Memetic Comp. (2009) 1:191–203
DOI 10.1007/s12293-009-0013-z

REGULAR RESEARCH PAPER

Selecting survivors in genetic algorithm using tabu search strategies

Chuan-Kang Ting · Cheng-Feng Ko · Chih-Hui Huang

Received: 16 April 2009 / Accepted: 17 August 2009 / Published online: 1 September 2009

Abstract Genetic algorithm (GA) is well-known for its effectiveness in global search and optimization. To balance selection pressure and population diversity is an important issue of designing GA. This paper proposes a novel hybridization of GA and tabu search (TS) to address this issue. The proposed method embeds the key elements of TS—tabu restriction and aspiration criterion—into the survival selection operator of GA. More specifically, the tabu restriction is used to prevent inbreeding for diversity maintenance, and the aspiration criterion is activated to provide moderate selection pressure under the tabu restriction. The interaction of tabu restriction and aspiration criterion enables survivor selection to balance selection pressure and population diversity. The experimental results on numerical and combinatorial optimization problems show that this hybridization can significantly improve GAs in terms of solution quality as well as convergence speed. An empirical analysis further identifies the influences of the TS strategies on the performance of this hybrid GA.

Keywords Genetic algorithm · Tabu search · Hybridization · Survival selection

C.-K. Ting (✉) · C.-F. Ko · C.-H. Huang
Department of Computer Science and Information Engineering, National Chung Cheng University, Chia-Yi 621, Taiwan
e-mail: ckting@cs.ccu.edu.tw

C.-F. Ko
e-mail: kcf95m@cs.ccu.edu.tw

C.-H. Huang
e-mail: hch96p@cs.ccu.edu.tw

1 Introduction

Genetic algorithm (GA) has dealt successfully with a variety of search and optimization problems. The basic idea of GA is mimicking the process of natural evolution, consisting of selection, reproduction, and mutation, to manipulate candidate solutions [16]. In addition, GA implements Darwinian evolution theory in the selection operation. Based on the principle 'survival of the fittest', GA is expected to evolve candidate solutions toward the optima.

Selection pressure and population diversity play a key role in the performance of GA [31]. The former leads GA to dig into the promising regions, while the latter drives the variation of candidate solutions during evolution. These two factors, nevertheless, form a trade-off. Emphasis on selection pressure causes a fast improvement on candidate solutions but risks premature convergence due to the hastened loss of population diversity. Maintenance of population diversity, on the other hand, helps to explore the search space and yet retards the convergence. Several approaches have been proposed to handle the trade-off between selection pressure and population diversity [29]. Some of them are based on the hybrid of GA and another heuristic algorithm for enhanced performance. A paradigm of this hybridization is to run the two algorithms by turns; the results from one algorithm will be passed to another, and vice versa.

This study presents a novel hybridization of GA and tabu search (TS) [10] to address the issue of balancing selection pressure and population diversity. Instead of running the two algorithms alternately, the tabu list and aspiration criterion of TS are embedded into the survivor selection of GA. The tabu restriction is used to prevent inbreeding for diversity maintenance, and the aspiration criterion is activated to provide moderate selection pressure under the tabu restriction. Consequently, selection pressure and population diversity are controlled by the two TS components in the survivor selection process of GA. A series of experiments on numerical and combinatorial optimization problems is conducted to evaluate the performance of the proposed method.

The rest of this paper is organized as follows. Section 2 reviews related work on hybrids of GA and TS. Section 3 describes the proposed method in detail and Sect. 4 presents the experimental results for performance evaluation. Finally, conclusions are drawn in Sect. 5.

2 Related work

Several hybrids of GA and TS have been proposed to enhance the search ability of algorithms. Glover et al. [9] inceptively designed the scatter search, which integrates GA and TS for a better performance than running GA and TS alone. Handa and Kuga [13] considered the difference between the convergence speeds of GA and TS in the two halves of the search process. They proposed the concatenation of GA and TS that switches these two algorithms to avoid premature convergence. Abdinnour-Helm [1] solved an incapacitated hub location problem using a hybrid of GA and TS, where GA is used to determine the number and the location of hubs, and TS is adopted to assign each spoke to the closest hub. Nara [20] combined GA with simulated annealing (SA) and TS to deal with the generator maintenance scheduling problem. The proposed method employs SA to improve the convergence speed of GA and adopts TS to strengthen neighborhood search. Furthermore, this hybrid method uses the tabu list to avoid mating of chromosomes that have low hamming distance in order to keep diversity. Ozdamar and Birbil [23] introduced a nonrestrictive TS/SA to GA to enhance the convergence speed. In addition, they claimed that high-quality initial chromosomes can affect the performance of the GA and proposed a restrictive TS/SA (RTSSA) to address this issue. The experimental results demonstrated that using RTSSA into GA provides feasible solutions to improve convergence ability. Chin et al. [5] proposed increasing the search intensity of TS when the population of GA converges. Based on this idea, they devised a hybrid approach that applies TS on the individual solutions in the later generations of GA.

Moreover, Shin et al. [27] proposed the genetic-tabu algorithm, wherein the best solution obtained from the GA population is checked with the tabu list to adjust the mutation probability. Liaw [19] integrated a local improvement procedure based on TS into GA for the open shop scheduling problem. This integration enables genetic search over the subspace of local optima. Vilcot and Billaut [30] compared two GAs that initialize the population at random and using TS, respectively. The experimental results show that the latter can lead to improvement in solution quality and computation time. Some researchers applied the combination of GA and TS to resolve real-world problems. Gandomkar et al. [8] adopted GA to optimize dispersed generation allocation and used TS to avoid the local optima and premature convergence of GA. Their experimental results show that this hybrid approach can greatly reduce distribution power loss and obtain better solution accuracy and convergence speed. Hageman et al. [12] designed a hybrid of GA and TS to solve the multilayer optical coatings optimization problem. This approach uses TS to enhance the solutions generated by GA. Xian et al. [32] incorporated GA with TS for the principal component analysis of three-dimensional molecular similarity problem. They carried out GA to align two molecules based on the evaluation results of molecular electrostatic potentials and employed TS to decrease the probability of falling into local optima. The two algorithms are performed alternately to enhance candidate solutions. Jiang et al. [18] hybridized GA and TS to solve the protein folding problem based on a hydro-phobic-hydrophilic lattice model, where TS is also used to improve the solutions obtained from GA.

In general, most of the above methods run GA and TS alternately, wherein TS serves as an enhancement in local search for GA. Restated, the best solution of the GA population is performed with TS to search its neighborhood. The result of TS is then returned to GA as a new member of the population. In such hybridization, the original structures of GA and TS are not altered. Rather than run GA and TS alternately, Ting et al. [28,29] proposed a new hybridization method, called tabu genetic algorithm (TGA), in which TS are employed as the strategy for selecting parents in GA. The experimental results show that TGA can lead to significantly better solution quality than GA and the method of running GA and TS alternately.

However, TGA suffers from the issue of spending much time in finding a valid parent in the select-and-check process. This issue becomes particularly serious as the population diversity is low. To address this issue, this paper devises a new scheme for the hybridization of GA and TS by embedding the TS strategies into the survivor selection, instead of parent selection, for algorithmic compactness and performance efficiency. The new method not only overcomes the time-consuming issue of TGA but it also preserves the advantage of balance between selection pressure and population diversity.

3 Tabu genetic algorithm 2 (TGA2)

This study proposes the *tabu genetic algorithm 2* (TGA2), which embeds the strategies of TS into the survivor selection of GA. TGA2 relies on GA for the adaptation and robustness of genetic operators and integrates with the memory structure and search strategy of TS.

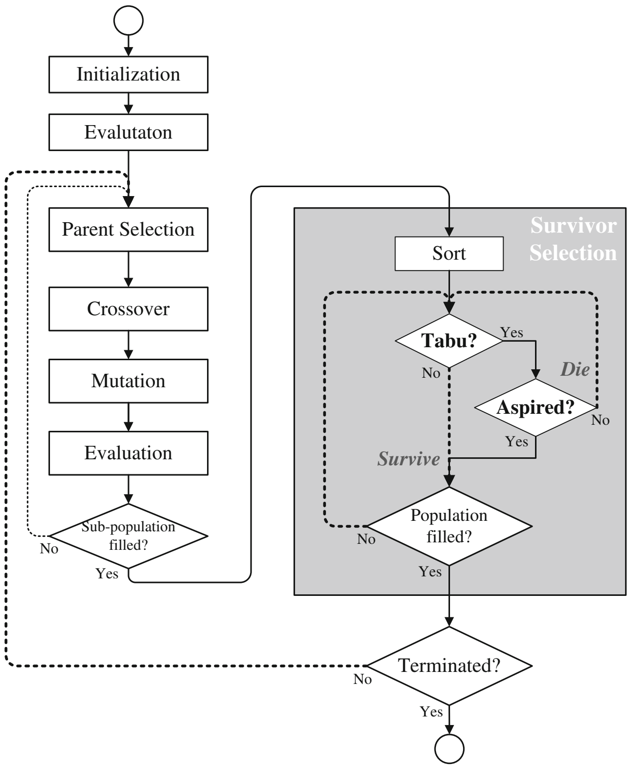

Fig. 1 Flowchart of TGA2

Figure 1 presents the flowchart of TGA2. The framework of TGA2 principally follows that of GA, except the survivor selection embedded with the TS strategies. In the survivor selection, TGA2 disallows the offspring reproduced from tabu mating to survive, unless they can satisfy the aspiration criterion. To adapt these TS strategies to GA, several modifications are required. The modified elements of GA in TGA2 are elaborated below.

3.1 Representation

The chromosome representation in TGA2 consists of the chromosome structure of GA and, especially, an additional memory structure based on TS to guide the search. Formally, a chromosome is represented as a three tuple $(\mathbf{g}, \phi, \boldsymbol{\tau})$, where $\mathbf{g} = (g_1, \ldots, g_l)$ includes genes, ϕ is the clan number, and $\boldsymbol{\tau} = (\tau_1, \ldots, \tau_T)$ is the tabu list.

The first part, $\mathbf{g}$, originated from GA, is concerned with the information about candidate solutions. Therefore, the representation is problem-dependent. This study uses binary-coded and order-based representations for numerical and combinatorial optimization problems, respectively.

The second part is a memory structure comprised of clan number ϕ and tabu list $\boldsymbol{\tau}$. The clan number acts as a signature for chromosome identification. Each chromosome in the initial population of TGA2 is assigned a unique clan number.

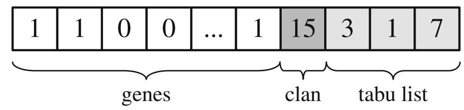

Fig. 2 Example of representation in binary-coded TGA2

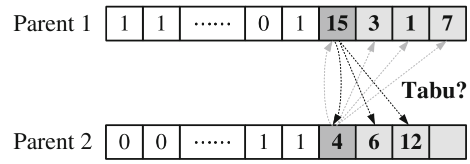

Fig. 3 Check of mating validity in TGA2

Similar to the family name in human society, offspring in TGA2 inherit the clan number from one of their parents. Additionally, the tabu list records the clan numbers that are forbidden to mate with. The functionality of tabu list will be introduced in the next section.

Figure 2 illustrates the chromosome structure in binary-coded TGA2. The first part, consisting of genes in GA, encodes the information about the candidate solution. The second part is composed of clan number 15 and tabu list {3, 1, 7}, which indicates that the mating of this chromosome with a chromosome of clan 3, 1, or 7 will yield tabu offspring.

3.2 Tabu restriction

The proposed TGA2 utilizes the tabu restriction of TS to prevent inbreeding. Inbreeding represents the mating (breeding) of close relatives and is known to be harmful to the health and fertility of individuals in nature. In evolutionary algorithms, several studies have proven that inbreeding causes a rapid loss of population diversity and raises the risk of premature convergence. To prevent this, TGA2 uses the clan number to identify chromosomes and the tabu list to record mating history. Whenever TGA2 selects a pair of parents for reproduction, it checks the validity of this mating. As Fig. 3 illustrates, if either parent finds its clan number exists in the clan number or the tabu list of its partner, this mating is invalid and their offspring are labeled with "tabu". That is to say, the mating of related chromosomes will yield tabu offspring according to the comparison result of clan and mating history. The following function determines whether the mating of two chromosomes $\mathbf{c}_1 = (\mathbf{g}_1, \phi_1, \boldsymbol{\tau}_1)$ and $\mathbf{c}_2 = (\mathbf{g}_2, \phi_2, \boldsymbol{\tau}_2)$ yields tabu offspring:

$$\text{Tabu}(\mathbf{c}_1, \mathbf{c}_2) = \begin{cases} \mathit{true}, & (\phi_1 = \phi_2) \text{ or} \\ & (\exists k : \phi_1 = \tau_{2,k} \text{ or } \phi_2 = \tau_{1,k}) \\ \mathit{false}, & \text{otherwise} \end{cases} \quad (1)$$

where $\tau_{i,k}$ denotes the k_{th} element of tabu list $\boldsymbol{\tau}_i$.

A chromosome updates its tabu list when it undertakes crossover. The crossover of parents in TGA2 includes two

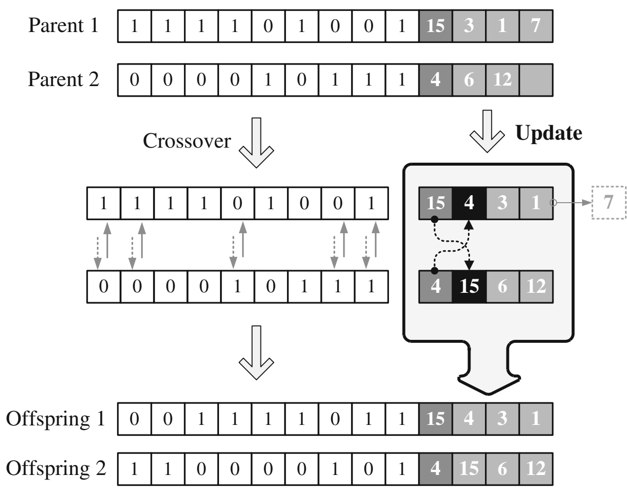

Fig. 4 Crossover and updating of tabu list in TGA2

SurvivorSection(P_t, P'_t)

$P^s \leftarrow$ Sort$(P_t \cup P'_t)$
$i \leftarrow 0$
$P_{t+1} \leftarrow \emptyset$
while (P_{t+1} is not filled)
  repeat
    $i \leftarrow i + 1$
  until ($\mathbf{c}_{(i)}$ is not tabu) or ($\mathbf{c}_{(i)}$ is aspired)
  $P_{t+1} \leftarrow P_{t+1} \cup \{\mathbf{c}_{(i)}\}$
return P_{t+1}

Fig. 5 Pseudocode of survivor selection of TGA2

parts. As Fig. 4 shows, first, the crossover is performed on genes to exchange information about the solution variables. This part is analogous with the crossover in GA. Second, both parents add the clan number of their partner to the tabu list, which works as a queue—first in, first out. The clan numbers that are removed out of the tabu list regain the validity to mate with. Each offspring inherits the clan number and tabu list from one of its parents.

3.3 Aspiration criterion

In addition to tabu restriction, TGA2 takes the aspiration criterion into account in the survivor selection. The tabu restriction helps to maintain population diversity; however, it may weaken the exploitation of promising solutions. TGA2 enables the aspiration criterion of TS to provide a chance to release from the tabu restriction so as to moderately reinforce exploitation.

Aspiration criterion is defined to allow superior offspring to override the tabu restriction. In TGA2, a tabu offspring $\mathbf{c}'$ is said to be *aspired* if its fitness is better than the best-so-far fitness f^*. For a minimization problem, the aspiration criterion is defined by

$$\text{Aspired}(\mathbf{c}') = \begin{cases} \mathit{true}, & f(\mathbf{c}') < f^* \\ \mathit{false}, & \text{otherwise} \end{cases} \tag{2}$$

The aspired offspring will survive while the tabu offspring will be discarded in the survivor selection. The interaction between tabu restriction and aspiration criterion is expected to balance the diversity maintenance and selection pressure of TGA2.

3.4 Survivor selection

The survivor selection is the key of TGA2 in hybridization of GA and TS. Traditionally, the survivor selection of GA considers only the chromosome fitness. In nature, the survivability of an individual is, however, concerned with many factors in addition to fitness. For example, inbreeding depression states that inbreeding, i.e., breeding of related individuals, will cause a decrease in fitness. To avoid that, humans generally conduct eugenics to avoid mating of parties related by blood (consanguinity) or marriage (affinity) or both.

TGA2 implements this idea through the components and strategies of TS. The clan number and tabu list enable TGA2 to identify chromosomes and their relatives. The tabu restriction, furthermore, forbids mating of related chromosomes. Analogous with the eugenics used in human society, the tabu offspring will be eliminated unless they are good enough to defeat the best chromosome so far, i.e., to satisfy the aspiration criterion. The survivor selection in TGA2 thus considers three kinds of chromosomes as survivors for the next generation: (1) parents in the current population, (2) offspring that are generated from valid mating, and (3) aspired offspring.

Suppose we have parental population P_t and offspring population P'_t, both having m chromosomes, at generation t. Figure 5 gives the procedure of $(\mu + \lambda)$ survivor selection of TGA2, which uses a temporary set $P^s = \{\mathbf{c}_{(1)}, \ldots, \mathbf{c}_{(2m)}\}$ of sorted chromosomes ($\mathbf{c}_{(1)}$: best, $\mathbf{c}_{(2m)}$: worst) and returns population P_{t+1} for the next generation. Notably, the population P_{t+1} will always be filled by qualified solutions since only offspring can be classified as tabu. In the extreme case that all λ offspring in P'_t are tabu, the μ parents in P_t will then survive into P_{t+1} through the proposed survivor selection.

4 Experimental results

This study conducts a series of experiments on numerical as well as combinatorial optimization problems. To evaluate the proposed TGA2, we compare its performance with GA and TGA in terms of solution quality and convergence speed.

Table 1 Test numerical optimization problems

Function	Range of x_i	N	Bits of x_i	l	Opt.
$f_{\mathrm{F2}}(x) = \sum_{i=1}^{N-1}\left[100\left(x_{i+1} - x_i^2\right)^2 + (x_i - 1)^2\right]$	$[-2.048, 2.047]$	2	12	24	0
$f_{\mathrm{RAS}}(x) = 10N + \sum_{i=1}^{N}\left[x_i^2 - 10\cos(2\pi x_i)\right]$	$[-5.12, 5.11]$	10	10	100	0
$f_{\mathrm{SCH}}(x) = 418.9829N - \sum_{i=1}^{N} x_i \sin\left(\sqrt{\lvert x_i\rvert}\right)$	$[-512, 511]$	10	10	100	0
$f_{\mathrm{GRI}}(x) = 1 + \sum_{i=1}^{N}\frac{x_i^2}{4{,}000} - \prod_{i=1}^{N}\cos\left(\frac{x_i}{\sqrt{i}}\right)$	$[-512, 511]$	10	10	100	0
$f_{\mathrm{ACK}}(x) = -20\exp\left(-0.2\sqrt{\frac{1}{N}\sum_{i=1}^{N}x_i^2}\right) - \exp\left(\frac{1}{N}\sum_{i=1}^{N}\cos(2\pi x_i)\right) + 20 + e$	$[-32.768, 32.767]$	10	16	160	0
$f_{\mathrm{LAN}}(x) = -\sum_{i=1}^{M} c_i \exp\left[-\frac{1}{\pi}\sum_{j=1}^{N}(x_j - a_{ij})^2\right]\cos\left[\pi\sum_{j=1}^{N}(x_j - a_{ij})^2\right]$	$[0, 10.48575]$	5	20	100	-1.4

Table 2 Parameter setting of GA, TGA, and TGA2 for numerical optimization problems

	GA	TGA	TGA2
GA type		Generational	
Representation	Bit-string	Bit-string + clan + tabu list	
Population size		100	
Parent selection	2-Tournament	2-Tournament + TS	2-Tournament
Crossover		Uniform crossover	
Crossover rate		1.0	
Mutation		Bit-flip mutation	
Mutation rate		$1/l$	
Tabu list size	–	2, 4, 6, and 10	
Survivor selection	$(\mu + \lambda)$	$(\mu + \lambda)$	$(\mu + \lambda)$ + TS
Termination		5,000 generations	
Number of runs		100	

Specifically, this study evaluates solution quality according to the mean best fitness (MBF)

$$\mathrm{MBF} = \frac{\text{Sum of the best fitness of each run}}{\text{Number of runs}}, \tag{3}$$

and assesses convergence speed by observing the anytime behavior of test algorithms. Several parameter settings for these algorithms are also considered in the experiments.

4.1 Numerical optimization problems

The test suite for numerical optimization includes six common test functions: De Jong's second function (F2) [6], the Rastrigin function (RAS) [24], the Schwefel function (SCH) [26], the Griewank function (GRI) [11], the Ackley function (ACK) [2], and the Langermann function (LAN) [4]. Table 1 summarizes these test functions and their parameters used in our experiments, where N denotes the number of dimensions and l denotes the chromosome length. Note that all these functions are minimization problems. The parameter setting of GA, TGA, and the proposed TGA2 is listed in Table 2. This study experiments with four tabu list sizes: 2, 4, 6, and 10. Each experiment includes 100 independent runs of test algorithms concerning their stochastic nature.

Table 3 compares the solution quality for GA, TGA, and TGA2 on the six test functions. The one-tailed t test with confidence level $\alpha = 0.05$ is further performed to examine the statistical significance. The results show that, using proper tabu list sizes, both TGA and TGA2 can lead to significant improvement over GA on F2, SCH, and LAN in terms of solution quality. This confirms the advantage of introducing TS strategies into selection operators of GA, i.e., parent selection and survivor selection. The results also show that TGA performs worse than GA does on RAS, GRI, and ACK; and yet TGA2 can significantly outperform or, at least, is comparable with GA on these three test functions. This demonstrates

Table 3 Average (MBF), SD, and t test result (p value) of the best fitness over 100 runs for GA, TGA, and TGA2 on the six test functions

		GA	TGA				TGA2			
			$T=2$	$T=4$	$T=6$	$T=10$	$T=2$	$T=4$	$T=6$	$T=10$
F2	MBF	1.39E-04	4.39E-04	1.29E-03	1.92E-04	4.81E-05	8.84E-05	3.38E-05	7.16E-05	4.10E-05
	SD	2.42E-04	1.98E-03	6.34E-03	1.11E-03	1.49E-04	2.05E-04	1.14E-04	1.91E-04	1.42E-04
	p value	–	6.87E-02	3.71E-02	3.21E-01	**9.01E-04**	5.86E-02	**7.36E-05**	**1.58E-02**	**3.43E-04**
RAS	MBF	3.200	9.406	7.036	6.320	5.512	2.722	2.686	2.244	2.551
	SD	2.138	3.379	2.761	2.326	2.344	1.684	1.746	1.494	1.810
	p value	–	2.58E-34	4.14E-22	4.27E-19	4.61E-12	**4.13E-02**	**3.28E-02**	**1.76E-04**	**1.11E-02**
SCH	MBF	161.570	98.188	102.747	91.661	69.998	134.479	101.168	81.492	102.992
	SD	124.753	85.943	97.457	92.093	72.256	100.890	93.437	83.222	104.726
	p value	–	**2.46E-05**	**1.43E-04**	**6.42E-06**	**1.26E-09**	**4.73E-02**	**7.97E-05**	**1.65E-07**	**2.19E-04**
GRI	MBF	0.216	0.667	0.452	0.412	0.325	0.208	0.208	0.201	0.215
	SD	0.077	0.324	0.194	0.140	0.130	0.077	0.075	0.069	0.088
	p value	–	2.65E-25	3.24E-21	1.41E-24	1.23E-11	2.21E-01	2.25E-01	7.65E-02	4.56E-01
ACK	MBF	2.79E-03	2.54E+00	1.43E+00	9.32E-01	8.29E-01	2.82E-03	2.80E-03	2.86E-03	2.91E-03
	SD	4.43E-04	1.31E+00	9.00E-01	1.04E+00	1.04E+00	4.54E-04	5.09E-04	4.65E-04	5.58E-04
	p value	–	1.16E-35	3.58E-29	1.54E-14	1.73E-12	3.15E-01	3.93E-01	1.42E-01	4.47E-02
LAN	MBF	−0.763	−0.867	−0.911	−0.902	−0.923	−0.848	−0.880	−0.873	−0.888
	SD	0.188	0.133	0.086	0.101	0.069	0.154	0.126	0.130	0.114
	p value	–	**5.81E-06**	**2.61E-11**	**5.65E-10**	**3.89E-13**	**2.99E-04**	**3.24E-07**	**1.60E-06**	**2.82E-08**

Four tabu list sizes (T) are adopted for TGA and TGA2
The boldface denotes statistical significance that the test algorithm outperforms GA in fitness

that TGA2 is capable of advancing the improvement of TGA to GA. Precisely, the improvement rate of TGA2 on GA in MBF can be 76% on F2, 30% on RAS, 50% on SCH, 7% on GRI, and 16% on LAN. In view of the tabu list size, TGA2 with $T = 6$ performs satisfactorily amongst the experimental results. Nonetheless, there is no consistent tendency between tabu list size and MBF.

Next, we look into the anytime behavior regarding MBF, number of tabu events, and number of aspiration events in the course of evolution on the six numerical test functions. Figures 6 and 7 show that, in the light of variation in MBF, GA converges fastest, TGA2 follows, and TGA converges slowest. The convergence speed of TGA is relatively slow in that it repeats selecting parents for a valid mating and thus requires more fitness evaluations for generating the same amount of offspring. TGA2 overcomes this issue by filtering the tabu offspring in survivor selection, instead of in parent selection. Moreover, TGA2 keeps the advantage of TS strategies in balancing selection pressure and diversity maintenance, which is reflected in its superior solution quality to GA. Notably, TGA2 with $T = 2$ has a similar convergence speed but achieves higher fitness than GA does. As the tabu list size increases, the severer tabu restriction makes TGA2 more difficult to yield valid offspring. This intensifies TGA2 in prevention of inbreeding for diversity maintenance and results in better solution quality. However, as Figs. 6 and 7 show, this advantage in solution quality is gained at the cost of convergence speed.

Figures 6 and 7 further show the effects of TS strategies on the behavior of TGA2. The occurrence frequency of tabu events increases rapidly as evolution goes. For all the six test functions, the number of tabu events achieves its maximum before 200 generations, namely 2×10^4 fitness evaluations. Larger tabu list sizes result in larger numbers of tabu events and therefore a stricter restriction on survivor selection, which encourages maintaining population diversity. Aside from tabu events, the figures demonstrate that the number of aspiration events increases in the beginning of evolution and decreases afterward. The increase of aspiration events is due to the increasing number of tabu offspring and the ease of surpassing the best solution in the early phase. Such an increase in aspiration frequency intensifies selection pressure and can then balance the increasingly strict tabu restriction. However, as evolution goes, the number of tabu events stays at its maximum while the difficulty to excel the best solution becomes severe. Thus the number of aspiration events gradually decreases with time. As the aspiration event diminishes, the survivor selection of TGA2 is guided mainly by the tabu restriction and the progress in MBF converges owing to the lack of selection pressure.

Table 4 compares the solution quality of TGA2 for different mutation rates. Due to similarity, only the results on LAN

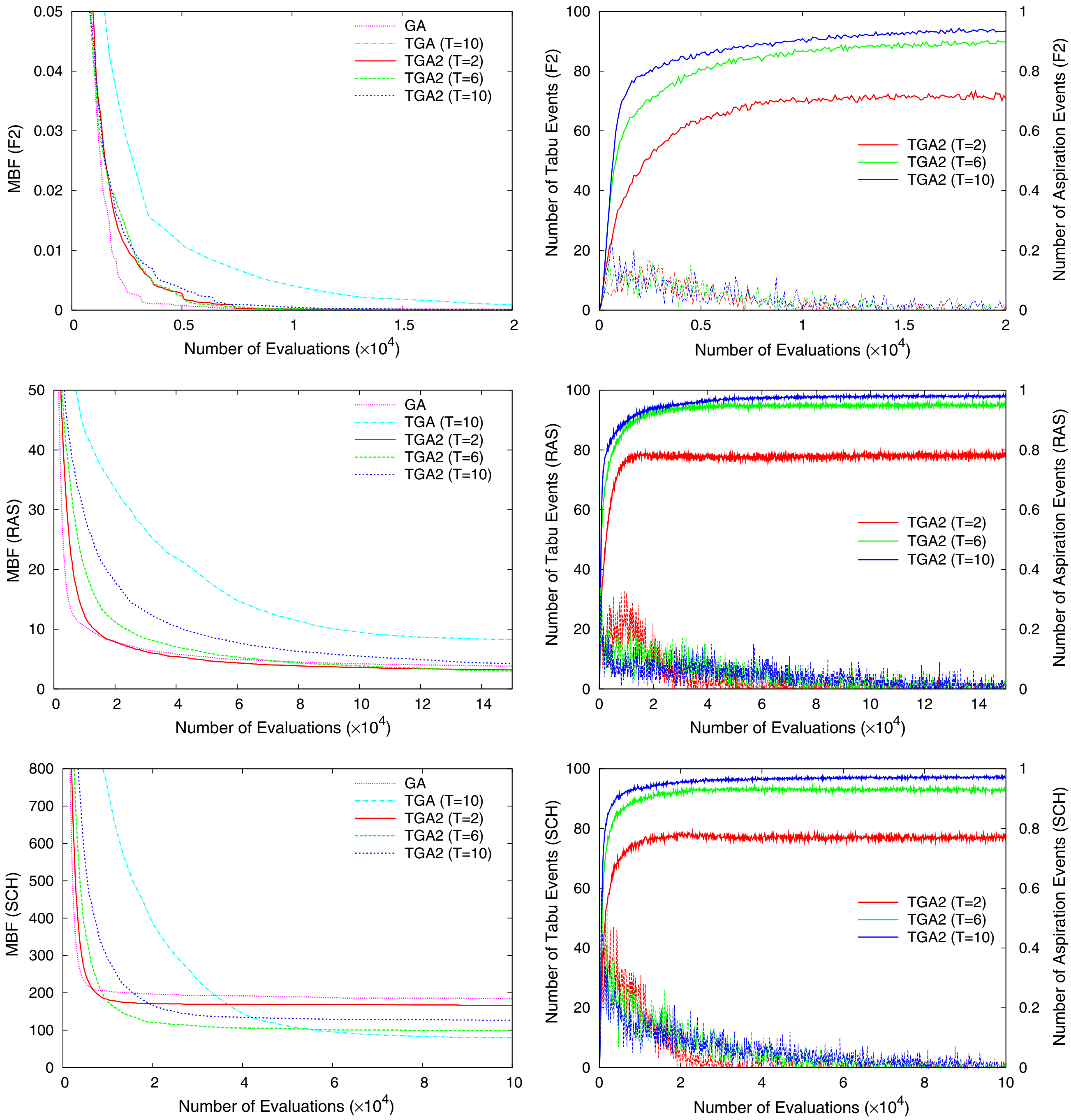

Fig. 6 Variation of MBF (*left*), numbers of tabu events (*right thick solid lines*), and number of aspiration events (*right thin dashed lines*) during evolution for GA, TGA, and TGA2 with tabu list size $T = 2$, 4, 6, and 10 on F2, RAS, and SCH

are presented here. The results show that the best tabu list size, i.e. the size corresponding to the lowest MBF, decreases as mutation rate increases. This phenomenon reflects that, with the twofold intensification of diversity from large tabu list and high mutation rate, TGA2 may focus too much on diversification and then detract from solution quality. Hence, the tabu list size for TGA2 should be increased as mutation rate is small or population diversity is low. On the other hand, for a diverse population or with high mutation rate, a small tabu list size can lead to better performance.

4.2 Combinatorial optimization problems

This study evaluates the performance of TGA2 on combinatorial, in addition to numerical, optimization problems. The well-known traveling salesman problem (TSP) is adopted

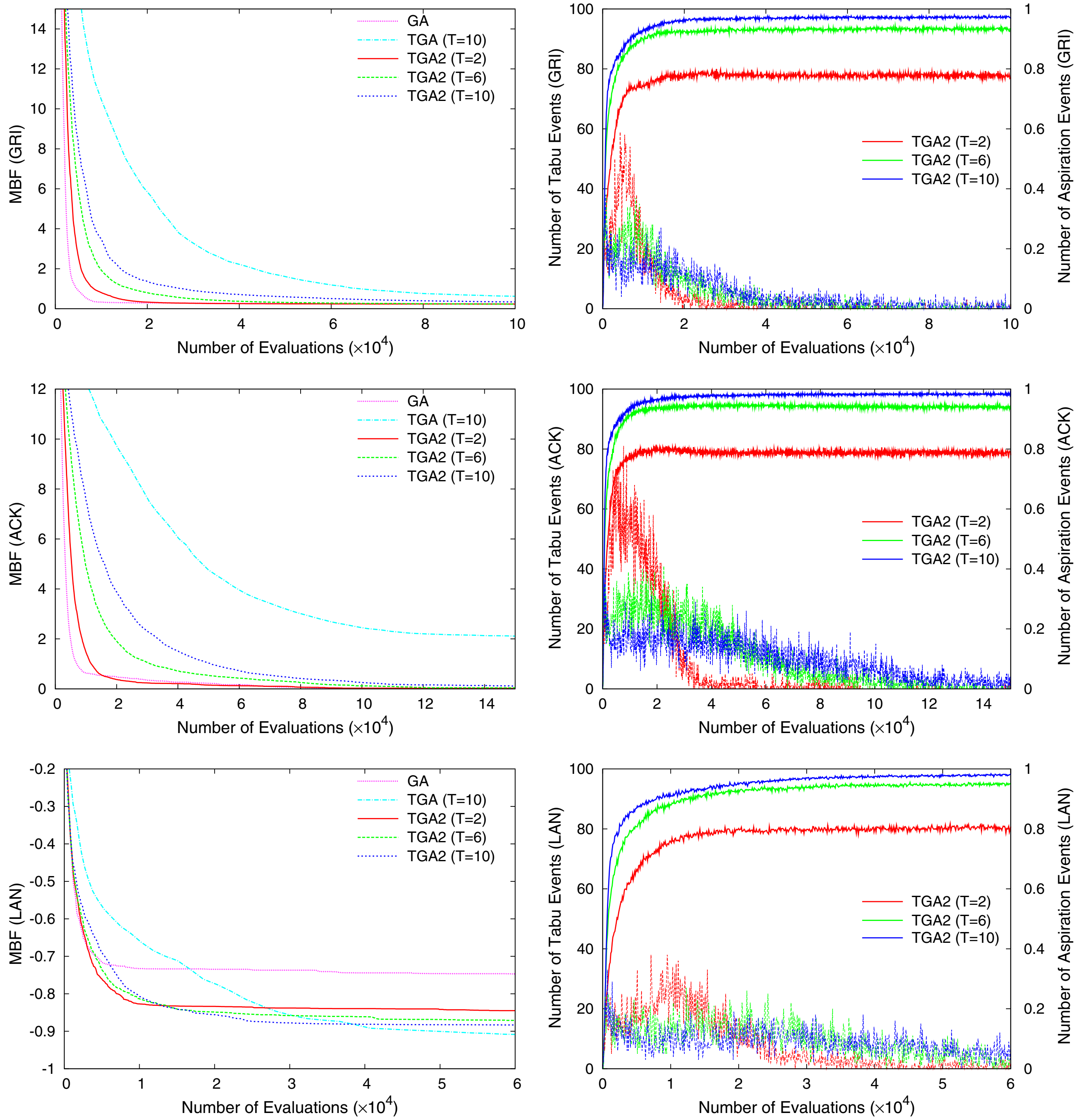

Fig. 7 Variation of MBF (*left*), numbers of tabu events (*right thick solid lines*), and number of aspiration events (*right thin dashed lines*) during evolution for GA, TGA, and TGA2 with tabu list size $T = 2$, 4, 6, and 10 on GRI, ACK, and LAN

as the testbed for combinatorial optimization. The test suite includes four TSP instances [25]: eil51, pr73, lin105, and bier127. Table 5 lists the operators and parameters of GA, TGA, and TGA2 used in our experiments for the TSP. Here the mutation rate for order-based representation is defined as the probability for the whole chromosome to be mutated rather than for a single gene [7].

In terms of solution quality, the experimental results on Table 6 show that both TGA and TGA2 outperform GA on the four TSP instances. The t test results further validate the statistical significance of their superiority over GA, which demonstrates the advantage of embedding TS strategies in the selection of GA. Concerning the impacts of the tabu list size, TGA with $T = 10$ performs best among the four test

Table 4 Average (MBF), SD, and t test result (p value) of the best fitness over 100 runs for GA and TGA2 with different mutation rates p_m on LAN test function

p_m		GA	TGA2			
			$T = 2$	$T = 4$	$T = 6$	$T = 10$
0.005	MBF	−0.764	−0.810	−0.848	−0.866	**−0.902**
	SD	0.178	0.173	0.161	0.146	0.100
	p value	–	3.50E-02	3.04E-04	8.96E-06	**1.44E-10**
0.01	MBF	−0.763	−0.848	−0.880	−0.873	**−0.888**
	SD	0.188	0.154	0.126	0.130	0.114
	p value	–	2.99E-04	3.24E-07	1.60E-06	**2.82E-08**
0.02	MBF	−0.789	−0.876	−0.906	**−0.914**	−0.902
	SD	0.182	0.136	0.108	0.098	0.099
	p value	–	9.41E-05	6.95E-08	**6.25E-09**	1.06E-07
0.05	MBF	−0.898	−0.937	**−0.946**	−0.941	−0.939
	SD	0.128	0.058	0.031	0.047	0.022
	p value	–	4.01E-03	**2.28E-04**	1.08E-03	1.12E-03

The boldface denotes the best result among the tabu list sizes with respect to each mutation rate

Table 5 Parameter setting GA, TGA, and TGA2 for combinatorial optimization problems

	GA	TGA	TGA2
GA type		Generational	
Representation	Permutation	Permutation + clan + tabu list	
Population size		100	
Parent selection	2-Tournament	2-Tournament + TS	2-Tournament
Crossover		Partially mapped crossover (PMX)	
Crossover rate		1.0	
Mutation		Swap mutation	
Mutation rate		0.1	
Tabu list size	–	2, 4, 6, and 10	
Survivor selection	$(\mu + \lambda)$	$(\mu + \lambda)$	$(\mu + \lambda)$ + TS
Termination	10,000 generations (eil51, pr76); 15,000 generations (lin105, bier127)		
Number of runs		100	

sizes on all test instances; TGA2, however, shows no clear relation between the tabu list size and the resulting solution quality. Further, TGA2 achieves better solution quality than TGA does in comparison of their MBF values for the best tabu list size on the four test instances.

Figure 8 compares the anytime behavior of GA, TGA, and TGA2. The variations of MBF, number of tabu events, and number of aspiration events show a high consistency on the four test TSP instances. In terms of convergence, GA has the fastest convergence speed but the worst solution quality. TGA spends much time in repeating parent selection and thus has the slowest convergence; nonetheless, this effort on diversity maintenance leads to better solution quality than GA. The convergence speed of TGA2 is dependent upon the adopted tabu list size: the speed slows down as the size increases. However, as Table 6 exhibits, the increase in tabu list size does not always bring about superior solution quality. Considering both the solution quality and convergence speed, TGA2 with $T = 2$ is a satisfactory choice, which has GA-like convergence speed and TGA-like solution quality. This superiority reconfirms the advantage of TS strategies embedded in survivor selection.

Further, the figure shows that the number of tabu events increases monotonically with time and tabu list size on the four TSP instances, which is similar to the situation on the numerical optimization problems. This outcome

Table 6 Average (MBF), SD, and t test result (p value) of the tour length over 100 runs for GA, TGA, and TGA2 on eil51, pr76, lin105, and bier127 instances

		GA	TGA				TGA2			
			$T = 2$	$T = 4$	$T = 6$	$T = 10$	$T = 2$	$T = 4$	$T = 6$	$T = 10$
eil51	MBF	501.71	486.63	485.12	482.65	479.49	478.69	473.78	472.24	470.78
	SD	24.65	23.17	22.57	21.76	19.56	18.42	19.13	19.86	20.02
	p value	–	**7.65E-06**	**8.40E-07**	**1.56E-08**	**1.89E-11**	**1.86E-12**	**2.35E-16**	**2.20E-17**	**1.34E-18**
pr76	MBF	145404	140559	135871	134873	130571	128931	127477	127650	133909
	SD	8120	9949	9976	10469	7312	6564	6925	6706	8313
	p value	–	**1.16E-04**	**2.48E-12**	**1.11E-13**	**4.14E-30**	**1.97E-36**	**1.11E-39**	**1.01E-39**	**3.69E-19**
lin105	MBF	22322	21363	20600	20232	19673	19291	19076	19081	20213
	SD	1515	1814	1588	1613	1233	1362	1195	1333	1515
	p value	–	**3.92E-05**	**1.71E-13**	**7.19E-18**	**7.94E-30**	**4.71E-34**	**2.22E-39**	**1.46E-37**	**5.21E-19**
bier127	MBF	162118	160098	154452	154031	149388	144396	145069	147416	157207
	SD	7292	10180	8659	10553	7341	5417	6833	7135	9193
	p value	–	**5.52E-02**	**9.06E-11**	**1.33E-09**	**2.53E-26**	**1.39E-46**	**9.98E-41**	**9.51E-33**	**2.38E-05**

Four tabu list sizes (T) are adopted for TGA and TGA2
The boldface denotes statistical significance that the test algorithm outperforms GA in fitness

demonstrates that the variation in the number of tabu events depends upon the tabu list size but is independent of problems. The number of aspiration events, on the other hand, decreases with time monotonically and does not show a peak as that on numerical optimization problems. The figure also indicates that the progress of TGA2 in MBF retards as aspiration events diminishes, which reflects the influences of aspiration over the performance of TGA2.

Table 7 presents the MBF of GA and TGA2 for different mutation rates. Due to similarity, only the results on lin105 instance are presented here. The table shows that the best tabu list size decreases with the increase of mutation rate, which is consistent with the tendency on numerical optimization problems in Table 4. The size of tabu list, therefore, should also increase for low mutation rate and low population diversity on combinatorial problems. For TGA2 using high mutation rate or holding diverse population, by contrast, the tabu list can be reduced for better solution quality.

5 Conclusions

This study proposes the TGA2, which embeds the strategies of TS, namely, tabu restriction and aspiration criterion, into the survivor selection of GA. The tabu restriction forbids inbreeding for maintaining population diversity; the aspiration criterion enables superior offspring to overcome the tabu restriction so as to supply moderate selection pressure. The survivor selection of TGA2 filters chromosomes according to these two TS strategies.

The proposed TGA2 gains three advantages by embedding TS strategies into the survivor selection of GA. First, tabu restriction and aspiration criterion control the population diversity and selection pressure in consideration of diversification and intensification. Second, this hybridization of GA and TS is independent of the adopted parent selection, crossover, and mutation operators of GA. This extends the applicability of TGA2 to diverse variants of GA. Third, TGA2 significantly reduces the computation cost of TGA because the potentially trivial repeats of select-and-check process in TGA is replaced with a single step of survivor selection in TGA2.

The performance of TGA2 is evaluated on both numerical and combinatorial optimization problems, in comparison with GA and TGA. The experimental results show that both TGA and TGA2 can significantly outperform GA in solution quality. However, TGA converges much slower than GA does whereas TGA2 can lead to superior solution quality with a similar convergence speed as GA. The experimental results also present the effects of TS strategies on the performance of TGA2. On the whole, these outcomes validate the effectiveness of the proposed hybridization of TS and GA in enhancing GA in terms of solution quality and convergence speed.

Future work includes some directions. First, local search has proved to be effective in improving evolutionary algorithms [3,14,15,17,21,22,33]. Adopting local search in TGA2 is promising to enhance its solution quality and convergence speed, especially for complex and real-world applications. Second, the performance of TGA2 is evaluated through comparison with GA and TGA. More algorithms

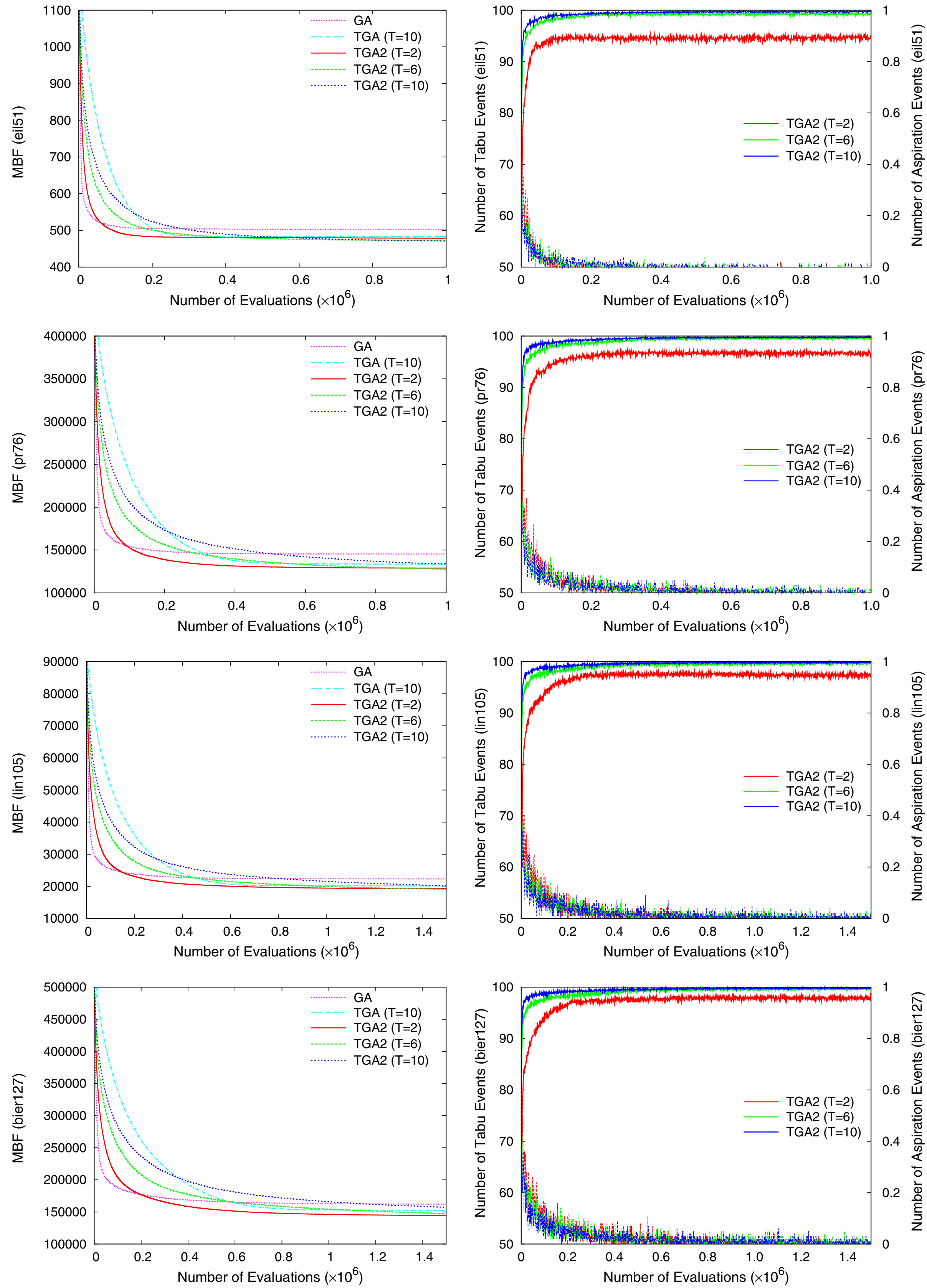

Fig. 8 Variation of MBF (*left*), numbers of tabu events (*right thick solid lines*), and number of aspiration events (*right thin dashed lines*) during evolution for GA, TGA, and TGA2 with tabu list size $T = 2, 4, 6$, and 10 on eil51, pr76, lin105, and bier127

Table 7 Average (MBF), SD, and t test result (p value) of the best fitness over 100 runs for GA and TGA2 with different mutation rates p_m on lin105 instance

p_m		GA	TGA2			
			$T = 2$	$T = 4$	$T = 6$	$T = 10$
0.05	MBF	22038	19397	19166	**18802**	20059
	SD	1783	1251	1474	1143	1625
	p value	–	3.65E-25	1.85E-26	**9.93E-34**	1.97E-14
0.1	MBF	22322	19291	**19076**	19081	20213
	SD	1515	1362	1195	1333	1515
	p value	–	4.71E-34	**2.22E-39**	1.46E-37	5.21E-19
0.2	MBF	22375	**19345**	19352	19436	20807
	SD	1455	1326	1661	1486	1706
	p value	–	**1.18E-35**	2.09E-30	6.87E-32	2.63E-11
0.5	MBF	22981	**19891**	20244	20829	22821
	SD	1403	1365	1674	1790	2084
	p value	–	**6.37E-37**	7.69E-27	8.81E-18	2.63E-01

The boldface denotes the best result among the tabu list sizes with respect to each mutation rate

and benchmark problems should be considered to validate the advantage of TGA2. In addition to empirical study, theoretical analysis is an important direction to investigate the impact of TS strategies on the behavior of TGA2.

Acknowledgments The authors would like to thank the reviewers and the Editor for their valuable comments and suggestions.

References

1. Abdinnour-Helm S (1998) A hybrid heuristic for the uncapacitated hub location problem. Eur J Oper Res 106:489–499
2. Ackley DH (1987) A connectionist machine for genetic Hillclimbing. Kluwer, Boston
3. Aranha C, Iba H (2009) The memetic tree-based genetic algorithm and its application to portfolio optimization. Memetic Comput 1(2):139–151
4. Bersini H, Dorigo M, Langerman S, Seront G, Gambardella L (1996) Results of the first international contest on evolutionary optimisation. In: Proceedings of the Third IEEE conference on evolutionary computation, pp 611–615
5. Chin AJ, Kit HW, Lim A (1999) A new GA approach for the vehicle routing problem. In: Proceedings of the 11th IEEE international conference on tools with artificial intelligence, pp 307–310
6. De Jong K (1975) An analysis of the behavior of a class of genetic adaptive systems. Ph.D. thesis, University of Michigan
7. Eiben AE, Smith JE (2003) Introduction to evolutionary computing. Natural Computing. Springer, Heidelberg
8. Gandomkar M, Vakilian M, Ehsan M (2005) A genetic-based tabu search algorithm for optimal DG allocation in distribution networks. Elect Power Compon Syst 33(12):1351–1361
9. Glover F, Kelly JP, Laguna M (1995) Genetic algorithms and tabu search: hybrid for optimization. Computers Ops Res 22(1):1–134
10. Glover F, Languna M (1997) Tabu search. Kluwer, Boston
11. Griewank AO (1981) Generalized descent for global optimization. J Optim Theory Appl 34(1):11–39
12. Hageman JA, Wehrens R, van Sprang HA, Buydens LMC (2003) Hybrid genetic algorithm-tabu search approach for optimising multilayer optical coatings. Anal Chim Acta 490:211–222
13. Handa K, Kuga S (1995) Pollycell placement for analog lsi chip designs by genetic algorithms and tabu search. In: IEEE int. conf. evol. comput., pp 716–721
14. Hart WE, Krasnogor N, Smith JE (2004) Recent advances in memetic algorithms. Springer, Heidelberg
15. Hasan SMK, Sarker R, Essam D, Cornforth D (2009) Memetic algorithms for solving job-shop scheduling problems. Memetic Comput 1(1):69–83
16. Holland J (1975) Adaptation in natural and artificial systems. University of Michigan Press, Michigan
17. Ishibuchi H, Yoshida T, Murata T (2003) Balance between genetic search and local search in memetic algorithms for multiobjective permutation flowshop scheduling. IEEE Trans Evol Comput 7(2):204–223
18. Jiang T, Cui Q, Shi G, Ma S (2003) Protein folding simulations of the hydrophobic–hydrophilic model by combining tabu search with genetic algorithms. J Chem Phys 119(8):4592–4596
19. Liaw CF (2000) A hybrid genetic algorithm for the open shop scheduling problem. Eur J Oper Res 124:28–42
20. Nara K (1997) Genetic algorithm for power systems planning. In: Proceedings of the fourth international conference on advances in power system control, operation and management, pp 11–14
21. Neri F, Tirronen V (2009) Scale factor local search in differential evolution. Memetic Comput 1(2):153–171
22. Ong YS, Lim MH, Zhu N, Wong KW (2006) Classification of adaptive memetic algorithms: a comparative study. IEEE Trans Syst Man Cybern Part B Cybern 36(1):141–152
23. Ozdamar L, Birbil SI (1998) Hybrid heuristics for the capacitated lot sizing and loading problem with setup times and overtime decisions. Eur J Oper Res 110:525–547
24. Rastrigin LA (1974) Systems of extremal control. Nauka, Moscow
25. Gerhard Reinelt. TSPLIB http://www.iwr.uni-heidelberg.de/groups/comopt/software/TSPLIB95/ (1995)
26. Schwefel HP (1981) Numerical optimization of computer models. Wiley, New York
27. Shin DJ, Kim JO, Kim KT, Choo JB, Singh C (2004) Optimal service restoration and reconfiguration of network using genetic-tabu algorithm. Elect Power Syst Res 71:145–152
28. Ting CK, Li ST, Lee C (2001) Tga: a new integrated approach to evolutionary algorithms. In: Proc. of the IEEE congress on evolutionary computation, pp 917–924
29. Ting CK, Li ST, Lee CN (2003) On the harmonious mating strategy through tabu search. Inform Sci 156:189–214

30. Vilcot G, Billaut JC (2008) A tabu search and a genetic algorithm for solving a bicriteria general job shop scheduling problem. Eur J Oper Res 190:398–411
31. Whitley D (1989) The genitor algorithm and selection pressure: Why rank-based allocation of reproductive trials is best. In: Proc. of the 3rd international conference on genetic algorithms, pp 116–121, San Mateo, CA
32. Xian B, Li T, Sun G, Cao T (2004) The combination of principal component analysis, genetic algorithm and tabu search in 3d molecular similarity. Mol Struct 674:87–97
33. Yu X, Tang K, Chen T, Yao X (2009) Empirical analysis of evolutionary algorithms with immigrants schemes for dynamic optimization. Memetic Comput 1(1):3–24