Expert Systems with Applications 36 (2009) 6276–6281

Contents lists available at ScienceDirect

Expert Systems with Applications

journal homepage: www.elsevier.com/locate/eswa

Population declining ant colony optimization algorithm and its applications

Zhilu Wu, Nan Zhao *, Guanghui Ren, Taifan Quan

School of Electronics and Information Technology, Harbin Institute of Technology 342#, Harbin, Heilongjiang 150001, China

ARTICLE INFO

Keywords:
Artificial intelligence
Ant colony optimization
Traveling salesman problem
CDMA
Multiuser detection

ABSTRACT

Population declining ant colony optimization (PDACO) algorithm is proposed and applied to the traveling salesman problem (TSP) and multiuser detection in this paper. Ant colony optimization (ACO) algorithms have already successfully been used in combinatorial optimization, however, as the pheromone accumulates, we may not get a global optimum because it stops searching early. PDACO can enlarge searching range through increasing the initial population of the ant colony, and the population declines in successive iterations. So, the performance of PDACO is superior with the same computational complexity. PDACO is applied to TSP and multiuser detection. Via computer simulations it is shown that PDACO has better performance in solving these two problems than ACO algorithms.

1. Introduction

Swarm intelligence is a relatively new approach to problem solving that takes inspiration from the social behaviors of insects and of other animals (Kennedy, Eberhart, & Shi, 2001). In particular, ants have inspired a number of methods and techniques among which the most studied and the most successful one is the general purpose optimization technique known as ant colony optimization (ACO). ACO algorithms take inspiration from the foraging behavior of some ant species (Goss, Aron, Deneubourg, & Pasteels, 1989). These ants deposit pheromone on the ground in order to mark some favorable path that should be followed by other members of the colony. ACO exploits a similar mechanism for solving optimization problems.

The first ACO algorithm, ant system (AS), is proposed as a means of solving the traveling salesman problem (TSP) (Dorigo, Maniezzo, & Colornim, 1996). AS has gained a great success in solving combinatorial optimization problems, however, the performance of it is still worse than some other metaheuristic algorithms (Laguna & Glover, 1993; Zhen-Ping & Bavarian, 1992). So, many other ACO algorithms are proposed inspired by AS, the performance of which is improved remarkably (Dorigo, Birattari, & Stützle, 2006). The main ACO algorithms presented in the literatures are: ant-Q (Dorigo & Gambardella, 1996), ant colony system (ACS) (Dorigo & Gambardella, 1997), MAX–MIN ant system (MMAS) (Stützle & Hoos, 2000), rank-based ant system (Bullnheimer, Hartl, & Strauss, 1999), ANTS (Maniezzo, 1999), hyper-cube ant system (Blum, Roli, & Dorigo, 2001), and KCC-Ants (Naimi & Taherinejad, 2009). The development of bio-inspired methodologies based on ant colony inspired algorithm systems is an emergent research area with applications in areas such as robotics (Lerman, Galstyan, Matinolli, & Ijspeert, 2002), quadratic assignment problems (Colorni, Dorigo, & Maniezzo, 1991), TSP (Li & Gong, 2003), and feature subset selection (Sivagaminathan & Ramakrishnan, 2007).

The most successful ACO algorithms are MMAS and ACS. Though MMAS and ACS can overcome the drawbacks of AS and achieve much better performance, the population of the ant colony in them does not change during the iterations. So it is a waste of calculation in the late stage of the algorithms because the population is still large while the searching range becomes smaller. In this paper, population declining ant colony optimization (PDACO) algorithm is proposed, which can improve the global searching capability without increasing the computational complexity. In PDACO, the population of the ant colony is much larger in the beginning, which can enlarge searching range, and as the iterations carried on, the population declines, which can decrease the computational complexity. So the performance of PDACO is much better than the corresponding ACO algorithm with the same computational complexity. For PDACO will not change the working function of ACO, it can be combined with many ACO algorithms to improve the performance of them. So, population declining MMAS (PDMMAS) and population declining ACS (PDACS) are proposed by combining PDACO with MMAS and ACS, respectively, and then applied to TSP. Through simulations it is shown that PDMMAS and PDACS have better performance in solving TSP than their corresponding ACO algorithms.

Code division multiple access (CDMA) has been the subject of extensive research in the field of mobile radio communications. This technique permits a large number of users to communicate simultaneously on the same frequency band; however, it also

* Corresponding author. Tel./fax: +86 0451 86418284.
E-mail addresses: wuzhilu@hit.edu.cn (Z. Wu), zhaonan@hit.edu.cn (N. Zhao), renguanghui@hit.edu.cn (G. Ren).

0957-4174/$ - see front matter © 2008 Elsevier Ltd. All rights reserved.
doi:10.1016/j.eswa.2008.07.013

creates multiple access interference (MAI). The MAI makes the conventional detector (CD), which can demodulate only one spread-spectrum signal without considering other signals, unreliable and insensitive to near-far effect in a multiuser environment. For this reason multiuser detection, which can overcome this problem, is a hot topic now for CDMA systems (Moshavi, 1996). The optimal multiuser detector (OMD) (Verdu, 1986) proposed by Verdu, is shown to be near-far resistant and has the optimal performance, however, the exponential complexity in the number of users makes it impractical to use in current CDMA systems. Therefore, research efforts have been concentrated on the development of suboptimal detectors, which exhibit good near-far effect resistant properties, have low computational complexity and achieve relatively high performance, such as MMSE detector (Xie, Short, & Rushforth, 1990), Hopfield neural network detector (Kechriotis & Manolakos, 1996), and stochastic cellular neural network detector (Wu, Zhao, Zhao, & Ren, 2007).

ACO can also be used in multiuser detection as a kind of suboptimal detectors, in which the length of the tour in TSP is related to the objective function of the OMD (Hijazi & Natarajan, 2004). In this paper, PDACO multiuser detector is proposed by applying PDACO to multiuser detection. Via simulations, it is shown that the PDACO multiuser detector has a much better performance in reducing the near-far effect than the ACO multiuser detector, as well as a superior performance in bit-error rate (BER).

The remainder of this paper is organized in four sections. In Section 2, some preliminaries about ACO are reviewed. In Section 3, PDACO is introduced, and its key characteristics are reported. In Section 4, PDACO is combined with ACS and MMAS, and applied to TSP. Simulation results are also presented. In Section 5, PDACO is applied to multiuser detection. The performance of the PDACO multiuser detector is compared with the ACO multiuser detector and some other detectors.

2. Ant colony optimization

Many ACO algorithms have been proposed. Here we present the original AS, and the two most successful variants: MMAS and ACS. In order to illustrate the differences between these three algorithms, we use the TSP as a concrete example.

2.1. Ant system

AS is the first proposed ACO algorithm. Its main characteristic is that, after each iteration, the pheromone values are updated by all the M ants that have built solutions. The pheromone τ_{ij}, associated with the edge joining cities i and j, is updated as follows:

$$\tau_{ij} \leftarrow (1-\rho)\cdot\tau_{ij} + \sum\nolimits_{m=1}^{M}\Delta\tau_{ij}^{m}, \tag{1}$$

where ρ is the evaporation rate, M is the number of ants, and $\Delta\tau_{ij}^{m}$ is the quantity of pheromone laid on edge (i,j) by ant m

$$\Delta\tau_{ij}^{m} = \begin{cases} Q/L_m & \text{if ant } m \text{ used edge}(i,j) \text{ in its tour,} \\ 0 & \text{otherwise,} \end{cases} \tag{2}$$

where Q is a constant, and L_m is the length of the tour constructed by ant m.

In the construction of a solution, ants select the following city to be visited through a stochastic mechanism. When ant m is in city i and has so far constructed the partial solution s^{p}, the probability of going to city j is given by

$$p_{ij}^{m} = \begin{cases} \dfrac{[\tau_{ij}]^{\alpha}[\eta_{ij}]^{\beta}}{\sum\limits_{c_{il}\in \mathrm{N}(s^{\mathrm{p}})}[\tau_{il}]^{\alpha}[\eta_{il}]^{\beta}} & \text{if } c_{ij}\in \mathrm{N}(s^{\mathrm{p}}), \\ 0 & \text{otherwise,} \end{cases} \tag{3}$$

where $N(s^{\mathrm{p}})$ is the set of feasible components; that is, edges (i,l) where l is a city not yet visited by the ant m. The parameters α and β control the relative importance of the pheromone versus the heuristic information η_{ij}, which is given by

$$\eta_{ij} = \frac{1}{d_{ij}}, \tag{4}$$

where d_{ij} is the distance between cities i and j.

AS has gain a great success in solving TSP, however, as the scale of TSP increases the performance of AS decreases seriously compared with other metaheuristic algorithms. So, most of the research on ACO has been focused on the methods to improve AS. The most successful ones are ACS and MMAS.

2.2. MAX–MIN ant system

MMAS is an improvement on the original AS. Its characterizing elements are that only the best ant updates the pheromone trails and that the value of the pheromone is bounded. The pheromone update is implemented as follows:

$$\tau_{ij} \leftarrow \left[(1-\rho)\cdot\tau_{ij} + \Delta\tau_{ij}^{\mathrm{best}}\right]_{\tau_{\min}}^{\tau_{\max}}, \tag{5}$$

where $\tau_{\max}$ and $\tau_{\min}$ are, respectively the upper and lower bounds imposed on the pheromone; the operator $[x]_b^a$ is defined as

$$[x]_b^a = \begin{cases} a & \text{if } x > a, \\ b & \text{if } x < b, \\ x & \text{otherwise,} \end{cases} \tag{6}$$

and $\Delta\tau_{ij}^{\mathrm{best}}$ is

$$\Delta\tau_{ij}^{\mathrm{best}} = \begin{cases} 1/L_{\mathrm{best}} & \text{if } (i,j) \text{ belongs to the best tour,} \\ 0 & \text{otherwise,} \end{cases} \tag{7}$$

where L_{best} is the length of the tour of the best ant. This may be either the best tour found in the current iteration (iteration-best, L_{ib}) or the best solution found since the start of the algorithm (best-so-far, L_{bs}) or a combination of both.

2.3. Ant colony system

The most interesting contributions of ACS are the introduction of a local pheromone update in addition to the pheromone update performed at the end of the construction process (called offline pheromone update).

The local pheromone update is performed by all the ants after each construction step. Each ant applies it only to the last edge traversed:

$$\tau_{ij} \leftarrow (1-\varphi)\cdot\tau_{ij} + \varphi\tau_0, \tag{8}$$

where $\varphi \in (0,1]$ is the pheromone decay coefficient, and τ_0 is the initial value of the pheromone.

The main goal of the local update is to diversify the search performed by subsequent ants during one iteration: by decreasing the pheromone concentration on the traversed edges, ants encourage subsequent ants to choose other edges and, hence, to produce different solutions. This makes it less likely that several ants produce identical solutions during one iteration.

The offline pheromone update, similarly to MMAS, is applied at the end of each iteration by only one ant, which can be either the iteration-best or the best-so-far. However, the update formula is slightly different

$$\tau_{ij} = \begin{cases} (1-\rho)\cdot\tau_{ij} + \rho\cdot\Delta\tau_{ij} & \text{if } (i,j) \text{ belongs to the best tour,} \\ \tau_{ij} & \text{otherwise,} \end{cases} \tag{9}$$

as in MMAS, $\Delta\tau_{ij} = 1/L_{\mathrm{best}}$, where L_{best} can be either L_{ib} or L_{bs}.

Another important difference between ACS and AS is in the decision rule used by the ants during the construction process. In ACS, the so-called pseudorandom proportional rule is used: the probability for an ant to move from city i to city j depends on a random variable q uniformly distributed over [0,1], and a parameter q_0; if $q \leqslant q_0$, then $j = \arg\max_{c_{il} \in N(s^p)}\{\tau_{il}\eta_{il}^{\beta}\}$, otherwise Eq. (3) is used.

3. Population declining ant colony optimization

ACO algorithms have already successfully been used in combinatorial optimization; however, as the pheromone accumulates, we may not get a global optimum because it stops searching early and the population of the ant colony in them does not change during the iterations. So it is a waste of calculation in the late stage of the algorithms because the population is still large while the searching range becomes smaller. So, PDACO algorithm is proposed, which can improve the global searching capability without increasing the computational complexity.

In PDACO, the population of the ant colony is much larger in the beginning, which can enlarge the searching range, and as the iterations carried on, the population declines, which can decrease the computational complexity. So the performance of PDACO is much better than the corresponding ACO algorithm with the same computational complexity.

Assuming the population of the ant colony in the initial stage is M and the number of iterations is N_c. PDACO can be discussed in two cases as follows.

(1) $N_c < M$.

The PDACO in this case is usually applied to the problems with high real-time requirement, such as multiuser detection. Assuming the population of the ant colony decreases by ΔM each iteration, so the population of the ant colony M_n after the nth iteration turns to

$$M_n = M - n \times \Delta M$$

(2) $N_c > M$.

The PDACO in this case is usually applied to the problems which are difficult to solve and without the real-time requirement, such as TSP. Assuming the population of the ant colony decreases by only one every n iterations, so the population of the ant colony M_n after the nth iteration turns to

$$M_n = M - \left[\frac{n}{\Delta n}\right].$$

So, the population of the ant colony decreases in successive iterations following Eq. (10) or Eq. (11), and ΔM or Δn in them can be set different values according to the initial population, the number of iterations and the kind of the problems to be solved.

For PDACO only controls the population of the ant colony and will not change the working function of ACO, it can be combined with many ACO algorithms to improve the performance of them.

4. PDACO algorithm for TSP

Since PDACO can be combined with many ACO algorithms to improve the performance of them, PDMMAS and PDACS are proposed by combing PDACO with MMAS and ACS, respectively, and applied to TSP.

4.1. PDMMAS and its application to TSP

PDMMAS is proposed by combining PDACO with MMAS, and it follows the principles of MMAS described in Section 2.2 except that the population of the ant colony is declining during the successive iterations as in Eq. (11).

To compare PDMMAS with MMAS, they are applied to two TSP problems, respectively (http://www.iwr.uni-heidelberg.de/groups/comopt/software/TSPLIB95/). The first is Eil51, and there are 51 cities in it with the best solution 425. The MMAS runs for 1000 iterations using 10 ants while in PDMMAS several cases with different initial population, which is increased by 5 from 15 to 50, are considered, and the number of iterations in each case of PDMMAS can be calculated with the same computational complexity as MMAS. Through simulations, the average length of the tour in Eil51 versus the population of MMAS and the initial population in each case of PDMMAS curves are depicted in Fig. 1.

In Fig. 1, it is shown that the average lengths of Eil51 solved by PDMMAS in all cases with the different initial population are all shorter than that solved by MMAS.

The second TSP problem is St70, and there are 70 cities in it with the best solution 675. The MMAS runs for 1500 iterations using 10 ants while in PDMMAS several cases with different initial population, which is increased by 5 from 15 to 70, are considered, and the number of iterations in each case of PDMMAS can be calculated with the same computational complexity as MMAS. Through simulations, the average length of the tour in St70 versus the population of MMAS and the initial population in each case of PDMMAS curves are depicted in Fig. 2.

In Fig. 2, it is shown that the average lengths of St70 solved by PDMMAS in all cases with the different initial population are all shorter than that solved by MMAS.

Through solving these two typical TSP problems, Eil51 and St70, using MMAS and PDMMAS, respectively, we can see that the performance of PDMMAS in solving combinatorial optimization problems is superior to that of MMAS.

4.2. PDACS and its application to TSP

PDACS is proposed by combining PDACO with ACS, and it follows the principles of ACS described in Section 2.3 except that the population of the ant colony is declining during the successive iterations as in Eq. (11).

To compare these two algorithms, ACS and several cases of PDACS with different initial population are applied to the Eil51 and St70 TSP problems, respectively. The parameters are the same as those described in 4.1. Through simulations, the average lengths of the tour in Eil51 and St70 versus the population of ACS and the initial population in each case of PDACS curves are depicted in Figs. 3 and 4.

In Figs. 3 and 4, it is shown that the average lengths of Eil51 or St70 solved by PDACS in all cases with the different initial population are all shorter than that solved by MMAS. So, we can see that the performance of PDACS in solving combinatorial optimization problems is superior to that of ACS.

For PDACO only controls the population of the ant colony and will not change the working function of ACO, it can be combined with many ACO algorithms to improve the performance of them. In Section 4.1 and Section 4.2, PDMMAS and PDACS are proposed by combining PDACO with MMAS and ACS, respectively, and then

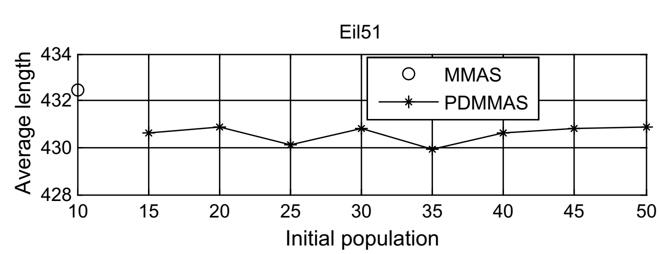

Fig. 1. Average length vs. initial population of PDMMAS in Eil51.

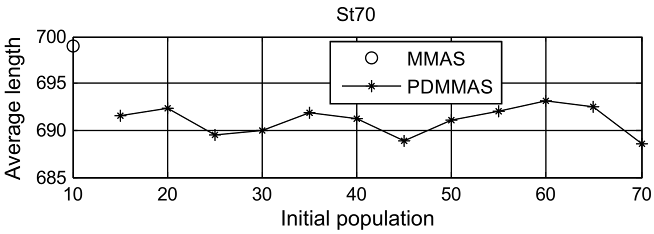

Fig. 2. Average length vs. initial population of PDMMAS in St70.

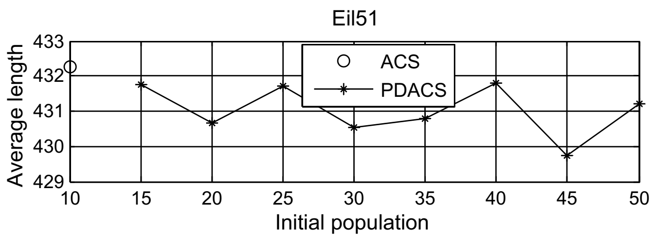

Fig. 3. Average length vs. initial population of PDACS in Eil51.

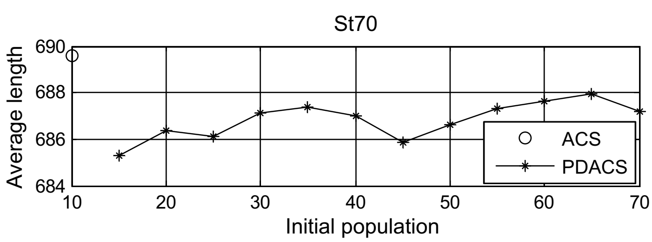

Fig. 4. Average length vs. initial population of PDACS in St70.

they are used in solving TSP. Simulation results show that through increasing the initial population of the ant colony the searching range is enlarged while through declining the population in successive iterations the computational complexity is reduced. So the performance of ACO algorithms improved by PDACO in solving combinatorial optimization problems is superior to that of the corresponding algorithm with the same computational complexity.

5. PDACO multiuser detector

5.1. Multiuser detection

CDMA has been the subject of extensive research in the field of mobile radio communications. This technique permits a large number of users to communicate simultaneously on the same frequency band. However, this creates MAI, which makes the CD of demodulating a spread-spectrum signal in a multiuser environment unreliable and insensitive to near-far effect. For this reason multiuser detection, which can overcome this problem, is a hot topic now for CDMA systems.

5.1.1. Conventional detector

Assuming there are K users of a CDMA system in a synchronous single-path channel, the received signal can be expressed as

$$r(t) = \sum_{k=1}^{K} A_k(t) g_k(t) d_k(t) + n(t), \tag{13}$$

where $A_k(t)$, $g_k(t)$, and $d_k(t)$ are the amplitude, signature code waveform, and information of the kth user, respectively. $n(t)$ is additive white Gaussian noise (AWGN), with a two-sided power spectral density of $N_0/2$ W/Hz.

The CD is composed of a bank of K matched filters, and can be shown in Fig. 5.

In Fig. 5, the existence of MAI has a significant impact on the capacity and performance of the CD system because the CD follows a single-user detector strategy. As the number of interfering users increases, the amount of MAI increases.

5.1.2. Optimal multiuser detector

Verdu has shown that the OMD may be achieved by producing an estimate for the information vector transmitted based on the maximization of the logarithm of the likelihood function. The objective function of the OMD is given as

$$b_{\text{opt}} = \arg\max\{2Y^{\mathrm{T}}Ab - b^{\mathrm{T}}Hb\}, \tag{14}$$

where $b \in \{+1, -1\}$, $Y^{\mathrm{T}} = (y_1, \ldots, y_k)$ is the row vector consisting of the sampled outputs of the matched filters, A is the diagonal matrix consisting of the corresponding received amplitudes, and $H = A^{\mathrm{T}}RA$, in which R is a $K \times K$ uniform correlation matrix.

Despite the huge performance and capacity gains over the CD, the OMD is not practical. The exponential complexity in the number of users makes the cost of this detector too high. Consequently, research efforts have been concentrated on the development of suboptimal multiuser detectors that exhibit good near-far resistance, reasonable implementation complexity and comparable BER performance to that of the OMD.

5.2. ACO multiuser detector

As the speciality of multiuser detection in the CDMA system, the ACO algorithms should be adjusted if we want to apply them to the multiuser detection. The adjustments are as follows.

(1) For the K users in the systems are independent, without losing generality, we can let each ant travel in the fixed order from the 1st user to the Kth user. In this case, the ants should not decide whether the user has been traveled.

(2) As there is not any heuristic information, we can discard parameters η_{ij}, α and β. Because the transmitted information by each user can only be +1 or -1, the probability of what value the kth user transmitted decided by the ant m at the time t is given by:

$$p_{kj}^{m}(t) = \frac{\tau_{kj}(t)}{\sum_{s\in(-1,+1)} \tau_{ks}(t)}, k = 1, 2, \ldots, K, j = +1 \text{ or } -1. \tag{15}$$

(3) The decision of which is the best solution is based on the values of the objective function in Eq. (14), and the solution that has the largest value is the best.

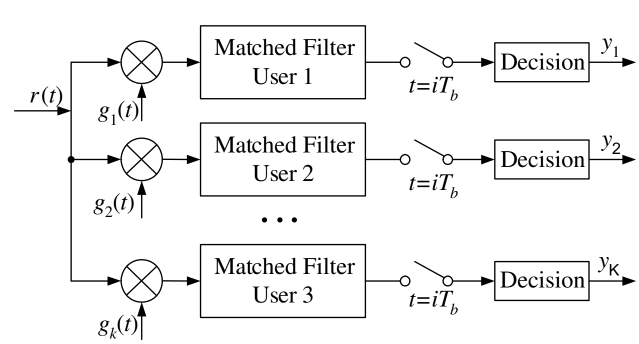

Fig. 5. The conventional detector.

(4) Because the datum in multiuser detection should be processed in real time, only the best ant in the current iteration deposits the pheromone after each iteration, but the pheromone on all the paths still evaporates.
(5) Through the rules set above, multiuser detection can be described as a path-choosing problem which can be solved by ACO algorithms.

5.3. PDACO multiuser detector

ACO algorithms have been applied to multiuser detection successfully; however, the performance of the ACO multiuser detector still can be improved. PDACO has been proved to have better performance than ACO in Section 4.1 and Section 4.2, so PDACO can be used in this field and PDACO multiuser detector is proposed.

PDACO is applied to multiuser detection following the rules described in Section 5.2, and the PDACO multiuser detector can be got. It is represented by the following steps:

Step 1: Initialize of parameters, including the number of iterations N_c, initial population of the ant colony M, the decrement of the population after each iteration ΔM, the evaporation rate ρ, and the initial value of the pheromone τ_0.
Step 2: Set the outputs of the matched filters in Fig. 5 as the initial global best solution.
Step 3: Calculate the population of the ant colony before the nth iteration M_{n-1}. M_{n-1} ants travel from the 1st user to the Kth user following Eq. (15), and then we can get M_{n-1} solutions.
Step 4: Compare the M_{n-1} solutions based in Eq. (14), and set the solution that has the largest value (equal to C_n) of Eq. (14) as the local best solution in this iteration.
Step 5: Update the pheromone as follows:

$$\tau_{kj}(t+1) = (1-\rho)\cdot\tau_{kj}(t) + \Delta\tau_{kj}, \quad k = 1,2,\ldots,K, \\ j = +1 \text{ or } -1, \tag{16}$$

$$\Delta\tau_{kj} = \begin{cases} C_n/r & \text{if } (k,j) \text{ belongs to the local best solution}, \\ 0 & \text{otherwise}, \end{cases} \tag{17}$$

where r is a constant.
Step 6: Compare the local best solution with the global best solution. If the local best solution is better than the global best solution, set the local best solution as the global best solution.
Step 7: Output the global best solution if stopping criterion is satisfied, or return to Step 3.

5.4. Experimental results

In order to evaluate the performance of the PDACO multiuser detector, a CDMA system using it is designed as Fig. 6.

In Fig. 6, the number of users $K = 10$ in the CDMA system and the length of PN sequences used is 15. The received signal $r(t)$ is handled in the matched filter bank, the outputs of which are fed into the PDACO multiuser detector, and then we can get the estimate of the baseband information transmitted of each user.

Based on the system described in Fig. 6, the performance of the PDACO multiuser detector, ACO multiuser detector, CD, and OMD is compared. In the PDACO multiuser detector, considering the high real-time requirement, we set the initial population of ant colony $M_{\text{PDACO}} = 4$ K, the decrement of the population after each iteration $\Delta M = 4$, and the number of iterations $N_{c\text{-PDACO}} = K = 10$.

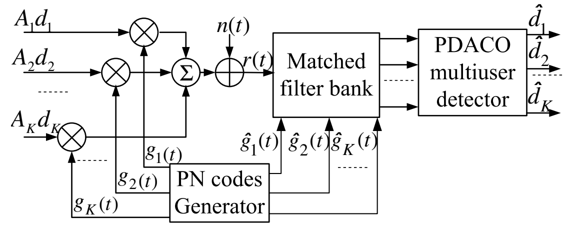

Fig. 6. A CDMA system uing PDACO multiuser detector.

Table 1
Computational complexity comparison

Multiuser detector	Calculation number
PDACO multiuser detector	$2\,K(K+1)$
ACO multiuser detector	$2\,K(K+1)$
OMD	2^K

To ensure that the computational complexity is the same as that of PDACO multiuser detector, in ACO multiuser detector the population of ant colony is set as $M_{\text{ACO}} = K = 10$, and the number of iterations $N_{c\text{-ACO}} = 2\,K + 2 = 22$. The total number of calculating the value of the objective function in Eq. (14) in PDACO multiuser detector, ACO multiuser detector and OMD is listed in Table 1.

As shown in Table 1, the computational complexity of PDACO multiuser and ACO multiuser detector is the same, and is much less than that of OMD when K is large.

The BER versus signal-noise ratio (SNR) curves with equal energy of each user, are depicted in Fig. 7.

It is shown in Fig. 7 that if the near-far effect is not considered, the BER of the PDACO multiuser detector is much lower than that of the CD and the ACO multiuser detector, and is close to the BER of OMD.

We also deal with the near-far effect problem. The BER curves of the first user are compared when its energy is increasing with the energy of the other users unchanged, and the results are shown in Fig. 8.

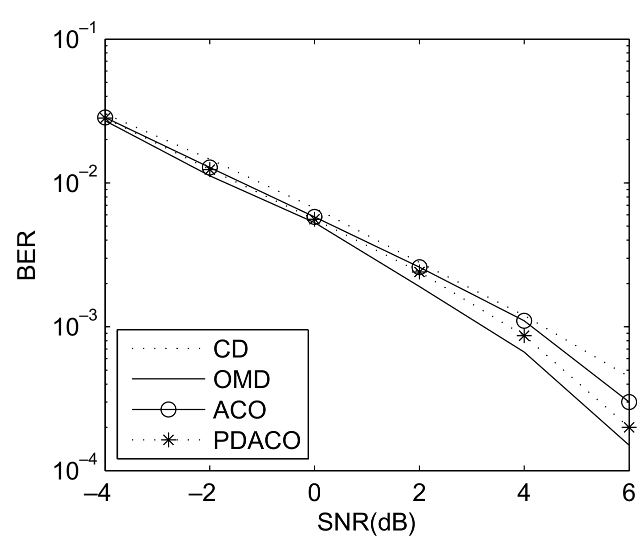

Fig. 7. Bit-error rate vs. signal-noise ratio.

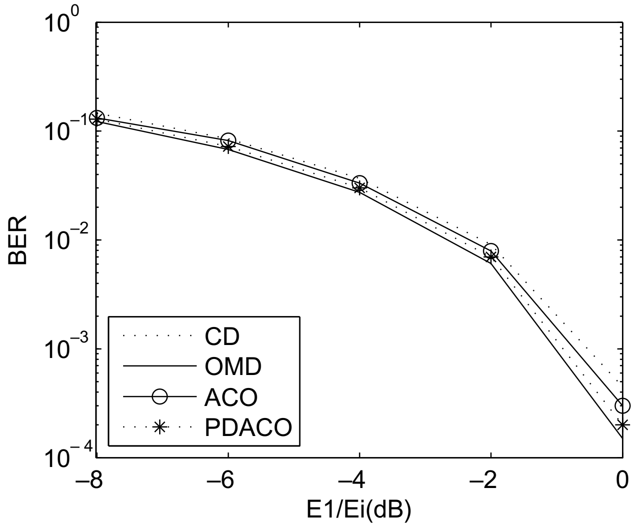

Fig. 8. Bit-error rate vs. near-far ratio.

From the simulation results, we can see that near-far effect resistant performance of the PDACO multiuser detector is much better than the CD and the ACO multiuser detector, and is close to OMD.

Therefore, the overall performance of the PDACO multiuser detector is much better than the ACO multiuser detectors and it is more suitable as a suboptimal multiuser detection scheme in CDMA systems.

6. Conclusion

In this paper we have proposed a PDACO algorithm, which enlarge searching range through increasing the initial population of the ant colony, and the population declines in successive iterations. So, the performance PDACO is superior to that of ACO with the same computational complexity. The applications of PDACO are discussed and it is used in TSP and multiuser detection. Since PDACO can be combined with many ACO algorithms to improve the performance of them, PDMMAS and PDACS are proposed by combining PDACO with MMAS and ACS, respectively, and applied to TSP. Simulating results show that PDMMAS and PDACS have much better performance than the corresponding ACO algorithms with the same computational complexity. PDACO multiuser detector is proposed and computer simulations show that the performance of both the BER and near-far resistant is superior to that of ACO multiuser detector.

References

Blum, C., Roli, A., & Dorigo, M. (2001). HC-ACO: The hyper-cube framework for ant colony optimization. In *Proceedings of MIC'2001-metaheuristics international conference* (pp. 399–403). Porto.

Bullnheimer, B., Hartl, R. F., & Strauss, C. (1999). A new rank-based version of the ant system: A computational study. *Central European Journal for Operations Research and Economics, 7*(1), 25–38.

Colorni, A., Dorigo, M., & Maniezzo, V. (1991). Distributed optimization by ant colonies. In *Proceedings of the European conference on artificial life* (pp. 134–142). Paris, France.

Dorigo, M., Birattari, M., & Stützle, T. (2006). Ant colony optimization. *IEEE Computational Intelligence Magazine, 1*(4), 28–39.

Dorigo, M., & Gambardella, L. M. (1996). A study of some properties of Ant-Q. In *Fourth international conference on parallel problem solving from nature* (pp. 656–665). Berlin.

Dorigo, M., & Gambardella, L. M. (1997). Ant colony system: A cooperative learning approach to the traveling salesman problem. *IEEE Transactions on Evolutionary Computation, 1*(1), 53–66.

Dorigo, M., Maniezzo, V., & Colornim, A. (1996). Ant system: Optimization by a colony of cooperating agents. *IEEE Transactions on Systems, Man, and Cybernetics, 26*(1), 29–41.

Goss, S., Aron, S., Deneubourg, J. L., & Pasteels, J. M. (1989). Self-organized shortcuts in the Argentine ant. *Naturwissenschaften, 76*(12), 579–581.

Hijazi, S. L., & Natarajan, B. (2004). Novel low-complexity DS-CDMA multiuser detector based on ant colony optimization. In *60th Vehicular technology conference* (pp. 1939–1943).

http://www.iwr.uni-heidelberg.de/groups/comopt/software/TSPLIB95/.

Kechriotis, G. I., & Manolakos, E. S. (1996). Hopfield neural network implementation of the optimal CDMA multiuser detector. *IEEE Transactions on Neural Networks, 7*(1), 131–141.

Kennedy, J. F., Eberhart, R. C., & Shi, Y. (2001). *Swarm intelligence*. San Francisco, CA, USA: Morgan Kaufman.

Laguna, M., & Glover, F. (1993). Integrating target analysis and tabu search for improved scheduling systems. *Expert Systems with Applications, 6*(3), 287–297.

Lerman, K., Galstyan, A., Matinolli, M., & Ijspeert, A. (2002). A macroscopic analytical model of collaboration in distributed robotic systems. *Artificial Life, 7*(4), 375–393.

Li, Y., & Gong, S. H. (2003). Dynamic ant colony optimization for TSP. *International Journal of Advanced Manufacturing Technology, 22*(7–8), 528–533.

Maniezzo, V. (1999). Exact and approximate nondeterministic tree-search procedures for the quadratic assignment problem. *INFORMS Journal on Computing, 11*(4), 358–369.

Moshavi, S. (1996). Multi-user detection for DS-CDMA communications. *IEEE Communications Magazine, 34*(10), 124–136.

Naimi, H. M., & Taherinejad, N. (2009). New robust and efficient ant colony algorithms: Using new interpretation of local updating process. *Expert Systems with Applications, 36*(1), 481–488.

Sivagaminathan, R. K., & Ramakrishnan, S. (2007). A hybrid approach for feature subset selection using neural networks and ant colony optimization. *Expert Systems with Applications, 33*(1), 49–60.

Stützle, T., & Hoos, H. H. (2000). MAX–MIN ant system. *Future Generation Computer Systems, 16*(8), 889–914.

Verdu, S. (1986). Minimum probability of error for asynchronous Gaussian multiple-access channels. *IEEE Transactions on Information Theory, 32*(1), 85–96.

Wu, Z. L., Zhao, N., Zhao, Y. Q., & Ren, G. H. (2007). Stochastic cellular neural network for CDMA multiuser detection. In *Fourth International Symposium on Neural Networks* (pp. 651–656). Nanjing, China.

Xie, Z., Short, R. T., & Rushforth, C. K. (1990). A family of suboptimum detectors for coherent multiuser communications. *Journal on Selected Areas in Communications, 8*(4), 683–690.

Zhen-Ping, L., & Bavarian, B. (1992). Optimization of job scheduling on parallel machines by simulated annealing algorithms. *Expert Systems with Applications, 4*(3), 323–328.