in this paper, we do not present this here.

3 Solving the wave equations

The hyperbolic (quasi-linear and linear) equations given above are also called wave equations, because their solutions describe the propagation of waves in spacetime. Here, we start by solving the linear wave equation (6) so as to understand the principle of wave propagation, then we solve the quasi-linear wave equation system to obtain the wave solution for the particle density ρ. After that, we use these solutions to analyze the search process and convergence of evolutionary algorithms.

3.1 Solving the linear wave equation

For simplicity, we start with the simple linear wave equation. First, we define two new variables [22]:

$$\xi = x - at, \quad \eta = x + at, \tag{7}$$

and substitute these into (6), using the chain rule for the derivatives of compound functions to transform it into the following:

$$\frac{\partial^2 \rho}{\partial \xi \partial \eta} = 0. \tag{8}$$

By integrating this with respect to ξ and η, we can easily derive the following formal solution:

$$\rho(\xi, \eta) = F(\xi) + G(\eta). \tag{9}$$

Next, by substituting (7) into (9), we can derive general solution:

$$\rho(t, x) = F(x - at) + G(x + at), \tag{10}$$

where $F(\cdot)$ and $G(\cdot)$ are two arbitrary differentiable functions.

After initializing the population for a given evolutionary algorithm, we can estimate the initial particle density distribution on the interval $[0, M]$, denoted by $\phi(x)$. In addition, the particles are stationary when the algorithm begins, so the initial conditions are given by

$$\rho|_{t=0} = \phi(x), \quad u|_{t=0} = 0, \quad \left.\frac{\partial \rho}{\partial t}\right|_{t=0} = 0. \tag{11}$$

Then, applying these initial conditions to the general solution (10) gives us

$$F(x) + G(x) = \phi(x), \tag{12}$$

$$a(-F'(x) + G'(x)) = 0. \tag{13}$$

Next, integrating (13) with respect to x yields

$$a(-F(x) + G(x)) = C, \tag{14}$$

where C is the constant of integrational. Solving (12) and (14) simultaneously enables us to determine the functions $F(\cdot)$ and $G(\cdot)$ as

$$F(x) = \frac{1}{2}\phi(x) - \frac{C}{2a}, \quad G(x) = \frac{1}{2}\phi(x) + \frac{C}{2a}. \tag{15}$$

Substituting (15) into (10) then produces the solution to (6), namely,

$$\rho(t, x) = \frac{1}{2}(\phi(x - at) + \phi(x + at)). \tag{16}$$

3.2 Solving the quasi-linear wave equation system

Using the acoustic speed a, we can rewrite (3) as

$$\frac{\partial u}{\partial t}+\frac{a^2}{\rho}\frac{\partial \rho}{\partial x}+u\frac{\partial u}{\partial x}=0. \tag{17}$$

Then, we can define the following first-order quasi-linear hyperbolic equation system by combining (1) and (17):

$$\begin{cases}\frac{\partial \rho}{\partial t}+u\frac{\partial \rho}{\partial x}+\rho\frac{\partial u}{\partial x}=0,\\ \frac{\partial u}{\partial t}+\frac{a^2}{\rho}\frac{\partial \rho}{\partial x}+u\frac{\partial u}{\partial x}=0.\end{cases} \tag{18}$$

Since this system can be solved by a standard application of the characteristic theory of hyperbolic equations, we will omit the details here. Instead, we will only state the main steps, providing necessary intermediate results and the final solutions [20, 22].

First, in order to determine the characteristic directions (or characteristic lines) in the x-t plane, we calculate the eigenvalues of P, the coefficient matrix of system (18) with respect to the spatial variable x, given by

$$P=\begin{bmatrix} u & \rho \\ \frac{a^2}{\rho} & u\end{bmatrix}. \tag{19}$$

The eigenvalues of P can easily be obtained as $\lambda_1=u+a$ and $\lambda_2=u-a$. Then, the characteristic directions in the x-t plane are

$$\frac{\mathrm{d}x}{\mathrm{d}t}=u+a, \quad \frac{\mathrm{d}x}{\mathrm{d}t}=u-a. \tag{20}$$

Second, by using the eigenvectors corresponding to these eigenvalues, we can transform equation system (18) into the two relevant characteristic relations, along with their characteristic lines.

The eigenvector corresponding to λ_1 is simply (a,ρ). Multiplying the first equation in system (18) by a and the second equation by ρ, and then adding these, yields the first characteristic relation:

$$a\left[\frac{\partial \rho}{\partial t}+(u+a)\frac{\partial \rho}{\partial x}\right]+\rho\left[\frac{\partial u}{\partial t}+(u+a)\frac{\partial u}{\partial x}\right]=0. \tag{21}$$

Likewise, the eigenvector corresponding to λ_2 is $(a,-\rho)$, and we can take the same approach to generate the second characteristic relation:

$$a\left[\frac{\partial \rho}{\partial t}+(u-a)\frac{\partial \rho}{\partial x}\right]-\rho\left[\frac{\partial u}{\partial t}+(u-a)\frac{\partial u}{\partial x}\right]=0. \tag{22}$$

Third, we derive the characteristic equations, along with their characteristic lines. To do this, we introduce two new variables r and s, called the Riemannian invariants, which are written as

$$r=\frac{1}{2}u+\frac{1}{2}\int_{\rho_0}^{\rho}\frac{a}{\rho}\mathrm{d}\rho, \quad s=-\frac{1}{2}u+\frac{1}{2}\int_{\rho_0}^{\rho}\frac{a}{\rho}\mathrm{d}\rho. \tag{23}$$

Using these Riemannian invariants, we can transform the characteristic relations (21) and (22), respectively, into two characteristic equations in the Riemannian invariants r and s:

$$\frac{\partial r}{\partial t}+(u+a)\frac{\partial r}{\partial x}=0, \tag{24}$$

$$\frac{\partial s}{\partial t}+(u-a)\frac{\partial s}{\partial x}=0. \tag{25}$$

According to the characteristic theory of hyperbolic equations, Eq. (24) indicates that the Riemannian invariant r is constant along the characteristic line $\frac{\mathrm{d}x}{\mathrm{d}t}=u+a$. Similarly, the invariant s is constant along the characteristic line $\frac{\mathrm{d}x}{\mathrm{d}t}=u-a$. In addition, we can easily verify that

$$r-s=u, \quad r+s=\int_{\rho_0}^{\rho}\frac{a}{\rho}\mathrm{d}\rho. \tag{26}$$

Without loss of generality, we can assume that a is constant, enabling us to simplify the second formula in (26) to

$$r + s = a(\ln \rho - \ln \rho_0). \tag{27}$$

As mentioned above, we concentrate in this paper on the changes in particle density while executing the algorithm, so we will focus on finding a solution for the density ρ below.

Fourth, we solve the characteristic equations. As we did when solving the linear wave equation, we derive general solutions for r and s by a similar substitution of variables with respect to (24) and (25), namely,

$$r = f(x + (u + a)t), \quad s = g(x + (u - a)t), \tag{28}$$

where $f(\cdot)$ and $g(\cdot)$ are two arbitrary differentiable functions.

Finally, we use the initial conditions to determine the functions $f(\cdot)$ and $g(\cdot)$, and obtain the particle density. Substituting the initial conditions (11) into (26)–(28), the functions $f(\cdot)$ and $g(\cdot)$ can straightforwardly be obtained, as follows:

$$f(x) = g(x) = \frac{1}{2}a(\ln \phi(x) - \ln \rho_0). \tag{29}$$

Then, by substituting the formulas (29) into those in (28) and employing the relationships given by (26) and (27) for r, s, ρ, and u, we can derive expressions for the two components of ρ, ρ_L, and ρ_R, as follows:

$$\rho_L(t, x) = \mathrm{e}^{-\frac{u}{a}}\phi(x + (a + u)t), \quad \rho_R(t, x) = \mathrm{e}^{\frac{u}{a}}\phi(x - (a - u)t). \tag{30}$$

3.3 Particle density wave propagation

Before attempting to understand nonlinear waves, it is helpful to first discuss the principles involved in linear wave propagation by analyzing the propagation of the density wave. The solution (16) to the linear wave equation (6) can be expressed as the superposition of the following two parts:

$$\rho_R(t, x) = \frac{1}{2}\phi(x - at), \quad \rho_L(t, x) = \frac{1}{2}\phi(x + at). \tag{31}$$

Now, we focus on ρ_R. At time $t_0 = 0$, $\rho_R(0, x) = \frac{1}{2}\phi(x)$ and $\phi(x)$ is the initial density distribution on the interval $[0, M]$, as shown by the example in Figure 1(a). At time $t = t_1$, $\rho_R(t_1, x_1) = \frac{1}{2}\phi(x_1 - at_1)$, showing that the initial density at position $x_0 = x_1 - at_1$ at time $t_0 = 0$, namely, $\rho_R(0, x_0) = \frac{1}{2}\phi(x_0)$, simply propagates to position $x_1 = x_0 + at_1$ at time t_1. This means that the initial density distribution moves a distance of at_1 to the right at a speed of a in the x-ρ plane, as shown by the dotted line in Figure 1(a). Over time, the distribution continues to move to the right, i.e., ρ_R simply propagates to the right, with a propagation speed of a, so it is called the rightward wave. Similarly, ρ_L propagates to the left at a speed of a, so it is called the leftward wave. The waves propagating in both directions combine linearly at $x = \frac{1}{2}(x_L + x_R)$ (where $x_L > x_R$) to form, the solution given by (16).

Furthermore, if we draw clusters of straight lines in the x-t plane such that $x = x_0 + at$ and $x = x_0 - at$, $x_0 \in [0, M]$ (Figure 1(b)), which we call characteristic lines clusters, then ρ_R propagates to the right along the first line cluster while ρ_L propagates to the left along the second line cluster.

Now that we have analyzed linear wave propagation, we turn to the solution to the quasi-linear wave equation system. Although the solution given in (30) also consists of two parts itself, these two parts cannot be overlaid because of the multi-solution resulting from the nonlinear phenomena. The multi-solution also gives an expression of diversity in nature. As for our attention, the two parts describe the nonlinear particle density wave and the way it propagates, which is different to the linear wave discussed above.

Nonetheless, we can still explain this nonlinear wave by analogy with the linear wave. As before, ρ_L, expressed by the first formula in (30), propagates at a speed of $a + u$ along the characteristic lines clusters $x = x_0 - (a + u)t$, $x_0 \in [0, M]$, although these lines are now no longer straight. The particle density increases (or decreases) by a factor of $\mathrm{e}^{-\frac{u}{a}}$ due to the nonlinear coherence. Likewise, ρ_R propagates at a

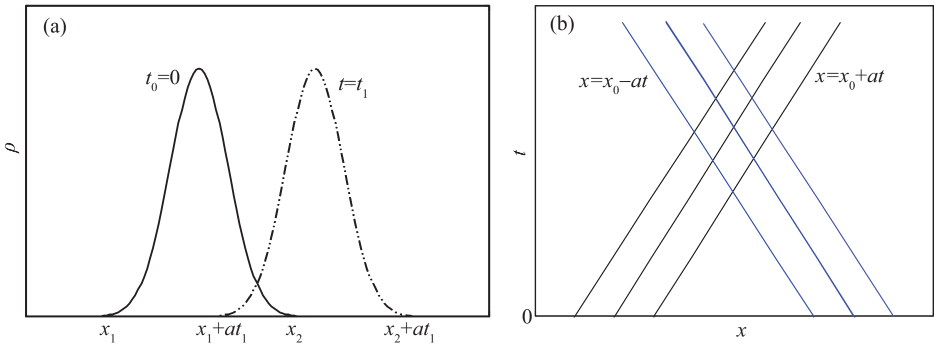

Figure 1 (Color online) Illustration showing (a) the rightward linear wave and (b) clusters of characteristic lines.

speed of $a-u$ along the cluster of characteristic lines $x=x_0+(a-u)t$ (which are again not straight), and the particle density increases (or decreases) by a factor of $\mathrm{e}^{\frac{u}{a}}$. Looking at the exponential functions $\mathrm{e}^{-\frac{u}{a}}$ and $\mathrm{e}^{\frac{u}{a}}$, we can clearly see that these increases or decreases are determined by the sign of u. The absolute value of their exponents $M_a=\frac{|u|}{a}$ is called the Mach number in aerodynamics.

The assumptions used when building the wave models in Section 2 imply that $|u|<a$, so ρ_L and ρ_R are waves that propagate to the left and right, respectively. Based on aerodynamics theory, they propagate into different areas and do not combine. One wave is called the compression wave while the other is called the sparse wave, depending on the sign of u. Relating this to the particle population of an evolutionary algorithm, the compression wave represents the tendency of particles to gather in a certain area, while the sparse wave represents the way particles are drawn away from other areas.

4 Explanation to evolutionary algorithms

The wave (change in particle density) appears to be caused by particle movement driven by evolutionary operators. Hence, the propagation of the particle density wave describes the characteristics of the evolving population, and reflects the effects the evolutionary operators have on the evolution process. Of course, different evolutionary algorithms involve different evolutionary operators, for example, the PSO algorithm's update step aims mainly to learn from the global best. However, in all evolutionary algorithms, the selection pressure drives the population to search for the optimal solutions. In principle, wave models ignore the details of the algorithms' evolutionary operators, focusing instead on the behavior they produce in the population (or particle system) as a whole. In other words, these models describe the population's evolution at a macro level. Thus, we will now analyze the macroscopic behavior of evolutionary algorithms based on the solutions to their corresponding wave models.

4.1 Analyzing the search mechanism and convergence

Based on our analysis of the particle density wave's behavior in Subsection 3.3, there are two main waves in both the linear and nonlinear model. These propagate synchronously and are known as the leftward and rightward waves. They correspond to two critical mechanisms involved in evolutionary algorithm: the leftward wave represents optimization or exploitation, while the rightward wave represents extensive search or exploration to maintain diversity. These two behaviors are vitally important to an evolutionary algorithm's search process, because they indicate the global optimization ability of the algorithm as a whole.

The solution for linear waves (31) shows that these two factors are equivalent. However, for nonlinear waves, the solution (30) indicates that they are opposite: one must be increasing while the other must be decreasing. The following experiments in Section 5 will show that the behavior of most evolutionary algorithms is governed by nonlinear waves, with linear waves being simply an ideal case. The pressure formula proposed in Section 2 can be used to adjust these two behaviors, and could contribute to the design of evolutionary algorithms. This will be the focus of our future work.

Now, we turn to analyzing the convergence of evolutionary algorithms. As described above, the leftward wave represents the algorithm's optimization and illustrates its progress toward convergence. For simplicity and clarity, we use the linear wave to concretely express this convergence without loss of generality.

After initializing the particle population, we can compute the particle density distribution $\phi(x)$, $x \in [0, M]$. Over time (generation t), the leftward wave becomes $\rho_L(t, x) = \frac{1}{2}\phi(x + at)$. Setting $x = 0$, namely, the minimum of the objective function, gives $\rho_L(t^*, 0) = \frac{1}{2}\phi(at^*)$. If we let $x^* = at^* \in [0, M]$, then $\frac{1}{2}\phi(at^*)$ will propagate along the characteristic line $x = x^* - at$ to position $x = 0$ after time $t^* = \frac{x^*}{a}$, meaning that some particles with function values x^* at time $t_0 = 0$ move to $x = 0$ at time t^*, that is, they find the minimum of the function. This demonstrates the global convergence of evolutionary algorithms, with a convergence time t^* that approximately equals the evolution generations used by the algorithm, and a convergence speed a that is defined relation to a concrete algorithm.

Analyzing the nonlinear leftward wave in the same way produces a similar result, but the convergence speed is now $a + u$ (no longer a constant), which is like adding wind with velocity u. Considering the case where the density wave propagates leftward, we find that the particle micelles move left as well, meaning that their velocities u are negative, and hence that $\mathrm{e}^{-\frac{u}{a}} > 1$. Thus, the leftward wave ρ_L is a compression wave and the particle density is increasing. This indicates that many particles gather on the left, behavior that guarantees the algorithm's global convergence.

Thinking about this in more detail, the particle density distribution generated through evolution at any given generation can generally be taken as the initial density distribution, so evolution up to that generation can be replaced by initializing the particle distribution differently. Likewise, the particle density wave then propagates in the same ways, namely, in the form of leftward and rightward waves. Consequently, some of the potential particles generated by extensive exploration will be propagated and preserved. In addition, most current evolutionary algorithms use an elite preservation. Hence, these potential particles and good genes must be propagated to the objective function's minimum point with the leftward wave, behavior guarantees the algorithm's global convergence.

However, this spreading or compression process cannot continue infinitely, because the density ρ must be finite, either in a real gas system or the particle system of an evolutionary algorithm. Thus, continuous compression will eventually lead to a discontinuity where, at some time t and point x, ρ_L jumps to a larger value, which is called a shock wave in aerodynamics. After that, the shock wave will continue to propagate to leftward. In this situation, many particles will also gather on the left, guaranteeing the algorithm's global convergence. The condition for producing such a shock wave is that the Mach number $M_a \geqslant 1$, this is called a supersonic flow in aerodynamics, with the opposite case being a subsonic flow.

4.2 Estimating the volume elasticity coefficient and convergence speed

So far, we have not considered two important and closely related parameters, namely, the volume elasticity coefficient κ, and the wave propagation speed a. In aerodynamics theory, these are macroscopically observable values. The volume elasticity coefficient κ indicates the ability of a gas system to generate pressure when it is being compressing and reflects certain internal and natural characteristics of the gas, such as the organization of the gas molecules and their interactions. With regard to our evolutionary algorithm analogy, it represents the relevance of the particles and their interactions, and also the ability of the evolutionary operators to drive the particle system's search process. The wave propagation speed a, which we define as the convergence rate, is an external reflection of the coefficient κ, since they are positively correlated. Whereas, theoretical aerodynamics equations apply to general gas systems, the coefficient κ and speed a, which are measured by aerodynamics experiments, characterize a specific gas system. Similarly, the particle system (or population) of a general evolutionary algorithm is governed by the wave equations established in Section 2, by analogy with a gas system, while the coefficient κ and speed a characterize different algorithms. In the same way, these can only be determined by numerical experiments. Thus, we will now derive formulas for estimating them based on the pressure formula introduced in Section 2 and the conservation law.

Going form generation $t-1$ to generation t, we select K particles to take part in evolutionary operations, with coordinates are $x_i(t-1)$. The particle density at $x_i(t-1)$ is $\rho_i(t-1)$, $i=1,\ldots,K$. These particles are updated via the evolutionary operations, which is the same as saying that they move in the interval $[0,M]$ driven by these operations. If, at generation t, the ith particle moves to $x_i(t)$, then its velocity is $u_i(t)=x_i(t)-x_i(t-1)$ and its momentum is $m_i=u_i$ (assuming its mass is 1). Employing the pressure formula given in Section 2 shows that the local pressure at $x_i(t-1)$ is $p_i=\frac{\kappa(\rho_i-\rho_0)}{\rho_0}$, which drives the particle's motion. The impulse applied to the ith particle by the pressure is $I_i=\frac{\kappa(\rho_i-\rho_0)}{\rho_0}\Delta t$, where for simplicity, we have set $\Delta t=1$. Since κ describes the nature of the particle population, conservation of momentum enables us to estimate its value when evolving from generation $t-1$ to t as

$$\kappa_t=\rho_0\frac{|\sum_{i=1}^{K}u_i^t|}{\sum_{i=1}^{K}|\rho_i^t-\rho_0|}. \tag{32}$$

To avoid any issues with the sign, we take the absolute value of the right hand side in the experiments below.

Having obtained the volume elasticity coefficient, it is now easy to obtain an estimate of the wave propagation speed as

$$a_t=\sqrt{\frac{\kappa_t}{\rho_0}}. \tag{33}$$

This expresses the convergence speed of an evolutionary algorithm in solving a problem, but only for the subsonic flow. In the supersonic case, shock wave propagation increases the convergence rate. By considering the shock wave mechanism and mass conservation [23], we can estimate propagation speed in this case as follows:

$$a_{sw}=\frac{\rho(x_f,t)}{\rho(x_f,t)-\rho(x_r,t-1)}u_r, \tag{34}$$

where $\rho(x_f,t)$ and $\rho(x_r,t-1)$ are the peak particle densities at generations t and $t-1$, respectively, obtained at the points x_f and x_r. While u_r (called the rear wave velocity) is the velocity of the particle micelle at x_r.

Now, we give a simple example to explain the wavelike nature of an evolutionary algorithm's search process and illustrate the roles played by the volume elasticity coefficient κ and wave propagation speed a that occur in wave models. The experimental methods will be described in more detail in the next section, along with further experiments. Consider a classical multimodal function, namely, Rastrigin's function, expressed as $f(x)=\sum_{i=1}^{n}(x_i^2-10\cos(2\pi x_i)+10)$, with a search domain of $-100\leqslant x_i\leqslant 100$ and $n=10$ dimensions. This function has many local minima, and the global minimum is $x^*=(0,0), f(x^*)=0$. We will use the standard PSO algorithm to solve this problem, with the parameters set as in global PSO (GPSO) optimizer [24].

Figure 2 illustrates the particle density distributions over the function value range after several different numbers generations (Figure 3(a)). Here, the particle density distributions generally continue to move leftward as the number of generations increasing, meaning that particles are gathering on the left, due to the leftward (compression) particle density wave. Inevitably, the wave also propagates rightward occasionally, which may be due to stochastic noise or the multimodal problem's landscape: for example, the green dashed line ($t=2002$) has moved to the right of the blue line ($t=2000$). However, the overall trend is for the wave to propagate leftward, causing compression, as the algorithm's selection pressure is always driving the population in search of the optimal solution. Figures 3(b) and (c) show the changes in the volume elasticity coefficient κ and wave propagation speed a, respectively, over time, these characterize the evolution process of the standard PSO algorithm's evolution process when optimizing the function $f(x)$. The volume elasticity coefficient κ denotes evolutionary operators' ability to generate pressure that drives local particle movement, which is then reflected macroscopically in the particle system's ability to search in the optimal direction. Figure 3(b) shows that κ is large in the early stages, because the particles are initially uniformly distributed over the function's range and the particle system is compressible. However, we can also see that its value declines quickly in the later stages, as particles

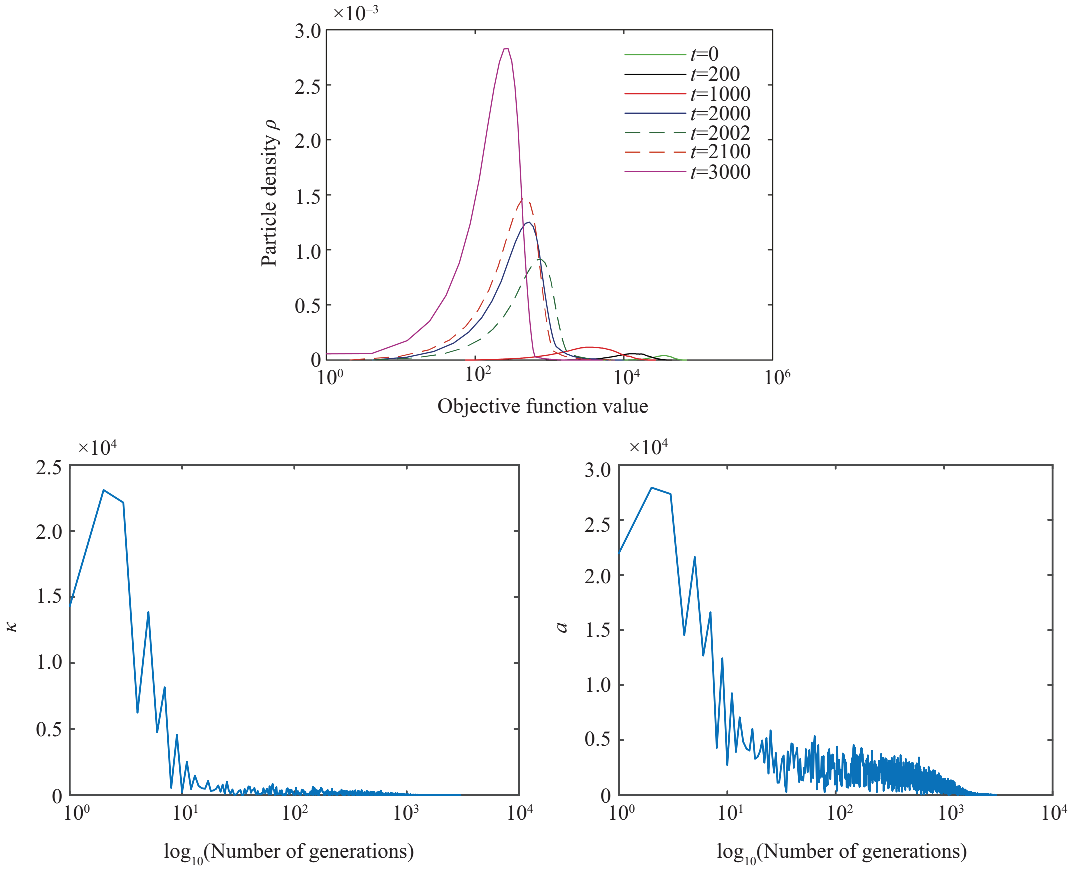

Figure 2 (Color online) Wave propagation for $f(x)$, showing the (a) particle density distribution, (b) volume elasticity coefficient κ, and (c) wave propagation speed a over time.

are gathering on the left and thus increasing the particle density towards the left-hand side of the interval, continually reducing the system's compressibility. Figure 3(c) shows that wave propagation speed a is positively correlated with κ, following a similar trajectory. This simple example shows that wave models can explain the dynamics of the evolution process by estimating the particle density ρ, elasticity coefficient κ, or wave speed a.

5 Experiments

Although we have obtained several important theoretical results about evolutionary algorithms via the analysis discussed above, these are still only theoretical. Although the toy example in Subsection 4.2 has helped to demonstrate the model's feasibility, more experimental results are needed to verify whether wave models can be used to analyze most evolutionary algorithms. In addition, we will exhibit the wave characteristics of different evolutionary algorithms via these experiments, and then propose suitable improvement strategies for different types of algorithms based their experimental characteristics and our previous theoretical analysis.

5.1 Algorithms and test problems

In this subsection, we verify the theoretical results by studying three evolutionary algorithms, namely, the classic genetic algorithm (GA), particle swarm optimization (PSO), and the differential evolution (DE) approach that is currently being widely studied and used. We chose three typical problems, suited to the characteristics of each algorithms. To test the GA, we selected the traveling salesman problem (TSP), a classical NP-hard problem. Meanwhile, for the PSO and DE methods, we selected two numerical

Table 1 Test functions used for PSO and DE

Test function	D	Search range	$f_{\min}$	Name
$f_1(y) = \sum_{i=1}^{D-1}(100(y_i^2 - y_{i+1}^2) + (y_i - 1)^2) + F_4^*$, where $y = M(\frac{2.048z}{100}) + 1$, M is a rotation matrix, $z = x - o$, and $o = [o_1, o_2, \dots, o_n]$ is the shifted global optimum.	30	$[-100, 100]^D$	400	Shifted and Rotated Rosenbrock [25]
$f_2(y) = \sum_{i=1}^{D} \frac{y_i^2}{4000} - \prod_{i=1}^{D} \cos(\frac{y_i}{\sqrt{i}}) + 1 + F_7^*$, where $y = M(\frac{600z}{100})$, M is a rotation matrix, $z = x - o$, and $o = [o_1, o_2, \dots, o_n]$ is the shifted global optimum.	30	$[-100, 100]^D$	700	Shifted and Rotated Griewank [25]

optimization problems, taken from the opening CEC2014 test case [25], namely, two multimodal functions with minima 400 and 700, respectively (described in Table 1).

Briefly, the TSP can be described as follows. Suppose there are n cities, with a given distance matrix D such that D_{ij} denotes the distance between city i and j. The problem is to find the shortest path that visiting every city once and only once before returning to the start, i.e., to minimize the objective function TSP(n). In our experiments, the number of cities n was 50 and 100, and the coordinates of the n cities were randomly generated with the square $[0, 1000] \times [0, 1000]$.

The aim of this experimental study was to demonstrate the wavelike behaviors of three typical evolutionary algorithms and then verify that they agree with the theoretical results. The experiments also illustrate the algorithm's different wavelike characteristics, which reflect their different search mechanisms. This will ultimately yield ideas and methods for further developing the theoretical analysis and directing algorithm design.

Since the main emphasis is on verifying the wave models and associated theory, we use the original forms of the algorithms below. Each algorithm was used to solve the corresponding problem 50 times, and the mean of these 50 solutions was taken as the final solution.

To compute the particle density, the interval between the minimum and maximum of the objective function at a certain generation was first divided into L equal smaller intervals. Then, considering each of these as an elementary volume, we calculated the number of particles and density for each small interval to drive the overall particle density distribution. The reference value, ρ_0 was taken as the mean of the initial density distribution.

5.2 Nonlinear compression and sparse waves

First, we used the classic GA to solve the TSP. Figure 3(a) and (b) show the particle density distributions for TSP(50) and TSP(100), respectively. Here, the horizontal and vertical axis denote the objective function value and the particle density, respectively. Snapshots of the distribution taken at several different evolution generations t are shown using different colors. Both figures (Figure 3) clearly show the particle density distributions narrowing and moving leftward over time (generation t), meaning that they represent leftward-propagating compression waves. Meanwhile, an invisible wave is also propagating rightward, namely, the sparse wave. The interval to the right of the distribution, where the particle density ρ is almost equal to 0, grows continuously as the distribution moves left, so the sparse wave can be seen as a leftward attraction, that draws particles from the right-hand side to the left, in agreement with the behavior of the theoretical solution (30).

Next, we applied elementary PSO to find the minima of the test functions f_1 and f_2 shown in Table 1. Figures 4(a) and (b) show the particle density distributions for f_1 and f_2, respectively, presented similarly to the GA results above. Again, both figures (Figure 4) clearly show leftward compression waves in the particle density, and we can infer the invisible sparse waves from these distribution lines, thus verifying the theoretical results presented in Section 4.

We have now tested two common algorithms by solving two typical problems. Comparing the two sets of figures (Figures 3 and 4) show that the particle density distributions for PSO are smoother than those

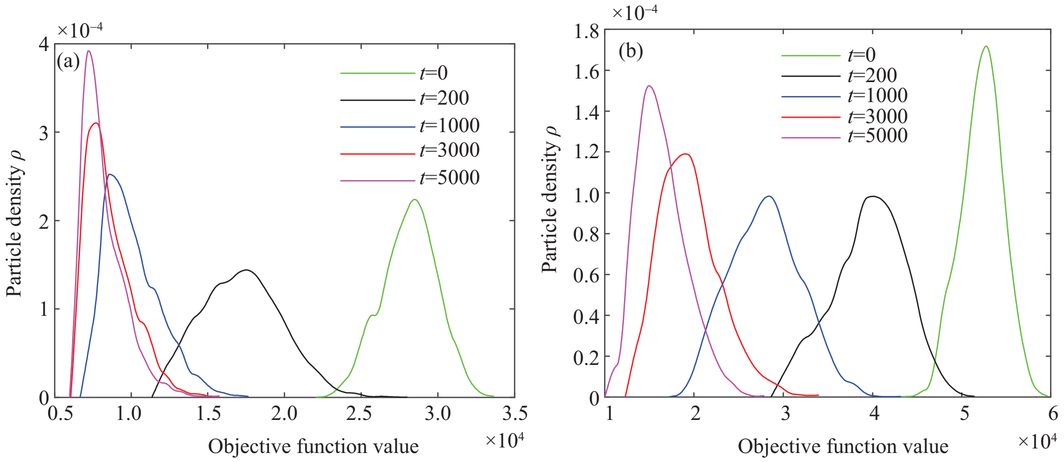

Figure 3 (Color online) (a) TSP ($n = 50$); (b) TSP ($n = 100$).

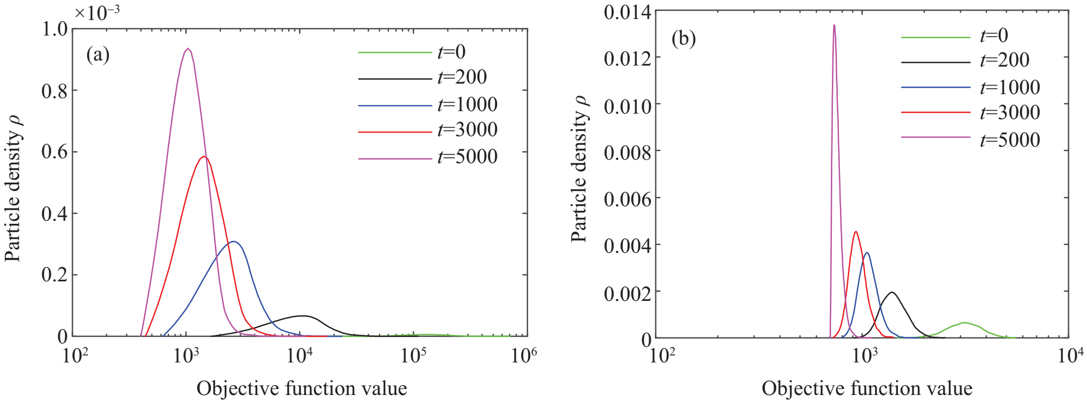

Figure 4 (Color online) PSO applied to (a) f_1 and (b) f_2.

for the GA because of the numerical optimization problems used. However, they still exhibit similar features and tendencies, namely, leftward compression waves search for the minima of the objective functions and sparse rightward waves that help to maintain the particle system's diversity. Overall, these experiments have demonstrated that the evolution processes of the GA and PSO, at least when solving these test problems, can be regarded as subsonic flows with continuous motion [20, 21].

5.3 Nonlinear shock waves and phase transitions

By contrast, DE's evolution process reveals different type of nonlinear wave, i.e., discontinuous motion. We applied elementary DE to find the minima of the functions f_1 and f_2, producing the particle density distributions shown in Figures 5(a) and (b), respectively. Inset into Figure 5(b) is an enlarged view of the particle density distribution over a small interval, presented for clarity. Both figures (Figure 5) clearly show a sudden jump in the particle density at a generation t and function value x, representing a shock wave. Unlike the GA and PSO, DE's evolution process can be seen as a supersonic flow [20, 23]. Afterwards, the peak particle density value continues to propagate left at a speed of a_{sw} until it runs into a barrier, such as the left-hand edge of the interval. This barrier then absorbs or reflects the shock wave, (or both), weakening. Again, the experimental results confirm the predictions of the theoretical analysis, and also demonstrate that shock wave related behavior can emerge when running some evolutionary algorithms. The phenomena of shock wave or discontinuities are called phase transitions in dynamical systems theory. Thus, these wave models have revealed an important aspect of some evolutionary algorithms, both theoretically and experimentally, namely, phase transitions. This will guide our deeper research into evolutionary algorithms.

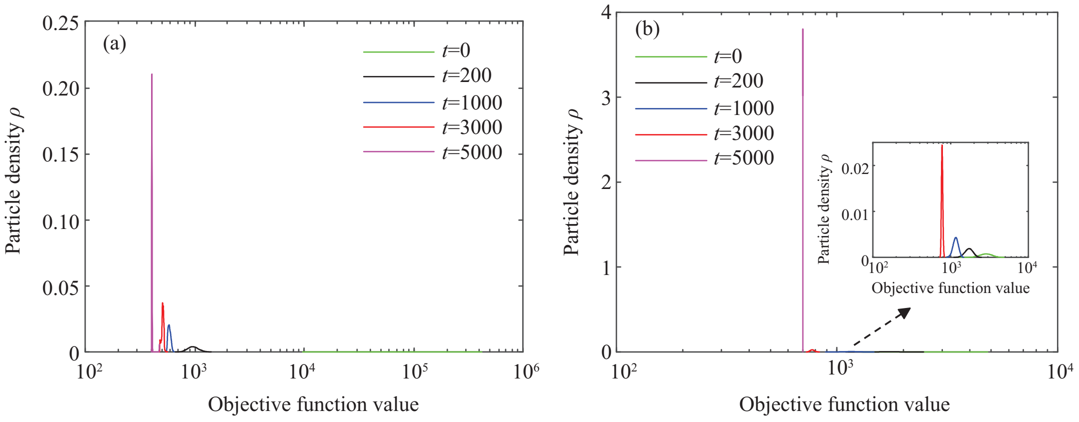

Figure 5 (Color online) DE applied to (a) f_1 and (b) f_2.

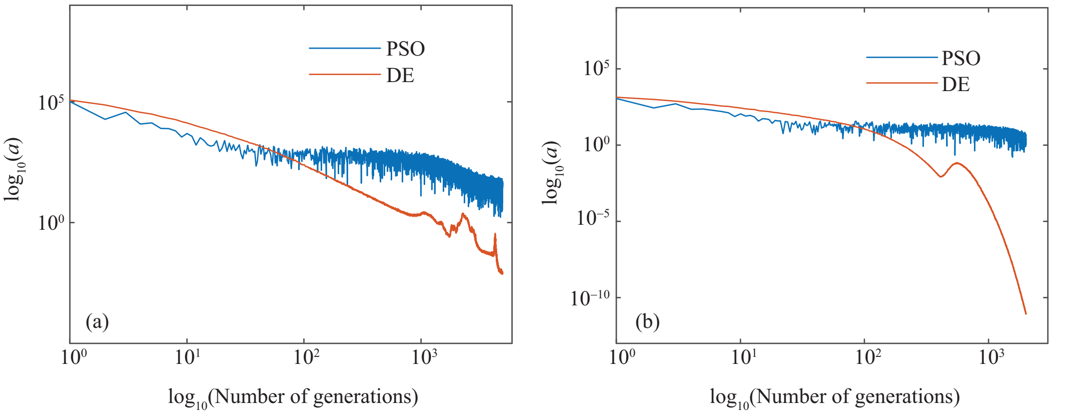

Figure 6 (Color online) PSO vs. DE, applied to (a) f_1 and (b) f_2.

To estimate the convergence speed, we compared the PSO and DE examples discussed above. Figures 6(a) and (b) show their convergence rates a when solving f_1 and f_2, respectively. Here, the horizontal and vertical axis give the number of generations and the convergence speed, respectively, and the red and blue lines represent DE and PSO, respectively. To make the differences clearer, both the horizontal and vertical use logarithmic scales. Comparing the two results using either figure indicates that DE initially converges rapidly, but soon appears to be slower than PSO. However, DE's evolution in fact undergoes shock wave phase during the middle and final periods, where its convergence rate should actually be a a_{sw} (see Figures 7(a) and (b)). By comparing the PSO and DE experiments, we find that DE usually converges faster than PSO, but PSO's convergence rate is nearly stable, meaning that its population diversity is better. This analysis is in general agreement with our knowledge of DE and PSO.

In a real gas system, shock waves propagate faster than the speed of sound. As discussed in Subsection 4.2, we usually have that $a_{sw} > a$. To confirm this, we computed a and a_{sw} when using DE to solve the same problem. Figures 7(a) and (b) show the way they changed over time when solving f_1 and f_2, respectively. Here, the red and blue lines represent a_{sw} and a, respectively. Figures 7(a) and (b) indicate that a_{sw} is always greater than a throughout the evolution process, again confirming the presence of supersonic flows [20, 23]. The shock wave is not present in the early stages, but this phase occurs very soon thereafter. Considering Figures 6(a) and (b), the shock wave appears after about generation 100. These results show that DE's convergence speed is initially a, becoming a_{sw} during the middle and later stages. Overall, the convergence rate slows over time, so the initial rapid convergence could be evidence of local convergence, with the algorithm performing slower local search during the middle and later stages. Here, both the theoretical and experimental results provide clues to improving such algorithms, namely, how to detect an earlier shock wave phase in the evolution process and look for ways of preventing premature

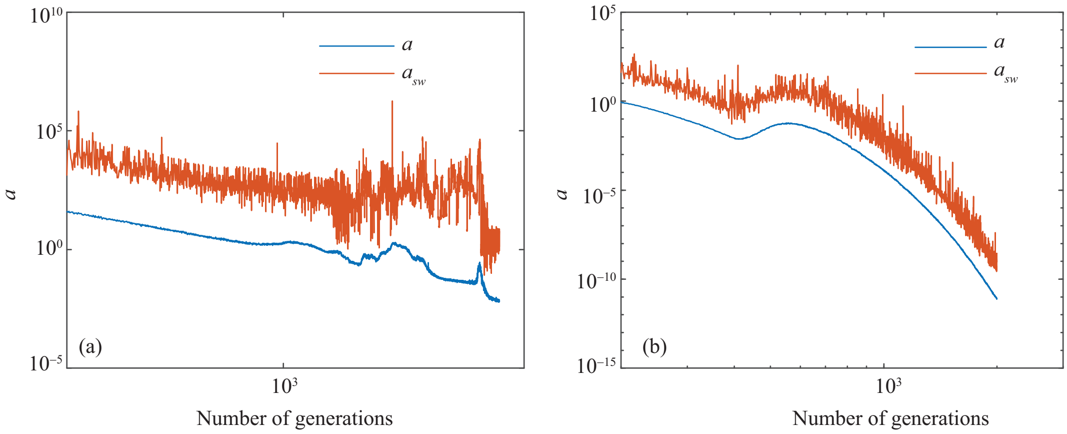

Figure 7 (Color online) a vs. a_{sw}, for (a) f_1 and (b) f_2.

convergence. These ideas will inspire our future research.

As is well known, we must consider the balance between exploration and exploitation in order to design a good evolutionary algorithm. A key criterion with regard to exploration is population diversity. Significantly, there are many metrics for measuring the population diversity of an evolutionary algorithm [26], such as entropy-based [27], probability distribution-based [28], and the distance-based diversity metrics [29]. The particle density from the wave models essentially coincides with the entropy-based metric. As well as the particle density, wave models give us other potentially useful parameters, such as κ and a, mentioned above, and the Mach number M_a. For example, we could use κ or a to measure the selection pressure, enabling us to adjust an algorithm's evolutionary operators and selectors so as to better balance exploration with exploitation in the population. Larger κ values are accompanied by larger a values, and hence higher selection pressure and more rapid diversity loss. With regard to the particle density ρ, we have experimentally confirmed the wave behavior seen in the theoretical solutions derived in Section 3 and analyzed in Section 4. In conjunction, with the Mach number M_a, this could also offer a new criterion for detecting earlier shock waves and thus preventing premature convergence. In summary, combining these theoretical models with experimental analysis will help us to design and improve algorithms.

6 Conclusion

Although many theoretical results have been achieved by building Markov models of evolutionary algorithms based on stochastic process theory, we have chosen to consider a different theoretical approach in the hope of obtaining new ideas that can assist with algorithm design. It is natural to consider an analogy between an evolutionary algorithm's population, or particle system, and a gas system. In this paper, we have therefore built wave models of evolutionary algorithms based on aerodynamics theory. By solving the models' hyperbolic equations, we have found wave solutions by applying the characteristic theory of hyperbolic equations. In addition, we have found significant results by studying the behavior of the particle density distribution waves, namely, the leftward compression wave and the rightward sparse wave. We then demonstrated the convergence of evolutionary algorithms by analyzing the mechanism underlying the leftward wave, and discussed population diversity by analyzing the rightward wave. In addition, we estimated the wave's propagation speeds while running a given algorithm, which could be used to measure their convergence rates in practice. To confirm these theoretical results, we carried out experiments that applied three typical evolutionary algorithms to common benchmark problems, finding agreement between the experimental and theoretical results. Furthermore, these theoretical and experimental analyses will inspire our future research, as they yielded several new clues and ideas, that could assist in the design and improvement of evolutionary algorithms.

First, analyzing both the theoretical and experimental results indicates that both the compression and sparse waves are relevant to the performance of evolutionary algorithms. It is thus worth investigating in more detail how to balance exploration (or diversity) and exploitation (or convergence) by adjusting the two waves so as to control an algorithm's evolution process. This would depend on the Mach number M_a.

Second, in Subsection 4.2 we only provided a rough estimate of the wave (or convergence) speed. This appears to be related to the orders of magnitude of the objective function values, and a more careful and accurate estimate would eliminate this relationship. In addition, the time (number of generations) needed for convergence should be better estimated in order to derive practical stopping criteria for algorithms. In other words, these estimates could offer valuable parameters and practical assistance to guide the evolution process of evolutionary algorithms.

Third, although the shock wave (or phase) transition is captured by both theoretical and experimental analysis, we still do not know when and where the phase transition will emerge in practice. This is an interesting and attractive question, for which both theoretical analysis and experiments will be important. Such research could potentially discover more algorithms with phase transitions, as well as the transitions' features and conditions, thereby slowly revealing the characteristics of this class of algorithms within the framework of dynamical systems theory.

In summary, this study has shown that dynamical systems theory can be used to analyze evolutionary algorithms, but this is just the beginning. Recent discussion has suggested some possible research directions, and we expect that dynamical systems theory will play an important role in the design and analysis of evolutionary algorithms.

Acknowledgements This work was supported by National Natural Science Foundation of China (Grant No. 61672391).

References

1 Back T. Evolutionary Algorithms in Theory and Practice: Evolution Strategies, Evolutionary Programming, Genetic Algorithms. Oxford: Oxford University Press, 1996

2 Wang F, Zhang H, Li K, et al. A hybrid particle swarm optimization algorithm using adaptive learning strategy. Inf Sci, 2018, 436–437: 162–177

3 Guo S M, Yang C C. Enhancing differential evolution utilizing eigenvector-based crossover operator. IEEE Trans Evol Computat, 2015, 19: 31–49

4 Wegener I. Methods for the analysis of evolutionary algorithms on pseudo-boolean functions. In: Evolutionary optimization. Boston: Springer, 2003. 349–369

5 Beyer H G. Convergence analysis of evolutionary algorithms that are based on the paradigm of information geometry. Evolary Computat, 2014, 22: 679–709

6 Derrac J, García S, Hui S, et al. Analyzing convergence performance of evolutionary algorithms: a statistical approach. Inf Sci, 2014, 289: 41–58

7 Tan C J, Neoh S C, Lim C P, et al. Application of an evolutionary algorithm-based ensemble model to job-shop scheduling. J Intell Manuf, 2019, 30: 879–890

8 Wu H, Kuang L, Wang F, et al. A multiobjective box-covering algorithm for fractal modularity on complex networks. Appl Soft Comput, 2017, 61: 294–313

9 Goldberg D E, Segrest P. Finite Markov chain analysis of genetic algorithms. In: Proceedings of the 2nd International Conference on Genetic Algorithms, Cambridge, 1987. 1: 1

10 Rudolph G. Finite Markov chain results in evolutionary computation: a tour d'horizon. Fund Inform, 1998, 35: 67–89

11 He J, Yao X. Drift analysis and average time complexity of evolutionary algorithms. Artif Intell, 2001, 127: 57–85

12 Sudholt D. A new method for lower bounds on the running time of evolutionary algorithms. IEEE Trans Evol Computat, 2013, 17: 418–435

13 Yu Y, Qian C, Zhou Z H. Switch analysis for running time analysis of evolutionary algorithms. IEEE Trans Evol Computat, 2015, 19: 777–792

14 Bian C, Qian C, Tang K. A general approach to running time analysis of multi-objective evolutionary algorithms. In: Proceedings of 27th International Joint Conference on Artificial Intelligence (IJCAI), Stockholm, 2018. 1405–1411

15 Mori N, Yoshida J, Tamaki H, et al. A thermodynamical selection rule for the genetic algorithm. In: Proceedings of IEEE International Conference on Evolutionary Computation, Perth, 1995. 1: 188

16 Cornforth T W, Lipson H. A hybrid evolutionary algorithm for the symbolic modeling of multiple-time-scale dynamical systems. Evol Intel, 2015, 8: 149–164

17 Li Y X, Zou X F, Kang L S, et al. A new dynamical evolutionary algorithm based on statistical mechanics. J Comput Sci Technol, 2003, 18: 361–368

18 Li Y X, Xiang Z L, Xia J N. Dynamical system models and convergence analysis for simulated annealing algorithm (in Chinese). Chin J Comput, 2019, 42: 1161–1173

19 Li Y X, Xiang Z L, Zhang W Y. A relaxation model and time complexity analysis for simulated annealing algorithm (in Chinese). Chin J Comput, 2019. http://kns.cnki.net/kcms/detail/11.1826.TP.20190425.1042.002.html

20 Zhou Y L. One-Dimensional Unsteady Hydrodynamics. Beijing: Science China Press, 1998

21 Lamb H. Hydrodynamics. Cambridge: Cambridge University Press, 1993

22 Gu C H, Li D Q. Mathematical Physics Equations. Beijing: People's Education Press, 1982

23 Zhang Y. Expansion Waves and Shock Waves. Beijing: Peking University Press, 1983

24 Shi Y, Eberhart R C. Empirical study of particle swarm optimization. In: Proceedings of IEEE International Conference on Evolutionary Computation, Washington, 1999. 3: 1945–1950

25 Liang J J, Qu B Y, Suganthan P N. Problem Definitions and Evaluation Criteria for the CEC 2014 Special Session and Competition on Single Objective Real-Parameter Numerical Optimization. Zhengzhou University and Nanyang Technological University, Technical Report. 2013

26 Črepinšek M, Liu S H, Mernik M. Exploration and exploitation in evolutionary algorithms. ACM Comput Surv, 2013, 45: 1–33

27 Liu S H, Mernik M, Bryant B R. To explore or to exploit: an entropy-driven approach for evolutionary algorithms. Int J Knowledge-based Intell Eng Syst, 2009, 13: 185–206

28 Tang K, Yang P, Yao X. Negatively correlated search. IEEE J Sel Areas Commun, 2016, 34: 542–550

29 Ursem R K. Diversity-guided evolutionary algorithms. In: Proceedings of International Conference on Parallel Problem Solving from Nature, Berlin, 2002. 462–471