FOI-R--0837--SE
March 2003
ISSN 1650-1942
Scientific report

Mattias Karlsson

Comparison of local and global search algorithms for inverse electromagnetic problems

Systems Technology
SE-172 90 Stockholm

SWEDISH DEFENCE RESEARCH AGENCY

Systems Technology
SE-172 90 Stockholm

FOI-R--0837--SE

March 2003

ISSN 1650-1942

Scientific report

Mattias Karlsson

Comparison of local and global search algorithms for inverse electromagnetic problems

Issuing organization	Report number, ISRN	Report type
FOI – Swedish Defence Research Agency Systems Technology SE-172 90 Stockholm	FOI-R--0837--SE	Scientific report
	Research area code 4. C4ISR	
	Month year March 2003	**Project no.** E6051
	Customers code 5. Commissioned Research	
	Sub area code 43 Underwater Sensors	

Author/s (editor/s)	Project manager
Mattias Karlsson	Peter Krylstedt
	Approved by Monica Dahlén
	Sponsoring agency Swedish Armed Forces
	Scientifically and technically responsible Sven Ivansson

Report title

Comparison of local and global search algorithms for inverse electromagnetic problems

Abstract (not more than 200 words)

Inverse problems of electromagnetic sounding often aim at extracting a one-dimensional piecewise constant conductivity profile of the marine environment from electromagnetic field measurements. Generally, solving inverse problems like this, local Newton like search methods have been shown to be sufficient, as long as the parameter search space is reasonably simple. In this work a performance comparison between a local algorithm and two global methods, a genetic algorithm and a simulated annealing, is made for inverse electromagnetic problems. The comparison is based on both simulated data and experimental results from field trials in the Stockholm archipelago. All results indicate that the one dimensional inverse problems are simple enough to be solved by a local algorithm, which performs better than the global methods. For more complicated searches, involving e.g. multidimensional environment models, the global algorithms will be more competitive.

Keywords

Inversion, Levenberg-Marquardt method, genetic algorithm, simulated annealing

Further bibliographic information	Language English
ISSN 1650-1942	**Pages** 25 p.
	Price acc. to pricelist

Utgivare	Rapportnummer, ISRN	Klassificering
Totalförsvarets Forskningsinstitut - FOI Systemteknik 172 90 Stockholm	FOI-R--0837--SE	Vetenskaplig rapport
	Forskningsområde 4. Spaning och ledning	
	Månad, år Mars 2003	**Projektnummer** E6051
	Verksamhetsgren 5. Uppdragsfinansierad verksamhet	
	Delområde 43 Undervattenssensorer	

Författare/redaktör	Projektledare
Mattias Karlsson	Peter Krylstedt
	Godkänd av Monica Dahlén
	Uppdragsgivare/kundbeteckning FM
	Tekniskt och/eller vetenskapligt ansvarig Sven Ivansson

Rapportens titel (i översättning)

Jämförelse mellan lokala och globala sökalgoritmer för tillämpningar på inversa elektromagnetiska problem

Sammanfattning (högst 200 ord)

Inversa elektromagnetiska problem går ofta ut på att beräkna en endimensionell styckvis konstant konduktivitetsprofil av den omgivande undervattensmiljön från elektromagnetiska fältmätningar. Allmänt när inversa problem liknande detta ska lösas, har lokala Newtonliknande sökmetoder visat sig tillräckliga om bara variationerna i det undersökta parameterrummet inte är alltför stora. I detta arbete görs en jämförelse mellan en lokal algoritm och två globala sökmetoder, en genetisk algoritm och en sk simulated annealing, för inversa elektromagnetiska problem. Jämförelsen är baserad på både simulerade data och experimentella resultat från fältförsök i Stockholms skärgård. Alla resultat indikerar att de endimensionella inversa problemen är tillräckligt enkla för att lösas med den lokala algoritmen som fungerade bättre än de global metoderna. För mera komplicerade problem, tex innehållande flerdimensionella miljömodeller, kommer dock de globala metoderna att vara mera konkurrenskraftiga.

Nyckelord

Inversion, Levenberg-Marquardt metod, genetisk algoritm, simulated annealing

Övriga bibliografiska uppgifter	Språk Engelska
ISSN 1650-1942	**Antal sidor:** 25 s.
Distribution enligt missiv	**Pris: Enligt prislista**

Contents

1 Introduction **2**

2 Search Algorithms **3**
2.1 The Levenberg-Marquardt algorithm 3
2.2 The genetic algorithm 4
2.3 Simulated annealing 5

3 Performance for electromagnetic inverse problems **6**
3.1 Initial example 6
3.2 Electromagnetic problems with simulated data 9
3.3 Electromagnetic problems with experimental data 15

4 Discussion **20**

A Appendix **21**
A.1 Detailed description of the applied genetic algorithm 21
A.2 Detailed description of the Very Fast Simulated Reannealing 22

1 Introduction

Most of the remote sensing methods in marine electromagnetics are focused on very or extremely low frequency fields. Due to the highly conductive seawater this is the frequency range of largest importance. The measured fields are apart from artificial sources or scatterers mainly determined by the conductivity profile of the surrounding marine environment. Inverse problems of electromagnetic sounding [1] therefore often aim at recovering a one-dimensional conductivity profile from electromagnetic field measurements. Based on these profiles accurate field predictions from artificial sources can be calculated. It is important to describe the surrounding environment well, since an inaccurate description will have a strong impact on the electromagnetic field predictions in the area. An insufficient environment model is likely to cause major errors when calculating variables like detection ranges.

To extract the most appropriate model for a general inverse problem, local Newton-like optimization methods are often applied. These methods have shown to be sufficient, at least as long as the parameter search space is reasonably simple. If the search space only contains a few local minima the local method will find the optimal model if it is just restarted a number of times. For more complicated parameter spaces however, the local optimization methods may get stuck in local minima too often. Obviously, this will make the problems very time consuming to solve, and a need for more efficient search algorithms arises. For example in the inverse acoustic problems, which are similar to the inverse electromagnetic problems, global search methods like the genetic algorithms [2] and hybrid methods [3], have been shown good to apply. Further examples of this can be seen in [4], [5]. For the electromagnetic problem there are many examples of inversions with local search methods [6], [7], [8]. However, there are not so many studies which apply global search algorithms. Furthermore, the electromagnetic inverse problems which have been studied with global methods, see e.g. [10], most often involve structures significantly different from those in our problem. It therefore seems motivated to explore if the global methods, which worked well for the related acoustic inverse problem, also can be successfully applied for the inverse electromagnetic problem.

In this study the restarted local Levenberg-Marquardt method used in [8] is compared to two global methods, a genetic algorithm and a simulated annealing method, for several inverse electromagnetic problems. The genetic algorithms have been shown to be superior to other global search methods, like simulated annealing [11] for certain applications. However, there are also studies which show the simulated annealing to be better than the genetic algorithms [12]. It seems dependent on the type of the studied problem which search method that is preferable. Both these methods are therefore included in this study. Obviously there are many more global search methods, but it would be an extensive task to try to find the most

suited method of them all. This study is therefore limited to a comparison between the local Levenberg-Marquardt method, a genetic algorithm, and a simulated annealing method.

Normally in studies which compare different search algorithms the result is presented as how the quality of the best found model increases with the number of models evaluated. However, if a method is applied a great number of times a few representative examples of such a function must be selected. For methods whose performance strongly varies from time to time this could be difficult. Here the result is therefore shown as the probability that a method finds a model with a certain quality as a function of models evaluated. By using this way to present the results, the three mentioned methods are compared for several problems. Initially the methods are compared for a problem which only aims at illustrating the strength of the global methods. Later follows a comparison for some electromagnetic problems with simulated data. All these problems are set up to resemble the situation at a ranging station in the Stockholm archipelago. Finally the methods are compared for real experimental input data from a sea trial.

2 Search Algorithms

The studied electromagnetic environment problem is a typical inverse problem in which a set of parameters shall be optimized to minimize an objective function. This function can also be called the cost function or the fitness value. To find the parameter values which correspond to the optimal objective function value there are numerous methods. Here the three well known algorithms used in this study are briefly described. More detailed descriptions of the global algorithm and the simulated annealing method are found in the appendix.

2.1 The Levenberg-Marquardt algorithm

The Levenberg-Marquardt based algorithm [8], [17] is a Newton like search method which uses the derivative of the objective function to find a local minimum. In contrast to the global methods this will make it impossible for the method to escape from a local minimum. In order to find the global optimum the method is therefore restarted a number of times. It would be sufficient to restart the method with a uniformly distributed starting point. However in [9] it is indicated that for the inverse electromagnetic problems another generation of the starting points is more efficient. The idea of the algorithm is to generate starting points far from earlier starting points, and earlier found local minima. Each parameter value of the new starting point is therefore randomly chosen in the largest interval between the values of that parameter for earlier starting points and earlier found local minima. If

the more distant starting points are more likely to lead to a new minimum this will increase the search speed.

2.2 The genetic algorithm

The genetic algorithm is a search algorithm inspired by the natural evolution. A set of models, referred to as a population, is used to create new models, which are normally called children. Similar to nature the population becomes too big and the weakest members are eliminated. The idea is to repeat this natural selection procedure until the population includes a model which is a good solution to the studied problem.

There are a number of different ways to construct a genetic algorithm. Similar for all algorithms though is that the parameter values, which describe the models, are limited by bounds, based on prior knowledge. Within these bounds the parameters are often discretized. There are also methods in which the parameter can take any value between the bounds, but a discretization is definitely more common. The discretization steps are always chosen to give 2^{n_i}, $n_i \in N$, possible values, where n_i is a number which corresponds to the i:th parameter's required sensitivity. A binary bit string with n_i bits is then enough to encode the value of each parameter. All models can then be described by long bit strings containing different parts corresponding to each parameter. These bit strings are referred to as chromosomes. In this study we use a genetic algorithm which is based on GENITOR [18], a method which applies the described chromosome like data structure. Our genetic algorithm has shown successful in acoustic problems similar to the electromagnetic problems in this study [16]. The basic idea is, as mentioned in [4], [5], [14]:

1. Randomly generate an initial model population, encoding each model to a binary string, a chromosome. Compute the fitness value and rank each chromosome in the population.

2. Randomly select two parent chromosomes using a selection probability proportional to the rank in the current population.

3. Generate two children which inherit their characteristics from their parents.

4. Randomly delete one of the children and evaluate the other child's fitness value, and rank in the population.

5. Eliminate the chromosome with the worst fitness value in the population.

6. Stop if the maximum number of allowed evaluations is reached, otherwise go to 2.

The method is designed not to get stuck in a local optimum, However, it can happen that all the chromosomes in a population describe models close to each other in the parameter space, all gathered around a local optimum. To avoid such a scenario, several populations can be randomly initiated, and developed according to the following, as also described in [16].

1. Develop each population for a certain time according to the scheme (2 to 6) described above.
2. Copy the best chromosomes in each population to the other populations.
3. Eliminate the weakest individuals in each population.
4. Stop if the maximum number of allowed evaluations is reached, otherwise go to 1.

These schemes define the structure of our genetic algorithm. However, to get a good result it is important to choose appropriate values on population sizes etc. To understand all the choices that have to be done, a closer description of the different parts in our genetic algorithm is required. This is found in the appendix.

2.3 Simulated annealing

In contrast to the genetic algorithms, simulated annealing only has one starting point [12]. This is similar to the local optimizing methods, but for simulated annealing the derivative of the objective function is not calculated. Instead the next point is randomly chosen from a probability distribution which continuously decreases with the distance from the starting point. An additional difference from the local methods is that the new point is not necessarily taken as a new starting point. The chance that it is taken is instead related to the difference in objective function values between the points. If the new point corresponds to a lower objective function value it is taken, but it can also be taken when it is higher than the original. Hence, the method is capable of climbing out of local optima. This procedure is then repeated but the probability distributions are changed. As the search goes on the distribution which controls the pick of the next point, gets narrower and new points far away from the starting point are less likely. Furthermore the probability that a point with a worse objective function value is accepted decreases with time. The idea is that the starting point will jump around over the entire search space in the beginning, avoiding getting stuck in local minima, until it finally finds the global minimum in the end. This algorithm has got it name from the roughly analogous physical process of heating and then slowly cooling a material, to obtain a strong crystalline structure. Just as in the physical process a system temperature controls the search algorithm, which is described in detail in the appendix.

3 Performance for electromagnetic inverse problems

In this chapter the results of the comparison between the three methods are shown. As previously mentioned there are many earlier studies in which different search algorithms have been compared. For all of these studies the compared methods are firstly each applied to a certain problem a number of times. Then there is however no obvious way of how to compare the results. Naturally it is both the quality of the produced solutions and the time required to find them that are important. One method may however work extremely well every hundred time it is used, but otherwise it may only give decent solutions. It will then be dependent on the studied problem if that method is better than an algorithm which mostly often extracts parameter values corresponding to a quite low objective function value, but which almost never performs extremely well. In most of the studies the result is shown as how the objective function value corresponding to the best found model decreases with the number of evaluated models. An example of this can be seen in figure 1. Since it is only possible to show a few functions for each method the authors have to select a few representative examples for each algorithm. Even though it sometimes can be done, it seems almost impossible to do if the evaluated method has large variations in the results. Here we therefore introduce a way to compare the methods which eliminates the need for the human capacity to select representative results. As before each of the compared methods are applied to a problem a great number of times. Based on the results, the probability that an algorithm will find a model corresponding to a objective function value lower than a bound C_{sol} can be estimated as a function of the number of evaluated models. The bound C_{sol} are varied to investigate which methods that find decent solutions quickly, and which methods that find the extremely good models.

3.1 Initial example

To illustrate the strength of the global search algorithms the following objective function is studied

$$F(x_1, x_2) = -0.5 + \frac{\sin^2(\sqrt{x_1^2 + x_2^2}) - 0.5}{(1 + 0.001(x_1^2 + x_2^2))^2}, \tag{1}$$

where x_1 and x_2 are two parameters in $[-100, 100]$. As shown in figure 2 the objective function has many local minima around the global minimum at $[0, 0]$. A locally restarted optimization method will only spin around and will clearly not work very well for this problem. For the global search algorithms however, this problem is quickly solved. A genetic algorithm with a single population is sufficient to solve the problem. The result is shown in figure 3.

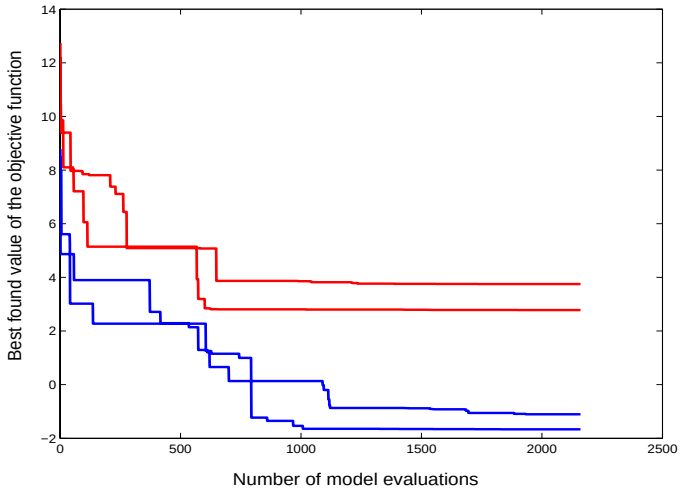

Figure 1: An example of the conventional way to compare two methods.

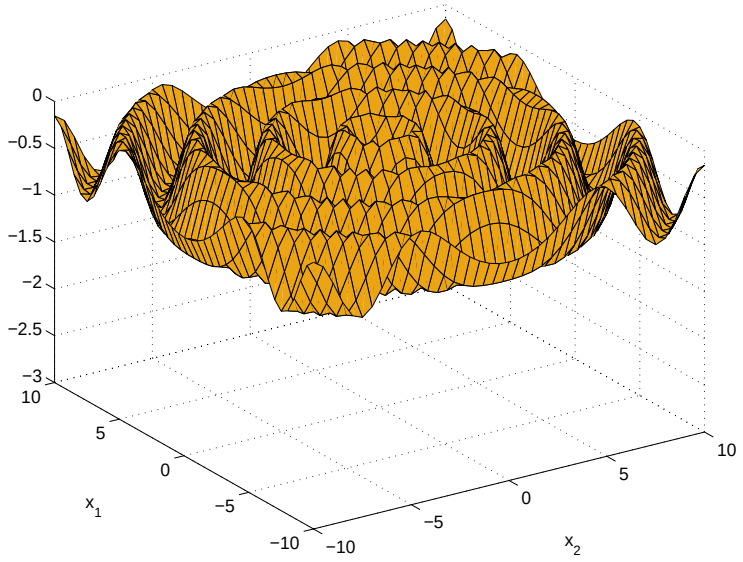

Figure 2: The objective function in equation (1) as a function of x_1 and x_2.

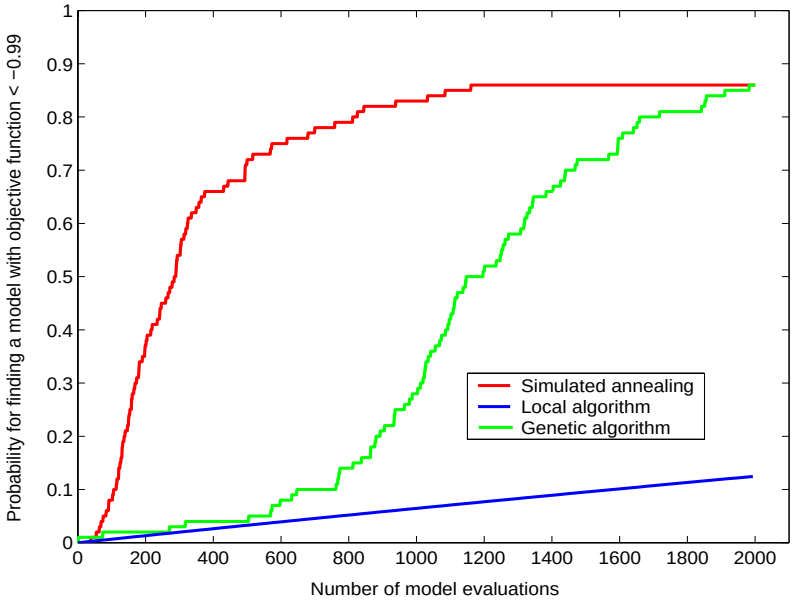

Figure 3: The probability for the search methods to find a model with a corresponding objective function value below -0.99 is plotted against the number of evaluated models. The search space was limited to $x_1 \in [-100, 100]$ and $x_2 \in [-100, 100]$. The parameter values for the genetic algorithm were $N_{evol} = 2750$, $N_{popsize} = 250$, $N_{pop} = 1$, $N_{emig} = 0$, $N_{migr} = 0$, $k_g = 3$, $c_{m1} = 0.15$, $c_{m2} = 0.6$, and uniform crossover was applied.

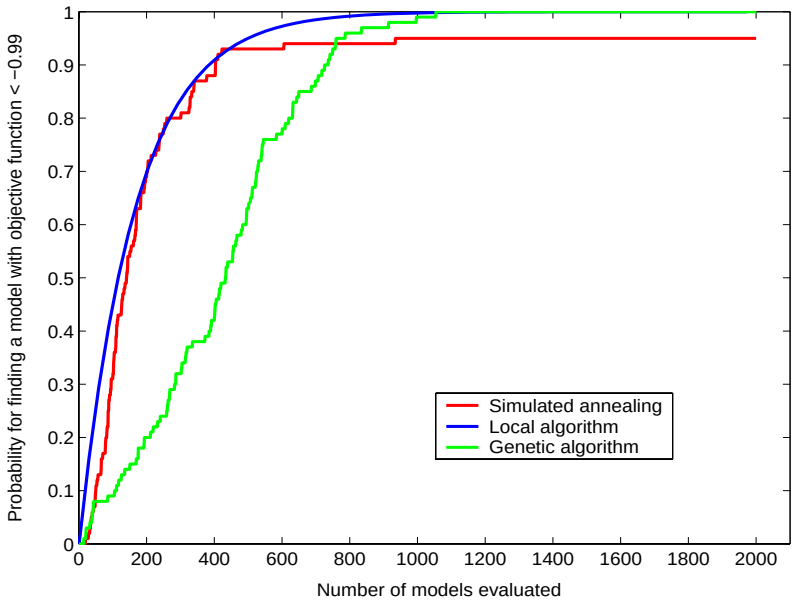

Figure 4: The probability for the search methods to find a model with a corresponding objective function value below -0.99 is plotted against the number of evaluated models. The search space was limited to $x_1 \in [-10, 10]$ and $x_2 \in [-10, 10]$ The parameter values for the genetic algorithm were $N_{evol} = 2750$, $N_{popsize} = 250$, $N_{pop} = 1$, $N_{emig} = 0$, $N_{migr} = 0$, $k_g = 3$, $c_{m1} = 0.15$, $c_{m2} = 0.6$, and uniform crossover was applied.

If the search space is limited to $x_1 \in [-10, 10]$ and $x_2 \in [-10, 10]$ the problem becomes significantly simplified. As seen in figure 4, the local algorithm then works better than the global methods for this problem. This is typical for the search algorithms. The global search methods tend to be more competitive for complicated search spaces. For simple objective functions with a limited number of minima the local methods are often preferable. Hence, the choice between applying a local or a global method strongly depends on the studied problem.

3.2 Electromagnetic problems with simulated data

Many electromagnetic inverse problems aim at describing the surrounding environment with a one dimensional conductivity profile, like the one in figure 5. To estimate a conductivity model a controlled current source and one or several sensors are used. The source is either used at a fixed position transmitting signals of several frequencies, or it is towed while sending only one frequency. Normally, this is referred to as a frequency sounding problem and a space sounding problem, respectively. To evaluate a conductivity

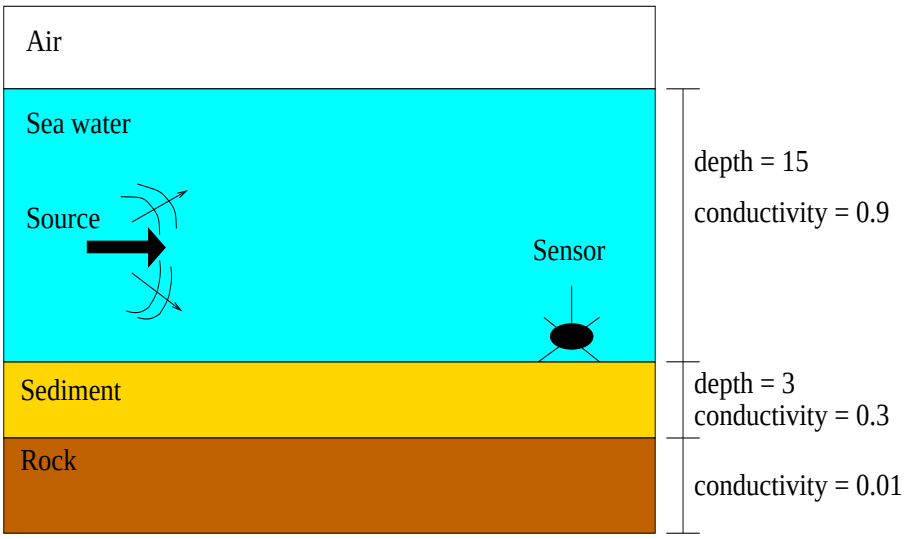

Figure 5: An example of a one dimensional conductivity profile of the environment. The thicknesses are in m, and the conductivities are in S/m.

model the electromagnetic wave propagation code NLAYER [15] is applied. Given a one dimensional environment the code can calculate the electric fields at the sensors. Hence the goal of the inverse problem is to extract the conductivity and the thickness for each layer in the model which will make the difference between the measured and calculated field as small as possible. The best model is defined as the one which minimizes the sum of the squares of the differences in field amplitudes, i.e.

$$F(x) = \sum_{j=1}^{n} \sum_{i=1}^{m} (r_i^j(x))^2$$

where $r_i^j(x)$ is the difference between the model and the reference field for the j:th sensor in the i:th direction. Hence the function $F(x)$ is the objective function.

By applying the program NLAYER it is also possible to create synthetic data for an environment with a one dimensional conductivity profile. This has been done for a frequency sounding problem. The studied configuration is very similar to a ranging station in the Stockholm archipelago [19]. Assume that the source is frequency limited and can not send signals with frequencies above 200 Hz. Furthermore, the source is fixed close to the bottom, which eliminates the possibility to use space sounding. Besides the source there are two sensors connected to the ranging station. Both of them are also close to the bottom. The problem consists of determining a one dimensional conductivity profile to describe the environment, by using the

Depth	Conductivity
0-12	0.80
12-x_1	x_2
$x_1 - \infty$	x_3

Table 1: Set up for the three layer inversion problem in the ranging station environment. The reference data was simulated with $x_1 = 27$, $x_2 = 0.3$, and $x_3 = 0.004$. The transmitted frequencies were 0.5, 5, 37, 87, and 195 Hz. The source position was (100, 50, 10.5) and the positions of the sensors were (0, -12, 11) and (0,12,11).

amplitudes of the 3-dimensional E-field measured in the sensors, when 5 signals of different frequencies are sent. The configuration is shown in detail in table 1.

In figure 6 the probability for the methods to find a model which corresponds to an objective function value below 0.3 is shown as a function of the number of model evaluations. As seen, the local method is superior to the global methods which is expected. The problem only has three parameters and does not involve any noise. If no noise is added to the data there is of course possible to find the correct environment model. Here it is easy for a local method to extract the correct solution. In the simulations it is actually seen that the local algorithm found the correct solution for 99% of the random starting points. This strongly indicates very few local minima in the objective function. In figure 7 the objective function is plotted as a function of the parameters x_1 and x_2, for four different values of the parameter x_3. The objective function only seems to depend weakly on the parameter x_3. As suspected the objective function is smooth and does not have many local minima. The local search algorithm therefore easily finds the global minimum. Both of the global methods also approaches the correct solution but they are much slower. They both found decent solutions with objective function values below 0.3 rather quickly, but the search for the correct solution would in average take a long time. Hence, for this problem it is clearly better to apply a local method instead of a global.

The no noise scenario can be illustrative but it will obviously never occur in real life. A more realistic problem is therefore the same configuration but with noise added. For all of the applied frequencies in the previous example the background noise intensity has been measured at the station the configuration is set to resemble. By adding these background fields to the simulated data from NLAYER a new more interesting problem is created. However, from the results in figure 8, it can be seen that the methods perform similarly to the noiseless case. The local method does however not find models with objective function values as low as before.

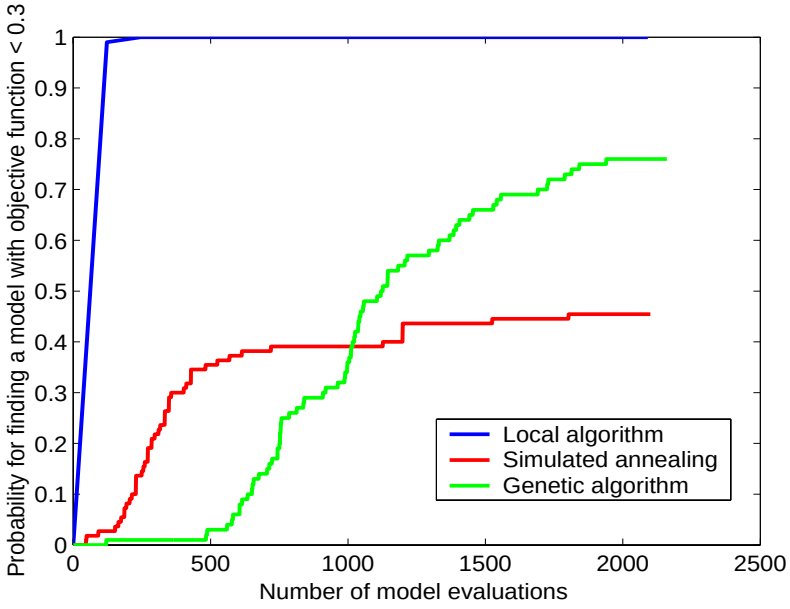

Figure 6: The probability that the search methods find a model with a corresponding objective function value below 0.3 is plotted against the number of evaluated models. The parameter values for the genetic algorithm were $N_{evol} = 100$, $N_{popsize} = 40$, $N_{pop} = 4$, $N_{emig} = 2$, $N_{migr} = 4$, $k_g = 3$, $c_{m1} = 0.15$, $c_{m2} = 0.6$, and uniform crossover was applied.

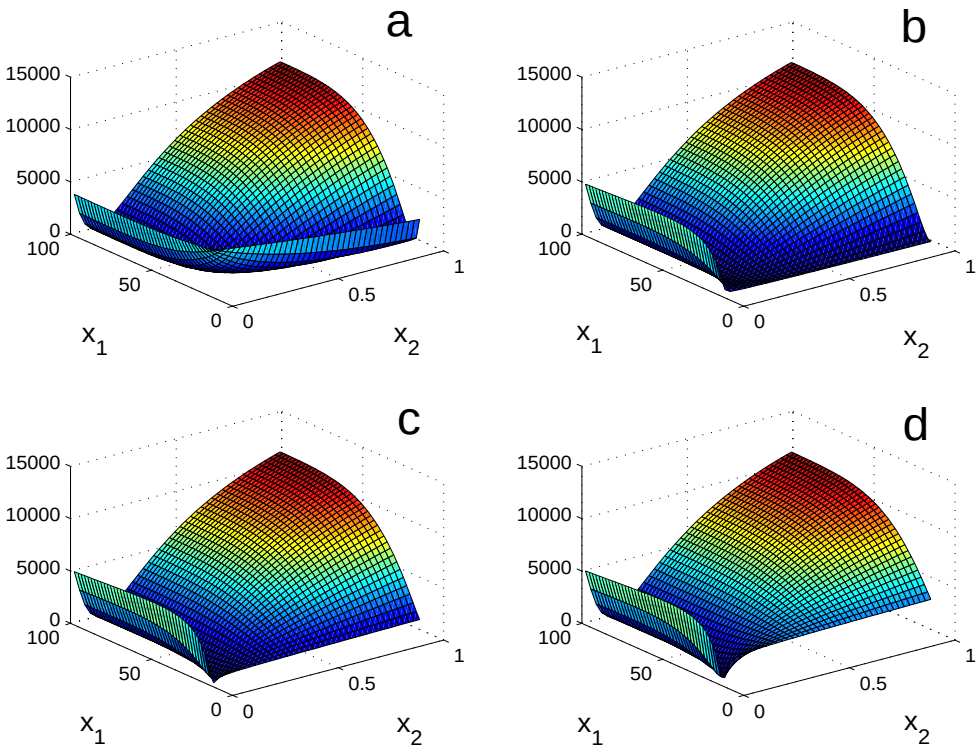

Figure 7: The objective function as a function of the parameters x_1 and x_2. The parameter x_3 is set to 0.004 (a), 0.1 (b), 0.2 (c), and 0.3 (d).

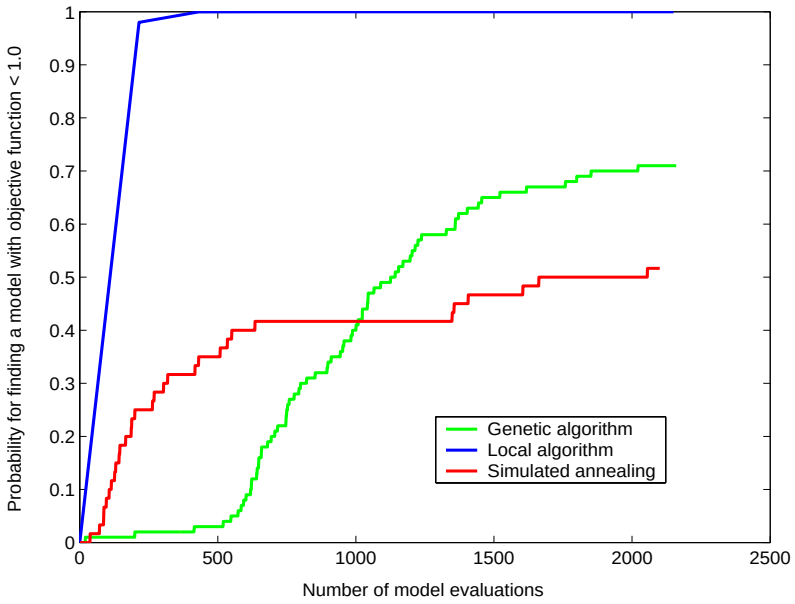

Figure 8: The probability for the search methods to find a model with a corresponding objective function value below 1 is plotted against the number of evaluated models. The parameter values for the genetic algorithm were $N_{evol} = 100$, $N_{popsize} = 40$, $N_{pop} = 4$, $N_{emig} = 2$, $N_{migr} = 4$, $k_g = 3$, $c_{m1} = 0.15$, $c_{m2} = 0.6$, and uniform crossover was applied.

The additional noise removes the possibility to find a correct solution and the best model will therefore correspond to a non zero objective function value. Compared to the global methods though, the local algorithm is still significantly faster in the search for decent solutions. From the simulations it can also be seen that the local method extracted the same model in 94 % of the restarts. This configuration corresponds to an objective function of 0.18. In an additional comparison of the ability of the methods to find solutions with even lower objective function values (e.g. less than 0.2) the local method would therefore come out even better.

As mentioned earlier the global methods are more competitive for more complicated problems. One problem that is definitely more difficult is the same ranging station but with a more detailed division of the conductivity layers. Assume that we describe the environment with a 10 layer configuration, which is shown in table 2, and 4 of the variables are tried to be estimated. The input data is as previously mentioned created by adding the components calculated in NLAYER to the measured noise levels at the station. From the results, shown in figure 9, the methods seem comparable. The simulated annealing is as fast as the local method in the beginning. If a decent model is not found in the 300 first model evaluations though, the

Depth	Conductivity
0-5	0.96
5-7	0.95
7-9	0.90
9-10	0.85
10-11	0.80
11-12	0.73
12-x_1	x_3
$x_1 - x_2$	x_4
$x_2 - \infty$	0.004

Table 2: Set up for the a 10 layer inversion problem in the ranging station environment. The reference data was simulated with $x_1 = 17$, $x_2 = 27$, $x_3 = 0.30$, and $x_4 = 0.10$. The transmitted frequencies were 0.5, 5, 37, 87, and 195 Hz. The source position was (100, 50, 10.5) and the positions of the sensors were (0, -12, 11) and (0,12,11).

possibilities to find such a model seems small. An alternative could therefore be to restart the algorithm after 300 model evaluations which would improve the results. The genetic algorithm works nearly as well as the local. Actually a more detailed study of the data shows that it is faster than the local method to find solutions with objective function values below 0.3. In the searches for really good solutions however, the local method is still better. The more detailed study also revealed that the objective function has numerous different local minima. Normally this would create significant problems for the local algorithm, but in this case many of the minima correspond to nearly the same objective function value. That is, it seems to exist many solutions that are approximately equally good. Therefore the local method still works well.

3.3 Electromagnetic problems with experimental data

All problems in the previous section were based on synthetic data from NLAYER, which assumes a one dimensional conductivity profile. The addition of measured noise to the data created a more relevant scenario but it is still not a real situation. For real experiments the conductivity profile will always be more complicated than the one dimensional profile used in NLAYER. Similar investigations as those in the previous chapter have therefore been performed also for experimental data from the Stockholm archipelago. Two experimental configurations were tested. The first one is a frequency sounding problem with one source and one sensor. The second configuration is a space sounding problem in which the source only transmitted signals with one frequency, but the signal was measured for seven

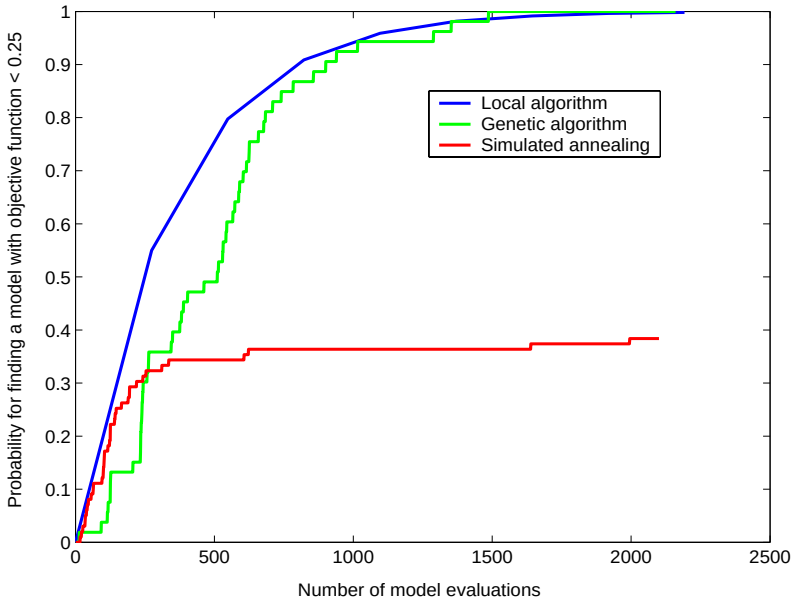

Figure 9: The probability for the search methods to find a model with a corresponding objective function value below 0.25 is plotted against the number of evaluated models. The parameter values for the genetic algorithm were $N_{evol} = 100$, $N_{popsize} = 40$, $N_{pop} = 4$, $N_{emig} = 2$, $N_{migr} = 4$, $k_g = 3$, $c_{m1} = 0.15$, $c_{m2} = 0.6$, and uniform crossover was applied.

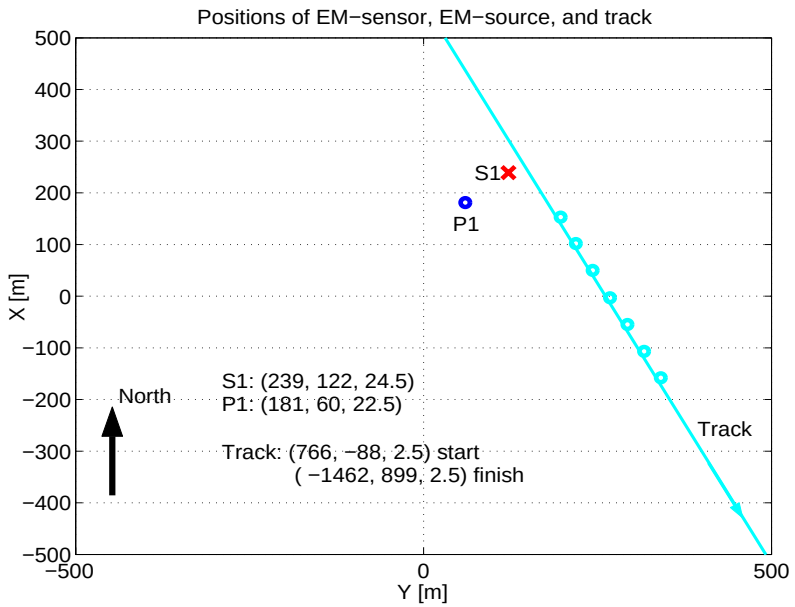

Figure 10: A map over the experimental set up. The source in the frequency sounding problem is referred to as P1 and the sensor in both problems is the one referred to as S1.

different source locations. The positions of the source and sensor for both cases are illustrated in figure 10.

In the first case the frequency sounding aimed at extracting three parameter values in a 9 layer model. The details for the problem are shown in table 3. As seen in figure 11 the local method worked well also for experimental data. The genetic algorithm also finds models which correspond to approximately the same objective function values, but it is slower than the local. For this problem the simulated annealing clearly performed worse than the other methods. A more detailed study of the data shows that the problem is rather simple since the local method converged to the same minimum for 62 % of the restarts.

In the second problem with the space sounding the goal was to describe the environment as well as possible with a two layer model. The set up is more closely described in table 4. In figure 12 the results of the searches are presented. The problem seems to be quite simple since all methods perform very well, and quickly find solutions with objective function values below 0.008. The value 0.008 is furthermore close to the best achieved objective function value for any found two layer model. Although all methods perform well the local method is the fastest, and for this application there does not seem to be any need for the global algorithms.

Depth	Conductivity
0-6.5	0.973
6.5-8.5	0.947
8.5-10.5	0.900
10.5-12.0	0.850
12.0-13.6	0.800
13.6-15.0	0.765
15.0-17.5	0.757
17.5-x_1	x_2
$x_1 - \infty$	x_3

Table 3: Assumed set up for the 9 layer frequency sounding inverse problem. The transmitted frequencies were 1, 23, 77, 177, 377, 577, 777, 979, 1270, 1570, 1871, 2172, 2472, 2772, 3374, and 3975 Hz.

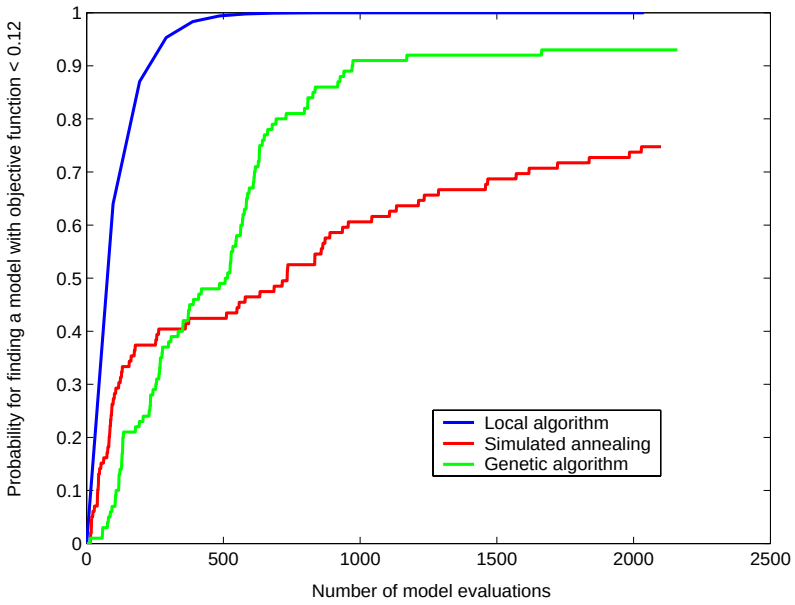

Figure 11: The probability for the search methods to find a model with a corresponding objective function value below 0.12 is plotted against the number of evaluated models.The parameter values for the genetic algorithm were $N_{evol} = 100$, $N_{popsize} = 40$, $N_{pop} = 4$, $N_{emig} = 2$, $N_{migr} = 4$, $k_g = 3$, $c_{m1} = 0.15$, $c_{m2} = 0.6$, and uniform crossover was applied.

Depth	Conductivity
0-x_1	x_2
$x_1 - \infty$	x_3

Table 4: Assumed set up for the two layer space sounding inverse problem. The transmitted frequency was 5 Hz.

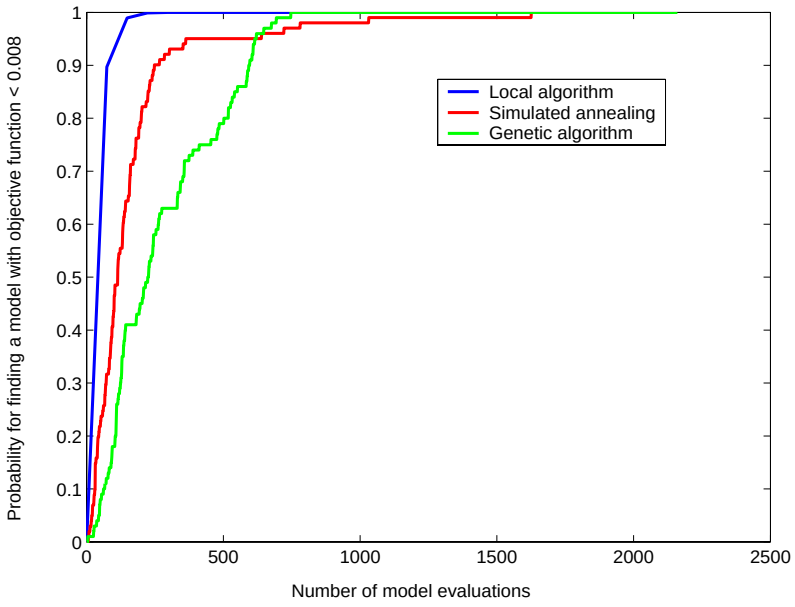

Figure 12: The probability for the search methods to find a model with a corresponding objective function value below 0.008 is plotted against the number of evaluated models. The parameter values for the genetic algorithm were $N_{evol} = 100$, $N_{popsize} = 40$, $N_{pop} = 4$, $N_{emig} = 2$, $N_{migr} = 4$, $k_g = 3$, $c_{m1} = 0.15$, $c_{m2} = 0.6$, and uniform crossover was applied.

4 Discussion

For almost all of the inverse electromagnetic problems studied so far the local method seems to be better than the global. Possibly the global methods could perform better if even more efficient parameter settings are found. The results from the simulated annealing show that it tends to get stuck in local minima too often which indicates that another annealing schedule for the system temperature could be preferable. Another possibility would be to combine the global algorithm with the local. The genetic algorithm often finds models close to the global minimum, and could therefore be used to extract the starting point for the local method. However, it is clear from the results that the inverse problems involving a one dimensional conductivity profile have rather simple objective functions. As shown in the examples in section 3.1 this normally makes the local search methods sufficient to apply. Hence, as long as the problems are this simple it does not seem to be necessary to replace the local algorithm with global methods. The only problem in which the methods were comparable was the 10 layer inversion with 4 unknown parameters. As mentioned earlier this problem created numerous local minima, but all corresponding to nearly the same objective function value. It seems like there could exist similar problems where one or at least only a few minima are clearly better than the others. This would create significantly problems for the local method which would get stuck too often on suboptimal models. Earlier studies [20],[21], have shown that there is often no meaning to apply too complicated divisions of the conductivity profile. The electromagnetic background, which will always exist, is often strong enough to make simple three or four layer models as efficient as the nine or ten layer models. Hence, the one dimensional inverse problems often only consist of a few parameters, and involve objective functions with a quite simple structures. Based on this assumption it can be concluded that the local methods probably will be sufficient for the one dimensional electromagnetic inverse problems, and there is no need for global methods like the genetic algorithms.

An important remark for future studies is that the conclusion is only made for the one dimensional inverse problems. Similar to the acoustic problems there is a need for E-field predictions in two and three dimensional environments. This will automatically also require extractions of multi dimensional conductivity profiles from data. In these problems the parameter search space will expand significantly compared to the one dimensional problems. It is then likely that the local methods will not be sufficient and that global methods like the genetic algorithms will be required. An important question for the future is to find out how complicated these inverse problems may be before a need for global search methods arises.

A Appendix

A.1 Detailed description of the applied genetic algorithm

Our genetic algorithm starts by randomly initiate N_{pop} populations with $N_{popsize}$ members. Each population is then evoluted N_{evol} times. These generation updates follow step 2-6 in the earlier showed scheme and are described in detail below. When all populations are updated the N_{emig} best individuals in each population are copied to the other populations. The least fitted chromosomes are then deleted until all populations have recovered their original size, $N_{popsize}$. Depending on the number of function evaluations which are to be done, this procedure is repeated N_{migr} number of times.

Selection of parents The first step in the generation update is to choose two chromosomes as parents. To imitate nature the most fit members of the populations should be the likeliest parents. However all chromosomes should have a chance to become a parent and the parents are therefore chosen with a probability related to their rank in the population. Two random, independent, uniformly distributed numbers (u_i), between 0 and 1, define the parents' index. The indicies are set to the smallest integers which are greater than or equal to the numbers s_i respectively,

$$s_i = N_{pop} - (u_i)^{(1/k_g)} \cdot N_{pop}, \quad i = 1, 2$$

where k_g is a natural number and N_{pop} is the number of chromosomes in the population. Here index 1 corresponds to the chromosome with the best fitness value, index 2 corresponds to the second best chromosome etc. If k_g is chosen to 1 every chromosome in the population will have the same probability to be chosen, but normally k_g is set higher, e.g. $k_g = 3$. This will make the choice of more fitted parents likelier.

Production of new chromosomes There are three standard methods to produce the children to the selected parents, one point crossover, two point crossover, and uniform crossover. In the one point crossover an index c_1 between one and the length of the chromosomes minus one are randomly chosen, where all indices are equally likely. The first child then inherits all bits corresponding to indicies from 1 to c_1 from the first parent and the rest of the chromosome is inherited from the second parent. For the second child it is vice versa. It inherits the beginning bits from the second parent and the last bits from the first parent. For the two-point crossover two indices are chosen (c_1 and c_2) and the first child inherit bits 1 to c_1 from parent one, bits $c_1 + 1$ to c_2 from parent two, and the last bits again from the first parent. For the second child it is vice versa similar to the one point crossover case. Finally, in the uniform crossover there is for each bit 50 percent chance

that it shall be inherited from parent one for the first child. If the first child inherits bit one from the first parent, the second child automatically inherits that bit from the second parent and so on.

Mutation One problem which often occurs in genetic algorithms is that all chromosomes in a population become almost identical. Parallel populations with chromosomes which migrate between them improve the results, but the problem still occurs. Since this limits the possibilities to escape local minima, mutation is often applied. All produced children are mutated with a certain probability which increases in relation to how much the children resemble their parents. The chance that a child is mutated is

$$p_{mutation} = 20 * e^{(-1.2 * c_{m1} * p_{diff})} + c_{m2}, \tag{2}$$

where c_{m1} and c_{m2} are constants and p_{diff} is the maximum percentage of bits the child differs from one of the parents. If a child chromosome gets mutated, each bit has a certain probability (half of the one in equation (2)) to get changed. Even though the mutation prevents the chromosomes in the population to be too similar, it is not sufficient to completely eliminate the problem. Further steps are therefore taken, such as no child which is a copy of an already existing chromosome is accepted in the population. Together these steps at least limit the risk of getting stuck in local optima.

A.2 Detailed description of the Very Fast Simulated Reannealing

Similar to the genetic algorithms there are many different simulated annealing methods. Here we have applied the Very Fast Simulated Reannealing (VFSR) [12], [13], which has been applied in areas such as combat analysis, finance, and neuroscience. This product includes software developed by Lester Ingber and other contributors. The presentation of the VFSR closely follows that of [12]. Simulated annealing generally includes three functional relationships.

1. $g(x)$: Probability density function of the parameter search space of D parameters $x = x_i; i = 1, D$.
2. $h(x)$: Probability density function for the acceptance of a new starting point given the just previous point.
3. $T(k)$: schedule of how the temperature T varies with the annealing time steps k. Both of the functions $g(x)$ and $h(x)$ are dependent on this temperature.

The function $g(x)$, which is the probability density function for the point which possibly could be the new starting point, is usually chosen to be Gaussian. An easy calculation however [12], show this approach to be rather slow.

If the Gaussian distribution is replaced by e.g. a D-dimensional Cauchy distribution, the method becomes faster. The fatter tail of the Cauchy distribution facilitate searches of many local minima which improves the result. A disadvantage though is that the distribution is symmetric and it is therefore not adjustable for each parameter. Also, there is no quick algorithm which produces random numbers from this distribution. All these issues are solved in the VFSR, for which a new probability density function is introduced. The distribution gives a faster search compared to the method which applies the Cauchy distribution. Furthermore it is easy to produce numbers from the distribution. All parameter values for the possibly new starting point in the VFSR are generated as

$$p_{k+1}^i = p_k^i + y^i(B_i - A_i),$$

where p_k is a vector with the values of the parameters, A_i and B_i are the limits for each parameter, and y^i is a random number in $[-1, 1]$. The function g is here defined as

$$g^i(y^i; T_i) = \frac{1}{2(|y^i| + T_i)\ln(1 + 1/T_i)},$$

where T_i is the temperature for parameter i. The functions $g^i(y^i; T_i)$ are repeatedly used until every parameter value has a value between its limits. The generated point p_{k+1} is then accepted as a new starting point, instead of p_k, with a probability

$$h(x) = \min\{\exp(-x/T_{cost}), 1\},$$

where $x = C(p_{k+1}) - C(p_k)$ and $C(p)$ is the objective function value for a point p. Hence, if the generated point corresponds to a lower objective function value compared to objective function value for the starting point, it will always be accepted. The cost temperature in the described acceptance probability density function varies as

$$T_{cost}(k_{cost}) = T_{0cost}\exp(-c_{cost}k_{cost}^{1/D}).$$

This is very similar to the annealing schedule for the temperatures T_i which are changed according to

$$T_i(k) = T_{0i}\exp(-c_i k_i^{1/D}),$$

where c_i is a constant which can be tuned to improve the search speed. The main difference between the temperatures is that the index k_{cost} is determined by the number of accepted points, instead of the number of generated points which are used to determine all indices k_i. Since the parameters often have different sensitivities the temperatures are then "reannealed". This

rescaling of the annealing times k_i is made to stretch out the ranges over which insensitive parameter values are being searched. A similar reanneling is also made for the cost temperature.

The above described VFSR algorithm can be tuned in many ways to improve the speed of the searches. Generally for simulated annealing methods, it can be guaranteed that the global optima will be found if the temperatures decrease slowly enough. In this study we have only tried to set different rates for this temperature decreasing, although other variations possibly could have given better results.

References

[1] Inverse problems, Vol. 16, No. 5, October 2000.

[2] P. Gerstoft, "Inversion of seismoacoustic data using genetic algorithms and a posteriori probability distribution", J. Acoust. Soc. Am. 95, pp. 770-782, 1994.

[3] M.R. Fallat and S.E. Dosso, "Geoacoustical inversion via local, global and hybrid algorithms", J. Acoust. Soc. Am. 105, pp. 3219-3230, 1999.

[4] V. Westerlin, "Multi-frequency inversion of syntetic transmission loss data using a genetic algorithm", J. of Comp. Acoust., Vol 6, Nos. 1 & 2, pp. 205-221, 1998.

[5] S. Ivansson, P. Moren, E. Parastrates, and V. Westerlin, "Hydroacoustical experiments for determination of sediment parameters", Oceans 94, vol. III, pp. 207-212, 1994.

[6] Inverse problems, Vol. 17, No. 6, December 2000.

[7] P. Krylstedt, J. Mattsson, and A. Timonov, "Inverse modelling in marine electromagnetics", IEEE Transactions on geoscience and remote sensing, submitted.

[8] P. Krylstedt, J. Mattsson, and A. Timonov, "A sequential approach to inverse modelling in marine electromagnetics: Recovering the sub bottom profile from measurements of the electromagnetical field", in the proceedings of MARELEC 2001, July 11-13, Stockholm.

[9] J. Schiöld, "Electromagnetic conductivity profile estimation by quasi-global restarting techniques of local Newton-like methods", FOI-R-0493-SE, 2002.

[10] E. Haber, U. M. Asher, and D. Oldenburg, "On optimization techniques for solving nonlinear inverse problems", Inverse problems, Vol. 17, No. 6, December 2000.

[11] J.A. Scales, M.L. Smith, and T.L. Fisher, "Global optimization methods for multimodal inverse problems", J. Comput. Phys., 103, pp. 258-268, 1992.

[12] L. Ingber and B. Rosen, "Genetic algorithms and very fast simulated reannealing: A comparison", Mathematical and Computer Modelling, 16(11), 87-100, 1992. The VFSR code is available at http://www.ingber.com/.

[13] A MATLAB Gateway Routine for Lester Ingber's Adaptive Simulated Annealing (ASA) was applied. This product includes software developed by Shinichi Sakata.

[14] E. Parastrates, "Narrowband MFP of shallow water experimental data", FOA-R- 99-01284-409-SE, 1999.

[15] L. Abrahamsson and B.L. Andersson, "Nlayer - An ELFE code for horizontally stratified media", FOA-R-97-00586-SE, 1997.

[16] B.L. Andersson, "Analysis of transmission loss data with JEPE-S and a genetic algorithm", FOA-R-01375-409-SE, 1999.

[17] D.W. Marquardt, "An algorithm for least-squares estimation of nonlinear parameters", J. Soc. Indust. Appl. Math., 11, pp. 431-441, 1963.

[18] D. Whitley and T. Starkweather, "Genitor II: a distributed geenetic algorithm", J. Expt. Theor. Artif. Intell. 2, 189-214, 1990.

[19] J. Mattsson and J. Schiöld, "Manual för CAT - Conductivity Analysis Tool", to be published, FOI report, 2003.

[20] D. Asraf, H. Claesson, L. Crona, E. Dalberg, P. Krylstedt, R. Lennartsson, and J. Mattsson, "Analys av data från fältförsök 2001. Detektion, lokalisering och klassificering på elektromagnetiska signaler", FOI-R-0584-SE, 2002.

[21] L. Crona, P. Krylstedt, J. Mattsson, A. Olsson, and M. Tulldahl, "Environment assessment for underwater sensors in the Stockholm archipelago, Part 2 - Optical transmission and inversion of electromagnetic sub-bottom parameters", FOI-R-0707-SE, 2002.