Available online at www.sciencedirect.com
ScienceDirect
Progress in Natural Science 19 (2009) 1341–1347
Progress in Natural Science
www.elsevier.com/locate/pnsc

A novel quantum-inspired immune clonal algorithm with the evolutionary game approach

Qiuyi Wu *, Licheng Jiao, Yangyang Li, Xiaozheng Deng

Key Laboratory of Intelligent Perception and Image Understanding of Ministry of Education of China, Institute of Intelligent Information Processing, Xidian University, Xi'an 710071, China

Received 30 December 2008; received in revised form 2 February 2009; accepted 23 February 2009

Abstract

The quantum-inspired immune clonal algorithm (QICA) is a rising intelligence algorithm. Based on evolutionary game theory and QICA, a quantum-inspired immune algorithm embedded with evolutionary game (EGQICA) is proposed to solve combination optimization problems. In this paper, we map the quantum antibody's finding the optimal solution to player's pursuing maximum utility by choosing strategies in evolutionary games. Replicator dynamics is used to model the behavior of the quantum antibody and the memory mechanism is also introduced in this work. Experimental results indicate that the proposed approach maintains a good diversity and achieves superior performance.

Keywords: Game theory; Evolutionary game; Replicator dynamics; Quantum-inspired optimization; Artificial immune system

1. Introduction

The quantum-inspired optimization algorithm is a rising intelligence algorithm that merges quantum mechanics and computing intelligence of the classical computer [1]. As a novel optimization technique, the quantum-inspired immune clonal algorithm (QICA), which is based on merging quantum computing and clonal selection theory is introduced in Ref. [2]. QICA has been used extensively in many fields, including function optimization [3], communication system detection [4], pattern recognition [5], etc.

Game theory is a mathematical theory of socio-economic phenomena exhibiting interaction among decision-makers, whose actions affect each other. Smith introduced evolutionary game theory by applying traditional game theory to Biology [6,7]. Though evolutionary game theory originated from biological game, it is applied successfully in many fields besides biology, for example, economics, management science, information science [8–12] and so on. Ficici and Pollack introduced evolutionary game theory in the simple co-evolutionary algorithm [13]. Wiegand et al. used the evolutionary game-theoretic model to help analyze the dynamical behaviors of co-evolutionary algorithms [14]. The GA-evolutionary game is proposed to simulate the behavior of producers operating in the same electricity market by Menniti et al. [15]. In Ref. [16], a hierarchical Nash GA is presented for multi-objective optimization. Liu introduced evolutionary game theory in the particle swarm optimization algorithm [17].

In this paper, we introduce evolutionary game theory in the QICA and present a quantum-inspired immune algorithm embedded with evolutionary game (EGQICA). We use replicator dynamics to simulate the actions of quantum antibodies. The introduction of game strategy increases the diversity of candidates and realizes the information communication among individuals. Moreover, the memory mechanism is also introduced to improve the search

* Corresponding author. Tel.: +86 29 88209786.
E-mail address: diwudiwu@126.com (Q. Wu).

1002-0071/$ - see front matter
doi:10.1016/j.pnsc.2009.02.007

efficiency of the new algorithm. Simulation results proved the algorithm's betterment.

2. Quantum-inspired immune clonal algorithm

The quantum-inspired immune clonal algorithm's operation process is as follows. Each quantum antibody adopts qubit representation. During iteration, each qubit phase of the antibody in the population is compared with that of the current best one, and then modification of its appearing probability is made, which aims at the evolution towards the fitter antibody with a larger probability. Besides, every antibody produced by observation of the quantum antibody represents a possible solution to the optimization task.

First, the definitions of elements for the general optimization problem are described.

A general optimization problem can be formulated as the following model:

$$\text{minimize } f(x) \quad x = (x_1, x_2, \ldots, x_n) \in S \tag{1}$$

where $f(x)$ is an objective function, $S \subseteq R^n$ defines the search space which is an n-dimensional space. That means $S = [\underline{x}, \overline{x}]$, $\underline{x} = (\underline{x_1}, \underline{x_2}, \ldots, \underline{x_n})$, $\overline{x} = (\overline{x_1}, \overline{x_2}, \ldots, \overline{x_n})$.

The affinity for the general optimization problem is defined as follows.

Definition 1. An antibody, a, represents a candidate solution to the optimization problem. The value of its affinity is equal to the negative value of the objective function,

$$a \in S \quad \text{and} \quad D(a) = -f(a) \tag{2}$$

Next lets present the state of the quantum antibody at the tth generation:

$$q^t = \begin{pmatrix} \alpha_1^t & \alpha_2^t & \cdots & \alpha_m^t \\ \beta_1^t & \beta_2^t & \cdots & \beta_m^t \end{pmatrix} \tag{3}$$

where m is the length of the qubit antibody q^t, α_l^t and β_l^t ($l = 1, 2, \ldots, m$) are randomly generated between -1 and 1, satisfying $|\alpha_l^t|^2 + |\beta_l^t|^2 = 1$.

Then the quantum antibodies need to produce common antibodies for evaluating the affinity of them by the observing operator.

According to the affinity function, a qubit antibody q^t in the population will be copied into the C_i same qubit antibodies in the solution space by using the clone operator Θ, which is defined as $\Theta(q^t) = I \cdot q^t$, and I are identity matrices of dimensionality C. Generally, C is given by:

$$C = \left\lceil N_c \times \frac{D(q^t)}{\sum D(q^t)} \right\rceil \tag{4}$$

which can be adjusted self-adaptively by the affinity $D(*)$. N_c is a given value relating to the clone scale, $\sum D(q^t)$ is the total affinity of all the qubit antibodies in a population.

After the clone operator, the proliferative qubit antibodies are:

$$q^{t\prime} = \{q^t, q_1^t, q_2^t, \ldots, q_i^t, q_{c-1}^t\},$$

$$(q_i^t = q^t) = \left\{ \begin{pmatrix} \alpha_1^{t\prime} & \alpha_2^{t\prime} & \cdots & \alpha_m^{t\prime} \\ \beta_1^{t\prime} & \beta_2^{t\prime} & \cdots & \beta_m^{t\prime} \end{pmatrix}, \begin{pmatrix} \alpha_1^{t\prime} & \alpha_2^{t\prime} & \cdots & \alpha_m^{t\prime} \\ \beta_1^{t\prime} & \beta_2^{t\prime} & \cdots & \beta_m^{t\prime} \end{pmatrix} \cdots \begin{pmatrix} \alpha_1^{t\prime} & \alpha_2^{t\prime} & \cdots & \alpha_m^{t\prime} \\ \beta_1^{t\prime} & \beta_2^{t\prime} & \cdots & \beta_m^{t\prime} \end{pmatrix} \right\} \tag{5}$$

And is updated by

$$\begin{bmatrix} \alpha_t^{t\prime\prime} \\ \beta_l^{t\prime\prime} \end{bmatrix} = U(\theta_l^t)^* \begin{bmatrix} \alpha_t^{t\prime} \\ \beta_l^{t\prime} \end{bmatrix} \quad l = 1, 2 \ldots, m \tag{6}$$

The updated operator is

$$U(\theta_l^t) = \begin{bmatrix} \cos(\theta_l^t) & -\sin(\theta_l^t) \\ \sin(\theta_l^t) & \cos(\theta_l^t) \end{bmatrix} \tag{7}$$

where θ_l^t is the rotate step and θ_l^t is defined as $\theta_l^t = k^* f(\alpha_l, \beta_l)$, and the function $f(\alpha_l, \beta_l)$ focuses on guidance by the current best antibody in the population as shown in Ref. [2]. Judge the termination condition, if it is satisfied, output the best solution and end the process, otherwise perform the clonal selection operation and the quantum recombination operation [2].

3. Evolutionary game theory

We begin with the model of Taylor and Jonker [18], which restricts our view to the class of finite games in strategic form. Generally, a normal game consists of three key components: players, strategies space and payoff function, which is defined as:

$$G \triangleq \langle N, (S_i), (P_i) \rangle \tag{8}$$

where $N \triangleq \{1, 2 \ldots, n\}$ is a set of players and n is a positive integer. For each player $i \in N$, $(S_i) \triangleq (S_1, S_2, \ldots, S_n)$ denote a set of allowable actions. The choice of a specific action $s_i \in (S_i)$ of a player i is called pure strategy. The vector $s = (s_1, s_2, \ldots, s_n)$ is called a pure strategies profile. $(P_i) \triangleq (P_1, P_2, \ldots, P_n)$, P_i is the payoff to player $i \in N$, $P_i \triangleq u(i, s_i)$.

Replicator dynamics and evolutionary stable strategies are the key concepts in evolutionary game theory to express the adaptation of each population over time. Suppose strategies space (X_i), consider a discrete time process $t = 1, 2, \ldots$ and the proportion of the individuals in the population who will select action j is x_t^j, where $\sum_{j \in (X_i)} x_j = 1$.

The replicator dynamics equation is

$$\frac{dx_t^j}{dt} \triangleq (u(x_t^j, x) - u(x, x)) \cdot x_t^j \tag{9}$$

where

$$u(x, x) = \sum x_t^j u(s_j, x) \tag{10}$$

$u(x_t^j, x)$ is the payoff of the jth strategy and $u(x, x)$ is the average of payoffs' over all individuals. This equation

implies that the share of the jth strategy grows or shrinks in proportion to the difference between its payoff and the average payoff. Given two selective strategies of individuals denoted by "p" and "n", therefore, $x_t^p + x_t^n = 1$. From Eq. (9), we can conclude:

$$\frac{dx_t^p}{dt} \triangleq (u(x_t^p, x) - u(x, x)) \cdot x_t^p \tag{11}$$

$$\frac{dx_t^n}{dt} \triangleq (u(x_t^n, x) - u(x, x)) \cdot x_t^n \tag{12}$$

From the equations above leads to:

$$\frac{dx_t^p}{dt} = x_t^p(1 - x_t^p)\left[u(x_t^p, x) - u(x_t^n, x)\right] \tag{13}$$

where $u(x_t^p, x)$ and $u(x_t^n, x)$ are the corresponding payoffs of the strategies "p" and "n".

4. Quantum-inspired immune algorithm embedded with evolutionary game

4.1. The principle of EGQICA

The idea of the present work is to modify the QICA algorithm so that each quantum in the population is associated with a strategy taken from the evolutionary game framework that is used to compute its update in the next time step. In this paper, we make maps as indicated in Fig. 1.

Therefore, all quantum antibodies research the best state that can be looked at as players' pursuing the maximum payoff. Consider the bounded rationality of the quantum mechanism, we use replicator dynamics to model the research process of the EGQICA. First of all, the definition of the model is given as follows:

Definition 2. For each quantum antibody $i \in N$ in the population set $N \triangleq \{1, 2, \ldots, n\}$ (n is the size of the population), there are two allowable observing actions $(S_{Oi}) \triangleq (S_{O1}, S_{O2})$, P_i is the affinity of the quantum antibody $i \in N$, $P_i \triangleq u(i, s_i)$.

Definition of S_{O1}: For the binary coding problem, we observe the qubit antibody q^t and produce the binary strings population $p^t = \{x_1^t, x_2^t, \ldots, x_n^t\}$, where $x_i^t (i = 1, 2, \ldots, n)$ is numeric strings of length m derived from the amplitude α_l^t or β_l^t ($l = 1, \ldots, m$). The process is described as follows: (i) generate a random number $p \in [0, 1]$; (ii) if it is larger than $|\alpha_l^t|^2$, the corresponding bit in p^t takes '1', otherwise it takes '0'.

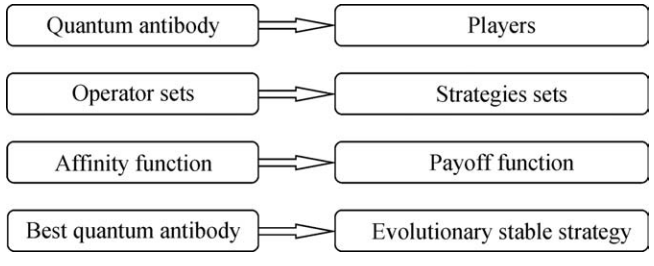

Fig. 1. Mapping relation.

Definition of S_{O2}: For the binary coding problem, we observe the qubit antibody q^t and produce the binary strings population $p^t = \{x_1^t, x_2^t, \ldots, x_n^t\}$, where x_i^t ($i = 1, 2, \ldots, n$) is numeric strings of length m derived from the amplitude α_l^t or β_l^t ($l = 1, \ldots, m$). The process is described as follows: (i) generate a random number $p \in [0, 1]$; (ii) if it is smaller than $|\alpha_l^t|^2$, the corresponding bit in p^t takes '1', otherwise it takes '0'.

From Eq. (13), we can conclude:

$$\frac{dx_t^{S_{O1}}}{dt} = x_t^{S_{O1}}(1 - x_t^{S_{O1}})\left[u(x_t^{S_{O1}}, x) - u(x_t^{S_{O2}}, x)\right] \tag{14}$$

$$\frac{dx_t^{S_{O2}}}{dt} = x_t^{S_{O2}}(1 - x_t^{S_{O2}})\left[u(x_t^{S_{O2}}, x) - u(x_t^{S_{O1}}, x)\right] \tag{15}$$

Therefore, in the search process, if $u(x_t^{S_{O1}}, x) \geq u(x_t^{S_{O2}}, x)$, then the proportion of the antibodies in the population who will select action S_{O1} will be increased. Let the best affinity of antibodies which select S_{O1} be denoted by B_{S1}, the qubit antibody q^t will be updated towards B_{S1} according to Eq. (6), and vice versa.

4.2. Memory cell

In this paper, the algorithm is composed of two populations: the memory cell M and the quantum population Q.

Definition 3. The quantum population $Q \triangleq \{q_1^t, q_2^t, \ldots, q_N^t\}$, N is the size of the quantum population.

Definition 4. The memory cell $M \triangleq \{q_{m1}^t, q_{m2}^t, \ldots, q_{mN_0}^t\}$, N_0 is the size of the memory cell.

4.3. Proposed algorithm

The quantum-inspired immune algorithm embedded with evolutionary game can be summarized in the following steps:

Step 1. *Initialization:* Enact the halting criteria and the parameters, and then generate the quantum population Q and the memory cell M at random. Set evolution generation $t = 0$.

Step 2. *Calculate affinities:* Produce a population of antibody p^t by observing Q and M with strategies S_{O1} and S_{O2}, then calculate the affinity. At generation $t = 0$, $x_t^{S_{O1}} = x_t^{S_{O2}} = 0.5$.

Step 3. *Clone operation:* Clone each quantum antibody in Q, the clone size is as shown in Eq. (4), giving rise to a proliferative group Q'.

Step 4. *Update operation:* Observing Q', and calculate the affinities, then choose the best quantum antibody then update Q' according to Eq. (6).

Step 5. *Game operation:* From Eqs. (14) and (15), update $x_t^{S_{O1}}$ and $x_t^{S_{O2}}$, respectively.

Step 6. *Clonal selection operation:* Note $\widetilde{q_i^t\prime}$ as the best quantum antibody in Q', replace q_i^t by $\widetilde{q_i^t\prime}$ as probability pp_i to form Q'', which is defined as:

$$pp_i = \begin{cases} 1 & \text{affinity}(q_i^t) \le \text{affinity}\left(\widetilde{q_i^{t\prime}}\right) \\ \exp\left(-\frac{\text{affinity}(q_i^t)-\text{affinity}\left(\widetilde{q_i^{t\prime}}\right)}{\alpha}\right) & \text{affinity}(q_i^t) \ge \text{affinity}\left(\widetilde{q_i^{t\prime}}\right) \text{ and } q_i^t \text{ is not the best antibody} \\ 0 & \text{affinity}(q_i^t) \ge \text{affinity}\left(\widetilde{q_i^{t\prime}}\right) \text{ and } q_i^t \text{ is the best antibody} \end{cases}$$

Step 7. *Memory operation:* This operation implements clone, update, and clone selection on memory antibodies in M. The best N_0 antibodies of the population should be saved in M after iteration once.

Step 8. Judge the termination condition, if it is satisfied, output the best solution and end the process, otherwise perform quantum recombination Q''.

4.4. Convergence of the algorithm

Definition 5. Assume that the size of a population is n, and X is a searching space to which all the antibodies belong. Let $X^t = (x_1^t, x_2^t, \ldots, x_n^t)$ in S^n be the population at time t and for $X(t)$, we define:

$$M = \left\{\vec{X} \,\middle|\, f(\vec{X}) = \max\{f(X_i^t), i \le n\}\right\} \tag{16}$$

$$M^* = \left\{\vec{X} \,\middle|\, f(\vec{X}) = \max\{f(X), X \in S^n\}\right\} \tag{17}$$

M is called the satisfied set of population X^t and M^* is defined as the global satisfied set of state S^n.

Definition 6. Let $f_t = \max\{f(x_i): i = 1, 2, \ldots, n\}$, for any initial distribution, if the following equation holds:

$$\lim_{t\to\infty} P\{f_t \subseteq M^*\} = 1 \tag{18}$$

where P stands for the probability, then we call the algorithm convergent with probability 1.

Theorem 1. *The population series of EGQICA $\{Q^t, t \geqslant 0\}$ is a finite homogeneous Markov chain.*

Proof. Like the evolutionary algorithms, the state transfer of EGQICA is processed on the finite space; therefore, the population is finite, since

$$Q^{t+1} = T(Q^t) = T_g \circ \Theta(Q^t) \circ T_u \circ T_s \circ T_m \circ T_r \tag{19}$$

Θ, T_g, T_u, T_s, T_m and T_r indicate the clonal selection operator, the immune genetic operator, the clone operator and directional operator, respectively. Note that the Θ, T_g, T_u, T_s, T_m and T_r operators have no relation with t, thus Q^{t+1} only relates with Q^t. Therefore, $\{Q^t, t \geqslant 0\}$ is the finite homogeneous Markov chain. □

Theorem 2. *The M of Markov chain of EGQICA is monotone increasing, namely, $\forall t \geqslant 0, f(Q^{t+1}) \geqslant f(Q^t)$.*

Proof. Apparently, the individual of EGQICA does not degenerate for our adopting holding best strategy in the algorithm. □

Theorem 3. *The EGQICA is convergent.*

Proof. Let $p_i(k) = P(X_k = s_i)$, where P stands for the probability and $p_k = \sum_{i\notin I} p_i(k)$. Let $|S|$ denote the number of the states in S. ($s_i \in S \quad (i = 1, 2 \ldots, |S|)$ is a certain state, then $s_i = \{x_1, x_2, \ldots, x_n\}$). Let $P_{ij}(k)$ be the probability of transition from X_k^i to X_k^j. Suppose that $I = \{i \mid s_i \cap s^* \neq \varnothing\}$, two especial cases will be discussed first:

1. If $i \in I, j \notin I$, according to the Theorem forenamed, then

$$P_{ij}(k) = 0 \tag{20}$$

2. If $i \notin I, j \in I$, then

$$P_{ij}(k) > 0 \tag{21}$$

Next, we discuss the general case except for the above two cases. According to the character of the Markov chain, we have

$$\begin{aligned} p_{k+1} &= \sum_{s_i\in S}\sum_{j\notin I} p_i(k)p_{ij}(k) \\ &= \sum_{i\in I}\sum_{j\notin I} p_i(k)p_{ij}(k) + \sum_{i\notin I}\sum_{j\notin I} p_i(k)p_{ij}(k) \end{aligned} \tag{22}$$

Since

$$\sum_{i\notin I}\sum_{j\in I} p_i(k)p_{ij}(k) + \sum_{i\notin I}\sum_{j\notin I} p_i(k)p_{ij}(k) = \sum_{i\notin I} p_i(k) = p_k \tag{23}$$

Therefore,

$$\sum_{i\notin I}\sum_{j\notin I} p_i(k)p_{ij}(k) = p_k - \sum_{i\notin I}\sum_{j\in I} p_i(k)p_{ij}(k) \tag{24}$$

Eq. (23) can be obtained according to Eqs. (20) and (22)

$$0 \le p_{k+1} < \sum_{i\in I}\sum_{j\notin I} p_i(k)p_{ij}(k) + p_k = p_k \tag{25}$$

Thus,

$$\lim_{k\to\infty} p_k = 0 \tag{26}$$

Because

$$\lim_{k\to\infty} P\{f_k = f^*\} = 1 - \lim_{k\to\infty}\sum_{i\notin I} p_i(k) = 1 - \lim_{k\to\infty} p_k \tag{27}$$

and according to Eq. (25), we have

$$\lim_{k\to\infty} P\{f_k = f^*\} = 1 \tag{28}$$

This implies that the EGQICA is convergent with the probability 1. □

4.5. Simulation experiments

In order to show the efficiency of EGQICA, two combinatorial optimization test problems are introduced. On the one hand, the satisfiability (SAT) problem is commonly recognized as a fundamental problem in artificial intelligence applications. On the other hand, there is a growing interest in the research of applying evolutionary algorithms to dynamic optimization problems recently [19].

Therefore, we use the SAT problem and the dynamic 0–1 knapsack problems to test the algorithm.

4.5.1. Experiment 1

Formally, the SAT problem can be formulated as follows:

$$\begin{aligned} &f(U) = C_1 \wedge C_2 \wedge \cdots \wedge C_m, U = \{u_1, u_2, \ldots u_n\} \\ &C_i = u_{i1} \vee u_{i2} \vee \cdots u_{ik} \vee \overline{u_{r1}} \vee \overline{u_{r2}} \vee \cdots \overline{u_{rl}} \\ &i = 1, 2, \ldots, m, i1, i2, \ldots, ik = 1, 2, \ldots, n, \\ &r1, r2, \ldots, rl = 1, 2, \ldots, n \end{aligned} \tag{29}$$

Let $U = \{u_1, u_2, \cdots u_n\}$ be a set of n Boolean variables. Corresponding to each variable u are two literals u_j and $\overline{u_j}$ $(j = 1, 2, \cdots n)$. A literal u_j is true and $\overline{u_j}$ is false. C_i $(i = 1, 2, \cdots m)$ of literals is called a clause. The goal of the SAT problem is to determine whether or not there exists an assignment of truth values to variables that makes the formula $f(U)$ true.

In this paper, the SAT problem is mapped to an optimization model:

$$\begin{aligned} &f(U) \quad \underset{\rightarrow}{F} \quad F(U) \\ &F(U) = \sum_{k=1}^{m} C'_k \\ &C'_k = \prod_{k=1}^{n} (1 - x_{ij}) * x_{rj} \end{aligned} \tag{30}$$

where $1 - x_{ij} = u_{ij}, x_{rj} = \overline{u}_{rj}$, $u_{ij}, \overline{u_{rj}}$ are arguments, and x_{ij}, x_{rj} are relevant real variables. When u_{ij} is true $(u_{ij} = 1)$, $x_{ij} = 0$ and vice versa. Therefore, the SAT problems are transformed to the minimum problem of $F(U)$.

In the experiments, "Uniform Random-3-SAT" problems are chosen from the SATLIB library, which are shown in Table 1.

In the experiments, the size of the quantum population is 10, N_C in Eq. (4) is 30, the probability of update is 0.5. The performance of the proposed EGQICA will be compared with the quantum-inspired immune clonal algorithm (QICA) and the simple immune clone selection algorithm (ICSA). In ICSA, the size of the population is 10, N_C is 30, the probability of mutation is $1/n$, where n is the dimension of the variables. The termination criterions are set to be 10^6 generations. Table 2 is the comparison of the success rate (success rate is defined as the times of the success/the total running times) and the mean number of function evaluations between the two algorithms.

Table 1
3-SAT problems in the experiments.

Problem sets	Number of problems	File names	Number of literals/each problem	Number of clauses/each problem
$URSAT_1$	1000	uf20-01, uf20-02, ..., uf20-01000	20	91
$URSAT_2$	1000	uf50-01, uf50-02, ..., uf50-01000	50	218
$URSAT_3$	100	uf75-01, uf75-02, ..., uf75-0100	75	325
$URSAT_4$	1000	uf100-01, uf100-02, ..., uf100-01000	100	430
$URSAT_5$	100	uf125-01, uf125-02, ..., uf125-0100	125	538
$URSAT_6$	100	uf150-01, uf150-02, ..., uf150-0100	150	645
$URSAT_7$	100	uf1750-01, uf175-02, ..., uf175-0100	175	753
$URSAT_8$	100	uf200-01, uf200-02, ..., uf200-0100	200	860
$URSAT_9$	100	uf225-01, uf225-02, ..., uf225-0100	225	960
$URSAT_{10}$	100	uf250-01,uf250-02, ..., uf250-0100	250	1065

4.5.2. Experiment 2

The model of the dynamic knapsack problem we used in our experiments is cited from Ref. [20].

Given a set of m items and a knapsack, the 0–1 knapsack problem can be described as

$$\begin{aligned} &\max \quad p(x) = \sum_{j=1}^{m} p_j x_j \quad j = 1, 2, \ldots, m \\ &\text{subject to} \quad \sum_{j=1}^{m} w_j x_j \le M \end{aligned} \tag{31}$$

where w_j, p_j are the weight and the profit of jth goods. In this paper, $M = 0.5 \times \sum_{j=1}^{m} w_j$, $m = 100$, $w_j, p_j \in [1, 50]$. The mechanism of greedy reparation is used to deal with constraint. That is let $r_j = p_j/w_j$, and rank r according to non-descending order then delete goods which could not satisfy the constraint condition. Construct the dynamic test

Table 2
Comparison among EGQICA, QICA and ICSA.

Problem sets	Success rate			Mean number of function evaluations		
	EGQICA	ICSA	QICA	EGQICA	ICSA	QICA
$URSAT_1$	1	1	1	355	611	380
$URSAT_2$	1	1	1	1652	2464	1860
$URSAT_3$	1	1	1	2450	4950	2950
$URSAT_4$	1	1	1	4790	9120	6823
$URSAT_5$	1	0.96	1	9980	16,200	11,950
$URSAT_6$	1	0.91	1	15,720	23,860	15,821
$URSAT_7$	0.99	0.77	0.98	18,970	36,360	23,371
$URSAT_8$	0.98	0.80	0.95	25,785	45,480	34,110
$URSAT_9$	0.95	0.72	0.93	35,285	54,980	39,302
$URSAT_{10}$	0.96	0.72	0.94	49,360	63,495	55,728

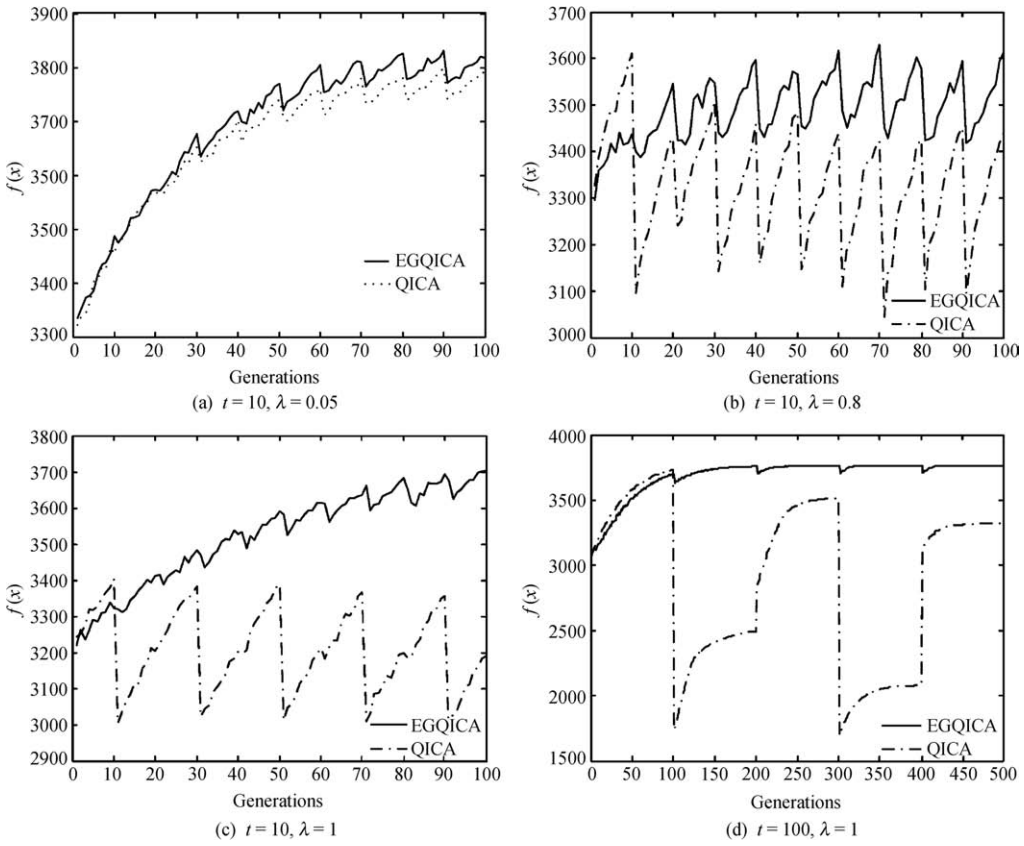

Fig. 2. Convergence performance versus generation with different t and λ.

algorithm as follows: given a general binary static function $f(x)$, create a binary mask Q with the same length of the antibody representation x and perform the operation $x \oplus Q$. Let $f(x)$ be changed at the tth generation, while $f(x)$ at the $(t+1)$th can be formulated as:

$$f(x, t+1) = f(x \oplus Q, t) \tag{32}$$

It can be seen that there are two parameters, one is t, which decides the change period; the other is the proportion of the number of "1" in Q denoted by λ, which controls the degree of environmental change.

In the experiments, the size of the quantum population is 10, N_C in Eq. (4) is 10. Experimental data are the statistical results of 50 times independent running. The performance of the proposed EGQICA will be compared with the quantum-inspired immune clonal algorithm (QICA).

Fig. 2 is a comparison of the performance of convergence achieved by different algorithms. The termination criterion is set to be 100 generations and 500 generations. The change period t is set to be 10 and 100. λ is changed from 0.05 to 1. With the problem changed periodically, the convergence curve dithers, respectively. When problem changes more acutely with increase in λ, EGQICA can still keep well the performance of convergence, as the appropriate strategy can be self-adaptively chosen in EGQICA by the evolutionary game approach. The advantage of EGQICA can be seen clearly in Fig. 2(c) and (d). Table 3 is the comparison of the mean function value and run time between the two algorithms. It can be seen that the proposed algorithm implemented satisfies performance with less mean running time. The simulations suggest that EGQICA still performs quite well in the dynamic optimization problems.

Table 3
Comparison between EGQICA and QICA.

t	λ	Mean running time (s)		Mean function value (standard deviation of function value)	
		EGQICA	QICA	EGQICA	QICA
10	0.05	23.04	25.12	3825.1 (0)	3751.9 (0)
	0.8	35.16	35.63	3610.4 (1×10^{-3})	3490.8 (3×10^{-4})
	1	36.25	38.64	3700 (4×10^{-4})	3405.2 (5×10^{-4})
100	1.0	325.32	322.87	3756.5 (6×10^{-5})	3385 (4×10^{-4})

5. Conclusion

In this paper, we combine game theory with the quantum-inspired immune optimization and use the memory technique to overcome the premature convergence. We present a quantum-inspired immune algorithm based on evolutionary game theory. The algorithm is applied to static and dynamic combinatorial optimization test problems. The good performances of EGQICA on dealing with the

test problems indicate that the proposed embedded game strategy and memory operator maintain good population diversity and less likely to be trapped in local optima. The abstraction of various model of game and application of this EGQICA to quantum intelligent computing deserves our further research.

Acknowledgements

This work was supported by the High-Tech R&D Program of China (No. 2006AA01Z107), the National Natural Science Foundation of China (Nos. 60703108 and 60703107), the Shaanxi Provincial Natural Science Foundation (No. 2007F32) and the China Postdoctoral Science Foundation (No. 20080431228).

References

[1] Luo ZY, Wang P, Li YG, et al. Quantum-inspired evolutionary tuning of SVM parameters. Prog Nat Sci 2008;18(4):475–80.
[2] Li YY, Jiao LC. Quantum-inspired immune clonal algorithm. In: Proceedings of the 4th international conference on artificial immune systems, Banff, Alberta, Canada, August; 2005. p. 304–17.
[3] Jiao LC, Li YY, Gong MG, et al. Quantum-inspired immune clonal algorithm for global optimization. IEEE Trans Syst Man Cybernet B 2008;38(5):1234–53.
[4] Li YY, Jiao LC, Gou SP. Quantum-inspired immune clonal algorithm for multiuser detection in DS-CDMA systems. In: Proceedings of 2006 simulated evolution and learning. Lecture notes in computer science, vol. 4247; 2006. p. 80–7.
[5] Gou SP, Jiao LC, Li YY, et al. Kernel matching pursuit based on immune clonal algorithm for image recognition. In: Proceedings of 2006 simulated evolution and learning. Lecture notes in computer science, vol. 4247, Hefei, China, October 15–18; 2006. p. 26–33.
[6] Smith JM, Price GR. The logic of animal conflict. Nature 1973;146:15–8.
[7] Smith JM. Evolution and the theory of games. New York: Cambridge University Press; 1982.
[8] Owen AL, Videras J. Trust, cooperation, and implementation of sustainability programs: the case of Local Agenda 21. Ecol Econ 2008;68:259–72.
[9] Fon V, Parisi F. Role-reversibility, stochastic ignorance, and social cooperation. J Socio-Econ 2008;37:1061–75.
[10] Breton M, Sokri A, Zaccour G. Incentive equilibrium in an overlapping-generations environmental game. Eur J Oper Res 2008;185:687–99.
[11] Nikiforakis N. Punishment and counter-punishment in public good games: can we really govern ourselves? J Public Econ 2008;92: 91–112.
[12] Chen SH. Computationally intelligent agents in economics and finance. Inform Sci 2007;177:1153–68.
[13] Ficici SG, Pollack JB. A game-theoretic approach to the simple co-evolutionary algorithm. Available from: http://www.portal.acm.org/citation.cfm?id=669093.
[14] Wiegand RP, Liles WC, De Jong KA. Analyzing cooperative co-evolution with evolutionary game theory. In: Proceedings of CEC2002. IEEE, Hilton Hawaiian Village, USA, May 12–17; 2002. p. 1600–5.
[15] Menniti D, Pinnarelli A, Sorrentino N. Simulation of producers behaviour in the electricity market by evolutionary games. Electr Power Syst Res 2008;78:475–83.
[16] Wang JF, Periaux J, Sefrioui M. Parallel evolutionary algorithms for optimization problems in aerospace engineering. J Comput Appl Math 2002;149:155–69.
[17] Liu WB, Wang XJ. An evolutionary game based particle swarm optimization algorithm. J Comput Appl Math 2008;214:30–5.
[18] Taylor PD, Jonker LB. Evolutionarily stable strategies and game dynamics. Math Biosci 1978;40:145–56.
[19] Mori N, Kita H, Nishikawa Y. Adaptation to changing environments by means of the memory based thermodynamical genetic algorithm. In: Proceedings of the 7th international conference on genetic algorithms. Morgan Kaufmann; 1997. p. 299–306.
[20] Yang SX. Non-stationary problem optimization using the primal-dual genetic algorithm. In: Proceedings of the 2003 congress on evolutionary computation, vol. 10, IEEE, Canberra, Australia, December 8–12; 2003, p. 2246–53.