Automatic velocity picking by simulated annealing

Yunyue (Elita) Li and Biondo Biondi

ABSTRACT

Manual velocity picking is an inevitable and tedious process in the petroleum industry. An ideal velocity model is both geologically significant and geophysically smooth. Velocity picking can be phrased as a nonlinear optimization problem with multiple contradictory objectives. In this paper, we develop an automatic velocity picking technique based on the Simulated Annealing (SA) Algorithm. Accuracy and smoothness of the velocity model are used as objective functions. To improve the convergence of the algorithm, we include prior knowledge of the velocity model in the initialization and the constraints. The algorithm is adapted for this problem and demostrated using a 2-D field example.

INTRODUCTION

Measuring errors in velocity is one of the key steps in the processing of seismic data. An accurate velocity model produces accurate depth migration and optimal stack response and can be used directly as the lithology indicator. Residual migration has been shown to be a powerful tool for performing velocity error analysis because of its low computational cost (Rothman et al., 1985; Sava, 2003). After residual migration, we have a cube of residual migration images as a function of ρ, the ratio of true velocity to current velocity. Semblance panels corresponding to each value of ρ are computed to evaluate the focusing of the residual migration. However, manual velocity picking is required to obtain the updated velocity model.

An ideal velocity model is both geologically significant and geophysically smooth. It is easy to pick the peaks (maximum values) of the semblance panels for each CMP at each depth. The peak ρ values correspond to the optimal focusing update of the velocity model; however, those values often have large variations both horizontally and vertically. To solve the nonlinear velocity inversion problem, Singh et al. (2008) proposed a customized, multiobjective evolutionary algorithm. The simulated annealing algorithm is also a global optimization method and is capable of coping with the nonlinear relationship between the seismic data and the velocity model.

The simulated annealing algorithm is a Monte Carlo approach for minimizing multivariate functions. The term "simulated annealing" derives from the roughly analogous physical process of heating and then slowly cooling a substance to obtain a strong crystalline structure. In the simulation, a minimum of the cost function corresponds to this ground state of the substance. The simulated annealing process

gradually lowers the temperature in stages until the system freezes and no further changes occur.

In this paper, we customize the simulated annealing algorithm to automatically pick the semblance panels and give an optimized velocity model which is both semblance focused and smooth. The algorithm is briefly explained and its objective functions are introduced. We perform experiments on different sets of initialization and constraint parameters, the results and the convergence of which are compared. To test the accuracy of the velocity models, we apply them to the residual migration cube.

SIMULATED ANNEALING ALGORITHM

The simulated annealing (SA) algorithm is the computational analog of slowly cooling a metal so that it adopts a low-energy, crystalline state. It is a provably convergent optimizer. Geman and Geman (1984) provided a proof that simulated annealing, if the annealing is sufficiently slow, converges to the global optimum. In geophysics, SA has been employed to solve the problems of statics (Rothman, 1985), waveform inversion (Sen and Stoffa, 1991) and ray tracing (Bona et al., 2009). Here we propose the application of the simulated annealing algorithm to automatic velocity picking.

We initialize the system at a high temperature. At this stage the particles are then free to move around by a small pertubation; as the temperature is lowered, however, they are increasingly confined due to the high energy cost of movement. At each temperature T, SA perturbs the system randomly until it reaches equilibrium. The new state of the perturbed system is accepted according to the metropolis algorithm:

$$P \;=\; min(1,\; e^{-\frac{(E(\rho')\;-\;E(\rho))}{T}}), \tag{1}$$

where ρ is the current velocity model, ρ' is the perturbed velocity model, and E is the objective function presented later in the chapter. As shown in the equation, a higher T introduces a higher probability that an uphill perturbation will be accepted, which means that SA can draw samples from the whole population. As T decreases, only perturbations leading to smaller increases in E are accepted, so that only limited exploration is possible as the system settles to the global minimum. The pseudo code of the algorithm is given in Table 1.

The simulated annealing algorithm is popular and has been well-developed for single-objective optimization. Traditionally, multiobjective optimization can be converted into a single-objective optimization by different fix-up approaches such as the weighted-sum or ϵ-constraint method. In our case, we use a composite-objective function:

$$E(\rho) = \omega_{semb}\;\frac{1}{Semb(\rho)+\epsilon}\;+\;\omega_{smooth}\;\nabla\rho, \tag{2}$$

Table 1: Algorithm - Simulated Annealing

Inputs:	
$\{T_k\}_{k=1}^K$	Sequence of temperature values
$\{L_k\}_{k=1}^K$	Sequence of epoch durations
ρ	Initial velocity model
SA loop:	
1:	do itemp = 1, K
2:	do iepoch = 1, L_K
3:	$\rho' := perturb(\rho)$
4:	δE: = $E(\rho') - E(\rho)$
5:	rand: = rand(0,1)
6:	if rand < min(1, exp(-δE/T_k))
7:	$\rho = \rho'$
8:	end if
9:	end do
10:	end do

where $Semb(\rho)^{-1}$ is the inverse of semblance and is a measurement of focusing; ϵ is a arbitrary small number to avoid diverging when $Semb(\rho) = 0$; and $\nabla\rho$ is the residual after passing the velocity model through the Laplacian operator and is a measurement of smoothness. ω_{semb} and ω_{smooth} are the weights for these two measurements, respectively, making $E(\rho)$ is the linear combination of these two objective functions.

There are different ways to compose this single objective function. By using the inverse of semblance and residual of Laplacian we cast the optimization as a minimization problem. This combined objective function is then used as the energy or cost to be minimized in an SA optimizer. For the case we discuss here, the fixed weights are chosen according to the relative importance of the objective functions.

REPRESENTATION OF THE VELOCITY MODEL

Although we are picking ρ instead of picking actual velocities, discussing the representation of the velocity model is still helpful to better understand the shape and size of model parameters. Kirkpatrick et al. (1983) report that the run time of simulated annealing is related almost linearly to the number of parameters being estimated. Thus, a proper presentation of velocity model which requires fewer parameters will greatly decrease the computational cost of SA. Generally, two widely used classes of velocity models are blocky and smooth velocity models. Blocky models represent the geologically stratified sedimentary rocks, while smooth models have many numerical advantages. Both have been utilized for velocity optimizations using global methods (Jervis et al., 1996; Docherty et al., 1997; Mansanné, 2000).

Here, we are optimizing the residual migration parameter ρ rather than velocity itself. Regarding the initial velocity model, ρ values at different location might be independent with each other, suggesting it is better for us to use a grid to represent the ρ model. At this stage, we optimize the ρ model point by point using grid samplings.

When perturbing the system, we select one sample in the velocity model randomly, and change the velocity at that point to a random velocity value within a reasonable range. The main obstacle for a practical application of such a global optimization method is the computational cost. The larger the parameter space that must be searched and the greater the number of parameters, the more expensive the method tends to be. Thus, the prior knowledge could be both useful for speed and convergence. There are two slots where the prior knowledge can be inserted into the algorithm: initialization and constrains. However, experiments show that incorporating the prior knowledge into initialization is more efficient than into constrains. Thus, we initialize the system by the semblance peaks and randomly perturbing the system by changing the value to any possible ρ value defined by residual migration. Weights for the two objective functions are chosen arbitrarily. The results are presented in the next section.

2-D VELOCITY MODEL EXAMPLE

We apply the simulated annealing algorithm to the ELF7D data set. This dataset is complicated not only because of the salt dome, but also the complicated geological structure caused by the rising of the salt.

To optimize the velocity model, we use grid samplings of 63×266 m, and the number of grid points is 832. We start from a residual migration semblance cube. The residual migration parameter ρ ranges from 0.925 to 1.0745. The aim of the algorithm is to produce a smooth and accurate velocity model in accordance with the semblance cube. We initialize the system with the ρ values corresponding to semblance peaks, and randomly perturb the system through changing the ρ value to any possible ρ value defined by the residual migration. To create a smooth velocity model, we choose a much larger weight for smoothness than for semblance.

The result of the simulated annealing algorithm is shown in Figure 1. The model generated by 3200 iterations is much smoother than the initial model, but its main structure is still present in the final velocity model.

The top panel of Figure 2 shows the composite cost function while cooling. The cost function fluctuates up and down when the temperature is high, but the general trend is always decreasing. When the temperature is lower, the cost function goes down more steadily. As seen in the plot, the uphill perturbations are accepted less and less frequently by the Metropolis algorithm along the cooling path. Generally, the cost decreases much faster during the first iterations. As the annealing proceeds, the cost becomes more stable, until complete stability is reached at crystallization. The bottom two plots in Figure 2 show the cost curve of semblance and smoothness,

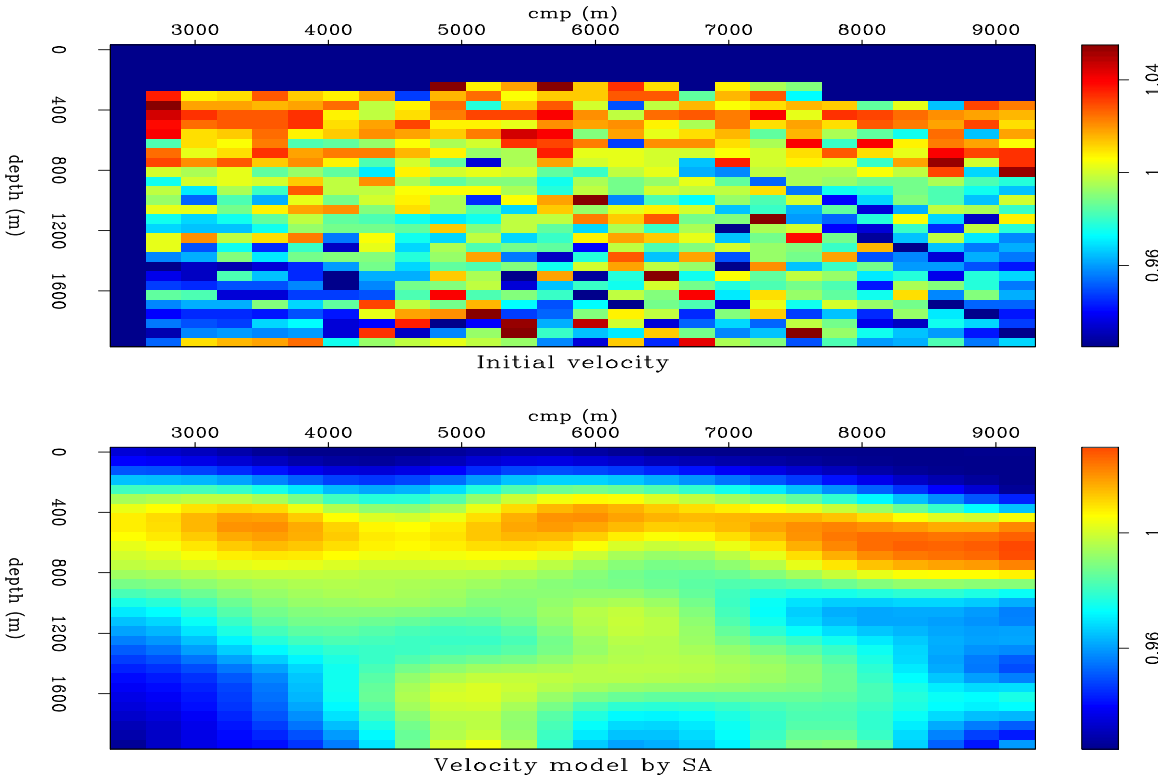

Figure 1: Initial velocity model (top). Velocity model after 3200 iterations (bottom). The SA velocity is much smoother and still has the structure of the initial velocity model. [**ER**]

respectively. Started from the minimum solution for semblance, simulated annealing maintains the increase of semblance within a factor of 2 while significantly enhancing the smoothness.

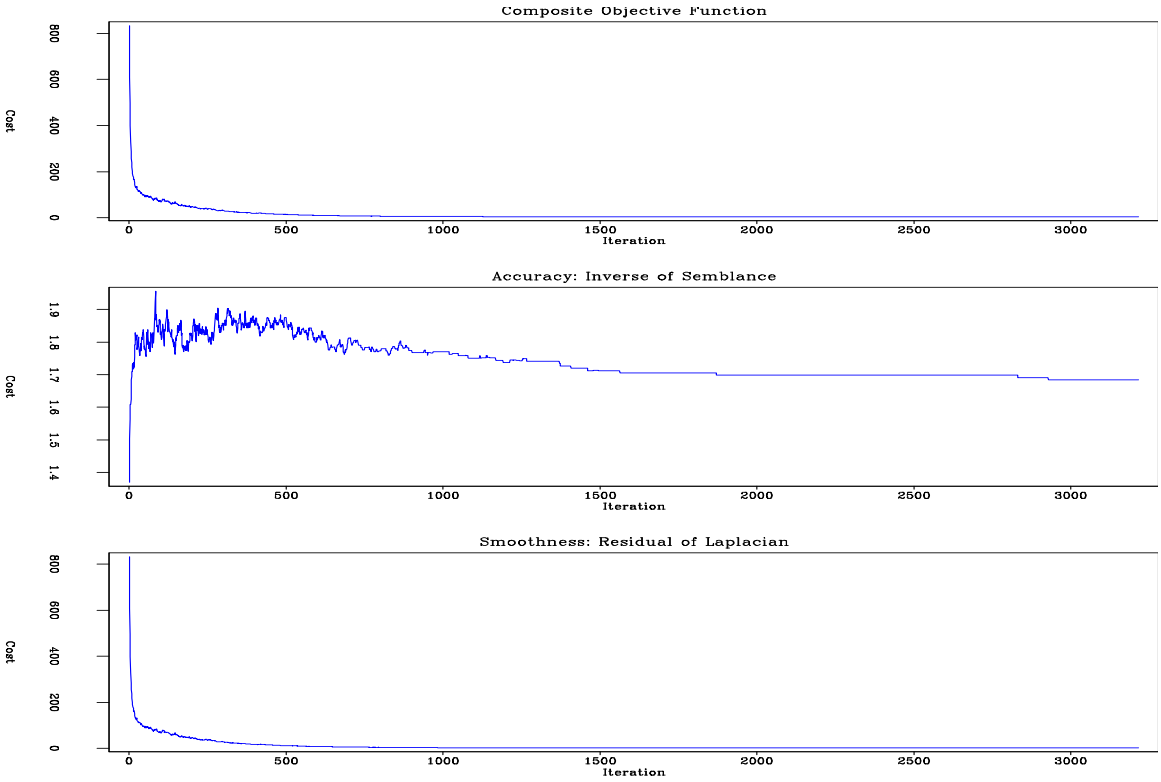

Figure 2: The plot on the top is the composite cost function; the plot in the middle is the cost of semblance; the plot at the bottom is the cost of smoothness. SA succeeds in controlling the increase of semblance when significantly enhancing smoothness. [**ER**]

Figure 3 and Figure 5 show the migration images before and after SA velocity analysis, respectively. Figure 4 and Figure 6 show the corresponding angle domain common image gathers (ADCIGs). It can be seen that the corners of the geological structure are focused and the boundaries of the salt body are connected after the velocity update.

FUTURE STEPS

Parallel computing

As Geman and Geman (1984) pointed out, simulated annealing can be implemented in parallel processes. In theory, using N processors would reduce run time by a factor of N. Additionally, parallel computing can also break the large problem down into

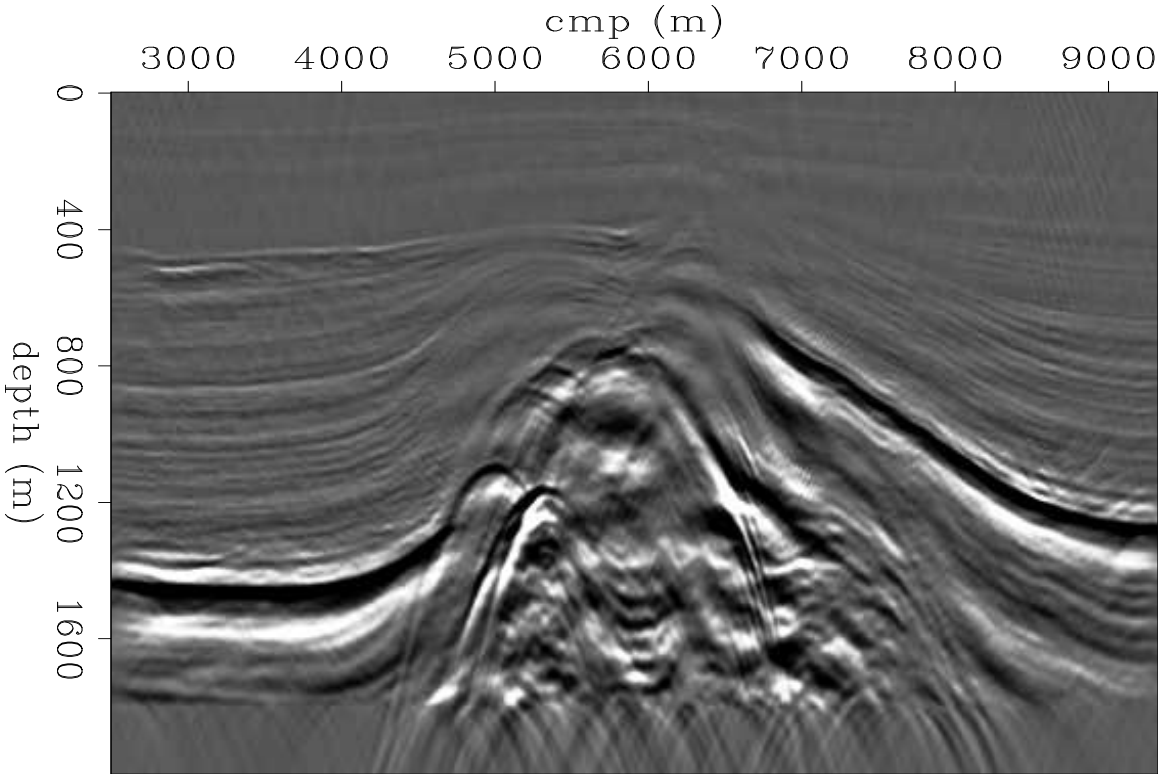

Figure 3: Migration image before performing SA velocity analysis. **[ER]**

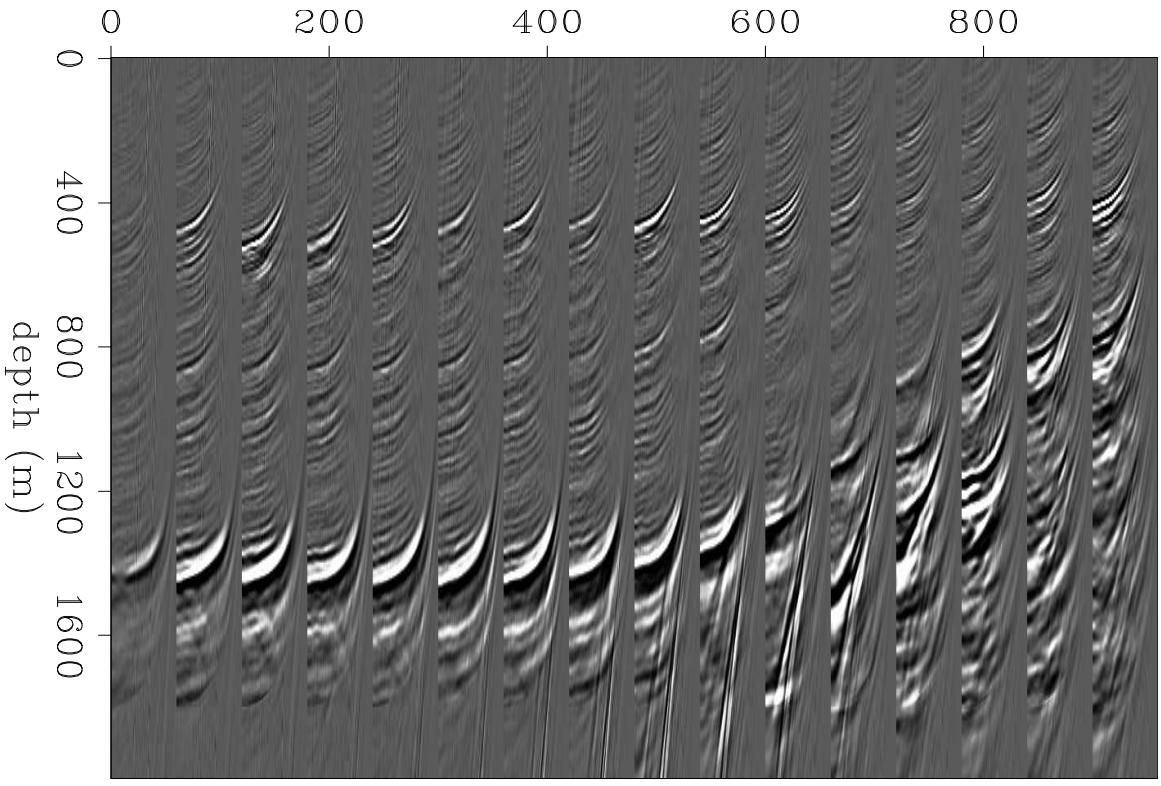

Figure 4: Corresponding angle domain common image gathers of figure 3. **[ER]**

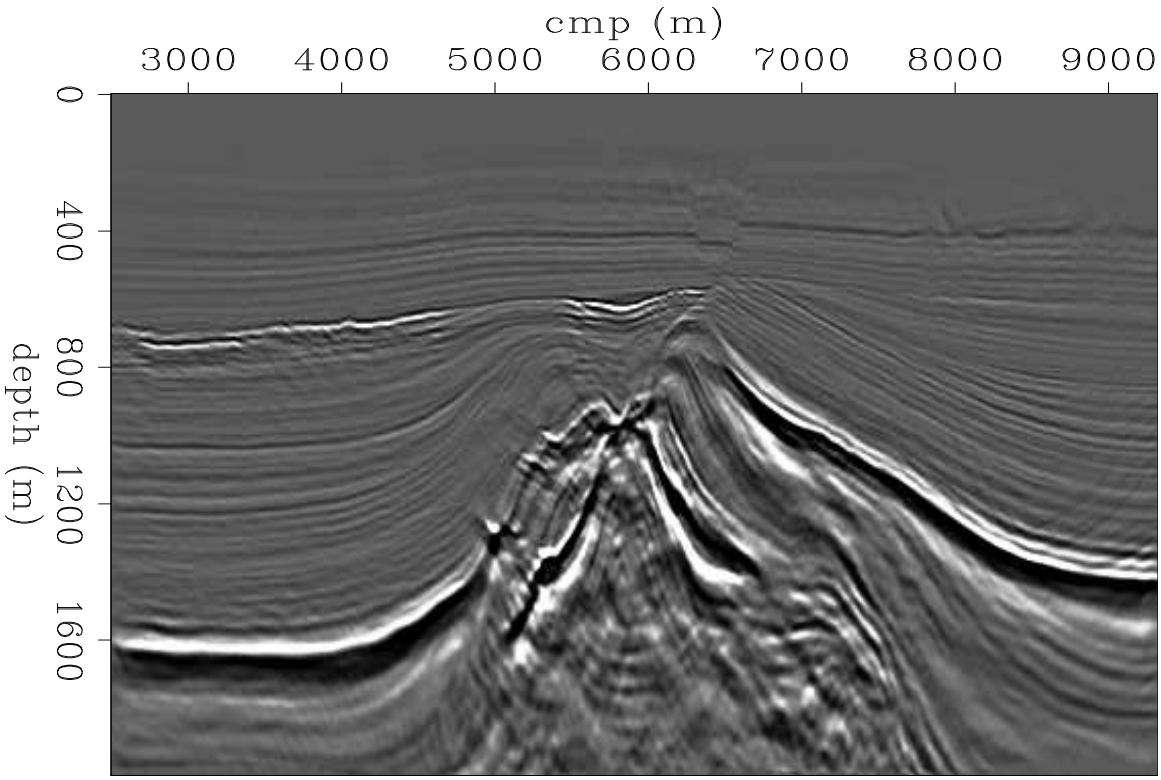

Figure 5: The residual migration image using the SA velocity model. Structures are well focused and the salt boundaries are connected. **[ER]**

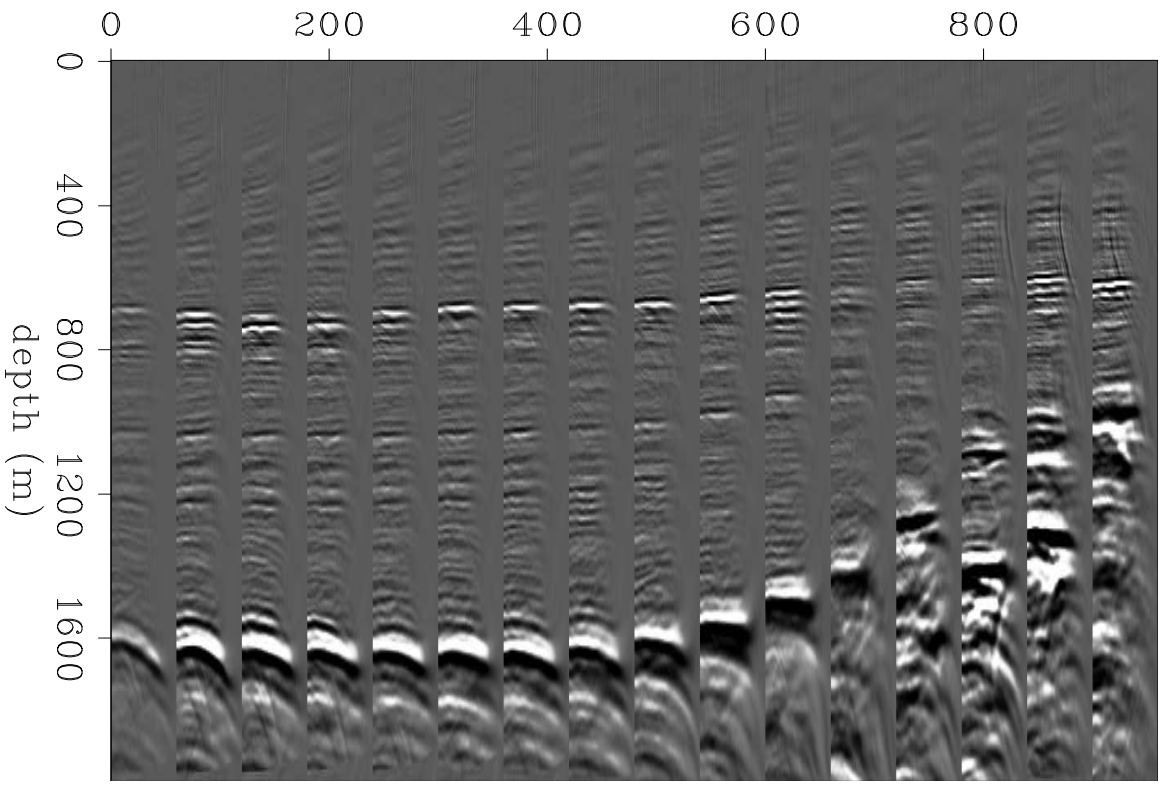

Figure 6: Angle domain common image gathers according to SA velocity. They are not flat in the complex area. The smiling events above water bottom can be corrected by further contrails. [**ER**]

small problems. Then the computation time is reduced to the same length of time that the small problems require.

Pareto optimal front

Automatic velocity picking is a multiobjective optimization problem. In this paper, we use a composite-objective function to find the optimal result. In fact, for multi-objective optimization, there exists a set of solutions known as Pareto optimal: no other solutions are feasible, which would decrease some objective without causing a simultaneous increase in at least one other objective. Several attempts (Pereyra, 2009; Nam and Park, 1999) have been made to explore the Pareto Front of multi-objective optimization. The next step of our project is to progress rapidly to the Pareto-optimal front, and then to finalize the result along the front in accordance with some constraints.

CONCLUSIONS

Simulated annealing is an effective global optimization method to cope with the non-linear velocity picking problem. We cast two contradictory objective functions into a single objective function by a weighted-sum criterion. The weights for each function are chosen according to their importance. Experiments show that including prior information into the initialization is important both speed and convergence. The results demostrate the robustness of the algorithm.

REFERENCES

Bona, A., M. A. Slawinski, and P. Smith, 2009, Ray tracing by simulated annealing: Bending method: Geophysics, **74**, T25–T32.

Docherty, P., R. Silva, S. Singh, Z. Song, and M. Wood, 1997, Migration velocity analysis using a genetic algorithm: Geophysical prospecting, **45**, 865–878.

Geman, S. and D. Geman, 1984, Stochastic relaxation, gibbs distributions, and the bayesian restoration of images: IEEE Transactions on Pattern Analysis and Machine Intelligence, **PAMI-6**.

Jervis, M., M. K. Sen, and P. L. Stoffa, 1996, Prestack migration velocity estimation using nonlinear methods: Geophysics, **61**, 138–150.

Kirkpatrick, S., C. Gelatt, and M. Vecchi, 1983, Optimization by simulated annealing: Science, **220**, 671–680.

Mansanné, F., 2000, Analyzse d'algorithmes d'évolution artificielle appliqués au domaine pétrolier: PhD thesis, Université de Pau.

Nam, D. and C. H. Park, 1999, Pareto-based cost simulated annealing for multiobjective optimization: 127–128.

Pereyra, V., 2009, Fast computation of equispaced pareto manifolds and pareto fronts for multiobjective optimization problems: Mathematics and Computers in Simulation, **75**, 1935–1947.

Rothman, D. H., 1985, Large near-surface anomalies, seismic reflection data, and simulated annealing: PhD thesis, Stanford University.

Rothman, D. H., S. A. Levin, and F. Rocca, 1985, Residual migration: Applications and limitations: Geophysics, **50**, 110–126.

Sava, P., 2003, Prestack residual migration in the frequency domain: Geophysics, **68**, 634–640.

Sen, M. K. and P. L. Stoffa, 1991, Nonlinear one-dimensional seismic waveform inversion using simulated annealing: Geophysics, **56**, 1624–1638.

Singh, V., B. Duquet, M. Leger, and M. Schoenauer, 2008, Automatic wave-equation migration velocity inversion using multiobjective evolutionary algorithms: Geophysics, **73**, VE61–VE73.