Automatic wave-equation migration velocity inversion using multiobjective evolutionary algorithms

Vijay Pratap Singh, Bertrand Duquet, Michel Léger, Marc Schoenauer

Automatic wave-equation migration velocity inversion using multiobjective evolutionary algorithms

Vijay Pratap Singh[1], Bertrand Duquet[2], Michel Léger[3], and Marc Schoenauer[4]

ABSTRACT

To solve nonlinear seismic velocity inversion problems, we have developed an automatic velocity-estimation technique based on the stochastic method called multiobjective evolutionary algorithms (MOEA). Semblance and differential semblance are used as objective functions. To cope with the high computational cost, we customized MOEA, added domain knowledge (velocity increases with depth, slowly varies along layers, and so forth), which improves the conditioning of the problem and accelerates convergence. This approach is robust because it can cope with large velocity errors. Computational cost of this algorithm is at least two orders of magnitude faster than other stochastic methods and comparable to that of direct gradient methods.

INTRODUCTION

A major interest in seismic exploration for hydrocarbons is to identify and localize the subsurface structure correctly. Therefore, seismic processing demands precise estimation of background velocity to obtain a good subsurface image. The conventional velocity-estimation technique of converting normal moveout (NMO) and/or stacking velocity into interval velocity is unstable for layers with lateral velocity variation (Lynn and Claerbout, 1982). However, migration velocity analysis (MVA), which adjusts the migration velocity model so there is minimal difference of migrated reflector images in common-image gathers (CIGs), is considered an attractive tool for velocity analysis because of its high sensitivity to the velocity model.

Generally, residual-moveout (RMO) curvature analysis (Al-Yahya, 1989) is used for MVA. This technique was developed further by Lee and Zhang (1992), Lafond and Levander (1993), Liu and Bleistein (1995), and Yan and Lines (2001). Geometric and mathematic characteristics of RMO for the prestack Kirchhoff migration are appraised for velocity by Zhu et al. (1998). RMO properties for wave-equation migration are analyzed by Biondi and Symes (2004) and by Shen (2004). RMO curvature analysis is based on the concept of the generation of a flat gather in CIGs by PSDM for the correct velocity model, regardless of structure.

Presently, iterative PSDM methods are used widely for velocity analysis in which flatness criteria have been measured on common-receiver gathers (Al-Yahya, 1989), common-offset gathers (Liu and Bleistein, 1995; Mulder and ten Kroode, 2002), common-scattering-angle gathers (Brandsberg-Dahl et al., 1999), and common-shot gathers (Symes and Carazzone, 1991) generated by Kirchhoff migration. Kinematic migration artifacts in CIG arise typically in prestack image gathers when the medium is a strong refractor (Stolk and Symes, 2004). As a result, the flatness principle might not be valid in CIGs produced by Kirchhoff-type migration. To avoid these artifacts for general velocity models, wave-equation migration is required for construction of CIGs. Attempts have been made to estimate velocity using wavefield methods, such as the method of full-waveform inversion (Tarantola, 1984; Pratt, 1999), diffraction tomography (Devaney and Oristaglio, 1984), wave-equation tomography (Luo and Schuster, 1991; Woodward, 1992), differential-semblance (DS) optimization (Symes and Carazzone, 1991), and wave-equation migration (Shen et al., 2003; Sava and Biondi, 2004; Sava et al., 2005).

Waveform-inversion schemes based on computation of the gradient of the misfit function are limited by the nonlinearity of the inverse problem. In addition, if phase differences between modeled and recorded wavefields are larger than a fraction of the wavelet, then the assumption made under the Born linearization is violated and velocity-inversion methods diverge (Woodward, 1992; Pratt, 1999; Dahlen et al., 2000). Consequently, either calculation of the gradient or nonlinearity of this problem becomes a severe obstacle. In addition, calculating the gradient is two (Sava and Biondi, 2004)

[1]Formerly IFP, Rueil-Malmaison, France, and École des Mines de Paris, France; presently ExxonMobil, Houston, Texas, U.S.A. E-mail: vijay.singh@exxonmobil.com, vijaypratap206@yahoo.com.
[2]Formerly IFP, Rueil-Malmaison, France; presently Total-CSTJF, Pau, France. E-mail: bertrand.duquet@total.com.
[3]IFP, Rueil-Malmaison, France. E-mail: michel.leger@ifp.fr.
[4]University Paris-Sud, Tao Project-Team, INRIA Futurs, Paris, France. E-mail: marc@lri.fr.

to four times (Shen, 2004) more computationally expensive than the migration itself. Hence, computation of gradients and lack of a good guess for initial velocity models in geologically complex regions inhibit the application of gradient-optimization methods for automatic velocity estimation.

Conversely, global optimization methods (genetic algorithms, simulated annealing, and the Monte Carlo method) are not constrained by local linearization of the wave equation (Born limitation) and are capable of coping with the nonlinear relation of the seismic data and velocity model (Jervis et al., 1996; Docherty et al., 1997; Mansanné, 2000). However, acute computational cost using global methods limits their routine use because a global method needs to search a large parameter space with little or no domain knowledge, i.e., problem-specific knowledge and guidance. Therefore, there is an urgent need to customize such an optimization method so it can use the domain knowledge and guide the optimization in the right direction without losing the generality of the method.

In this paper, we present a new global optimization algorithm based on multiobjective evolutionary algorithms (MOEA) for automatic velocity estimation. MOEA, like all other evolutionary methods, is a stochastic search technique inspired by principles of natural selection and natural genetics. It has attracted significant attention from researchers and technologists in various fields because of its ability to search for a set of Pareto-optimal solutions, i.e., a set of best-compromised solutions (mathematically defined in the next section) for a multiobjective optimization (MO) problem. In the next section, MO and evolutionary algorithms (EAs) are discussed briefly.

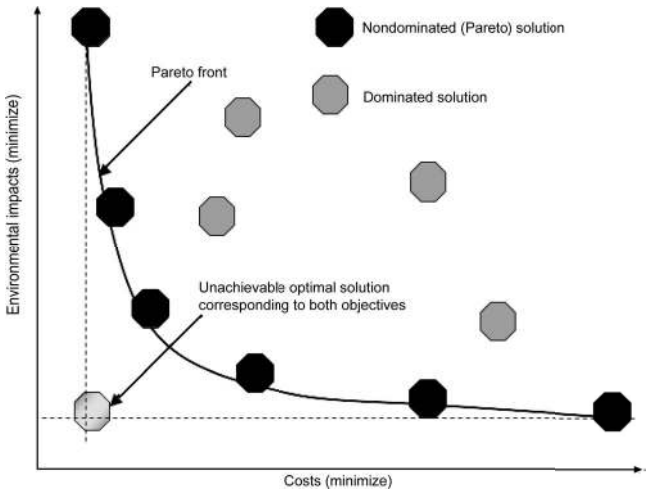

Figure 1. Concept of Pareto optimality. For instance, consider a problem in seismic data acquisition in an environmentally susceptible region. The objective is to find a solution that has minimum impact on the environment and costs less than alternatives. For this objective, many solutions exist (black and gray octagonals). Here, a solution can be considered Pareto optimal if there is no other solution that performs at least as well on every criterion (minimum environmental impact *and* minimum cost) and strictly better on at least one criterion (minimum environmental impact *or* minimum cost). Here, black solutions satisfy these criteria and hence are called Pareto-optimal solutions. These solutions cannot be improved without hurting at least one criterion. Solutions that are Pareto optimal also are known as nondominated. On the contrary, a solution is not Pareto optimal if one criterion can be improved without degrading any other. These solutions are known as dominated or inferior solutions (see gray solutions in the figure).

MOEA needs to search a large parameter space with little or no domain knowledge, i.e., problem-specific knowledge and guidance. Therefore, we customized MOEA so it could use the domain knowledge and guide the optimization in the right direction without losing the generality of the method. Customizing the MOEA according to domain knowledge accelerates convergence. This customized MOEA can cope with large velocity errors, and its computational cost becomes comparable to that of direct gradient methods.

In this paper, we first present a very brief introduction of MOEA along with its advantages and disadvantages. Then we discuss the objective functions, velocity-model representation, and information-extraction technique from migrated images and CIGs. After discussing these basic ingredients, we present customization of MOEA. Finally, we present results and conclusions of the current approach.

MULTIOBJECTIVE EVOLUTIONARY ALGORITHMS

Multiobjective optimization is concerned with finding a solution that optimizes (minimum or maximum) several contradictory or slightly contradictory objectives and eventually meets some additional constraints.

Suppose we want to optimize (in terms of minimization) a vector of functions: Minimize $[f_1(x), f_2(x), \ldots, f_k(x)]$ subject to the m inequality constraints of $g_i(x) \leq 0 \;\; i = 1,2,\ldots,m$, (inequality constraints) and the p equality constrains of $h_i(x) = 0$, $i = 1,2,\ldots,p$, (equality constraints), where k is the number of objective functions $f_i: \boldsymbol{R}^n \rightarrow \boldsymbol{R}$. Here, $x = [x_1, x_2, \ldots . x_n], \in \mathbf{X}$ is a vector of decision variables, and $\mathbf{X}$ is the feasible domain (the set of all vectors that satisfy constraints). We wish to determine in the set $\mathbf{X}$ the particular element (if unique) that yields the optimum value for all objective functions. Rarely is there a case that has a single point that optimizes all objective functions simultaneously. Therefore, we normally look for trade-offs or best compromises, usually by combining function f_i, into a single global function. Instead of searching for a single solution when dealing with MO problems, it is possible to look for a set of optimal solutions. First, a point x in X is said to dominate another point y in X only if $f_i(x) \leq f_i(y)$ for all i in $[1,k]$. This definition of *dominance* is written for the case in which objective function f_i is to be minimized, but it could be adapted easily to cases in which optimization means maximization or even to mixed cases. Now we simply define the Pareto front as the set of x^* in X that are Pareto optimal, with no y in X that dominates x^*.

In other words, this definition says that x^* is Pareto optimal if no feasible vector of decision variable $x \in X$ exists, which would decrease some criterion without causing a simultaneous increase in at least one other criterion. This concept almost always gives solutions called the Pareto front or Pareto-optimal sets (Figure 1).

Suppose we want to acquire seismic data in an environmentally susceptible area with a goal of minimum environmental impact, and we want to minimize the cost of data acquisition. If we use an environmentally good solution, it costs more for data acquisition. Conversely, if we minimize cost, there is more impact on the environment. In such a condition, our first goal is to find solutions that are optimal with respect to both criteria or at least with respect to one criterion.

Here, a solution can be considered Pareto optimal if there is no other solution that performs at least as well on every criterion (mini-

mum environmental impact *and* cost) and strictly better on at least one criterion (minimum environmental impact *or* cost). In Figure 1, dark-gray solutions are Pareto optimal. These solutions cannot be improved without hurting at least one criterion. Pareto-optimal solutions also are known as nondominated. A solution is not Pareto optimal if one criterion can be improved without degrading any others. These solutions are known as dominated or inferior solutions (see Figure 1, light-gray solutions).

Multiobjective optimizations are not only different in terms of the number of objective functions but also are different in terms of the number of goals number of search spaces, and solution type. In contrast to single-objective optimizations, which have the single goal of searching for a global optimal solution, multiobjective optimization has two goals: (1) to progress toward the Pareto-optimal front and (2) to maintain the diversity of solutions along the front. In addition, single-objective optimization has one search space, the decision-variable space, whereas multiobjective optimization has two search spaces, the decision-variable and objective spaces. Traditionally, multiobjective functions are converted into a single-objective function by different fix-up approaches such as the weighted-sum approach or ε-constraint method. Unfortunately, the outcome of this optimization strategy depends on chosen fix-up values. Multiobjective optimization for finding multiple Pareto-optimal solutions eliminates all such fix-ups and, in principle, finds a set of optimal solutions corresponding to different fix-up values. Interestingly, the diversity of solutions along the Pareto front gives an idea of the uncertainties about the solution of the problem.

After discussing multiobjective optimization, the common terminology and working procedure of evolutionary algorithms are discussed. Essentially, EAs are a method of breeding solutions of an optimization problem by simulating evolution. Because EAs are inspired by natural selection and genetics, they borrow much dialect from genetics, cellular biology, and evolutionary theory. In EAs, a candidate solution is known as an individual. The collection of current individuals in the system is known collectively as the population. The representation of an individual is known as a genotype. The way a solution operates when tested in the problem environment is known as the individual's phenotype. When individuals are modified to produce new individuals, they are said to be breeding. After the evaluation, an individual is given a rank known as its fitness, which indicates how good a solution the individual is. The entire process of finding an optimal solution is known as evolving a solution. In EA, first a set of random individuals (models) is generated. Then each individual in the initial population is evaluated, and a fitness is assigned to it. The better individuals from members of an initial or old population are selected for breeding and form a new population. This new population is evaluated and then mixed with the old population. An optimal new population is created from this assembly (old + new population). This process is continued until an ideal individual is discovered or resources are exhausted.

MOEA combined the concept of MO into EA. It has attracted significant attention from researchers and technologists in various fields because of its ability to search for a set of Pareto-optimal solutions for multiobjective optimization. Although the first MOEA was published in the mid-1980s (Schaffer, 1985), significant development and interest in MOEA are noticed after the mid-1990s (Deb, 2001; Coello et al., 2002).

Comprehensive information about working groups, books, papers, abstracts, and code can be obtained from Coello (2003). Below are the definition and a few important terminologies of multiobjective optimization, their differences from single-objective optimization techniques, and advantages and disadvantages.

First, positive and negative properties of MOEA are discussed to help in understanding customized MOEA. In the section titled "Optimization," we describe how we use positive and negative properties in customizing MOEA. This customized MOEA makes optimization very fast. Evolutionary algorithms, in principle, can find solutions to problems with nonsmooth nonlinear objective functions and constraints, whereas gradient-based methods require smooth and differentiable objective functions. Furthermore, because they evolve a population of candidate solutions, they offer additional advantages when tackling an MO problem, such as identifying many Pareto-optimal solutions in a single run. This explains why MOEAs are so popular today. Moreover, besides having no assumption with respect to problem space and being easy to hybridize with other approaches such as gradient or local methods, MOEAs add domain knowledge to improve their efficiency. MOEAs also can cope with noisy data. The choice of ε-MOEA was motivated by Deb et al. (2003), who demonstrate performances similar to the best-performing previous MOEAs but requiring much less computational effort. A flowchart and description of the original ε-MOEA (Deb et al., 2003) are given in Appendix A.

However, the disadvantage of MOEAs and even ε-MOEA is that they are usually much slower than local optimization methods, often by several orders of magnitude. As problem size increases (from, say, 10 to 100 or 1000 decision variables), an evolutionary algorithm often is overwhelmed by the dimensionality of the problem and is unable to find anything close to an optimal solution, whereas it is still possible to solve such large problems with local optimization methods.

The main reason for such high computational cost and the inability to perform on a great number of variables is the blindness (semi-randomness) of those algorithms, i.e., the lack of exploitation of domain-specific information. Any efficient global optimization algorithm must balance carefully between two possible strategies to find the global optimum: exploration to investigate new and unknown areas in the search space and exploitation to make use of knowledge from previous iterations. These two requirements are contradictory. Therefore, a good optimization algorithm must strive to find a balance between them. In the hybrid MOEA we designed (described in the section titled "Optimization"), an effort has been made to exploit and use domain-specific information without reducing the exploration property of the original underlying MOEA, the ε-MOEA. Such ad hoc use of available expert knowledge from previous iterations is mandatory for evolutionary algorithms to reach (fast) convergence toward the global solution, even in large search spaces.

Main issues that we resolved before applying MOEAs for velocity estimation were choosing a velocity-model representation, finding a robust and accurate objective function to measure the fitness of a velocity model, and extracting and deciding how and where to use the domain knowledge. These issues are discussed individually in the next sections. After resolving those issues, we applied MOEA in the customized ε-MOEA discussed in the section titled "Optimization." First we will discuss the representation of the velocity model.

Representation of the velocity model

Representation of the velocity model is an important issue in velocity optimizations because it controls the shape and size of model parameters. Classically, the two main classes of models generally

used are blocky and smooth models. Generally, geologists are influenced by stratified aspects of sedimentary rocks. Therefore, they usually recommend blocky models. Nevertheless, smooth representations have many numerical advantages, although they are not very suitable for complex velocity models. Both models have been used for velocity optimizations using global methods. The smooth velocity has been represented by grids, splines, or cubic B-splines and encoded into binary strings (Jervis et al., 1996; Docherty et al., 1997). Mansanné (2000) prefers blocky model representations and uses a Voronoi diagram to represent a geologic model. This Voronoi representation was encoded into real parameters for each domain, with two parameters (in two dimensions) for the location of the central point and one parameter for the velocity. The Voronoi representation needs fewer parameters to represent a velocity model in two or three dimensions than is needed for grid or spline representation. The lesser the number of parameters, the faster the optimization because of reduced dimensions. Mansanné (2000) observes that sometimes the objective function is optimized numerically, although obtained velocity models are geologically insignificant.

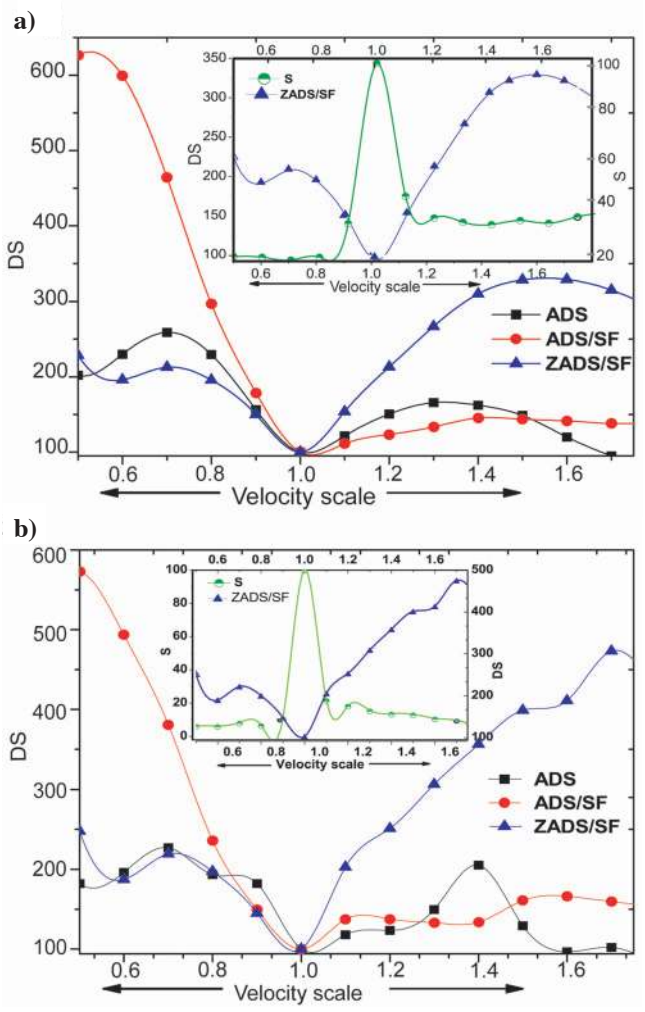

Figure 2. (a) The response of different differential-semblance (DS) functions for scaled Marmousi models in which velocities are multiplied by 0.5 to 1.75. Each scaled velocity model is migrated using every second shot and frequencies of 10–25 Hz. Corresponding semblance (S) and DS are plotted here. The combination of S and modified DS (ZADS/SF) is in the subplot. The S function has a narrow, steep valley of attraction near the true velocity model, whereas the ZADS/SF function has a wide, gentle valley of attraction. (b) The response of different DS functions for the Marmousi model at X = 6000 m. Modified DS function response is better than other DS functions.

Hence, we attempted to represent a velocity model by layers in place of grids, splines, or Voronoi representation (Singh et al., 2005). By using layers, we made the velocity model geologically significant and concise. We could optimize the velocity using the geologic representation, although we realize that this representation is model dependent. Generally, it is difficult to find a forward model and data that could invert a real geologic environment.

Therefore, we decided to use a grid parameterization to represent our velocity models because any geologic environment could be represented by a grid, although it requires a large number of parameters. In a grid representation, it is easy to incorporate information obtained after migration of the velocity model because the wave-equation migration is performed on a regular grid, whereas it is difficult with other representations (Voronoi or geologic). The usual drawback of having a large number of parameters (CPU time is long) will be balanced by the introduction of domain knowledge (good initialization, RMO correction, and dip smoothing).

Objective functions

For an automatic velocity estimation based on CIG, in which no picking is introduced, the choice of the objective function is vital. The least-squared difference between each pair of migrated shots (Jervis et al., 1996), semblance (S) (Jin and Madariaga, 1994; Docherty et al., 1997; Mansanné et al., 1999), and differential semblance (DS) (Symes and Carazzone, 1991; Chauris and Noble, 1998; Plessix et al., 2000; Pratt and Symes, 2002; Mulder and ten Kroode, 2002) on CIG are used widely for velocity estimation. The DS function also has been used for optimization of waveform tomography in the time domain (Plessix et al., 2000) and in the frequency domain (Pratt and Symes, 2002). Because we use shot-gather wave-equation migration (Rickett and Sava, 2002) for velocity optimization in this work, our discussion about the objective function is wavefront-oriented ray tracing (WRT) of the shot-gather wave-equation migration. In the following section, we summarize the properties of S and DS functions and provide arguments to substantiate the use of both functions in this work.

Semblance

Semblance function often is used for velocity optimization (Jin and Madariaga, 1994; Docherty et al., 1997; Mansanné et al., 1999) employing global methods. It is based on the flatness of semblance panels (flatness of CIG). A measure of flatness of events in the semblance panel is given by the sum of the total energy of traces. Semblance measured on a common-angle gather could be written as

$$S = \int_x dx \frac{\left[\int_z dz \left(\int_\theta I d\theta\right)^2\right]}{\left[\int_z dz \int_\theta I^2 d\theta\right]}, \qquad (1)$$

where I is the migrated image, z is the depth, and θ is the reflection angle. The response of the semblance function for the whole Marmousi model, at location X = 6000 m is shown in the Figure 2a and

b subplot by a green circle. One of the most attractive features of semblance function is its strong sensitivity near the true velocity, but it is not sensitive for large velocity errors.

Differential semblance

Differential-semblance objective function for velocity optimization is proposed by Symes and Carazzone (1991) to overcome the nonconvex property of semblance function. DS combines concepts from least-squares inversion, migration velocity analysis (flatness criteria of CIG), and traveltime tomography. DS is used with ray-trace-based migration and wave-equation migration. DS function is a measure of the sum of differences of near-trace energy. Mulder and ten Kroode (2002) use the following relation:

$$\mathrm{ODS} = \int_x dx \int_z dz \int_h dh (\partial_h I)^2. \tag{2}$$

Here, $I(x,z,h)$ is the migrated image; it is a function of abscissa x, depth z, and offset h. Chauris and Noble (1998) use this function (equation 2) by adding a scaling factor ($SF = \int_z dz \int_h I^2 dh$) in the denominator to reduce the effect of the migration amplitude:

$$ODS/SF = \int_x dx \frac{\left[\int_z dz \int_h dh (\partial_h I)^2\right]}{\left[\int_z dz \int_h I^2 dh\right]}. \tag{3}$$

Recently, we have proposed a modified offset-domain differential semblance (MODS) function for velocity analysis (Singh et al., 2006).

Equations 2 and 3 could be used in the angle domain by replacing offset parameter h by angle parameter θ.

$$\mathrm{ADS} = \int_x \int_z dz \int_\theta d\theta (\partial_\theta I)^2; \tag{4}$$

$$ADS/SF = \int_x dx \frac{\left[\int_z dz \int_\theta d\theta (\partial_\theta I)^2\right]}{\left[\int_z dz \int_\theta I^2 d\theta\right]}. \tag{5}$$

Because amplitude decays with depth, these relations will be more sensitive to shallow events than to deep events. To compensate for this effect, a factor of depth, Z, is introduced. This modified angle-domain differential semblance (MADS) function is presented here:

$$ZADS/SF = \int_x dx \frac{\left[\int_z dz \int_\theta Z d\theta (\partial_\theta I)^2\right]}{\left[\int_z dz \int_\theta I^2 d\theta\right]}. \tag{6}$$

We believe the depth factor (Z) compensates for the effect of spherical divergence (associated with the used velocity model) and the scaling factor (SF) reduces the effect of data noise. The response of these objective functions (equations 4–6) for the whole Marmousi model (Figure 2a) and the most complicated region of the Marmousi model (at X = 6000 m) (Figure 2b) is shown in Figure 2. These measurements are obtained when the Marmousi velocity model is scaled by a factor between 0.5 and 1.75, and every second shot and a frequency range of 10–25 Hz are used for migration. The modified differential-semblance function is nicely convex for a large range of high (1.75 times of the velocity model) and low velocities (0.5 times of the velocity model), and it is stable with respect to the choice of migration parameter. The modified function is not affected much by migration-parameter settings. Thus, smaller frequencies and lesser shots can be used to speed up velocity optimization.

In the beginning of optimization, initial models are generally far from true models, and the DS function can help in convergence because of its sensitivity for far velocity models. On the contrary, when models are near to the true model, the S function helps convergence because of its strong sensitivity for these models. Chauris and Noble (1998) suggest that DS function could be used in the first few iterations and followed subsequently by S function during optimization. This motivated us to use multiobjective evolutionary algorithms for employing the advantages of both objective functions simultaneously. Hence, we use S and DS objective functions to measure the goodness of a complete velocity model, and to correspond locally to each panel of gathers (Figure 3). The local measurements of goodness by S and DS objective functions are referred to hereafter as LS and LDS (L stands for *local*, i.e., corresponding to one panel of gathers). In the next section, we give a brief introduction to MOEA in which these global and local objective functions are used.

Extraction of domain knowledge

The method of velocity estimation using migration with global optimization is computationally very intensive. One main reason why global optimization is expensive is the lack of the exploitation of information present in the image and gathers. In other words, the lack of use of domain knowledge (i.e., problem-specific knowledge) is one of the main reasons why global optimization is expensive. To create a geologically feasible velocity model and to speed convergence of the migration-velocity-estimation problem, it is necessary to add extra information during inversion. This extra information can be taken from well logging, a geologic structural model, and preliminary stack or migration results. Other additional information could be from migrated images obtained with velocity models (even from a wrong velocity model), such as their gross velocity-error estimations and the structural trend of the geology. Thus we need a robust approach that could extract the approximate amount and direction of the gross velocity error and structral trend of the geology. We show how this additional information, when incorporated into MOEA, can lead to more efficient optimization.

Gross velocity-error estimation

Gross velocity errors from migrated images are estimated generally by either RMO or Stolt prestack residual-migration (Sava, 2003) techniques. The main advantages of Stolt prestack residual migration are its interpretative and structural dependencies. Prestack residual migration also reduces the effect of image dispersal between events imaged at the same physical location but with different aperture angles. However, the main difficulties with residual migration are its computational complexity and structural dependencies, thereby making its automation more challenging. On the

other hand, RMO inconsistency in image is measured by computing the semblance scan as a function of one RMO parameter and then picking the maximum of the semblance scan (Biondi and Symes, 2004). For flat events, nonlinear RMO function and residual migration are equivalent (Liu and Bleistein, 1995; Biondi, 2003).

In this work, the RMO approach is adopted because we are interested only in an approximate magnitude and direction of change of velocity required at any place for correcting the velocity model. Regardless of the domain (offset or angle) in which the prestack partial images are defined, the RMO function usually is parameterized by a single parameter denoted γ, which is the ratio of migration velocity and true velocity. Inaccurate velocities will cause moveout artifacts on the migrated image, and its shape also depends on the velocity contrast. Smile and frown shapes generally are produced for low-velocity and high-velocity contrasts, respectively.

Here, we first perform the Radon transform of a gather (offset or angle gather). This enables us to work in parameter space (z, γ) rather than the image space in which curve detection is tedious. The peaks thus determined from the Radon transform are basically parameters of a reflection that gives necessary information about the background velocity model. These peaks measure the integral of the velocity error, although these errors all should be spread above the measured peak location to update velocity. It was our practical choice to use this velocity error to update velocity locally above the measured peaks.

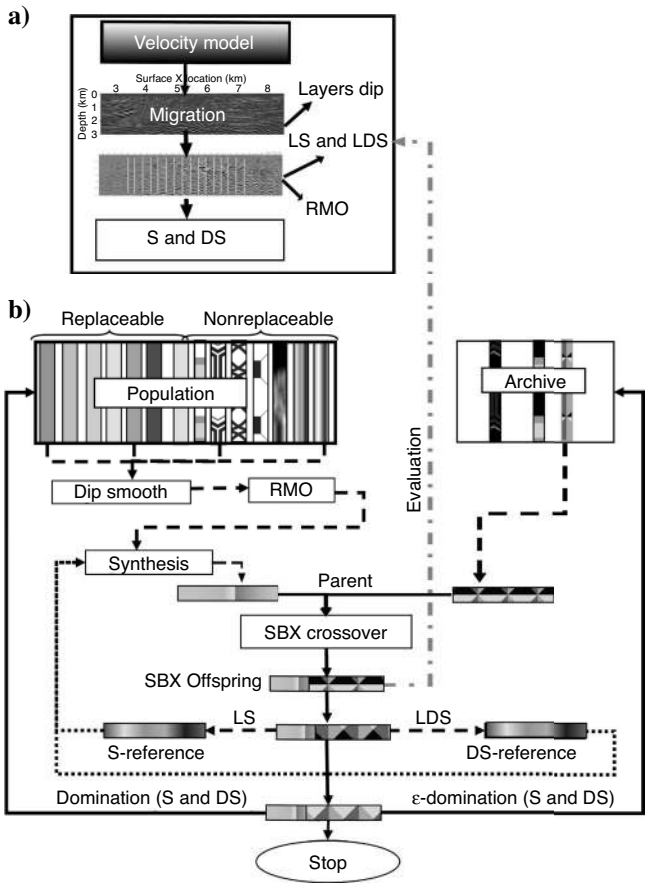

Figure 3. (a) Process of evaluation and information extraction from a velocity model. (b) Schematic diagram of our customized ε-MOEA. Four models are selected stochastically from the population or among reference models. Each selected model first is smoothed using dip information and then corrected using RMO information. A parent model is synthesized from these four models using LS or LDS criteria. A synthesized parent and one uniformly selected parent from the archive are used to generate an offspring using variation operation (simulated binary crossover, or SBX). This offspring is compared with the semblance reference and differential-semblance reference using LS and LDS criteria. If the offspring has better LS or LDS than the reference models for some panel, the column in the velocity grid corresponding to this panel is copied into the reference model. This offspring replaces one member in the population that it dominates, if any, or one random member otherwise, whereas it is placed in the archive only if it ε-dominates at least one of its members.

Structural trends and dip information

The structural trend of a geologic model also provides significant information and has been used differently for velocity estimation. The basic assumption (see Delprat-Jannaud and Lailly, 1992) is that velocity follows structural dips or some other known trend and hence can be incorporated as a term into the objective function (Delprat-Jannaud and Lailly, 1992). Kaipio et al. (1999) suggest using prior structural information to create conditional covariance matrices. This also has been used as a model-regularization operator and preconditioner (Clapp et al., 2004). It has been found that this information significantly improves convergence speed. This dip could be extracted from a migrated image using a prediction-error filter or geostatistical approach of covariance analysis (Clapp et al., 2004).

Here, we are not using dip information as an objective function, regularization term, or preconditioner. Our goal is to extract dip information at each point of the migrated image and use it to reduce the variation of velocity along the dip direction. This will help in reducing implicitly the effective number of unknown parameters.

To obtain dip information, we convolve Sobel operators to the migrated image and extract dip at each point. The Sobel operator (Sobel and Feldman, 1968) is used in image processing. A discrete differentiation operator computes an approximation of the gradient of the stacked migrated-image intensity function. At each point in the image, the result of the Sobel operator is either the corresponding gradient vector or the norm of this vector. The gradient approximation that the Sobel operator produces is relatively crude, particularly for high-frequency variations in the image.

Dip information is available on a fine-scale grid of migrated images, whereas we need dip information on a coarse-scale velocity grid. Therefore, we smoothed the migration dip map up to the scale of the velocity model. This dip smoothing can be viewed as some type of antialiasing. Once we obtained dip information up to the scale of the velocity model, we smoothed the velocity model along the obtained dip direction.

Optimization

Once we have represented our model, have found a better objective function, and have the tools for providing information during optimization, we need to develop a good optimization strategy. Because we use evolutionary algorithms, the choice of appropriate algorithms according to the problem is of prime importance. In addition, because we intend to use two objective functions, multiobjective evolutionary algorithms are the preferred choice. In MOEA, we have an option to select generational MOEA (nondominated sorting genetic algorithm-II, or NSGA-II [Deb et al., 2002], strength Pareto

evolutionary algorithm 2, or SPEA-2 [Zitzler et al., 2001], and so forth) or a steady-state MOEA such as ε-MOEA (Deb et al., 2003). In generational MOEAs, all new individuals are created first, and then they are selected to build the next generation of the population. In steady-state MOEAs, new individuals are created and processed one by one. Generally, steady-state EAs are faster than generational EAs. The superiority of ε-MOEA to NSGA-II and SPEA-2 in terms of convergence speed or diversity maintenance was established by Deb et al. (2003), and therefore, we decided to use it.

Because global optimization methods lack good exploitation properties, solving for a large number of parameters is extremely expensive and even impossible at times. If one has to solve a large number of parameters (say, 500 to 5000), one has to modify the ε-MOEA. This necessitates the development of a good exploitation operator and having an efficient information system to proceed in the right direction. Below, we explain our new customized ε-MOEA, in which a biased initial population (group of initial models), information exploitation models (reference model), and knowledge-specific crossover operators are introduced.

Initial population generation

In MOEA, evolution starts from a population of specific size of random models. There are two practical issues in this approach: (1) population size and (2) initial population generation. These two issues are discussed separately below.

Population size generally depends on the nature of the problem but typically contains several hundreds or thousands of possible initial models. Population size has a major effect on the efficiency and performance of MOEA. MOEA does not work well for a very small population size, and a very large population size impacts the performance of the MOEA. The purpose of large populations is to have diversity in the parameter space. However, a large population needs the evaluation of a large number of models. Because our evaluation process is costly, we decided to evaluate fewer numbers of models but still maintain diversity in parameter space. To fullfill the above goals, a small number of models is generated in initial population first, and then the initial population is divided into two subpopulations. The other imposed constraint is that one subpopulation is nonreplaceable and the other is replaceable during the optimization process. In this way, diversity in parameter space is maintained by the nonreplaceable subpopulation. A similar strategy is adopted by Coello and Pulido (2001) for micromultiobjective genetic algorithms (μ-MOGA). The replaceable part of the subpopulation will help in convergence.

The next important issue is generation of the initial population. The initial population is a good place to embed knowledge from the problem domain. One can take advantage of embedding domain knowledge into the initial population. As a result, a smart initial population can increase the likelihood of successfully composing global solutions through the iterative process of information exchange.

Generally, evolutionary algorithms start from random models. Such generation of completely random models is suitable for problems for which we lack any prior information, knowledge, or experience about the models. However, in our approach, we have prior information about the possible velocity from geologic studies, well logs, and seismic preprocessing, along with the well-known fact that velocity generally increases with depth. In addition, we have information from CIGs. Therefore, we should generate approximate solutions by leveraging resources that are generally available. Those resources might be problem-specific knowledge, experience, relations, or field data that characterize system behavior. Consequently, the generation of knowledge and experience-based semirandom models is more suitable. This process will speed up convergence in addition to providing information to the system. However, it should be pointed out that small populations with proper management, information, and diversity are sometimes better than or as good as larger populations.

Reference models

The objective functions (S and DS) measure the global goodness of a model. Although a generated model is good at many locations, it might not be good globally. Hence, it is not placed in the solution space. Therefore, to exploit this partially good local information, we created two reference models, one for semblance and one for differential semblance. The underlying physical idea is that the velocity-determination problem at some abscissa is not coupled or is coupled weakly to the same problem at some distant abscissa (a few kilometers). This allows us to build compound reference models with the best parts of previously processed models (according to S or DS criteria). For generating these reference models, semblance and differential semblance of each CIG are calculated (hereinafter called LS and LDS, respectively; as a reminder, L stands for *local*). Once the LS and LDS for each model are calculated, they are compared with each other. Velocities corresponding to the best CIG (LS or LDS), i.e., the column of the velocity grid with the same abscissa as that best CIG, are selected. One reference model copies the velocity corresponding to the best LS location, and the other copies the velocity corresponding to the best LDS. During the generation of models, each model will be compared with reference models. If some CIGs are better than the CIG of the reference model, then the corresponding velocity will be copied. Reference models then are used for crossover with other models. During crossover, such reference models help in generating better models and, according to our tests, they greatly speed up convergence. These reference models also introduce an exploitation property into the MOEA.

Guided crossover

The philosophy of MOEA is that good parent models generate good offspring models. The crossover of parents generates offspring. The main purpose of a crossover operator is to recombine partial good information from two or more models to generate better offspring models. Conventionally, models for crossover are selected from a population by some statistical technique (tournament selection, roulette-wheel selection, and so forth) (Deb and Goldberg, 1991) or randomly. Because we have a small number of models in the population, some of which are nonreplaceable, selecting a good solution each time for crossover will reduce the diversity of the population. Thus, traditional selection criteria of a model might lead toward premature convergence and reduce convergence speed drastically. Here, one of our goals is to find good models for crossover from a population. To accomplish this, we decided to synthesize a parent model from a few randomly selected parent models rather than statistically selecting one model.

There could be many strategies for synthesizing parent models. Our parent-generation method is based on the local fitness criteria (LS, LDS) of a model. For this, we randomly selected four models from the population. Each selected model first is corrected on the basis of available γ (RMO) information. Then these models are

smoothed along the dip direction, which reduces the variance of velocity along layers. Once this information is introduced, the next goal is to synthesize a parent model from these four selected models. For synthesizing a parent model from these models, a strategy again is adapted similarly to reference-model generation. First we compare the selected models for each CIG on the basis of LS or LDS. Then velocity models corresponding to the best LS and LDS are copied to the parent synthetic velocity model. To do this, there are three possibilities: (1) We could replace the model on the basis of the best LS, (2) we could replace it on the basis of LDS, or (3) we could adapt a Pareto-dominance strategy for replacing models. Figure 4 shows the schematic view of parent synthesis.

Once we synthesize a parent from four parents, we apply simulated binary crossover (SBX) (Deb and Agrawal, 1995) (see Appendix B for a detailed description). SBX emphasizes the generating of offspring near parents. Therefore, crossover guarantees that the diversity of children is similar to the diversity of parents and has the advantage that individuals near parents are more likely to be chosen as children than individuals distant from parents. By synthesizing one parent from several parents, we have created a parent of good attribute because SBX generates an offspring near parents. Hence, if parents of good attributes do the crossover, they should generate a good offspring. In this way, merely by using the philosophy of MOEA, we could guide a crossover to the right direction without disturbing the exploration property of MOEA. For example, with reference-model generation, the intuitive idea behind parent synthesis is to take advantage of the quasi-independence of the subsurface velocity-determination problem for parts of the model separated by distances of a few kilometers. This allows us, to some extent, to solve for these model parts as decoupled problems (Figure 4).

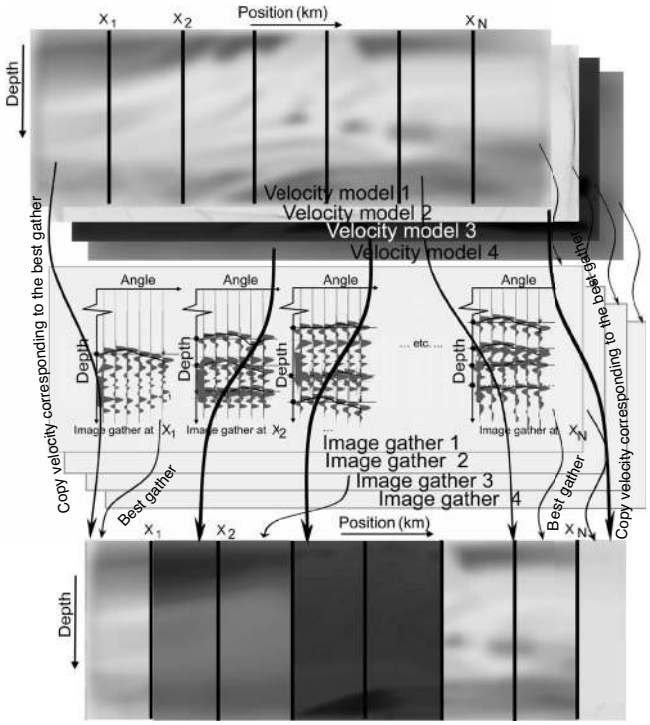

Figure 4. Synthesis of a parent. From the four selected velocity models, one new velocity model is synthesized. At each X position of all velocity models, corresponding local semblance (LS) or local differential semblance (LDS) are compared. Velocity columns corresponding to the best LS or LDS are copied into the synthesized parent model.

Fixed subpopulations are participating during the synthesis of a new parent; they also achieve crossover with archive populations. Because synthesis is based on the criteria of the local fitness of a model, it also is necessary to do crossover with global good and bad models. Crossover of nonreplaceable subpopulations (random models) with archive models produces a diverse model and maintains the diversity of models. Crossover with two reference models (collection of the best velocity panels) will add the positive attribute to offspring models and help in converging toward the global solution.

Guided crossovers explore most probable regions by searching good CIGs from four randomly selected models and synthesizing a parent. Guided crossovers exploit the good attributes of reference models and maintains the diversity of a fixed subpopulation. Therefore, the crossover operator has all the good attributes required for a good optimization operator.

RESULTS

This section discusses results based on the synthetic example of the Marmousi velocity model (Bourgeois et al., 1991).

Marmousi velocity model

We have taken the complicated example of the 2D Marmousi velocity model to demonstrate the robustness of the approach. The Marmousi synthetic data set (Bourgeois et al., 1991) first was released as a blind test for velocity estimation. It is a complicated acoustic 2D data set based on a profile of the North Quenguela Trough in the Cuanza Basin in Angola (Versteeg, 1993). The structural style is dominated by growth faults that arise from salt creep and cause the complicated velocity structure in the upper part of the model. The target zone is a reservoir located at a depth of about 2500 m. The model contains many reflectors, steep dips, and strong velocity variations in lateral and vertical directions (with a minimum velocity of 1500 m/s and a maximum of 5500 m/s). The synthetic data set consists of 240 shots with 96 traces each. Zero-phase source deconvolution was applied to the data used in this study as part of the preprocessing (Barut et al., 1991).

To optimize the velocity model, three types of parameter need to be set. The first type is related to representation of the model (i.e., grid size), the second to migration (number of shots to be migrated, frequency, propagation depth, etc.), and the third to optimization, with MOEA used here (number of models in population, crossover probability, εs, and εds). To optimize the Marmousi velocity model, we represent it on a regular grid. Here we present our results using grid samplings of 250×250 m and 100×100 m. The number of grid points is 432 for the 250-m grid sampling and 2700 for 100-m grid sampling. The main migration parameters are frequency range, number of shots, propagation depth sampling, maximum subsurface offset, and size of CIGs. These parameters control migration cost. Because global methods require a large number of evaluations, lower migration cost will help in reducing optimization cost.

In this work, for migration, we have used the frequency range of 10–25 Hz, every second shot, propagation depth sample of 20 m, maximum subsurface offset of 3000 m, and CIG at every 250 m. Because we are using every second shot for migration, the speed doubled. At the same time, use of a small-frequency bandwidth and a reasonable propagation depth sample also reduces migration cost significantly. As fewer shots are used, shot-related artifacts will be

present in CIGs. Here we use the robust objective function (MADS), which is affected less by data and parameter setting. As a result, these artifacts will have a negligible effect on optimization.

The MOEA-related parameters are the number of models in the population, crossover rate, coefficient of crossover, and ε-parameters in objective space. We have taken a population of 20 models and a 50% probability for crossover for synthetic and normal parents. ε-parameters, which are related to objective functions, divide the objective space into a grid and provide an opportunity to fine-tune the objective functions according to importance. The DS objective function is very effective when initial models are far from true models, but its effectiveness fades when models converge toward the true model. As a result, a coarse sampling (Figure 5) is chosen for the DS function, thereby resulting in the sorting of bad models and fast convergence. Conversely, semblance is very effective when models are close to the true one. Consequently, a fine sampling (Figure 5) is chosen for the semblance function. We consider the objective space to be the product of intervals $[0, S_{max}]$, and $[0, DS_{max}]$, where S_{max} and DS_{max} are the highest values of the semblance and differential-semblance objective functions in the initial population, respectively. Because of the contrasting behaviors of these functions, we decided to sample this space by a coarse $\varepsilon_S = S_{max}/10$ and a fine $\varepsilon_{DS} = DS_{max}/100$. Although S and DS functions measure the flatness criteria, a clever combination of the distinct properties of both functions with ε-MOEA will reduce computation cost.

For optimization initialization, first the velocity model was divided into many horizontal blocks in which random velocity was generated so that minimum velocity increases with depth, with a maximum velocity limit as high as 5500 m/s. Migration then is performed on these velocity models and the corresponding S, DS, LS, and LDS functions, and RMO and smoothing dips are calculated. The ε-dominant models are copied into the archive population from a population. The LS and LDS function also are used to create reference models. The first reference model is created from population models by the combination of panels of good LS objective functions and their velocities. The second reference model is created by the combination of panels of good LDS objective functions and their velocities of population models. After generation of the initial population, archive population, and reference models, optimization starts. The procedure is shown in Figure 3. In this process, we have not used the mutation operator because it perturbs a solution obtained by the crossover in the hope of creating an even better solution and maintaining the diversity of the solution. Because we have maintained a set of random models in the population to preserve diversity, there is no need to have a mutation operator. Hence, only the crossover operator is used for optimization in this work.

We are using two types of crossover, each of which has a 50% probability. In one crossover, we select four parents, apply RMO correction and dip smoothing on the models, and then synthesize a single new parent. This parent is a combination of panels of good LS or LDS functions and their corresponding velocities. After synthesizing this parent, simulated binary crossover is performed between the synthesized parents and archive-selected parents. SBX also is used for the second type of crossover, in which we select one random parent from the population or from reference models and cross it over with the archive solution. After crossover, new velocity models are generated, and migration is performed on them. Once migration is performed, we have the migrated image and CIGs (offset and angle). The migrated image is used for visualizing, and CIGs are used for the S or DS calculation. Here the migrated image is used to obtain information about the layers used for dip smoothing. This process reduces the variance of the velocity along the structure, thereby producing a smooth image. CIGs obtained after migration are used for RMO error estimation, and LS and LDS are calculated for each panel. We associate RMO errors and dip-smoothing information with models used before crossover. After a few hundred iterations, we end up with optimized velocity models.

If the Marmousi model is represented by a coarse grid sampling of 250- $\times$ 250-m grid size and if it is smoothed using dip information, the generated velocity models almost become equivalent to the 500- $\times$ 500-m grid size. Only 200–400 evaluations are needed to obtain an optimal population of velocity models. The computational cost for those evaluations is equivalent to or less than the cost for 50–100 migrations. Figure 6a shows the Marmousi velocity model smoothed with a 400-m window size and resampled for migration. Figure 6b shows the migrated image with this velocity model. By comparing smooth Marmousi velocity models (Figure 6a) and one of the optimized velocity models (Figure 7b), it can be seen that optimized results closely resemble the smooth Marmousi model. Here we can obtain more than one optimized solution, which helps in appraising the ambiguity and uncertainty of velocity models.

On the other hand, the 100- $\times$ 100-m migration velocity grid needs almost 2700 parameters to be optimized. One of the optimized velocity models and corresponding migrated images are shown in Figure 8a and b. Smoothing along dip reduces the variance, and the model becomes almost like a grid size of 200 m or 250 m. It requires almost 600–800 evaluations. The increase in the number of evaluations is a result of the increase in the number of unknown parameters.

No drastic changes are noticed in the migrated image obtained by the 250-m or 100-m grids, but the velocity model obtained by the 100-m grid produces more velocity variation information than that obtained from the 250-m grid. CIGs are flat in both models. The results discussed above led us to use a coarse grid representation. However, we would like velocity models to resemble geology so we

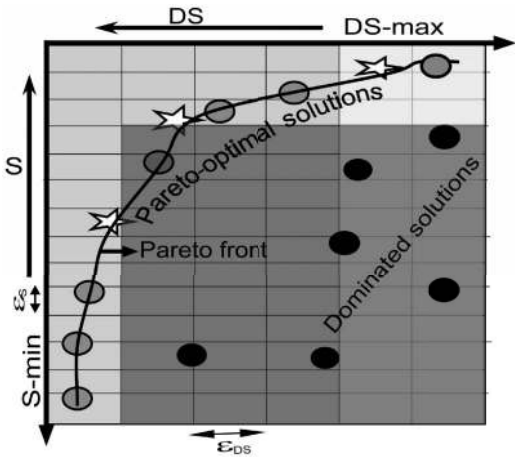

Figure 5. The ε-dominance concept and the grid of objective functions S and DS. The points represent solutions in objective space. Objectives S and DS are a grid with user-defined ε_S and ε_{DS} spacing. Allowing only one model in each cell helps to maintain diversity with a minimum number of models. Solutions shown by gray points are nondominated or Pareto-optimal solutions, and the gray curve represents the Pareto front. Stars represent ε-dominance solutions, and black points indicate dominated models. The goal of optimization is to find all ε-dominant solutions, which means obtaining a full discretization of the Pareto front.

can verify them. Although the model of 250-m grid size gives a relatively good depth image, having confidence in a model requires a profound trust in the method used to determine the model. Conversely, a 100-m grid size has produced a good image and a geologically appealing velocity model, which gives more confidence in the velocity model but at extra computational cost.

DISCUSSION

This approach differs from other global approaches (Jervis et al., 1996; Docherty et al., 1997; Mansanné, 2000) with respect to the following criteria: (1) representation of models, (2) evaluation of a model, (3) exploitation property, and (4) reproduction technique of a new model.

Jervis et al. (1996) and Docherty et al. (1997) use a binary-coded spline representation and Mansanné (2000) uses real-coded Voronoi to represent a model, whereas we use a real-coded regular grid to represent a model. We evaluate two objective functions simultaneously for a model, whereas others evaluate one objective function at a time. We have added an exploitation property by reference model, RMO correction, and directional smoothing, whereas other approaches have not exploited that information. We use only crossover without any mutation operator, whereas in other approaches, both operators were used. In our approach, before crossover, the model is corrected (RMO) and smoothed (along dip), and then only a crossover operator is used. This is a clever use of the philosophy of evolutionary algorithms, which says that good parents will produce a good child.

In Table 1, we made a comparison with respect to the number of unknown velocity parameters and the required number of PSDM for convergence. We avoided comparison on the basis of computational power. We note that although we have 10 to 50 times more unknown velocity parameters, our convergence is almost 100 times faster than that of other methods.

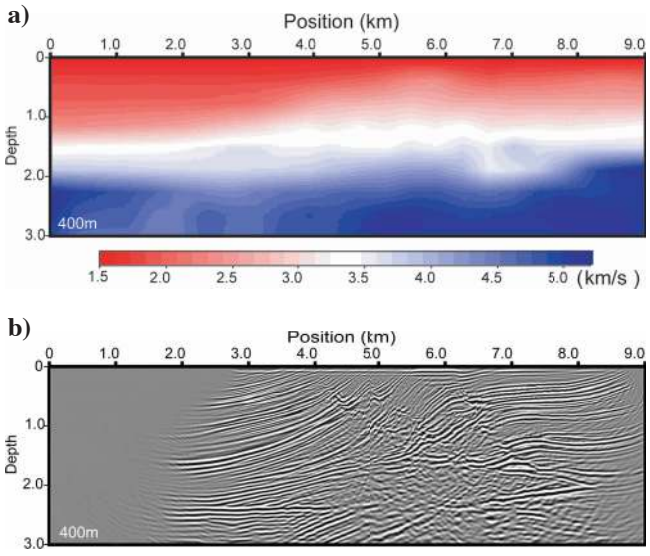

Figure 6. (a) Marmousi velocity model is smoothed with a 400-m window and then resampled for migration. (b) The corresponding migrated image.

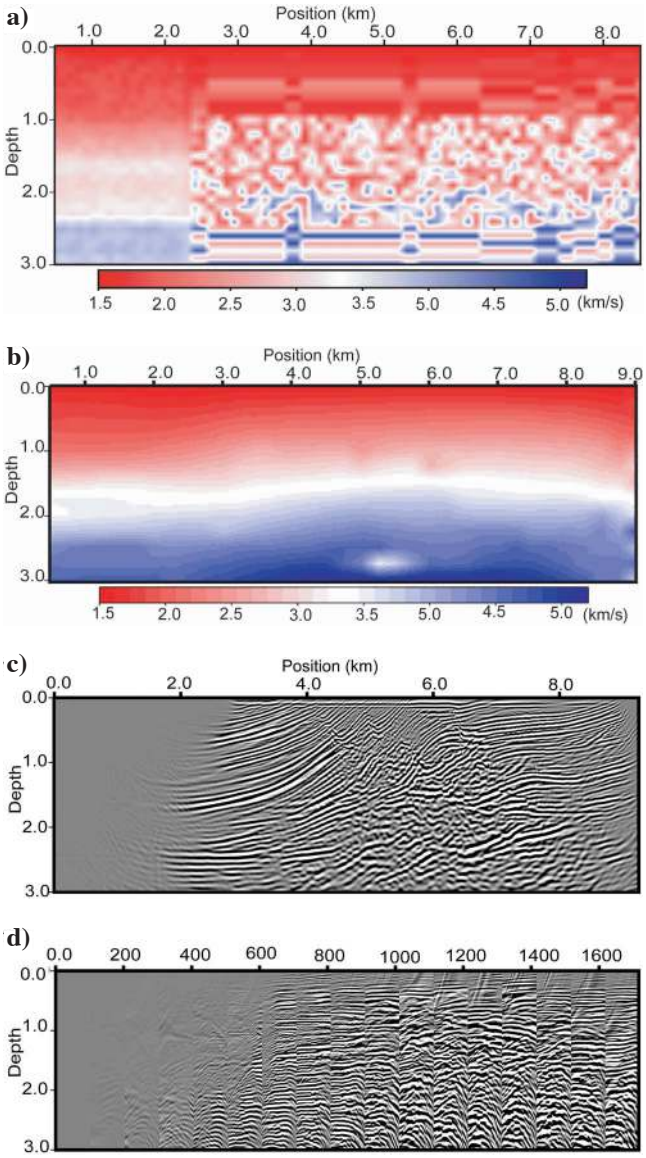

Figure 7. (a) One of the initial semirandom velocity models. (b) One of the optimized velocity models on 250-m grid sampling obtained from the initial random velocity model. This velocity model closely resembles the smoothed Marmousi model (Figure 6a). Computational cost is equivalent to that of 50–100 PSDM. (c) Corresponding migrated image. (d) Corresponding CIG does not appear flat in the complex part of the model.

Table 1. Performance comparison among various approaches.

Authors	Number of parameters	Required number of PSDM
Jervis et al. (1996)	50	2000
Docherty et al. (1997)	68	12,994
Mansanné and Schoenauer (2000)	100–200	20,000
250-m grid	457	200–400
100-m grid	2700	300–600

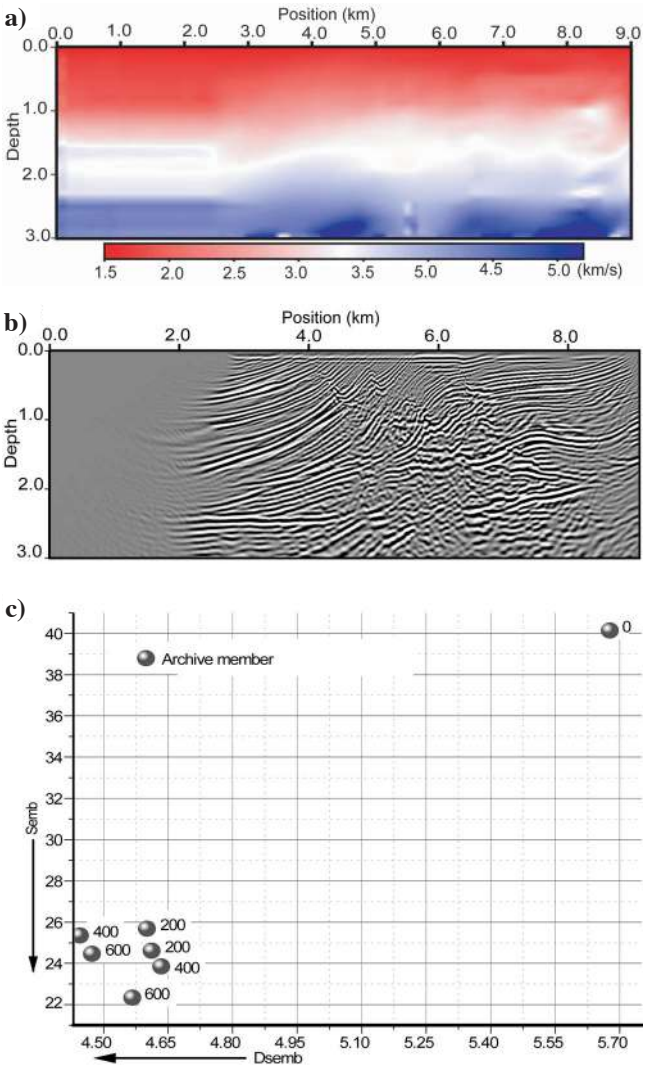

Figure 8. (a) One optimized velocity model on 100-m grid sampling obtained from the initial random velocity model. These velocity models give some geologic resemblance compared with a 250-m grid sampling (Figure 6a) at the cost of a few more PSDM (150–200 PSDM). (b) Corresponding migrated image. (c) Convergence of velocity models with respect to both criteria (semblance and differential semblance) and the number of model evaluations. Initial convergence is very fast, but after 200 evaluations, it becomes slow. We think this is the point at which it can be coupled with gradient methods to obtain a fast and refined velocity model.

CONCLUSION

Here we presented a technique of automatic velocity optimization that does not require picking of the migrated common-image gathers. To our knowledge, this is the first time multiobjective evolutionary algorithms have been used to estimate seismic velocity.

We modified the MOEA and introduced the exploitation property. We gave a new concept of the reference model and parent synthesis in MOEA to introduce the exploitation property. This property is introduced in such a way that it will not affect the global exploration property of MOEA. An effort also is made to use good properties of semblance and differential-semblance functions. This proposed technique uses RMO error and iterative methods. This technique is at least two orders of magnitude faster than global methods. By adding the exploitation property in MOEA and properly using information, the major issue of computational cost with global methods is resolved. The computational cost of this technique is equivalent to that of gradient methods. As far as we know, we have solved the highest number of geophysical parameters using global optimization methods.

The Marmousi example demonstrates that global optimization methods can be applied to realistic-scale seismic problems, at least in two dimensions. This example shows that the technique presented here is robust and can be applied to noisy data. The minor sensitivity of the objective function to data and noise adds extra robustness in optimization and boosts optimization speed.

In the future, we would like to extend our approach from two to three dimensions by using a computationally efficient migration technique. One other future extension of this research is to couple our global optimization approach with gradient methods. Our global approach can lead to the valley of global minima, whereas coupling with the gradient method can lead to the local minimum. In other words, our approach can provide robust initialization for gradient methods. It also can help in velocity model building in a geologically complex and unexplored basin.

ACKNOWLEDGMENTS

We thank Laurence Nicolétis and Karine Broto for fruitful discussions and kind assistance. We also acknowledge associate editor Tamas Nemeth and reviewers Gilles Lambaré and Gopal Palacharla for numerous comments and suggestions that improved this manuscript significantly. We thank IFP for allowing us to publish this research.

APPENDIX A

ε-MULTIOBJECTIVE EVOLUTIONARY ALGORITHMS

The ε-MOEA (Figure A-1) uses two coevolving populations: an EA population $P(t)$ and an archive population $A(t)$. The initial archive population $A(0)$ is assigned from the initial population $P(0)$ using a ε-nondominated solution. Thereafter, two solutions, one from the $P(t)$ and the other from $A(t)$, are selected using the population (tournament-selection) and archive (random-selection) procedures, and an offspring solution $O(t)$ is created. This offspring $O(t)$ then can enter into each population using the population-acceptance or archive-acceptance procedure. These loops continue until the criteria are satisfied or resources (number of iteration) are exhausted. The ε-MOEA has the following properties:

- It emphasizes a nondominated solution.
- It maintains diversity without redundancy in the archive by allowing only one solution in each hyperbox on the Pareto-optimal front.
- It is an elitist approach.
- By choosing the appropriate ε-vector, emphasis on the selected objective can be given.

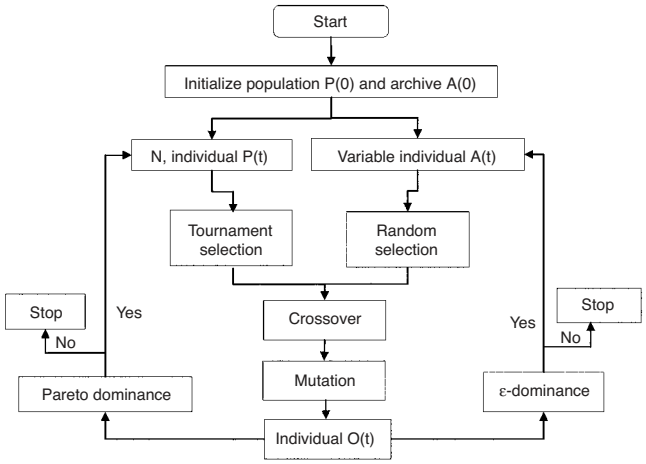

Figure A-1. Flowchart of original ε-MOEA (Deb et al., 2003). One solution from each population $P(t)$ and archive $A(t)$ is selected for a generation of offspring by crossover and mutation. After offspring $O(t)$ are evaluated, they are given their places in the population and archive only if the offspring is Pareto dominant or ε-dominant. These processes continue until the criteria are satisfied or resources are exhausted.

APPENDIX B

SIMULATED BINARY CROSSOVER (SBX)

The crossover operator combines genes of two or more parents to generate better offspring. The main purpose of a crossover operator is to recombine the partial good information from two or more parents to generate better offspring. Crossover occurs during evolution according to a user-defined crossover probability. Crossover plays a central role in evolutionary algorithms; in fact, it might be considered one of their defining characteristics.

The SBX operator for real variables was introduced by Deb and Agrawal (1995). SBX emphasizes the generating of offspring in proximity to parents. Therefore, crossover guarantees that the range of children is proportional to the range of parents and has the advantage that near-parent individuals are more likely to be chosen as children than individuals distant from parents. These crossovers are self-adaptive in the sense that the spread of the possible offspring solutions depends on the distance between parents, which decreases as the population converges.

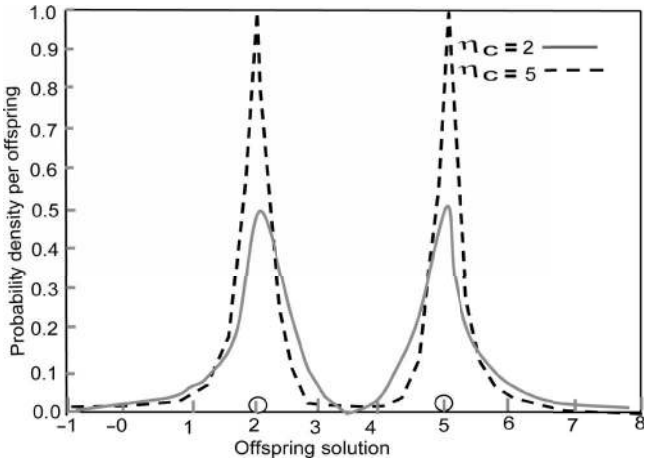

Figure B-1. A large value of η_c has a higher probability of generating offspring similar to the parents, and vice versa (Deb and Agrawal, 1995).

The procedure of computing offspring P_{1i}^{t+1} and P_{2i}^{t+1} from the parent solutions P_{1i}^{t} and P_{2i}^{t} is as follows: First a random number u_i is generated between 0 and 1. Thereafter a spread factor $\beta_i = |P_{2i}^{t+1} - P_{1i}^{t+1}|/|P_{2i}^{t} - P_{1i}^{t}|$ is calculated using the specified probability distribution function given below. These probability distribution functions are used to create an offspring using the following relation (Deb and Agrawal, 1995):

$$P(\beta_i) = 0.5(\eta_c + 1)\beta_i^{\eta_c}, \text{ if } \beta_i \leq 1;$$

$$P(\beta_i) = 0.5(\eta_c + 1)/\beta_i^{\eta_c}, \text{ otherwise.}$$

η_c (distribution index) is a nonnegative real number. β_i is calculated by equating the area under the curve equal to u_i using the following relation:

$$\beta_i = (2u_i)_c^{1/\eta+1} \text{ if } u_i \leq 0.5;$$

$$\beta_i = (1/2(1 - u_i))_c^{1/\eta+1} \text{ otherwise.}$$

After obtaining β_i from the above relation, the offspring is calculated as follows: A large value of η_c gives a higher probability to generate near-parent offspring (Figure B-1).

REFERENCES

Al-Yahya, K. M., 1989, Velocity analysis by iterative profile migration: Geophysics, **54**, 718–729.

Barut, M., E. de Bazelaire, J. Ravat, and G. Taveau, 1991, A methodology to image tectonically complex areas using polystack and prestack depth migration, *in* R. Versteeg, and G. Grau, eds., The Marmousi experience: EAGE, 159–167.

Biondi, B., 2003, 3-D seismic imaging, http://www.stanford.edu/sep/biondo/Lectures/index.html, accessed 1 May 2005.

Biondi, B., and W. Symes, 2004, Angle-domain common-image gathers for migration velocity analysis by wavefield-continuation methods: Geophysics, **69**, 1283–1298.

Brandsberg-Dahl, S., M. V. de Hoop, and B. Ursin, 1999, Velocity analysis in the common scattering-angle/azimuth domain: 69th Annual International Meeting, SEG, Expanded Abstracts, 1222–1223.

Bourgeois, A., M. Bourget, P. Lailly, M. Poulet, P. Ricarte, and R. Versteeg, 1991, Marmousi, model and data, *in* R. Versteeg, and G. Grau, eds., The Marmousi experience: EAGE, 5–16.

Chauris, H., and M. Noble, 1998, Testing the behaviors of differential semblance for velocity estimation: 68th Annual International Meeting, SEG, Expanded Abstracts, 1305–1308.

Clapp, R. G., B. Biondi, and J. F. Claerbout, 2004, Incorporating geologic information into reflection tomography: Geophysics, **69**, 533–546.

Coello, C. A., 2003, EMOO, http://www.lania.mx/~ccoello/EMOO/, accessed 14 November 2006.

Coello, C. A., and G. T. Pulido, 2001, Multi-objective optimization using a micro-genetic algorithm, *in* L. Spector, E. D. Goodman, A. Wu, W. B. Langdon, H. M. Voigt, M. Gen, S. Sen, M. Dorigo, S. Pezeshk, M. H. Garzon, and E. Burke, eds., Proceedings of the Genetic and Evolutionary Computation Conference: Morgan Kaufmann Publishers, 274–282.

Coello, C. A., D. A. Van Veldhuizen, and G. Lamont, 2002, Evolutionary algorithms for solving multi-objective problem: Morgan Kaufmann Publishers.

Dahlen, F., S. H. Hung, and G. Nolet, 2000, Fréchet kernels for finite-frequency travel times, I: Theory: Geophysical Journal International, **141**, 157–174.

Deb, K., 2001, Multi-objective optimization using evolutionary algorithms: John Wiley and Sons, Inc.

Deb, K., and R. B. Agrawal, 1995, Simulated binary crossover for continuous search space: Complex Systems, **9**, 115–148.

Deb, K., and D. Goldberg, 1991, A comparative analysis of selection schemes used in genetic algorithms, *in* G. J. Rawlins, ed., Foundations of genetic algorithms: Morgan Kaufmann Publishers, 69–93.

Deb, K., M. Mohan, and S. Mishra, 2003, A fast multi-objective evolutionary

algorithm for finding well-spread Pareto-optimal solutions: KanGAL, Internal Report No. 2003002.
Deb, K., A. Pratap, S. Agrawal, and T. Meyarivan, 2002, A fast and elitist multi-objective genetic algorithm: NSGA-II: IEEE Transactions on Evolutionary Computation, **6**, 181–197.
Delprat-Jannaud, F., and P. Lailly, 1992, What information on the earth model do reflection travel times provide?: Journal of Geophysical Research, **97**, 19827–19844.
Devaney, J., and L. M. Oristaglio, 1984, Geophysical diffraction tomography: 61st Annual International Meeting, SEG, Expanded Abstracts, 330–333.
Docherty, P., R. Silva, S. Singh, Z. Song, and M. Wood, 1997, Migration velocity analysis using a genetic algorithm: Geophysical Prospecting, **45**, 865–878.
Jervis, M., M. K. Sen, and P. L. Stoffa, 1996, Prestack migration velocity estimation using nonlinear methods: Geophysics, **61**, 138–150.
Jin, S., and R. Madariaga, 1994, Nonlinear velocity inversion by a two-step Monte Carlo method: Geophysics, **59**, 577–590.
Kaipio, J. P., V. Kolehmainen, M. Vauhkone, and E. Somersalo, 1999, Inverse problems with structural prior information: Inverse Problems, **15**, 713–729.
Lafond, C. F., and A. R. Levander, 1993, Migration moveout analysis and depth focusing: Geophysics, **58**, 91–100.
Lee, W. B., and Lin Zhang, 1992, Residual shot profile migration: Geophysics, **57**, 815–822.
Liu, Z., and N. Bleistein, 1995, Migration velocity analysis: Theory and an iterative algorithm: Geophysics, **60**, 142–153.
Luo, Y., and G. T. Schuster, 1991, Wave equation traveltime inversion: Geophysics, **56**, 645–653.
Lynn, W. S., and J. F. Claerbout, 1982, Velocity estimation in laterally varying media: Geophysics, **47**, 884–897.
Mansanné, F., 2000. Analyse d'algorithmes d'évolution artificielle appliqués au domaine pétrolier: Ph.D. thesis, Université de Pau.
Mansanné, F., F. Carrère, A. Ehinger, and M. Schoenauer, 1999. Evolutionary algorithms as fitness function debuggers, *in* Z. W. Ras, and A. Skowron, eds., Foundation of intelligent systems: Springer-Verlag, 639–647.
Mulder, W., and A. ten Kroode, 2002, Automatic velocity analysis by differential semblance optimization: Geophysics, **67**, 1184–1191.
Plessix, R. E., W. A. Mulder, and A. P. E. ten Kroode, 2000, Automatic crosswell tomography by semblance and differential semblance optimization: Theory and gradient computation: Geophysical Prospecting, **48**, 913–935.
Pratt, R. G., 1999, Seismic waveform inversion in the frequency domain, Part 1: Theory and verification in a physical scale model: Geophysics, **64**, 888–901.
Pratt, R. G., and W. Symes, 2002, Semblance and differential semblance optimization for waveform tomography: A frequency domain implementation: Journal of Conference Abstracts, **7**, 183–184.
Rickett, J., and P. Sava, 2002, Offset and angle-domain common image-point gathers for shot-profile migration: Geophysics, **67**, 883–889.
Sava, P., 2003, Prestack residual migration in the frequency domain: Geophysics, **67**, 634–640.
Sava, P., and B. Biondi, 2004, Wave-equation migration velocity analysis, I: Theory: Geophysical Prospecting, **52**, 593–606.
Sava, P., B. Biondi, and J. Etgen, 2005, Wave-equation migration velocity analysis by focusing diffractions and reflections: Geophysics, **70**, no. 3, U19-U27.
Schaffer, J. D., 1985, Multiple objective optimization with vector evaluated genetic algorithms, *in* J. J. Grefenstette, ed., Proceedings of the First International Conference on Genetic Algorithms: Lawrence Erlbaum Associates, 93–100.
Shen, P., 2003, Differential semblance velocity analysis by wave-equation migration: 73rd Annual International Meeting, SEG, Expanded Abstracts, 2132–2135.
——–, 2004, Seismic velocity analysis by wave-equation migration, Ph.D. dissertation, Rice University.
Singh, V. P., B. Duquet, M. Léger, and M. Schoenauer, 2006, Some practical aspects of migration velocity analysis: Modified differential semblance objective function: 68th Conference and Exhibition, EAGE, Extended Abstracts, G004.
Singh, V. P., M. Schoenauer, and M. Léger, 2005, A geologically sound representation for evolutionary multi-objective subsurface identification: IEEE Congress of Evolutionary Computation, **3**, 2325–2332.
Sobel, I., and G. Feldman, 1968, A 3×3 isotropic gradient operator for image processing: Presented at the Stanford Artificial Project. Published *in* R. Duda, and P. Hart, 1973, Pattern classification and scene analysis: John Wiley and Sons, 271–272.
Stolk, C. C., and W. Symes, 2004. Kinematic artifacts in prestack depth migration: Geophysics, **69**, 562–575.
Symes, W., and J. Carazzone, 1991, Velocity inversion by differential semblance optimization: Geophysics, **56**, 654–663.
Tarantola, A., 1984, Inversion of seismic reflection data in the acoustic approximation: Geophysics, **49**, 1259–1266.
Versteeg, R. J., 1993, Sensitivity of prestack depth migration to the velocity model: Geophysics, **58**, 873–882.
Woodward, M. J., 1992, Wave-equation tomography: Geophysics, **57**, 15–26.
Yan, L., and L. R. Lines, 2001, Seismic imaging and velocity analysis for an Alberta Foothills seismic survey: Geophysics, **66**, 721–732.
Zhu, J., L. R. Lines, and S. H. Gray, 1998, Smiles and frowns in migration/velocity analysis: Geophysics, **63**, 1200–1209.
Zitzler, E., M. Laumanns, and L. Thiele, 2001, SPEA2: Improving the performance of the strength Pareto evolutionary algorithm: Swiss Federal Institute of Technology (ETH), TIK Report no. 103.