

A COMPARATIVE STUDY OF GLOBAL OPTIMIZATION APPROACHES TO MEG SOURCE LOCALIZATION*

TIANZI JIANG^{a,†}, AN LUO^{a,‡}, XIAODONG LI^{a,¶} and F. KRUGGEL^{b,§}

^aNational Laboratory of Pattern Recognition, Institute of Automation, Chinese Academy of Sciences, PO Box 2728 Beijing 100080, P.R. China; ^bMax-Planck Institute of Cognitive Neuroscience, Stephanstrasse 1, D-04103, Leipzig, Germany

(Received 6 March 2002)

It is well-known that the problem of MEG source localization can be cast as an optimization problem. So far, there have been many works in which various optimization methods were adopted for source localization. In this paper, we compare the performance of three typical and widely used optimization techniques for a specific MEG source localization problem. We first introduce a hybrid algorithm by combining genetic and local search strategies to overcome disadvantages of conventional genetic algorithms. Second, we apply the tabu search, a widely used optimization method in combinatorial optimization and discrete mathematics, to source localization. To the best of our knowledge, this is the first attempt in the literature to apply tabu search to MEG/EEG source localization. Third, in order to further compare the performance of the above algorithms, simulated annealing is also applied to MEG source localization problem. The computer simulation results show that our local genetic algorithm is the most effective approach to dipole localization, and the tabu search method is also a very good strategy for this problem.

Keywords: Magnetoencephalogram (MEG); Dipoles; Global optimization; Genetic algorithms; Simulated annealing; Tabu search

C. R. Categories: G.1.6, I.2.8

1 INTRODUCTION

Measurements of the magnetoencephalogram (MEG) as well as the electroencephalogram (EEG) provide unique insights into the dynamic behavior of the human brain as they are able to follow changes in neural activity on a millisecond time-scale [9, 21]. In comparison, the other functional imaging modalities (positron-tomography (PET) and functional magnetic resonance imaging (fMRI)) are limited in temporal resolution to time scales on the order of, at best, several seconds by physiological and signal-to-noise considerations. In the study of MEG/EEG, we are confronted with the following inverse problem. Given magnetic field

* The earlier version of this paper was finished at Max-Planck Institute of Cognitive Neuroscience, Leipzig, Germany.

† Corresponding author. E-mail: jiangtz@nlpr.ia.ac.cn

‡ E-mail: aluo@nlpr.ia.ac.cn

¶ E-mail: xdli@nlpr.ia.ac.cn

§ E-mail: kruggel@cns.mpg.de

values at a limited number of measurement points, we have to reconstruct the sources generating these data. Generally, given a suitable source and head model, this inverse problem can be cast as a nonlinear optimization problem of computing the location and moment parameters of the set of dipoles whose field best matches the MEG measurements in a least squares sense [19]. Mathematically, it is a very difficult nonlinear optimization problem because its objective function is very complex and always has many local optima, especially when the number of dipole sources is large.

In order to solve this problem, various optimization techniques have been adopted. These optimization methods can loosely be classified into two groups: gradient based Newton-type methods such as Levenberg–Marquardt [18] and gradient-free search methods such as the Nelder–Mead downhill simplex method [20]. However, the gradient based methods are problematic for this specific problem because they will easily be trapped by the local optima, which probably result in incorrect estimates of the dipole parameters. Though Nelder–Mead downhill simplex method is better than the Newton-type technique in escaping local optima, Khosla *et al.* [14] demonstrated that it is sensitive to starting parameter estimates and can also converge to a suboptimal local optimum. Therefore, conventional gradient based optimization methods and Nelder–Mead downhill simplex method are hardly suitable to MEG source localization. The key requirement to any global optimization method is that it must be able to escape in local minima and continue the search to give a near-optimal final solution whatever the initial condition is. *Simulated annealing* (SA) and *genetic algorithm* (GA) meet this requirement, and a majority of work along this stream is focused on how to apply simulated annealing algorithms as an alternative to conventional gradient based optimization methods [10, 14, 23, 24]. Unfortunately, K. Uutela *et al.* [25] have shown that for the dipole localization, the performance of SA is the worst one among their chosen three types of global optimization. Besides GA and SA, several other global optimization techniques exist, the most popular in them is *tabu search* (TS) [6]. There are very few works for applying it to continuous problems [12, 13]. It has also been shown that stimulated annealing is a special case of tabu search [6]. A key feature of this algorithm is a tabu list which is included in the search process. This provides TS with some memory and endorses some intelligence to find the optimal solution. So it is reasonable to expect that TS should be superior to SA for the dipole localization. It remains natural to ask whether GA is the most effective algorithm. Moreover, in Ref. [25] only conventional GA was adopted, though some new features have been added to GA in its implementation. However, it has been shown that the conventional GA has a very poor local performance because of the random search of GA.

Although a lot of different optimization methods exist, the efficacy of an optimization method is always problem specific. In this paper, we compare the performance of these three typical and widely used optimization techniques for a specific MEG source localization problem. The computer simulation results show that our local genetic search algorithm is the most effective approach to dipole localization.

The paper is organized as follows. Section 2 is devoted to the formulation of the problem. In Section 3, we give a detailed description of our hybrid genetic algorithm, simulated annealing, and tabu search and their implementations. In Section 4, we describe the procedure of computer stimulation. Section 5 are experimental results and discussions. The final section presents a summary of our conclusions.

2 MEG SOURCE IMAGING AS AN INVERSE PROBLEM

Given a suitable source and head model, we always need to reconstruct the source from some given magnetic field values at a limited number of measurement points. This inverse problem

can be cast as the following least squares problem of estimating the location and moment parameter of current dipoles [19].

$$E(L, Q) = \|B - G(L)Q\|_F^2 \tag{1}$$

where B are current dipoles, L and Q are location and dipole moment parameters in the three dimensions respectively, $G(L)$ is the gain matrix, and $\|\cdot\|_F$ indicates the Frobenius norm. We refer to [19] for a detailed derivation of this formula. Thus the inverse problem is to find the set $\{L, Q\}$ to minimize this error function. In our implementation, we transferred this problem to maximize the reciprocal of the error function $f(L, Q) = E(L, Q)^{-1}$, which is our fitness function. We must use some iterative optimization algorithm to solve this nonlinear optimization problem. Assuming that there exist N dipoles and k time points, then there will be $3N$ location parameters and $3Nk$ moment parameters, for an overall of $3N(k + 1)$ parameters. In practice, it is a very difficult optimization problem due to its dimension. For example, if there are $N = 3$ dipoles and the time points $k = 80$, then we will have to search for a 729-dimension parameter space to find the global optimum for this problem. If this is a linear problem, the size of the problem is acceptable, while for a nonlinear problem, its computational complexity will be overwhelming.

Fortunately, the computational complexity of the problem can be greatly reduced by separating the linear and nonlinear parameters because B is a linear function of parameter Q . The method to factor out the linear moments has been used by many researchers, say, [19] and references therein, and has been mathematically justified in Gloub *et al.* [5]. First, for any location parameter L , the optimal Q that will maximize $f(L, Q)$ is

$$Q = G^*B \tag{2}$$

where G^* is Moore-Penrose pseudo-inverse which can be found by $G^* = V\Sigma^+U^T$, where $G = U\Sigma V^T$ is the singular value decomposition (SVD) and Σ^+ is the inverse of Σ . Then replace Q with this pseudo-inverse solution before solving for L , the cost function of the inverse problem then becomes

$$E = \|B - GQ\|_F^2 = \|B - GG^*B\|_F^2 = \|(I - GG^*)B\|_F^2 \tag{3}$$

Now we can see the cost function no longer depends on matrix Q . The number of parameters of cost function has been reduced to $3N$, for example, 9 in the case of three dipoles.

It has been shown by Mosher *et al.* [19] that the computation process can be further simplified by using some technique to compute $I - GG^*$, and the final result is as follows.

$$E = \|P_G^\perp B\|_F^2 = \|U_{m-r}^T B\|_F^2 \tag{4}$$

where P_G^\perp is projections which project the data into left null space of G and supposing $G = [U_r U_{m-r}]\Sigma V^T$, where r is the rank of G . For details, we refer to Mosher *et al.* [19].

After finding the value of L using some nonlinear optimization techniques, $3Nk$ moment parameters in Q can be estimated using Eq. (2). This is computationally cheap since it only needs to compute the pseudo-inverse of G .

3 OPTIMIZATION METHODS

In Section 2, the problem of MEG source localization is addressed as an optimization problem. Once it is understood that MEG source localization is such an optimization problem, the use of any technique for tackling optimization problems is obvious. In this section, we will give an introduction to genetic algorithm, simulated annealing, and tabu search, and the details of our implementation of these algorithms to this problem.

3.1 Genetic Algorithm

The Genetic algorithm is an iterative optimization technique, originating from the mechanics of genetics and natural selection. A group of candidate solutions, which are described as binary sequences (chromosome), is viewed as a generation of population. Operations such as selection, crossover and mutation are possessed upon these chromosomes to produce new offsprings with better fitness.

Genetic algorithms (GA) have been shown to be capable of searching for optima in function which cause difficulty for gradient based methods [7]. They have many attractive attributions and have been successfully applied to many nonlinear, multi-peak, continuous or combinatorial optimization problems. The principal attractions of GA are domain independence, non-linearity and robustness. Domain independence means that it is easy to write one general computer program for solving many different optimization problems. Non-linearity means that GA does not need the unrealistic assumptions of linearity, convexity, differentiability which are necessary for many conventional optimization techniques. The only requirement for GA is the ability to calculate the measure of performance, which may be highly complicated and non-linear. The above two characteristics of GA assume that GA is inherently robust. They can work with highly non-linear functions, and they can cope with a great diversity of problems from different fields. For details about GA, we refer to [7].

K. Uutela *et al.* [25] have applied GA to the problem of MEG source localization. They adopted the conventional GA though some new features have been added to GA in their implementation. However, it has been shown that the conventional GA has a very poor local performance because of the random search of GA. To achieve a good solution, great computational cost is inevitable. Many improvements can be made in methodological decisions and parametric choices to enhance the performance of GA. In this paper, several remarkable features are added and some important extensions are also made to improve the performance of the conventional GA. The computer simulation results have shown that our method is able to greatly reduce the computational cost and improve the precision of the solution. The implementation details of these features are described as follows.

3.1.1 Hybrid Algorithm

In many cases, hybridizing GA with another heuristic method can significantly enhance the effectiveness of a GA [16]. Since GA has poor local search performance and conventional local search methods have remarkable ability in finding local optima, we propose a hybrid algorithm, which combines GA and a modified local search procedure. In the hybrid algorithm, Hooke-Jeeves local search procedure [11], a simple but efficient local search method, is applied to new solutions generated by the genetic operations (*i.e.*, selection, crossover, and mutation) to maximize their fitness value f . The local search is also applied to elite solutions inherited from the previous populations. Some parameters are introduced into the algorithm to control the local search. The parameter *Step* is used to control the initial step of the search.

The parameter ε is used to control the distance from the solution to its nearest local optimum. If we use a large value of *Step* and ε , the local search will be fast and the computational cost will be little though the solution will not be precise. This is what we expect in the beginning of the algorithm. While in the end of GA, a high precise solution is needed, we should set *Step* and ε to be small.

3.1.2 Elitist Strategy

Two sets of solutions are stored in our algorithm: a current population and an elite set. After genetic operation and local search, the current population is replaced with the improved population and the elite set is updated by a new set. Local search procedure is applied to solutions in the elite set without applying genetic operations to them. By preserving the solutions appearing in one generation and adding them into next generation without any genetic operations, we can avoid losing some excellent solutions. It has been shown that only after one adopts the elitist strategy, the GA can mathematically converge to a global optimum [7]. In our algorithm, the parameter *EliteRate* is the proportion of the number of solutions of the elite set to that of the whole population. But we must be careful about the value of *EliteRate* because a large value of *EliteRate* will make GA trapped onto a sub-optimal solution when there are many poor solutions and just one or two outstanding ones in the population at the start of the GA. The one or two outstanding solutions will rapidly be dominant in the group and will lead to premature convergence of the algorithm. So we set the *EliteRate* a small number at the beginning and increase it as the optimization goes on.

3.1.3 Selection and Crossover Operation

In our algorithm, we adopt the roulette wheel selection mechanism. The selection probability $P(x)$ in the current population Φ can be expressed as:

$$P(x) = \frac{f^\lambda(x)}{\sum_{x \in \Phi} f^\lambda(x)} \quad (5)$$

where $f(x)$ is the fitness value of solution x and λ is the parameter to control the scaling procedure of selection operation. The parameter λ is very important in our algorithm because different values of λ can make GA result in different solutions. When the value of λ is large, the selection mechanism is strong and competition in the group is intense, and some outstanding solutions in the population have greater chance to survive. However, the GA is easy to be trapped in a non-optimal value because one or two outstanding solutions in the group will be rapidly dominant in the population and the evolution will stop at the sub-optimal solution. This is what we do not expect in the beginning. Thus in the beginning of GA, we give λ a small value to limit the competition. When we want to stimulate it to speed the evolution and make GA converge to a optimal solution after several generations, we can give λ a larger value. Another reason for us to adopt a scaling strategy is that the difference of the fitness value of different solutions may be very large for MEG source localization problem. For example, fitness of one or two solutions may be one hundred times larger than other solutions in the beginning of the evolution though the fitness of them is very low. In that case, naive fitness measurement may lead to a rapid takeover of the one or two solutions and premature convergence will occur. Using the scaling procedure, we can mitigate this problem.

3.1.4 Parameters EliteRate, λ , ε , and Step in our algorithm

The performance of the algorithm largely depends on the parametric choice. In order to reduce the computational cost, simultaneously to keep the convergence to a global optimal solution, we carefully adjust the parameters in our algorithm in different stages of the algorithm. We divide the algorithm into three stages and adjust the parameters in each stage. To ensure the algorithm to find global optimum and avoid premature convergence to a poor solution, the competition is limited early on, but it is stimulated later. The number of elite is limited at the start and increased later. Another problem is to reduce the computational cost, this has been realized by adjust the parameters of the local search since the main computation is spend on local search. The strategy we adopt to adjust the parameter is listed in Table I. Our GA is illustrated in Figure 1.

3.2 Simulated Annealing

Simulated annealing (SA) is a stochastic simulation method originally proposed by Kirkpatrick *et al.* [15]. It was heralded as a simple and robust algorithm and had been proved to be useful in a wide range of complex combinatorial optimization problems which was difficult for conventional optimization problems. The idea of SA mainly comes from the field of statistical mechanics. The process of annealing is analogous to the process of the optimization, in which the value of the cost function takes the role of the energy of the system, and the global optimum as the energy of the ground state of the system. Metropolis *et al.* [17] proposed a Monte Carlo method to simulate the evolution of the system at a fixed value of temperature T . If the system in energy state E_1 is perturbed to another energy state E_2 , the new state is accepted with the probability $\exp(-\Delta E/kT)$, where $\Delta E = E_2 - E_1$, that is to say when the perturbed state is of lower energy, it will always be accepted. While when perturbed to a higher energy state, the probability to accept the new energy state will depend on ΔE and T . By repeating the basic step many times, the system will evolve into thermal equilibrium and the temperature will be lowered. As the temperature decreases, the probability of accepting uphill steps will decrease and the algorithm will eventually converge into a global optimum. We refer to [1] for details about SA.

Because of its outstanding performance in finding global optimal solution, SA has been implemented in a variety of different ways to MEG/EEG source localization problem [10, 14, 23, 24]. The implementation details of the algorithm used by us to solve the source localization problem are provided below.

3.2.1 Move Strategy

The core of SA is the perturbation of the solution and then acceptance of the rejections based on the Metropolis criterion. A reasonable move strategy will lead to the efficient search of the parameter space. There are some move strategies that has been used, which include round-robin fashion [3], simplex move by Ref. [14]. It has been shown that round-robin scheme

TABLE I Configuration of parameters in different stages of the hybrid algorithm.

	<i>EliteRate</i>	<i>Step</i>	ε	λ
Stage 1	0.1	No local search	No local search	0.3
Stage 2	0.15	2	0.5	0.6
Stage 3	0.3	0.5	0.05	1

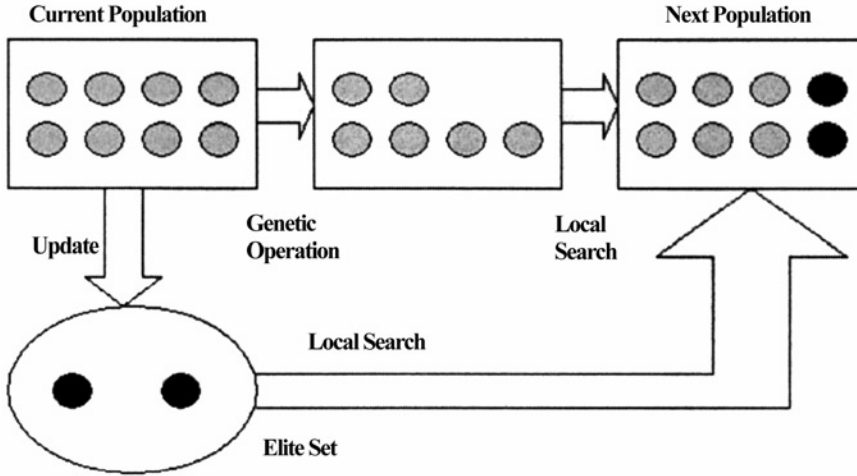


FIGURE 1 Genetic local search algorithm.

can get trapped in shallow valleys not parallel to the parameter axes [3]. We adopt a simple move strategy in which new configurations are selected from a normal distribution centered on the current configuration, in the hope this strategy will be the most suitable to find better solutions while at the same time have the potential to jump out of the local optima. We do not compare the relative merit of different move strategies, but our test results show this move strategy is rather effective.

3.2.2 Thermal Equilibrium Criterion

In order to ensure the SA can converge to a global optimum, it is important that thermal equilibrium must be reached at each value of temperature. If this is not done, the SA may get stuck in locally optimal value of the cost function. However, to reach thermal equilibrium, length of the Markov chain (the number of the trial configurations generated) should be theoretically infinite, this would result in a impractically large computational time. In practice, some approximate methods are often used. The work of K. Sekihara *et al.* [23] use η as a value to evaluate whether thermal equilibrium has been attained.

$$\eta = \frac{|\Gamma_1 - \Gamma_2|}{|\Gamma_1 + \Gamma_2|} \tag{6}$$

where Γ_1 is the number of accepted transitions that increase the cost function, Γ_2 is the number of accepted transitions that decrease the cost function. Given a small value ε , if $\eta < \varepsilon$, we deem the equilibrium to be attained. The parameter ε can be adjusted to get a trade-off between the computational cost and the possibility of the global minimum convergence. Another method is to give a fixed number L_k to determine the length of the Markov chain at each temperature [1]. The number can be determined empirically. We have compared the performance of the two methods by experiments. In our work, we use this method to determine the length of Markov chain because this way has been shown to be simple and effective and L_k is set to 3.

3.2.3 Annealing Schedule

Suitable anneal schedule is most important to ensure the SA converges to a global optimum. It includes the initial temperature (T_0), temperature decrement rule and the terminate criterion. In our work, we adopt the following strategies:

(a) *Initial Temperature:* Temperature is an important factor in the process of simulated annealing, initial temperature affects the trade off of the computational cost and possibility to find global optimum. The initial temperature should be high enough to melt the system at the beginning of the annealing; however an overly high initial temperature will consume too much time and computation. A principle for determining T_0 is that under T_0 , virtually all moves will be accepted, *i.e.*, $\exp(-\Delta E/T_0) \approx 1$, however this will make T_0 very large. In practice, given the predetermined acceptance ratio χ_0 , T_0 is often determined by calculating the average increase in cost function $\Delta\bar{E}$ for some times (*e.g.*, 1000) and then obtained through the following formula [1]

$$T_0 = \frac{\Delta\bar{E}}{\log(\chi_0^{-1})} \quad (7)$$

(b) *Temperature Decrement Rule:* In order to avoid an impractically long time of computation, the temperature decrement should not be too slow. However, if the decrement is too fast, thermal equilibrium cannot be reached at each temperature. The temperature should be decreased such that given the length of the Markov chain can establish quasi-equilibrium after each decrement. We adopt the following decrement rule

$$T_k = \alpha T_{k-1} \quad (8)$$

where T_k is the k th temperature, α is the parameter to control the decrement. The typical value for α is 0.8–0.95, and we set α as 0.85.

(c) *Terminate Criterion:* we set the number of the times of lowering the temperature in order to control the computational time more easily. We did a Hooke-Jeeves high precision local search at the end of the algorithm to make the localization more precise.

3.3 Tabu Search

Tabu search (TS) is a meta-heuristic optimization method that can be used to explore the solution space beyond local optimality [6]. The notion of exploiting certain forms of flexible memory to control the search process is the central theme underlying TS. Because of the memory TS has some intelligence. It is different from the well-known hill-climbing local search techniques in that it employs a somewhat different philosophy for going beyond the criterion of terminating at a local optimum. By allowing the moves out of a current solution and accepting a worse solution in the hope that it will eventually find a better solution, it does not become trapped in a local optimal solution. TS is a flexible framework. The history record gives TS some memory and the rules give TS some intelligence to find the optimal solution and avoid being trapped onto local optima. We refer to [6] for the details about tabu search.

Tabu search starts with a certain solution x_{now} , then, follows a certain set of candidate moves. A candidate set of moves $\text{Cand}(x_{\text{now}})$ can be obtained from its neighborhood $N(x_{\text{now}})$. Based on the history record of TS, some of the moves are tabu, some of the

moves are permitted. Then the aspiration criteria is applied to candidate set, each solution in the set is evaluated based on its value of cost function $c(x)$, the history record H , and the aspiration criteria. The best move in the set is selected to be the new solution x_{next} . Repeat the procedure until some stop criteria are satisfied in our tabu search procedure, x_{now} includes a set of solutions with the best fitness values in the optimization process and is called the best solution list. Our algorithm may be expressed in the following manner:

- Step 1 *Initialization*: Produce a set of solutions randomly, sort order of these solutions according to their fitness values, put the best ones in the best solution list, and initialize the history record (memory) with empty.
- Step 2 *Choice and Termination*:
- Step 2.1 In the neighborhood of the best solution list $N(x_{\text{now}})$, select the ones that satisfy the tabu restrictions as the candidate moves $\text{Cand}(x_{\text{now}})$.
- Step 2.2 Sort order of the solutions in $\text{Cand}(x_{\text{now}})$, do the local search to some good solutions, and use the best ones to update the best solution list.
- Step 2.3 Terminate by a chosen iteration cut-off rule.
- Step 3 *Update*: Perform the update for the history record H and go to Step 2.

Step 2 and Step 3 of process are called an iteration. TS is illustrated in Figure 2.

To the best of our knowledge, this is the first attempt to use tabu search to solve the MEG source localization problem. Conventional applications of TS mostly focus on combinatorial optimization problems. However the MEG/EEG source localization problem is a continuous optimization problem. In order to apply TS effectively to it, some strategies specific to this problem are carefully devised. These strategies include the definition of neighborhood, some suitable tabu restrictions and aspiration criteria and an appropriate local search method. The implementation details of TS can be described as follows.

3.3.1 Definition of Neighborhood

The strategy to produce a set of candidate moves from current solutions is important. It will influence the computational cost and the performance of the algorithm. In this paper,

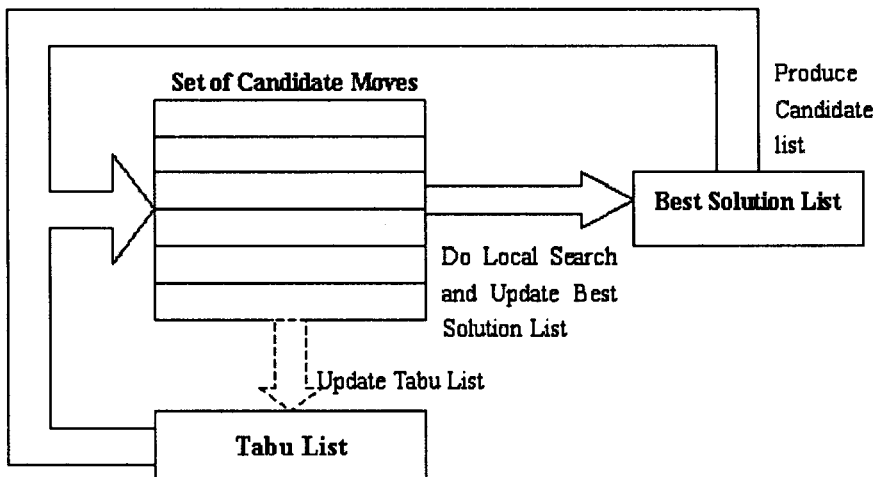


FIGURE 2 Tabu search procedure.

we generate the candidate list in this way: on the one hand, we combine the dipoles of the solutions in the best solution list to find a better one after a large scale of random search, for the good solutions have the good component (dipole); on the other hand, we still reserve the random search method to some extent to move out of a local optimum. Since the magnetic field is the sum of each field generated by each dipole, our method is especially suitable for this problem. The size of the neighborhood is closely related to the size of the best solution list, since a larger best solution list will result in more combinations of the dipoles.

3.3.2 *Tabu restrictions and Aspiration Criteria*

When candidate moves are produced, they are put in the tabu list so that the repeated search can be avoided. Tabu search method is typically used in discrete problems, and we need to define what is a repeated search in a continuous one. It is natural to use the distance between the corresponding dipoles in a newly generated solution and in the tabu list as the criterion. If the distance between all the corresponding dipoles in this newly generated solution and in some solution in the tabu list is within a given boundary value (this value will get smaller and smaller as the optimization process goes on), the candidate move is viewed as repeated move and is not permitted. When the next candidate moves are produced, the tabu list is updated and here aspiration criteria work on the solution in the old tabu list.

As the optimization process goes on, the search will be more precise, and it is very likely to find a better solution near the solutions that were taboo previously. By updating the tabu list and the boundary value, the TS has the potential to find global optimum.

3.3.3 *Local Search Strategy*

In the algorithm of Tabu Search, we also use the Hooke-Jeeves Local search method as our Local Search Strategy. For details about the Hooke-Jeeves method, we refer to Section 3.1. In our algorithm, as the number of iterations increases, the parameters of the local search are adjusted so that the solution will be more precise.

In this section, we have introduced different global optimization techniques and described our implementation details of them to MEG/EEG source localization problem. In the next sections, we will describe the design of our experiments and the results of them.

4 COMPUTER SIMULATIONS

In order to find the relative merit of the proposed optimization methods, we carefully devise some experiments and give some mathematical measure to compare performance of these algorithms. The performance of proposed algorithms is assessed and compared via a large number of computer simulations. In this section, we describe the procedure of our simulation in detail.

4.1 Assumptions

Multiple current dipoles model and a spherically symmetric conductor head model are adopted in our simulation. The human head is assumed to be a spherically symmetric conductor with an outer radius of 120 mm. The magnetic field data are measured by 128 evenly distributed sensors. Only the magnetic field component normal to the spherical surface is measured, thus only magnetic fields due to the primary tangential dipole currents

are computed, as we have discussed in Section 2. Our simulation can be easily extended to some more complex forward model. We should note that the number and location of the detectors can significantly influence the difficulty of source localization.

In the simulation, three dipoles ($N = 3$) are assumed to generate magnetic data. Since the radial component of a current dipole does not generate a measurable magnetic field on a spherical surface, this component is neglected and only the location and two tangential components are considered. Three cases of dipoles with different location, orientation, and time course are used to simulate. These three cases are designed to evaluate the performance of the hybrid GA, TS and SA. The three dipoles in Case III are the closest, and their time courses overlap most, then the dipoles in Case II, and the dipoles in Case I are relatively separated spatially and temporally. In all cases, we assume the three dipoles are all with fixed location and orientation, which is in accord with the real condition. The dipole parameters are listed in Table II. A total of 100 time samples are generated in our simulation. The shape of the dipole time courses is assumed to be double-sided Gaussian (Fig. 3(a)–(c)) since physiologically it is expected the wave forms will have smooth shapes like the ones chosen here. Under each case, a noiseless and 2 different noisy conditions are adopted in the comparison. We assume the noise to be Gaussian, and the power of noise to signal are 0%, 5%, 10%, respectively.

In this paper, the number of dipoles is assumed to be known. We do not consider the unknown number case so that we can focus on the comparison of the relative performance of different optimization techniques.

4.2 Methodology for Evaluation of Optimization Algorithms

The diversity of optimization techniques imposes a need to evaluate the relative performance of these algorithms and understand the relationship between their free parameters and their performances. However, the task to evaluate different optimization algorithms is much more difficult and complex than it appears. The relative performance of different algorithms is task specific. In a specific implementation of an algorithm, some choices have to be made on its many parameters. Such choices affect the performance of the algorithm. Thus in our experiments, we first use an objective evaluation and optimization procedure to give the best parameter choices for a particular task, then we compare the efficacy of different algorithms under the same computational time under the same computer, in another word, the same computational cost.

TABLE II Dipole parameters to generate magnetic field.

	L_x	L_y	L_z	M_x	M_y	M_z
<i>Case I</i>						
Dipole 1	−9.0	1.0	2.0	0.577	0.577	0.577
Dipole 2	1.0	9.0	−2.0	−0.577	0.577	0.577
Dipole 3	−1.0	3.0	9.0	0.577	−0.577	0.577
<i>Case II</i>						
Dipole 1	2.8	−1.7	8.3	0.577	0.577	0.577
Dipole 2	−2.9	8.3	0.0	−0.577	0.577	0.577
Dipole 3	8.1	3.3	−1.2	0.577	−0.577	0.577
<i>Case III</i>						
Dipole 1	2.8	−1.7	8.3	0.577	0.577	0.577
Dipole 2	−2.9	−1.6	8.3	−0.577	0.577	0.577
Dipole 3	0.0	3.3	8.4	0.577	−0.577	0.577

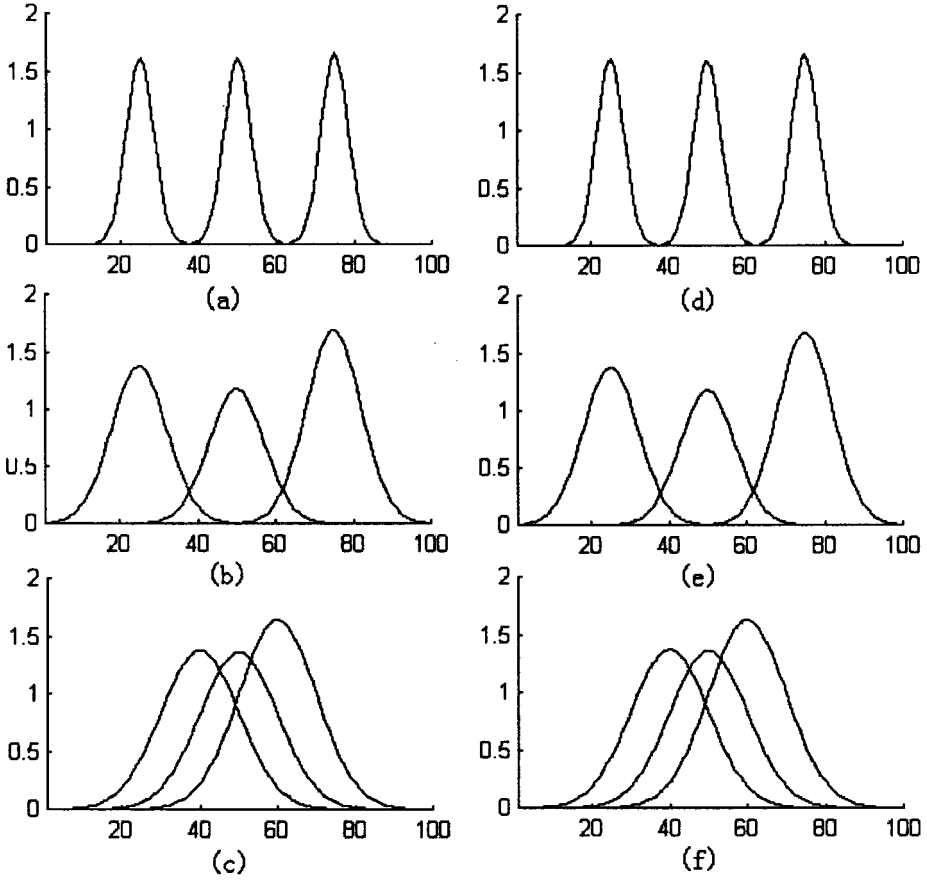


FIGURE 3 Dipole time courses for dipole 1, 2, and 3. (a), (b), and (c) are the time courses in Case I, II and III respectively. And (d), (e) and (f) are estimated dipole time courses. The location of dipoles is optimized using GA after the location of dipoles is successfully estimated. The time course of each dipole is estimated using rank one approximation from SVD for each dipole.

To compare different algorithms, some mathematical measures are needed. Such mathematical measures have been used in previous work such as localization error d and squared error of the magnetic field E in the paper of Haneishi *et al.* [10], root-mean-squares error, and relative difference measure (RMD) in the paper of Khosla [14]. In this paper, we use value of the localization error d as a measure for each dipole estimate, which is defined as follows.

$$d = \sqrt{(\bar{L}_x - L_x)^2 + (\bar{L}_y - L_y)^2 + (\bar{L}_z - L_z)^2} \quad (9)$$

where L_x , L_y and L_z are three coordinates of a dipole to be estimated, and \bar{L}_x , \bar{L}_y and \bar{L}_z are the estimated coordinates of this dipole. When a solution is obtained using an algorithm, the localization error d_i for each dipole can be calculated, and by repeating this process many times, the probabilistic distribution function of the localization error of this algorithm under each case can be obtained. We deem the estimation of a dipole to be correct when its localization error is less than 0.06 cm, which can satisfy the requirement in application very well.

4.3 Simulation Procedures

Our simulation consists of the following process:

- (1) Compute the magnetic field using the forward model and dipole parameters stated above.
- (2) Estimate source locations on the whole time window using hybrid GA, SA and TS respectively by maximizing the fitness function as we described in Section 2.
 - 1) Evaluate the performance of each algorithm when the parameter configuration is different and get the optimal parametric choice for each algorithm.
 - 2) Compare the performance of different algorithms when the computational cost of each algorithm is the same.
- (3) Using the estimated source location, compute best-fit dipole time course for each dipole.

5 EXPERIMENTAL RESULTS AND DISCUSSIONS

Before we present the results of the comparisons, we should mention once again that in our experiments, we assume that correct head and source models are known. Thus in our experiments, the errors due to incorrect modelling are eliminated. Our aim is to evaluate the relative performance of these algorithms when they are applied to MEG source localization. As mentioned above, the performance of an algorithm is task-dependent and depends on the configuration of parameters. In our experiments, three cases of dipoles with different intensity of noise are used to test the performance of GA, SA and TS. For each algorithm, a procedure to optimize the parameter value is executed before the algorithm is used to source localization. The results of the three algorithm under the same computational cost are listed in Tables III–V. And we can see the three algorithms' probabilistic distribution function of localization error in the three cases in Figures 4–6, respectively.

5.1 Genetic Algorithm

After a large number of simulations, the optimal parameter configuration has been selected. In our experiments, the parameters ε , λ , and *Step* have been selected according to the different

TABLE III Results using the three algorithms on different noise conditions in Case I.

<i>Noise/signal</i>	<i>GA</i>	<i>TS</i>	<i>SA</i>
0%			
Success	100%	100%	63.3%
d_1 cm	0.007 ± 0.003	0.009 ± 0.004	0.410 ± 0.257
d_2 cm	0.009 ± 0.005	0.011 ± 0.006	0.500 ± 2.319
d_3 cm	0.007 ± 0.003	0.008 ± 0.003	0.020 ± 0.055
5%			
Success	99.3%	97.3%	56.7%
d_1 cm	0.011 ± 0.004	0.014 ± 0.014	0.431 ± 0.282
d_2 cm	0.024 ± 0.013	0.197 ± 0.693	0.695 ± 2.442
d_3 cm	0.008 ± 0.003	0.009 ± 0.003	0.024 ± 0.061
10%			
Success	90.7%	90.7%	53.3%
d_1 cm	0.017 ± 0.008	0.017 ± 0.007	0.424 ± 0.275
d_2 cm	0.050 ± 0.024	0.214 ± 0.675	1.010 ± 3.273
d_3 cm	0.009 ± 0.003	0.009 ± 0.004	0.062 ± 0.218

Initial solutions are randomly generated in the parameter space.

The criteria for success and the measurement d_i are defined in Section 4.

TABLE IV Results using the three algorithms on different noise conditions in Case II.

<i>Noise/signal</i>	<i>GA</i>	<i>TS</i>	<i>SA</i>
0%			
Success	92.7%	94.0%	90.0%
d_1 cm	0.008 ± 0.004	0.009 ± 0.005	0.018 ± 0.048
d_2 cm	0.029 ± 0.071	0.029 ± 0.022	0.731 ± 2.886
d_3 cm	0.102 ± 0.444	0.082 ± 0.308	0.070 ± 0.117
5%			
Success	91.3%	91.3%	88.0%
d_1 cm	0.009 ± 0.004	0.011 ± 0.005	0.013 ± 0.029
d_2 cm	0.027 ± 0.015	0.033 ± 0.016	0.733 ± 2.889
d_3 cm	0.043 ± 0.028	0.044 ± 0.306	0.082 ± 0.273
10%			
Success	90.0%	89.3%	84.0%
d_1 cm	0.011 ± 0.014	0.011 ± 0.011	0.028 ± 0.079
d_2 cm	0.034 ± 0.020	0.227 ± 1.039	0.861 ± 3.271
d_3 cm	0.047 ± 0.031	0.248 ± 1.004	0.072 ± 0.876

configurations in Table I. We do 50 experiments for each configuration under three different noisy conditions and test the probability it can find the correct location and the localization error d .

5.2 Tabu Search

The strategy of TS algorithm we use has been stated in Section 3. It's obvious that one key factor for performance of TS is the number of iterations, since the more the number of iterations, the larger the number of points TS have searched. Another factor that influences TS's performance is the size of the tabu list, for an over small one will not avoid the repeated search, while an over large one will take too much time to search in this list. As the size of the tabu list is directly related to that of the best solution list, we do a large number of experiments to find the optimal size of the best solution list, which is 7.

TABLE V Results using the three algorithms on different noise conditions in Case III.

<i>Noise/signal</i>	<i>GA</i>	<i>TS</i>	<i>SA</i>
0%			
Success	100%	100%	85.3%
d_1 cm	0.008 ± 0.003	0.009 ± 0.004	0.493 ± 1.244
d_2 cm	0.009 ± 0.004	0.011 ± 0.006	0.657 ± 1.729
d_3 cm	0.007 ± 0.004	0.008 ± 0.003	0.103 ± 0.272
5%			
Success	100%	100%	84.0%
d_1 cm	0.011 ± 0.006	0.014 ± 0.014	0.703 ± 1.893
d_2 cm	0.011 ± 0.004	0.197 ± 0.693	0.179 ± 0.476
d_3 cm	0.008 ± 0.005	0.009 ± 0.003	0.568 ± 1.488
10%			
Success	100%	98.0%	77.3%
d_1 cm	0.014 ± 0.007	0.068 ± 0.398	0.823 ± 0.577
d_2 cm	0.014 ± 0.006	0.073 ± 0.423	1.083 ± 2.206
d_3 cm	0.012 ± 0.006	0.022 ± 0.078	0.261 ± 0.594

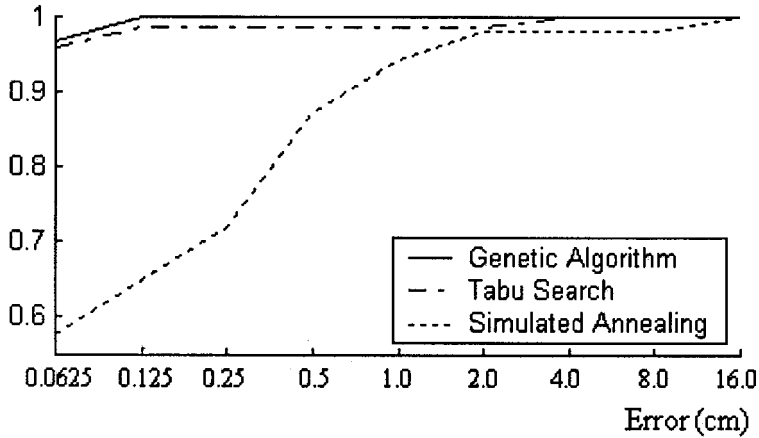


FIGURE 4 The three algorithms' probabilistic distribution function of the localization error requirements in Case I.

5.3 Simulated Annealing

The most important parameter in SA are T_0 , α and L_k . A suitable configuration of the parameters in SA is a key factor for its successful implementation. We have done a large number of experiments using different parameter to test the performance of SA. In our simulation, T_0 is set at 0.4 and the start location is randomly specified in the parameter space. For each parameter pair and each noisy condition, 50 simulation is conducted.

5.4 Comparison Among Three Cases

A lot of simulations are implemented in Case I, II and III. The results are listed in Tables III–V, and Figures 4–6. From the results of GA, SA and TS, we can see that they all have the potential to find the global optimum if their parameters are correctly configured and the computational cost is sufficient. GA demands that the population and the number of generations

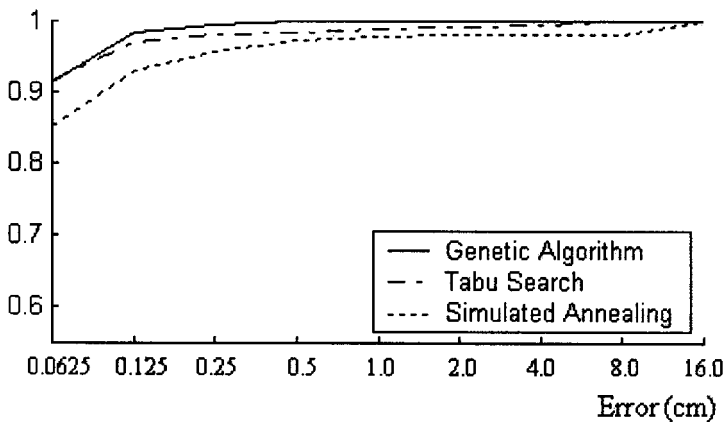


FIGURE 5 The three algorithms' probabilistic distribution function of the localization error requirements in Case II.

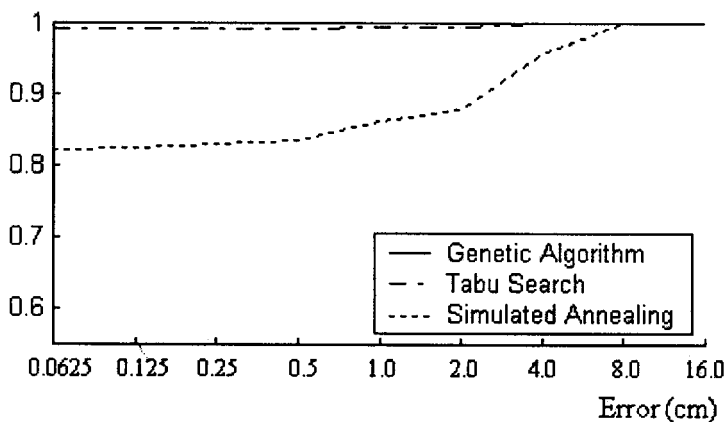


FIGURE 6 The three algorithms' probabilistic distribution function of the localization error in Case III.

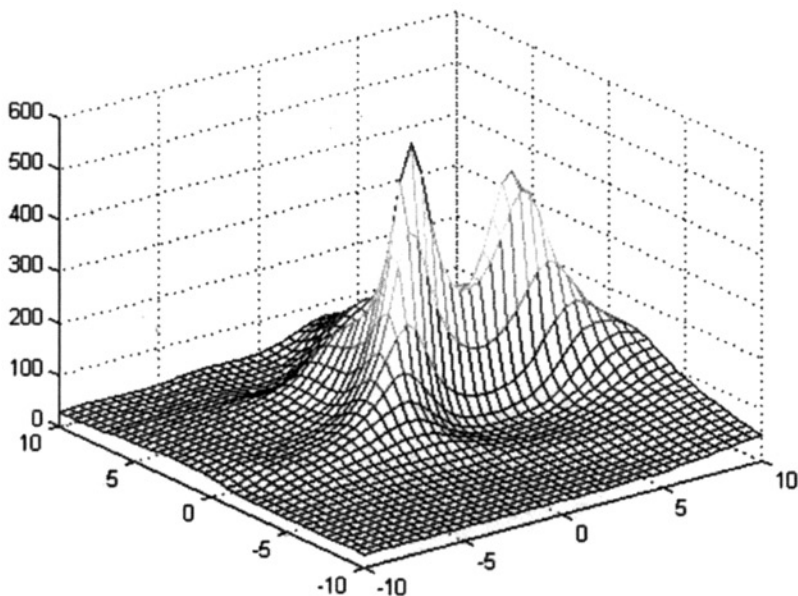
is large enough. SA requires the decrease of temperature is slow enough and the length of Markov chain is large enough. TS requires the iteration number to be large enough. However, in practice, the resource of computation is limited. In order to compare the performance of them, we limited the computational cost and the average computational cost is the same among the three algorithms.

Our experimental results show that our GA performs best in this problem. Under all three cases, the computational errors of GA are the smallest. Moreover, its value of probabilistic distribution function of localization error is the highest. This proves that our implementation of GA is very efficient and effective to localize the dipole, since the population in our algorithm is rather small compared with the previous implementation of GA to this problem [25]. We think that this results from the local search feature of our algorithm. It also illustrates that the strategy of hybridizing conventional GA with local search is very suitable to this kind of MEG source localization problem. Local search will make the solution close up to the global optimum rapidly and reduce the computational cost.

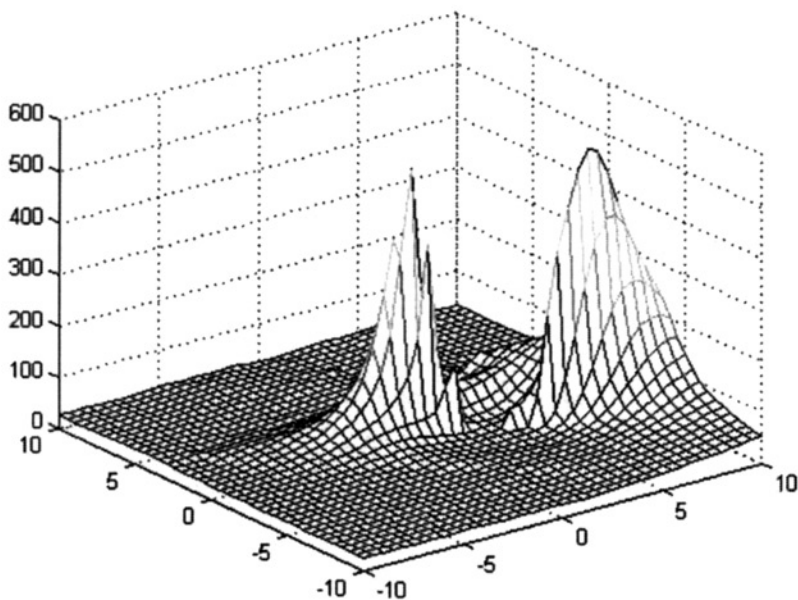
TS also performs very well. It ranks second among the three algorithms. Under all three cases, the computational errors of TS and its value of probabilistic distribution function of localization error are very similar to that of GA. So given the suitable search strategy and enough number of iterations, TS performs also very good to this problem. Moreover, TS is a flexible framework and its performance greatly depends on the strategy it adopts. Using the suitable search strategy will make the algorithm very efficient. Besides, new better strategies can be easily adopted to solve this problem.

SA can also find the global minimum in most cases if we use the suitable annealing schedule and a long enough Markov chain. But SA performs worst among GA, TS and SA, and it seems to be easier to be trapped into the local optima. Under all cases, the computational errors of SA are relatively large. This is probably because SA is a stochastic search method. Each step in SA is randomly produced.

Moreover, from Figures 4–6 we can see the different performance of GA, TS and SA under the three different cases. GA and TS are very similar with each other in that they both do best in Case III, and they perform better in Case I than in Case II. This similarity is probably because of the same local search strategy they use. We can see the reason for the different performance in the three cases in Figures 7–9, in which the fitness value near the global optimum is shown. Since in each case there are 3 dipoles to be localized, which is 9 variables in all, and we can only see the change of cost function with 2 variables in a three-dimension

FIGURE 7 Fitness function f in Case I.

figure, we must find the variables that their change is more likely to make the fitness function to trap into a local optimum. We can do this by analyzing the result in Tables III–V. For example, the effect of dipole 2's localization is worst in Case II, so we use the x and y values of dipole 2 as variables to draw this figure. Among the three cases, the cost function of Case III has the best shape, which means the number of local optima is small and the shape of cost

FIGURE 8 Fitness function f in Case II.

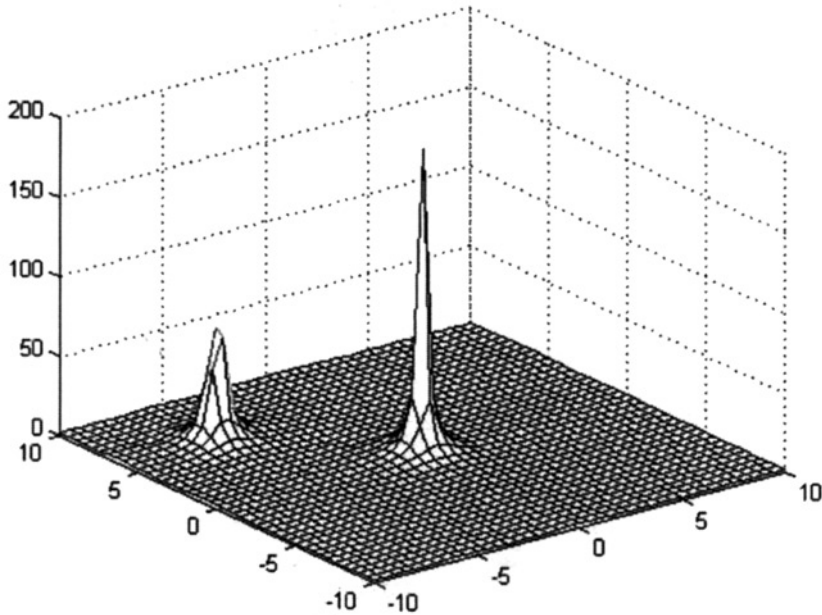


FIGURE 9 Fitness function f in Case III.

function is smooth. Comparing with Case III, the cost function in Case I and II is not so good, so we think this is the reason why GA and TS perform best in Case III, and we can logically get the result that the shape of the cost function in Case I is better than in Case II.

As for SA, there is a relatively large difference between its performance and the GA and TS's. Its ability to find the global optimum precisely is not so good. The performance of SA in Case III is especially the worst, we assume this is due to the overlapping of the time courses of the dipoles and the dipoles' nearer distance, but in GA and TS, the local search method plays an important role in the successfully localization of dipoles.

In sum, we find that our genetic local search algorithm is most effective compared with simulated annealing and tabu search when their computational cost is the same. Genetic local search algorithm is effective to find the global optimum and have a reasonable computational cost. It can be an alternative to the prevalent method for solving the MEG source localization problem. And tabu search is also a very good method for this problem. Simulated annealing has also the potential to give the correct estimation but it's performance is not so good as those of Genetic local search algorithm and tabu search algorithm. We think that maybe hybridizing SA with another heuristic will improve the efficiency of SA where future study is needed.

Evaluate the Dipole Moment and Dipole Time Courses After we have estimated the dipole location L , the gain matrix G can be calculated using the forward model. Q can be obtained using $Q = G^*B$. Then the moment and time courses of each dipole can be easily obtained by finding the best rank one approximation for each dipole. This has been described in detail in Mosher *et al.* [19]. In this paper, we give one example to calculate the dipole time courses using a correct estimation of L obtained in GA. The estimated time courses are displayed in (d), (e) and (f) in Figure 3.

6 CONCLUSIONS

Measurements of the magnetoencephalogram (MEG) provides unique insights into the dynamic behavior of the human brain as they are able to follow changes in neural activity on a millisecond time-scale. A central problem in the interpretation of MEG measurement data from evoked response experiments is the localization of the neural current dipoles. It is well-known that the problem of MEG source localization can be cast as an optimization problem. So far, there have been many works in which various ready optimization methods were adopted for the source localization. It is interesting to compare the performance of all existing approaches to this problem and find out which one is more suitable for source localization. In this paper, we compare the performance of three typical and widely used optimization techniques via using it to this specific MEG source localization problem. We first introduce a hybrid algorithm by combining genetic and local search algorithms to overcome some disadvantages of the conventional genetic algorithms. Then we make an attempt to apply tabu search, a widely used optimization method in combinatorial optimization and discrete mathematics, to source localization. To the best of our knowledge, this is the first attempt in the literature to apply tabu search to MEG source localization and compare simultaneously the performance of three typical global optimization methods. The computer simulation results show that our local genetic algorithm is the most effective approach to dipole location compared with simulated annealing and traditional tabu search. However, when the local search is incorporated into tabu search, the performance of tabu search to find the optimum can be dramatically improved. In fact, it is almost as good as that of the genetic algorithm. So tabu search is also a good choice for EEG/MEG source localization because it has the following advantages: (1) TS avoids entrapment in local minima and continues the search to give a near-optimal final solution; (2) TS is very general and conceptually much simpler than either simulated annealing (SA) or genetic algorithm (GA); (3) TS has no special space requirement and is very easy to implement (the entire procedure only occupies a few lines of code). Our method is also suitable for EEG source localization. In future we will consider the parallel genetic algorithm for the EEG/MEG source localization to improve the accuracy and speed.

Acknowledgement

This work was partially supported by Hundred Talents Programs of the Chinese Academy of Sciences, the Natural Science Foundation of China, Grant No. 697908001, and State Commission of Science and Technology of China, Grant No. G1998030503.

References

- [1] Aarts, E. H. L. and Korst, J. H. M. (1989). *Simulated Annealing and Boltzmann Machines*, John Wiley and Sons.
- [2] Achim, A., Richer, F. and Saint-Hilaire, J. (1988). Methods for separating temporally overlapping sources of neuroelectric data. *Brain Topograph*, **1**(1), 22–28.
- [3] Gerson, J., Cardenas, J. V. A. and Fein, G. (1994). Equivalent dipole parameter estimation using simulated annealing. *Electroenceph. Clin. Neurophysiol.*, **92**, 161–168.
- [4] Gloub, G. H. and van Loan, C. F. (1989). *Matrix Computations*, 2nd ed. The Johns Hopkins University Press, Baltimore, MD.
- [5] Gloub, G. H. and Pereyra, V. (1973). The differentiation of pseudo-inverses and nonlinear least squares problems whose variables separate. *SIAM J. Numerical Anal.*, **10**, 413–432.
- [6] Glover, F. (1993). In: Reeves, E. R. (Ed.), *Tabu Search, in Modern Heuristic Techniques for Combinatorial Problems*, John Wiley & Sons, Inc.
- [7] Goldberg, D. E. (1989). *Genetic Algorithms in Search, Optimization, and Machine Learning*. Addison-Wesley, Reading, Mass.

- [8] Guttman, I. and Pereyra, V. (1973). Least squares estimation for a class of nonlinear models. *Technometrics*, **15**, 209–218.
- [9] Hämäläinen, M., Hari, R., Ilmoniemi, P. J., Uutila, J. K. and Lounasmaa, O. V. (1993). Magnetoencephalography theory, instrumentation, and applications to noninvasive studies of the working human brain. *Rev. Mod. Phys.*, **65**(2), 413–497.
- [10] Haneishi, H., Ohyama, N., Sekihara, K. and Honda, T. (1994). Multiple current dipole estimation using simulated annealing. *IEEE Trans. Biomed. Eng.*, **41**(11), 1004–1009.
- [11] Ignizio, J. P. (1976). *Goal Programming and Extensions*, Lexington Books.
- [12] Jiang, T. (1999). A tabu search approach to optimal structuring element extraction for MST-based shapes description. *International J. Computer Mathematics*, **71**(3–4), 437–445.
- [13] Jiang, T. (1999). Geometric primitive extraction by the combination of Tabu search and subpixel accuracy. *J. Computer Science and Technology*, **14**(1), 74–80.
- [14] Khosla, D., Singh, M. and Don, M. (1997). Spatio-temporal EEG source localization using simulated annealing. *IEEE Trans. Biomed. Eng.*, **44**(11), 1075–1091.
- [15] Kirkpatrick, S., Gelatt, C. D. J. and Vecchi, M. P. (1983). Optimization by simulated annealing. *Science*, **220**(4598), 671–680.
- [16] Li, X., Jiang, T. and Evans, D. J. (2000). Medical image reconstruction using a multi-objective genetic local search algorithm. *International J. Computer Math.*, **74**(3), 301–314.
- [17] Metropolis, N., Rosenbluth, A., Rosenbluth, M., Teller, M. and Teller, E. (1953). Equations of state calculations by fast computing machines. *J. Chem. Phys.*, **21**, 1087–1092.
- [18] Marquardt, D. W. (1963). An algorithm for least-squares estimation of nonlinear parameters. *J. Soc. Indust. Appl. Math.*, **11**, 431–441.
- [19] Mosher, J., Leahy, R. and Lewis, P. (1992). Multiple dipole modeling and localization from spatio-temporal MEG data. *IEEE Trans. Biomed. Eng.*, **39**, 541–557.
- [20] Nelder, J. and Mead, R. (1965). A simplex method for function minimization. *Comput. J.*, **4**, 308–313.
- [21] Sarvas, J. (1987). Basic mathematical and electromagnetic concepts of the biomagnetic inverse problem. *Phys. Med. Biol.*, **32**(1), 11–12.
- [22] Scherg, M. (1989). Fundamentals of dipole source potential analysis. In: Hoke, M., *et al.* (Eds.), *Auditory Evoked Magnetic Fields and Potentials*, Vol. 6. Adv Audiol. Basel, Karger.
- [23] Sekihara, K., Haneishi, H. and Ohyama, N. (1992). Details of simulated annealing algorithm to estimate parameters of multiple current dipoles using biomagnetic data. *IEEE. Trans. Med. Imag.*, **11**(2), 293–299.
- [24] Sekihara, K. and Ohyama, N. (1990). Parameter estimation for *in vivo* magnetic resonance spectroscopy (MRS) using simulated annealing. *Magn. Reson. Med.*, **13**, 332–339.
- [25] Uutela, K., Hämäläinen, M. and Salmelin, R. (1998). Global optimization in the localization of neuromagnetic sources. *IEEE Trans. Biomed. Eng.*, **45**(6), 716–723.