

Simulated Annealing for Sequential Pattern Detection and Seismic Applications

Kou-Jen Huang, Kou-Yuan Huang, *Senior Member, IEEE*, I-Chieh Chen, and Luke K. Wang

Abstract—Sequential pattern detection with simulated annealing (SA) is adopted to estimate parameters and detect lines, ellipses, hyperbolas type by type, and patterns by patterns in each type. The motivation of the sequential detection method is to deal with multiple patterns. The parameters of a pattern are formed as a vector and used as a state, and adjusted in SA. A sequential detection algorithm using SA to detect patterns is proposed. It detects one or a small number of patterns at each step. SA has the capability of the global minimization. The six parameters of patterns are adjusted sequentially step by step. The computation can converge efficiently. In the experiment, the result of sequential detection is better than that of synchronous detection in detecting a large number of patterns. In sequential detection, detection of one pattern at each step can have less computation time and good convergence in total detection than using two or more pattern detections. In simulated seismic data, SA is applied to detect the hyperbolas in the common depth point (CDP) gather. In real one-shot seismogram, SA is applied to detect lines of direct wave and hyperbolas of reflection wave. The results can show that the proposed method is feasible.

Index Terms—Ellipse, hyperbola, line, parametric pattern detection, seismic patterns, sequential detection, simulated annealing (SA).

I. INTRODUCTION

PATTERN detection was very important in computer vision. Usually the Hough transform (HT) was used to detect patterns in images [1]. The generalized HT (GHT) and several efficient GHT methods were ever proposed to detect the patterns which can be lines, circles, ellipses, and arbitrary shapes [2], [3], but no one efficient method had both the less memory requirements and the less computational complexity. The hyperbolas were not detected.

In seismic exploration data, a one-shot seismogram contains line pattern from direct wave and hyperbolic pattern from reflection wave [4]–[6]. Huang *et al.* employed the HT to detect line pattern from direct wave and hyperbolic pattern from reflection wave [7]. However, there was a problem in

determining the peaks in the parameter space. The memory requirement was a problem in the parameter space also.

In 2002, the HT neural network (HTNN) was proposed to detect lines, circles, and ellipses, but the hyperbolas were not detected also [8]. In 2006, Huang *et al.* adopted the HTNN to detect the parametric patterns including hyperbolas [9], but the HTNN used the gradient descent method that had the local minimum problem.

The simulated annealing (SA) was first proposed by Kirkpatrick *et al.* [10]. Its global optimization capability relied on the Metropolis criterion [11]. There were some applications of SA to the geoscience and remote sensing and the results were good [12]–[15]. There were some seismic applications of SA and the results were good also [16]–[20]. Huang and Chen had ever applied the SA for pattern detection and seismic applications [21], [22]. A small number of patterns, at most four, were detected synchronously, but there was no experiment for a large number of patterns.

The SA is a global optimization method. Here, we use it for pattern detection and seismic applications. There are two kinds of processing. One is the synchronous detection that detects all parametric patterns at the same time, but there is a convergence problem and the detection result is not good for a large number of patterns. The other is the sequential detection that detects one or a small number of patterns at each step and the final detection result is good. Sequential detection method can overcome the convergence problem of synchronous detection. Taking the advantages of SA and sequential detection method, we propose the SA detection system to detect the patterns with sequential procedure. Also, we apply the SA to detect hyperbolas in the seismic common depth point (CDP) gather and lines from direct wave and hyperbolas from reflection wave in the real seismic data. The CDP gather was also called common midpoint (CMP) gather [5], [6].

II. SEQUENTIAL PATTERN DETECTION

We sequentially detect the parametric patterns type by type that includes lines, ellipses, and hyperbolas. The ellipse includes circle. Each pattern is represented by a set of parameters. The SA can estimate parameters of each pattern.

The procedure of sequential pattern detection is shown in Fig. 1. First, there are N points in the image. The SA parameter estimation system can detect a set of parameter vectors of one type. One or some patterns are detected. Then, the system deletes the detected patterns and the corresponding points that are close to them and repeats detection on the

Manuscript received January 26, 2014; revised June 04, 2014; accepted July 16, 2014. Date of publication August 12, 2014; date of current version January 21, 2015. This work was supported in part by the National Science Council, Taiwan, under Grant NSC98-2221-E-009-144 and Grant NSC101-2221-E-009-147.

K.-J. Huang and L. K. Wang are with the Department of Electrical Engineering, National Kaohsiung University of Applied Sciences, Kaohsiung 80778, Taiwan (e-mail: u568372@taipower.com.tw; lwang@mail.ee.kuas.edu.tw).

K.-Y. Huang is with the Department of Computer Science, National Chiao Tung University, Hsinchu 30010, Taiwan (e-mail: kyhuang@cs.nctu.edu.tw).

I.-C. Chen is with the Department of Electrical Engineering, National Tsing Hua University, Hsinchu 30013, Taiwan (e-mail: lawliet357@gmail.com).

Color versions of one or more of the figures in this paper are available online at <http://ieeexplore.ieee.org>.

Digital Object Identifier 10.1109/JSTARS.2014.2344756

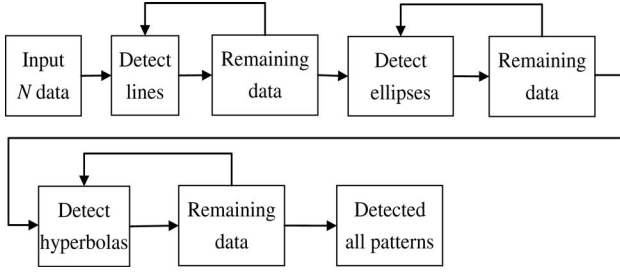


Fig. 1. Sequential pattern detection.

TABLE I
RELATION BETWEEN SHAPE AND PARAMETERS IN (2)

a	b	f	Shape
+	+	+	Ellipse
-	-	-	Ellipse
+	-	+	Hyperbola
-	+	+	Hyperbola
+	-	-	Hyperbola
-	+	-	Hyperbola
+	-	0	Asymptote
-	+	0	Asymptote
-	-	+	No graph
+	+	-	No graph
+	+	0	Point
-	-	0	Point

remaining points. Until all patterns of one type are detected, it changes to detect the next type. After detecting all types, the system stops.

We define system error. The SA algorithm can estimate the parameters of all patterns with minimum error.

A. Parametric Patterns

For ellipse or hyperbola detection, the standard equation is expressed as

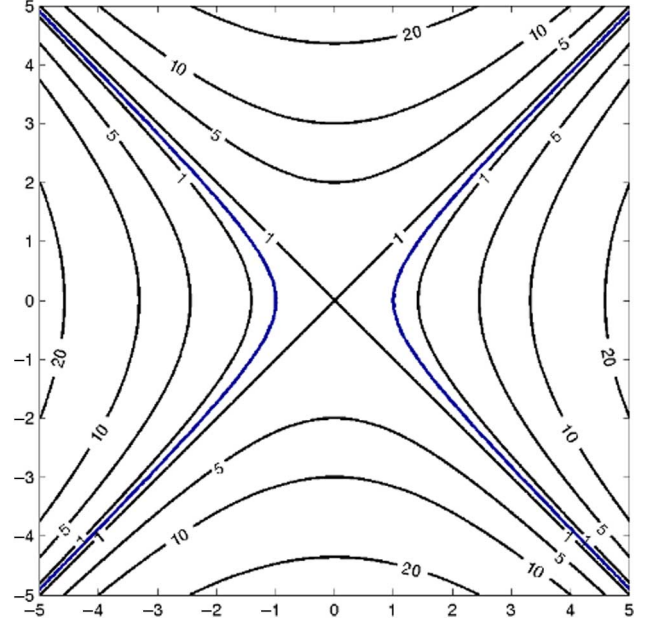
$$\frac{x'^2}{a'^2} \pm \frac{y'^2}{b'^2} = 1. \quad (1)$$

The shape is transferred through translation, rotation, and scaling. The equation is expressed as

$$a[(x - m_x)\cos\theta + (y - m_y)\sin\theta]^2 + b[-(x - m_x)\sin\theta + (y - m_y)\cos\theta]^2 = f \quad (2)$$

where (m_x, m_y) is the center, a and b are the shape parameters, θ is the rotation angle, and f is the size of the pattern. Table I lists the relation between the shape of the equation and parameters a , b , and f . The symbol + is the positive number and the symbol - is the negative number.

In vector form, a parameter vector $\mathbf{p} = [m_x \ m_y \ a \ b \ \theta \ f]^T$ represents a pattern (ellipse or hyperbola). For the k th pattern, $\mathbf{p}_k = [m_{k,x} \ m_{k,y} \ a_k \ b_k \ \theta_k \ f_k]^T$, and for all K patterns, the matrix $\mathbf{P} = [\mathbf{p}_1 \ \mathbf{p}_2 \ \dots \ \mathbf{p}_k \ \dots \ \mathbf{p}_K]$. We set f as a positive number. The parameters a , b , and f can show types of patterns as listed in Table I.

Fig. 2. Equal radial distance curves to a hyperbola $x^2 - y^2 = 1$.

For line, the equation is expressed as

$$ax + by + c = 0. \quad (3)$$

A parameter vector $\mathbf{p} = [a \ b \ c]^T$ represents a line pattern. For the k th pattern, $\mathbf{p}_k = [a_k \ b_k \ c_k]^T$, and for all K line patterns, the matrix $\mathbf{P} = [\mathbf{p}_1 \ \mathbf{p}_2 \ \dots \ \mathbf{p}_k \ \dots \ \mathbf{p}_K]$.

B. System Error

Considering N points from K patterns in an image, for the error (distance and energy) of the system, at first we calculate the distance from a point to a pattern. Then, we calculate the distance from a point to all K patterns. Finally, we calculate the distance from all N points to all K patterns as the system error.

1) *Distance from a Point to a Pattern:* For detection of ellipses and hyperbolas, the distance from a point $\mathbf{x}_i = [x_i \ y_i]^T$ to the k th pattern is defined as $d_k(\mathbf{x}_i)$

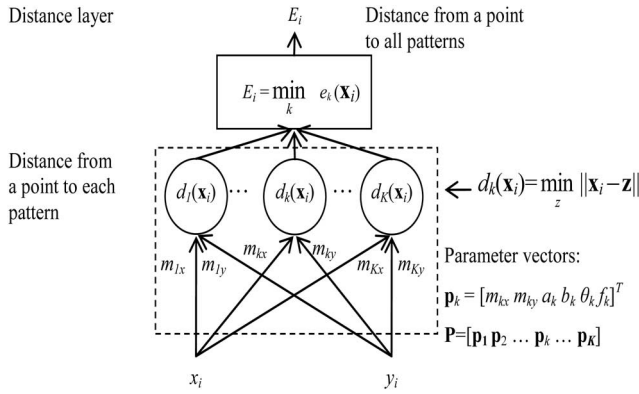
$$d_k(\mathbf{x}_i) = \min_{\mathbf{z}} \|\mathbf{x}_i - \mathbf{z}\| \quad (4)$$

where \mathbf{z} is a point on the (2) of the k th pattern. From (4), the point \mathbf{z} on the k th pattern is the nearest to \mathbf{x}_i . $d_k(\mathbf{x}_i)$ is the length of the perpendicular line segment from \mathbf{x}_i to (2) [23]. In the implementation, we estimate (4) by sampling sufficiently enough points, e.g., 1000 points, from (2), and find the nearest point \mathbf{z} to \mathbf{x}_i .

The other kind of distance from a point $\mathbf{x}_i = [x_i \ y_i]^T$ to the k th pattern could also be defined in [21] and [22] as

$$d_k(\mathbf{x}_i) = |a_k[(x_i - m_{k,x})\cos\theta_k + (y_i - m_{k,y})\sin\theta_k]^2 + b_k[-(x_i - m_{k,x})\sin\theta_k + (y_i - m_{k,y})\cos\theta_k]^2 - f_k|. \quad (5)$$

It was a radial distance [23]. Fig. 2 shows the equal distance curves of a hyperbola. The equal distance curves were very close to the four far away sides of a hyperbola. The point


 Fig. 3. Steps of distance calculation from a point to all K patterns.

was near to the side of hyperbola, but the distance was very big. Because the distance defined in (5) becomes sensitive at some particular positions, here we use (4) in the distance computation.

For line detection, the distance from a point $\mathbf{x}_i = [x_i \ y_i]^T$ to the k th line pattern is defined as

$$d_k(\mathbf{x}_i) = d_k(x_i, y_i) = \frac{|a_k x_i + b_k y_i + c_k|}{\sqrt{a_k^2 + b_k^2}}. \quad (6)$$

Also, we can use (2) to detect line. The line is one of the asymptotes of a hyperbola. We set $f = 0$. For a and b , one is positive and the other is negative.

2) *Error from a Point to All Patterns*: Error (distance and energy) from a point to all patterns is defined as the minimal distance from the point to all patterns. The error of the i th point \mathbf{x}_i to all K patterns is defined as E_i

$$E_i = \min_k e_k(\mathbf{x}_i) \\ e_k(\mathbf{x}_i) = -\exp(-d_k^2(\mathbf{x}_i)/q) \quad (7)$$

where q is a parameter to adjust the sensitivity of the k th pattern to the outliers. Equation (7) can reduce the effect of outliers and still keep the error for minimization in the following system error (8). Fig. 3 shows the calculating procedure for the distance of a point to all K patterns. The distance layer computes the distance from a point to each pattern by (4) or (6), and the error layer outputs the minimum error from a point to all K patterns by (7).

3) *System Error From All Points to All Patterns*: The system error is defined as the average of errors from all N points to all K patterns

$$E = \frac{1}{N} \sum_{i=1}^N E_i. \quad (8)$$

The calculating procedure of system error is shown in Fig. 4.

C. SA for Parameter Estimation

We use SA to estimate parameters and detect patterns. To detect lines, we use the line equation in (3). To detect ellipses or circles, we use (2) and set $a > 0$, $b > 0$, and $f > 0$. To detect hyperbolas, we use (2) and set $a < 0$, $b > 0$, and $f > 0$ in order to detect the seismic reflection patterns.

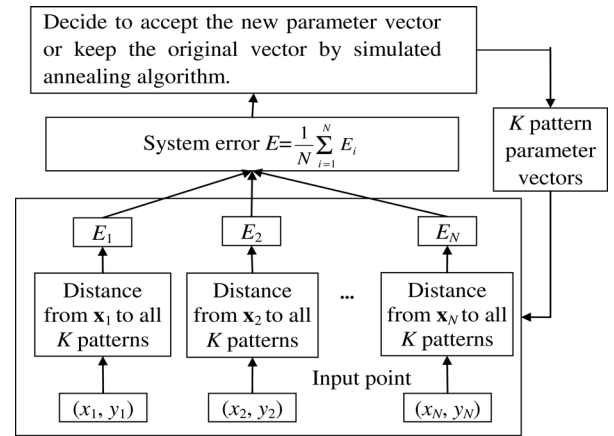


Fig. 4. Steps of calculating system error and SA.

The goal of using SA is to find a set of parameter vectors that can globally minimize the error of the system. We use the temperature decreasing function $T(t)$ by [10]

$$T(t) = T_{max} \times \beta^{(t-1)}, \quad \text{for } t = 1, 2, 3, \dots, \quad 0 < \beta < 1 \quad (9)$$

where t is the temperature step index.

Adjusting all parameters of a pattern at a time is not efficient in convergence [8]. We adjust parameters sequentially step by step. For line, the order is a and b for slope, and then c for intercept. For ellipse or hyperbola, the order is the center (m_x, m_y) , the shape parameters a and b , the rotation angle θ , and the size f .

Algorithm 1 SA algorithm to Estimate Parameters of K Patterns (Lines, Ellipses, or Hyperbolas)

Input: N points in an image. Set K as the number of patterns and N_s as the temperature cooling steps.

Output: A set of K parameter vectors.

Step 1: Initialization.

In the initial step $t = 1$, choose T_{max} as $T(1)$ at high temperature. Initialize β and $0 < \beta < 1$. Define the temperature decreasing function as in (9)

$$T(t) = T_{max} \times \beta^{(t-1)}.$$

Initialize parameter vectors of K patterns, $\mathbf{p}_1 \mathbf{p}_2 \dots \mathbf{p}_K \dots \mathbf{p}_K$, where $\mathbf{p}_k = [a_k \ b_k \ c_k]^T$ for lines and $p_k = [m_{kx} \ m_{ky} \ a_k \ b_k \ \theta_k \ f_k]^T$ for ellipses or hyperbolas. One \mathbf{p} is for one pattern and set $\mathbf{P} = [\mathbf{p}_1 \ \mathbf{p}_2 \ \dots \ \mathbf{p}_k \ \dots \ \mathbf{p}_K]$ for all K patterns.

Calculate energy $E(\mathbf{P})$ as that in (8).

Step 2: Randomly change parameter vectors and decide the new parameter vectors at a temperature or in one cooling step.

For $m = 1$ to N_t (N_t trials at a temperature)

For $k = 1$ to K (k is the index of the pattern)

To detect lines, go to Step 2A; to detect ellipses and hyperbolas, go to Step 2B.

Step 2A for line detection:

Start a trial, including the following steps (a) and (b).

(a) Randomly change the slope parameters:

$$[a'_k \ b'_k]^T = [a_k \ b_k]^T + \alpha_{ab} \mathbf{s} \quad (10)$$

where $\mathbf{s} = [s_1 s_2]^T$ is a 2×1 random vector, s_1 and s_2 are Gaussian random variables with $N(0, 1)$, and α_{ab} is a constant.

Now $\mathbf{p}'_k = [a'_k b'_k c'_k]^T$ and $\mathbf{P}' = [\mathbf{p}_1 \mathbf{p}_2 \dots \mathbf{p}'_k \dots \mathbf{p}_K]$. Calculate the new energy $E(\mathbf{P}')$ from N points to K patterns and use Metropolis criterion to decide whether or not to accept \mathbf{P}' . If the new energy is less than or equal to the original one $\Delta E = E(\mathbf{P}') - E(\mathbf{P}) \leq 0$, then accept \mathbf{P}' . Otherwise, the new energy is higher than the original one, $\Delta E = E(\mathbf{P}') - E(\mathbf{P}) > 0$. In this case, compute $prob = \exp[-\Delta E/T(t)]$. Generate a random number r uniformly distributed over $(0, 1)$. If $prob \geq r$, accept \mathbf{P}' ; otherwise, reject it and keep previous \mathbf{P} .

(b) Randomly change the intercept parameter:

$$c'_k = c_k + \alpha_c s \quad (11)$$

where s is a Gaussian random variable with $N(0, 1)$ and α_c is a constant.

Now $\mathbf{p}'_k = [a_k b_k c'_k]^T$ and $\mathbf{P}' = [\mathbf{p}_1 \mathbf{p}_2 \dots \mathbf{p}'_k \dots \mathbf{p}_K]$. Similar to Step 2A(a), calculate the new energy $E(\mathbf{P}')$ from N points to K patterns and use Metropolis criterion to decide whether or not to accept \mathbf{P}' .

Step 2B for ellipse and hyperbola detection:

Start a trial, including the following steps (a)–(d).

(a) Randomly change the center of the k th pattern:

$$[m'_{kx} m'_{ky}]^T = [m_{kx} m_{ky}]^T + \alpha_m \mathbf{s} \quad (12)$$

where $\mathbf{s} = [s_1 s_2]^T$ is a 2×1 random vector, s_1 and s_2 are Gaussian random variables with $N(0, 1)$, and α_m is a constant.

Now $\mathbf{p}'_k = [m'_{kx} m'_{ky} a_k b_k \theta_k f_k]^T$ and $\mathbf{P}' = [\mathbf{p}_1 \mathbf{p}_2 \dots \mathbf{p}'_k \dots \mathbf{p}_K]$. Similar to Step 2A(a), calculate the new energy $E(\mathbf{P}')$ from N points to K patterns and use Metropolis criterion to decide whether or not to accept \mathbf{P}' .

(b) Randomly change the shape parameters:

$$[a'_k b'_k]^T = [a_k b_k]^T + \alpha_{ab} \mathbf{s} \quad (13)$$

where $\mathbf{s} = [s_1 s_2]^T$ is a 2×1 random vector, s_1 and s_2 are Gaussian random variables with $N(0, 1)$, and α_{ab} is a constant.

Note that for detecting ellipses, we set $a'_k > 0$ and $b'_k > 0$. For detecting hyperbolas, we set $a'_k > 0$ and $b'_k < 0$. Now $\mathbf{p}'_k = [m_{kx} m_{ky} a'_k b'_k \theta_k f_k]^T$ and $\mathbf{P}' = [\mathbf{p}_1 \mathbf{p}_2 \dots \mathbf{p}'_k \dots \mathbf{p}_K]$. Similar to Step 2A(a), calculate the new energy $E(\mathbf{P}')$ from N points to K patterns and use Metropolis criterion to decide whether or not to accept \mathbf{P}' .

(c) Randomly change the angle:

$$\theta'_k = \theta_k + \alpha_\theta s \quad (14)$$

where s is a Gaussian random variable with $N(0, 1)$, and α_θ is a constant.

Now $\mathbf{p}'_k = [m_{kx} m_{ky} a_k b_k \theta_k f'_k]^T$ and $\mathbf{P}' = [\mathbf{p}_1 \mathbf{p}_2 \dots \mathbf{p}'_k \dots \mathbf{p}_K]$. Similar to Step 2A(a), calculate the new energy $E(\mathbf{P}')$ from N points to K patterns and use Metropolis criterion to decide whether or not to accept \mathbf{P}' .

(d) Randomly change the size:

$$f'_k = |f_k + \alpha_f s| \quad (15)$$

where s is a Gaussian random variable with $N(0, 1)$ and α_f is a constant.

Now $\mathbf{p}'_k = [m_{kx} m_{ky} a_k b_k \theta_k f'_k]^T$ and $\mathbf{P}' = [\mathbf{p}_1 \mathbf{p}_2 \dots \mathbf{p}'_k \dots \mathbf{p}_K]$. Similar to Step 2A(a), calculate the new energy $E(\mathbf{P}')$ from N points to K patterns and use Metropolis criterion to decide whether or not to accept \mathbf{P}' .

End for k (pattern)

End for m (trial)

Step 3: System Cooling.

Add 1 to t . Decrease temperature T according to the cooling function (9) $T(t) = T_{max} \times \beta^{(t-1)}$ and repeat Step 2 and 3 until $t = N_s$ cooling steps for the low temperature.

D. Sequential Pattern Detection

We detect lines, ellipses, and hyperbolas type by type sequentially. We must set the threshold of error λ_e and the threshold of point number λ_p . If the error (distance) from a point to a pattern as in (4) or (6) is less than λ_e , then the point belongs to that pattern. If a pattern contains enough points that are larger than λ_p , then the pattern is detected. This is to ensure that the pattern covers enough points instead of just being represented by a few points.

We input points to the sequential pattern detection system. One or some patterns are detected. We remove the detected patterns and their corresponding points. Then, we input the remaining data to the detection system for further detection. It repeats until all patterns of a type are detected. Then, we change to detect the patterns of the next type. When patterns of all types are detected, the system stops. The algorithm is as follows.

Algorithm 2 Sequential Pattern Detection System

Input: N points in an image. Set K_1 , K_2 , and K_3 as the number of lines, ellipses, and hyperbolas.

Output: A set of estimated parameter vectors for K_1 lines, K_2 ellipses, and K_3 hyperbolas.

Step 1: Initialization.

Set the order of pattern type: line, ellipse, and hyperbola.

Set threshold of error λ_e and threshold of point number λ_p .

Step 2: Detect one or some patterns in one type (line, ellipse, or hyperbola) by SA parameter estimation system of Algorithm 1.

Step 3: Remove the detected patterns and their corresponding points.

Step 4: Check the detection results.

If all patterns of a type are detected, go to Step 5. Otherwise, go to Step 2 and continue to detect the patterns of the same type.

Step 5: Check the detection type.

If all types are detected, stop the detection. Otherwise, change to detect the next type and go to Step 2 to detect the patterns of that type.

We use Fig. 5 as an example. We number each pattern in the image which consists of one line, two ellipses, and one hyperbola. We detect one pattern at each step. The sequence is line 1, ellipse 2, ellipse 3, and hyperbola 4 as shown in Fig. 5. Finally, all patterns can be detected type by type and pattern by pattern. The simulation is in the experiment.

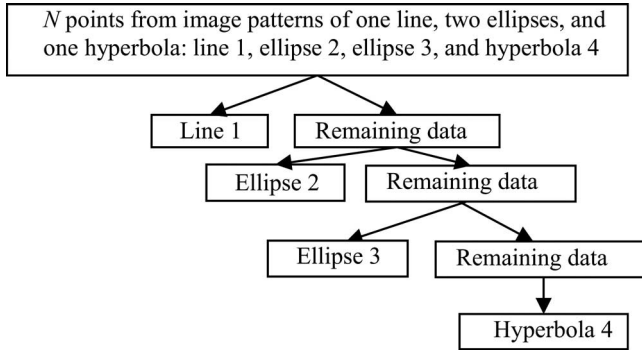


Fig. 5. Example of sequential detection.

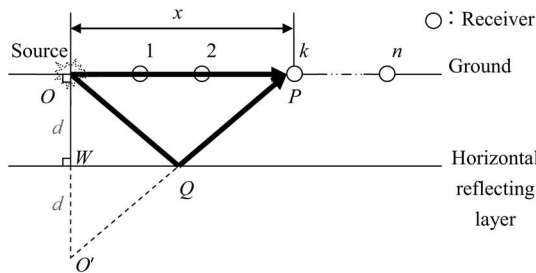


Fig. 6. Direct wave and reflection from the single horizontal layer.

III. SEISMIC PATTERNS

Seismic reflection method is important in oil and gas exploration. Seismic signal is generated by explosion. For a one shot, there are many receivers on the two sides. After explosion, there are direct P-wave along the ground surface and reflection P-wave from the subsurface reflection layers. At one-shot seismogram, we use SA to estimate parameters of line patterns of direct wave and hyperbolic patterns of reflection wave.

A. Direct Wave

From Fig. 6, the equation for the travel time t of the direct wave from a shot point to a receiver on the ground is in (16). The velocity of P-wave is v . The distance of receiver is x . It is a line in the time–distance space

$$t = \frac{x}{v}. \quad (16)$$

B. Reflection of Single Layer

After one shot, the wave propagates through the medium and is reflected back from the reflection layer. There are two cases of reflection: from the horizontal layer and from the dipping layer.

The first case is the horizontal reflection layer as shown in Fig. 6. The depth of reflection layer is d . The time of the wave reaching the k th receiver is derived in (17), and the time–distance curve is a hyperbola

$$\begin{aligned} t &= \frac{(\overline{OQ} + \overline{QP})}{v} = \frac{\overline{OP}}{v} = \frac{\sqrt{\overline{OO'}^2 + \overline{OP}^2}}{v} = \frac{\sqrt{(2d)^2 + x^2}}{v} \\ &= \sqrt{\left(\frac{2d}{v}\right)^2 + \left(\frac{x}{v}\right)^2}. \end{aligned} \quad (17)$$

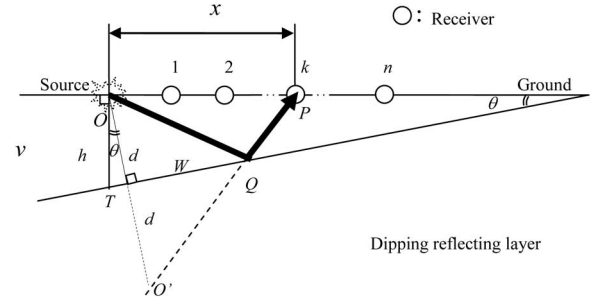


Fig. 7. Reflection from the dipping reflection layer.

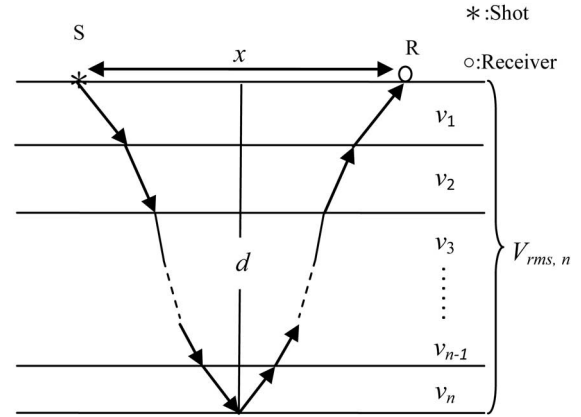


Fig. 8. Travel path of a reflected ray in a multilayer structure.

The other case is the dipping reflection layer as shown in Fig. 7. The layer dipping angle is θ . In this case, we use the trigonometric laws of cosine to derive the relation between receiving time t and the distance x in (18). After further algebraic calculation, it is also a hyperbola in (19). Note here, the hyperbola pattern from dipping reflection layer has a shift related to the hyperbola which is from horizontal reflection layer

$$\begin{aligned} t &= \frac{(\overline{OQ} + \overline{QP})}{v} = \frac{\overline{OP}}{v} = \frac{\sqrt{\overline{O'O}^2 + \overline{OP}^2} - 2 \cdot \overline{O'O} \cdot \overline{OP} \cdot \cos(\text{angle}(\angle O'OP))}{v} \\ &= \frac{\sqrt{\overline{O'O}^2 + \overline{OP}^2} - 2 \cdot \overline{O'O} \cdot \overline{OP} \cdot \cos(90^\circ - \theta)}{v} = \frac{\sqrt{(2d)^2 + x^2} - 4dx \sin \theta}{v} \end{aligned} \quad (18)$$

$$\begin{aligned} t &= \frac{\sqrt{(2d)^2 + (x - 2d \sin \theta)^2} - (2d)^2 \sin^2 \theta}{v} \\ &= \frac{\sqrt{(2d \cos \theta)^2 + (x - 2d \sin \theta)^2}}{v}. \end{aligned} \quad (19)$$

C. Reflection of Multilayer

According to Dix's equation [5], [6], [24], [25], the total travel time of the reflection $t_n(x)$ in Fig. 8 from the small offset $x(x \ll d)$ and the n layers at depth d is given by

$$t_n(x) = (4d^2 + x^2)^{1/2} / V_{rms,n} \quad (20)$$

where d is the depth for the n layers, $V_{rms,n}$ is the root-mean-square velocity from ground down to the n th layer. So the time–distance curve is also a hyperbola.

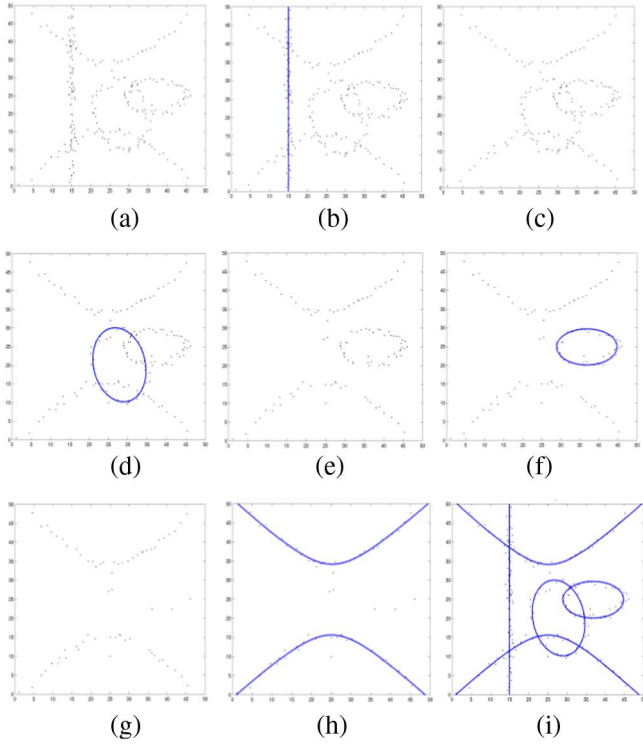


Fig. 9. Sequential detection on patterns of mixed types. First step: (a) input data and (b) detected line. Second step: (c) remaining data and (d) detected first ellipse. Third step: (e) remaining data and (f) detected second ellipse. Fourth step: (g) remaining data and (h) detected hyperbola. (i) Final detected patterns.

IV. EXPERIMENTS

In experiments, there are patterns of mixed types, 10 ellipses, real one-shot seismogram, and seismic CMP gather data. The synchronous and the sequential detections are used. There is a comparison for both detections.

A. Detection of Patterns in Mixed Types

In Fig. 9(a), there are four mixed type patterns. In this experiment, the sequential detection method is used. One pattern is detected at each step. The four steps are shown in Fig. 9. Each step shows input data, detected pattern, and remaining data. In the line detection, we set the parameters of SA system: $\alpha_{ab} = 0.2$, $\alpha_c = 4$, $\beta = 0.99$, $q = 0.2$, $T_{max} = 100$, $N_t = 20$, $N_s = 1000$, $\lambda_e = 1.2$, and $\lambda_p = 48$. In the ellipse detection, we set the parameters: $\alpha_m = 1$, $\alpha_{ab} = 1$, $\alpha_\theta = 2$, $\alpha_f = 2$, $\beta = 0.99$, $q = 0.2$, $T_{max} = 100$, $N_t = 20$, $N_s = 1000$, $\lambda_e = 1.2$, and $\lambda_p = 30$. In the hyperbola detection, we set the parameters: $\alpha_m = 1$, $\alpha_{ab} = 4$, $\alpha_\theta = 0.5$, $\alpha_f = 6$, $\beta = 0.99$, $q = 0.2$, $T_{max} = 100$, $N_t = 20$, $N_s = 1000$, $\lambda_e = 1.2$, and $\lambda_p = 30$. The detection results of four steps are shown in Fig. 9.

B. Detection in a Large Number of Patterns

We do experiments on detecting a large number of patterns using both synchronous detection and sequential detection

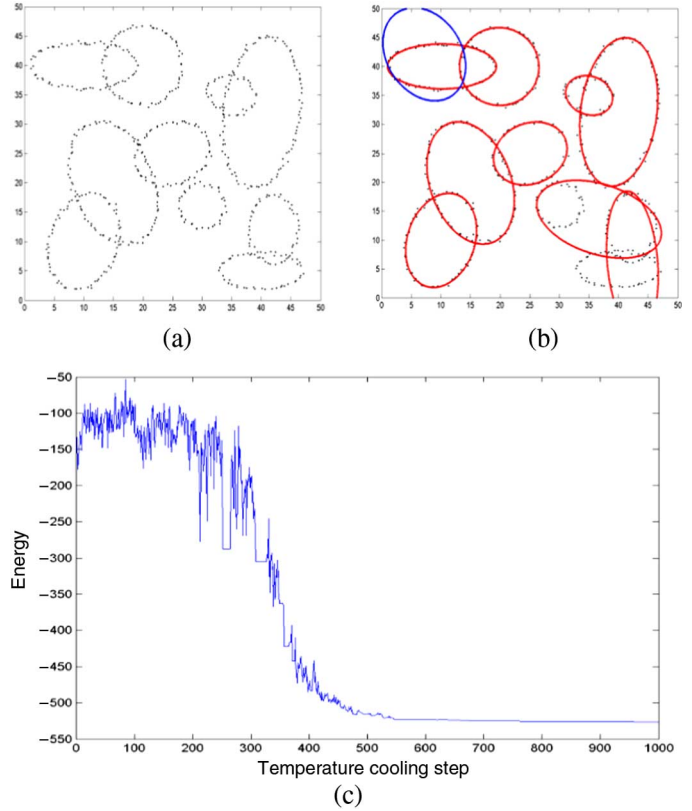


Fig. 10. Synchronous detection on 10 ellipses. (a) Input data. (b) Detected patterns. (c) Energy versus temperature cooling step.

and make a comparison. In Fig. 10, the simulated patterns include 10 ellipses. Each point is added with Gaussian noise $N(0, 0.2) \times N(0, 0.2)$, and every ellipse contains at least 30 points.

1) *Synchronous Detection*: We use the synchronous detection to detect 10 ellipses in Fig. 10(a). We set the parameters of SA system: $\alpha_m = 1$, $\alpha_{ab} = 1$, $\alpha_\theta = 2$, $\alpha_f = 2$, $\beta = 0.99$, $q = 0.2$, $T_{max} = 100$, $N_t = 20$, and $N_s = 1000$, $\lambda_e = 1.2$, and $\lambda_p = 30$. Fig. 10(b) shows the detection result. Fig. 10(c) shows the energy versus temperature cooling step. Only seven ellipses are correctly detected. The detection result is not good. The reason is that 60 parameters of 10 ellipses must be determined synchronously and there is also a problem of interference and the convergence.

We take the detected patterns from the minimum error in the 1000 temperature cooling steps, not at the last temperature. All the experiments have the same procedure.

2) *Sequential Detection*: We use the sequential detection to detect 10 ellipses in Fig. 11(a), same as Fig. 10(a). One pattern is detected at each step. We set that the parameters are the same as that of the synchronous detection. There are 10 steps in the sequential detection. We just show the results of two steps in the following. They are shown in Figs. 11–13.

Fig. 11(b) shows the result at the first step where the pattern close to the points is detected. Fig. 11(c) shows the plot of energy versus temperature cooling step.

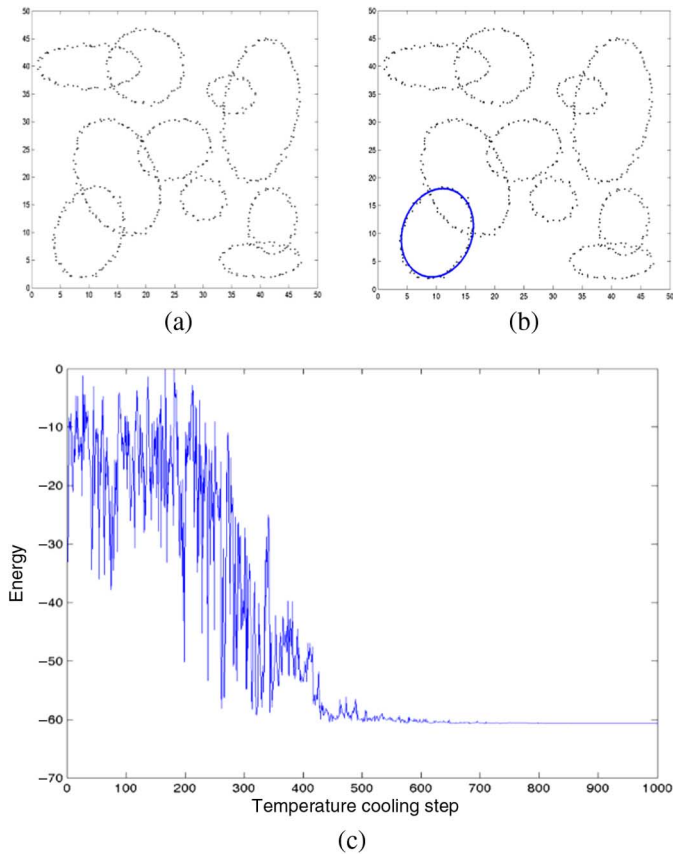


Fig. 11. Sequential detection at the first step on 10 ellipses. (a) Input data. (b) Detected pattern. (c) Energy versus temperature cooling step.

Fig. 12(a) shows the remaining points at the fifth step. Fig. 12(b) shows the detected pattern. Fig. 12(c) shows the plot of energy versus temperature cooling step.

Fig. 13 plots the final result. Every pattern is detected. The result is quite good. Comparing the detection results in Figs. 10(b) and 13, the sequential detection is better than the synchronous detection in detecting a large number of patterns.

In the experiments of sequential detection on 10 ellipses, we use one, two, and three patterns at each step. If we detect one pattern at each step, we must do 10 steps. Each ellipse has six parameters. If we detect two patterns at each step, we must do five steps. However, the CPU time in the two-pattern detection is more than two times in the one-pattern detection. So, one-pattern detection takes less CPU time in the detection of all 10 ellipses. It is the same result for the detection of three patterns. Moreover, for the detection of two and three patterns, the number of pattern parameters is 12 and 18, as there is more interference in the detection. The detection result may be not good. The one-pattern detection at each step has the less CPU time and can get good result.

C. Experiments on Real One-Shot Seismogram

The system is applied to detect line pattern from direct wave and hyperbolic patterns from reflection wave in real one-shot seismic data. We obtain seismic data from Seismic Unix System [5]. The real data of one-shot seismogram in Fig. 14(a) are from Canadian Artic. The receivers are at two sides. It has

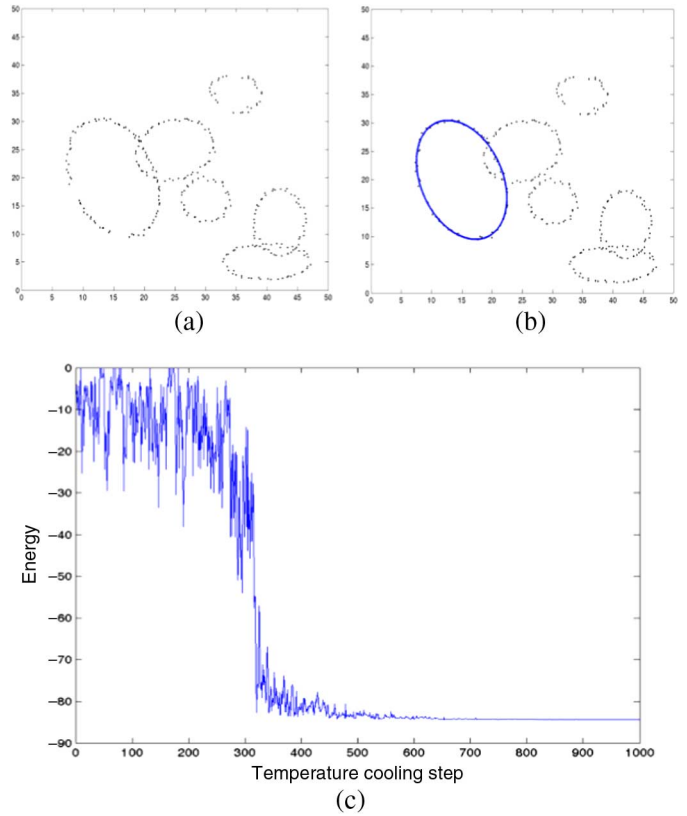


Fig. 12. Sequential detection at the fifth step. (a) Input data. (b) Detected pattern. (c) Energy versus temperature cooling step.

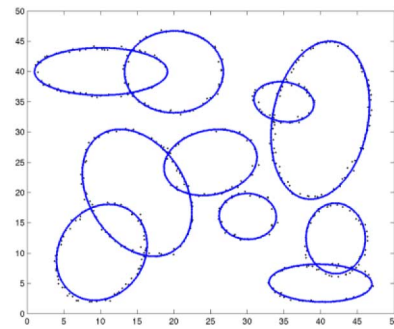


Fig. 13. Final detected patterns by sequential detection.

48 traces and 3100 samples per trace with sampling interval 0.002 s.

The seismogram is preprocessed by envelope processing, peak detection, and threshold processing [7]. Because after 1.4 s the signal is weak, we only choose points with $t < 1.4$ s, which includes pattern points from direct wave, first layer reflection wave, and second layer reflection wave as in Fig. 14(b), where there are 88 points. The image data in Fig. 14(b) are the input to the SA parameter detection system.

1) *Synchronous Detection*: We set the parameters of SA system: $\alpha_m = 1$, $\alpha_{ab} = 4$, $\alpha_\theta = 0.5$, $\alpha_f = 6$, $\beta = 0.99$, $q = 0.2$, $T_{max} = 500$, $N_t = 20$, $N_s = 1000$, $\lambda_e = 1.2$, and $\lambda_p = 10$. Fig. 14(c) shows the synchronous detection result. Three hyperbolas are detected. Because of random process, we do many experiments, but we cannot get good result every

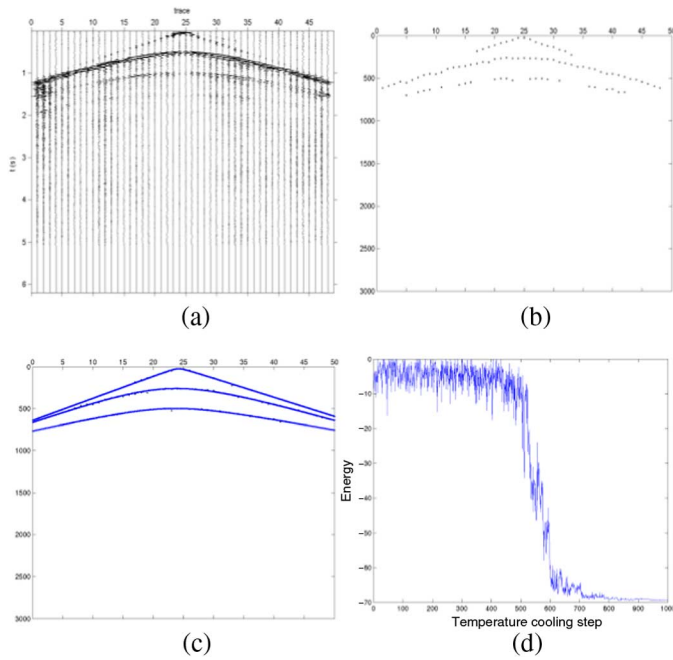


Fig. 14. Synchronous detection on real one-shot seismogram. (a) Real seismic data at Canadian Artic. (b) Image data after preprocessing. (c) Detected patterns. (d) Energy versus temperature cooling step.

time. The computation time takes much longer than that of sequential method in the following comparison.

2) *Sequential Detection*: We do sequential hyperbola detection in Fig. 14(b). We set the parameters of SA system: $\alpha_m = 1$, $\alpha_{ab} = 4$, $\alpha_\theta = 0.5$, $\alpha_f = 6$, $\beta = 0.99$, $q = 0.2$, $T_{max} = 100$, $N_t = 20$, $N_s = 1000$, $\lambda_e = 1.2$, and $\lambda_p = 10$. Fig. 15 shows the sequential detection steps. Fig. 15(a), (c), and (e) are the first to third detected patterns, and Fig. 15(b) and (d) are the remaining data after the first and second detection steps. Fig. 15(f) is the final detected patterns.

D. Experiments on Seismic CDP Gather

We generate the shot recording data from geologic model.

From many one-shot seismograms, we collect CDP gather data in the following. Then, we apply the SA to estimate the parameters of hyperbolic patterns on the simulated seismic CDP gather.

1) *Geologic Model, Seismic Data, and CDP Gather*: We use Seismic Unix System [5] to generate a geologic model of five layers as shown in Fig. 16. The interval velocity of each layer is listed in Table II. We have one-shot point and receivers at two sides and get one-shot seismogram. Then, we move shot point and receivers at the same time and get the other seismogram. Repeatedly, we have 40 shots to obtain 40 one-shot seismograms. The first shot point starts at 2 km distance from the origin. The sampling time is 0.004 s. Each shot point is in the middle of the 60 spread receivers. The distance between two receivers is 50 m. After finishing the first shot, the second shot point and all receivers move 50 m from left to right. The range of shot points is from 2 to 3.95 km and the range of receivers is from 0.525 to 5.425 km.

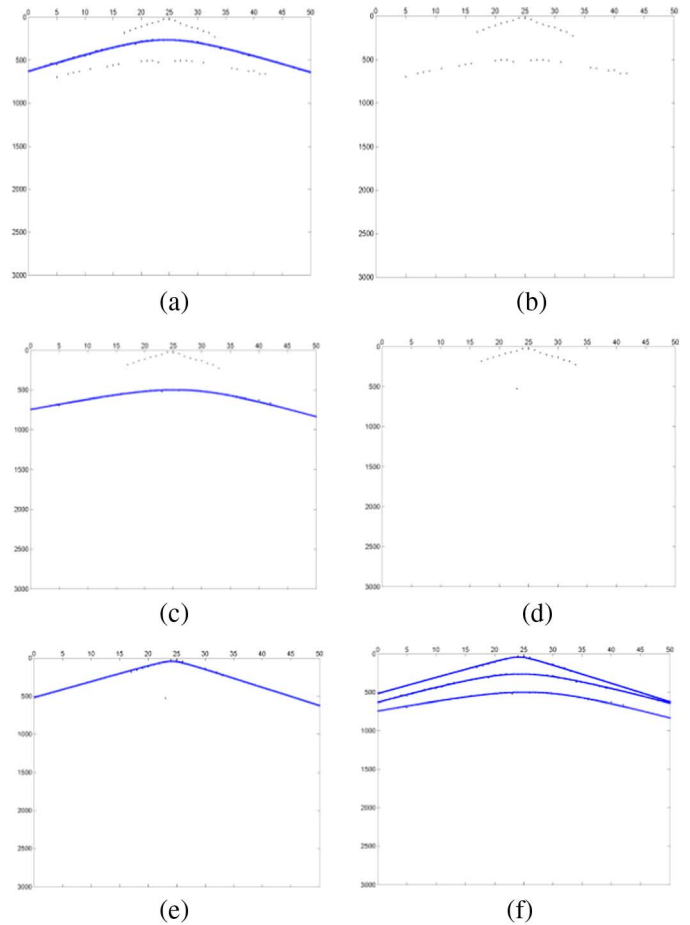


Fig. 15. Sequential detection on real one-shot seismogram. (a) First detected pattern. (b) Remaining data. (c) Second detected pattern. (d) Remaining data. (e) Third detected pattern. (f) Final detected patterns.

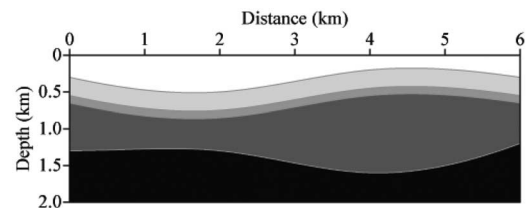


Fig. 16. Geologic model of five layers.

TABLE II
INTERVAL VELOCITY IN GEOLOGIC MODEL

Layer	1	2	3	4	5
V_{int} (m/s)	1508	1581	1690	1826	2000

We collect those reflection traces with the same reflection point from underground at different shots and receivers to become a CDP gather [5], [6]. One example of collecting CDP traces is shown in Fig. 17. The travel time of reflection is the same as (20). There are total 138 CDP gathers from 40 shot records. We choose one example of CDP gather #72 at 3.0375 km in Fig. 18(a). The patterns are hyperbolas from the reflection of CDP of different layer depths. Fig. 18(b) is the preprocessing result through envelope, threshold, and peak processing [7]. There are four hyperbolas.

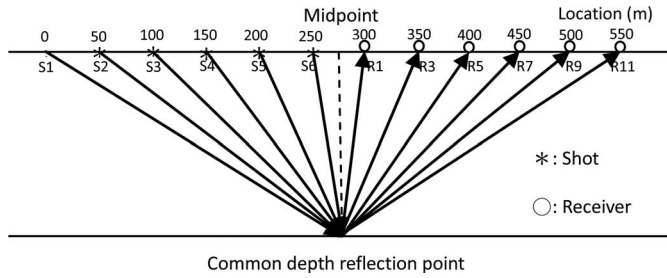


Fig. 17. Six traces of CDP gather 11 with midpoint at 275 m.

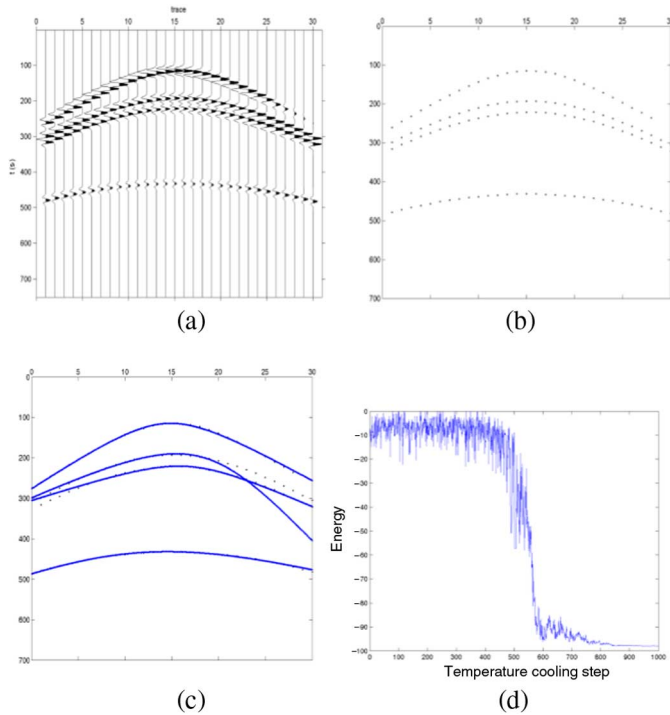


Fig. 18. Synchronous four hyperbolas detection on seismic CDP gather. (a) CDP gather #72 at 3.0375 km. (b) Image data after preprocessing. (c) Synchronous detection result. (d) Energy versus temperature cooling step.

2) *Synchronous Hyperbola Detection on CDP Gather:*

We use synchronous hyperbola detection in Fig. 18(b). We set the parameters of SA system: $\alpha_m = 1$, $\alpha_{ab} = 4$, $\alpha_\theta = 0.5$, $\alpha_f = 6$, $\beta = 0.99$, $q = 0.2$, $T_{max} = 500$, $N_t = 20$, $N_s = 1000$, $\lambda_e = 1.2$, and $\lambda_p = 10$. Fig. 18(c) shows the detected patterns. Only three hyperbolas are correctly detected. Because 24 parameters are computed at one time and there is more interference, the result is not good.

3) *Sequential Hyperbola Detection on CDP Gather:* We do sequential hyperbolic detection in Fig. 18(b). We set the parameters of SA system: $\alpha_m = 1$, $\alpha_{ab} = 4$, $\alpha_\theta = 0.5$, $\alpha_f = 6$, $\beta = 0.99$, $q = 0.2$, $T_{max} = 100$, $N_t = 20$, $N_s = 1000$, $\lambda_e = 1.2$, and $\lambda_p = 10$. There are four steps in the detection. We just show the results of two steps in the following. The first detection step is shown in Fig. 19. The third detection step is shown in Fig. 20. Final detection result is shown in Fig. 20(d). Four hyperbolas are correctly detected.

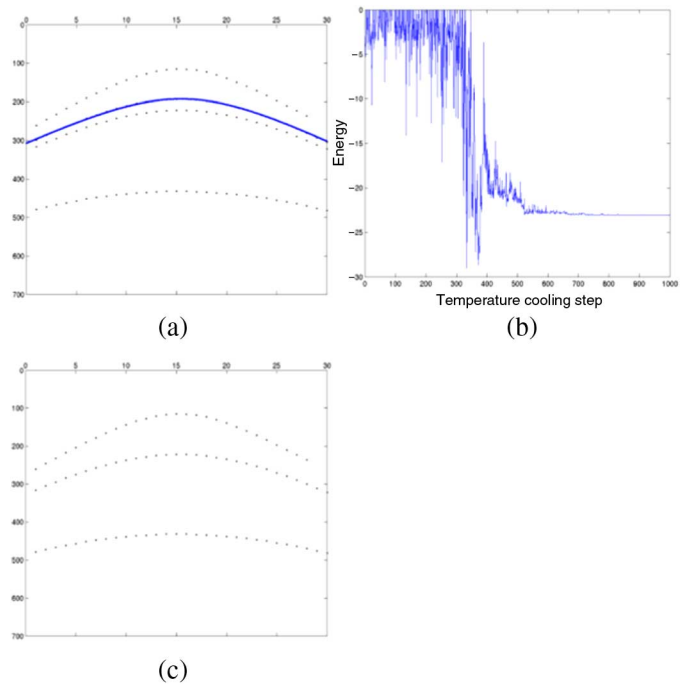


Fig. 19. Sequential four hyperbolas detection on seismic CDP gather. (a) First detection result. (b) Energy versus cooling step. (c) Remaining data.

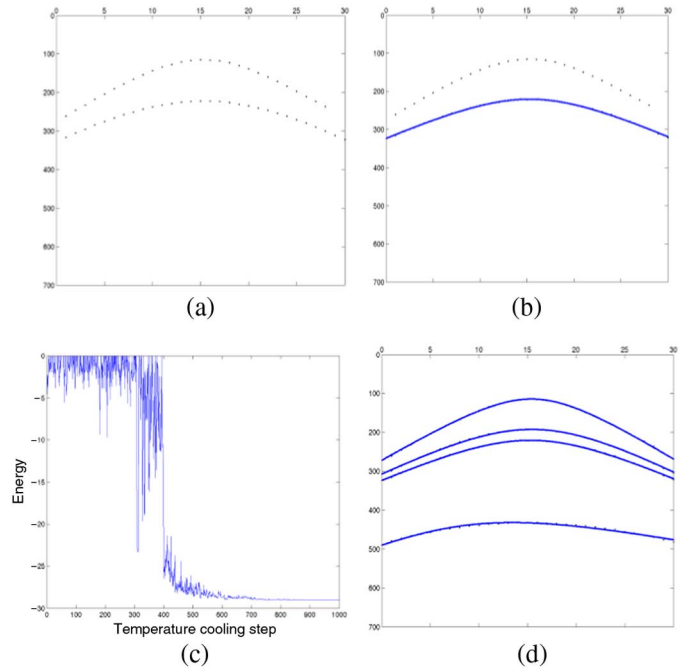


Fig. 20. Sequential four hyperbolas detection on seismic CDP gather. (a) Third detection on remaining data. (b) Detected pattern. (c) Energy versus cooling step. (d) Final detection result.

E. *Computation Time and Memory Space Requirements*

We compare the computation time and memory space between synchronous detection in Table III and sequential detection in Table IV. We use MATLAB R2013b programming to run the experiments by a computer with an Intel core-i7 CPU. Although there is no big difference in memory space requirements of two methods, the sequential detection takes much less computation time.

TABLE III
COMPUTATION TIME AND MEMORY SPACE REQUIREMENTS FOR
SYNCHRONOUS METHOD

Initial temperature	Pattern type	CPU time (s)	Memory space (kB)
100	10 ellipses	34 348	104 328
100	3 real seismic hyperbolas	1245	20 620
100	4 seismic hyperbolas	2250	28 056
500	10 ellipses	53 403	116 508
500	3 real seismic hyperbolas	1143	13 696
500	4 seismic hyperbolas	2220	48 056

TABLE IV
COMPUTATION TIME AND MEMORY SPACE REQUIREMENTS FOR
SEQUENTIAL METHOD

Initial temperature	Pattern type	CPU time (s)	Memory space (kB)
100	4 mixed patterns	560	39 232
100	10 ellipses	4063	28 330
100	3 real seismic hyperbolas	431	7464
100	4 seismic hyperbolas	412	22 692
500	4 mixed patterns	525	20 046
500	10 ellipses	2393	33 888
500	3 real seismic hyperbolas	287	15 244
500	4 seismic hyperbolas	438	39 296

V. CONCLUSION AND DISCUSSION

We adopt the previously available SA-based method and apply it in a new domain, i.e., the area of seismic data processing, the main innovation. We use SA with sequential steps to estimate parameters and detect lines, ellipses, hyperbolas type by type, and patterns by patterns in each type. The motivation of sequential detection method is to deal with a large number of patterns.

We adjust the parameters of a pattern sequentially step by step. The computation can converge efficiently than all parameters that are adjusted at one time.

In the simulation experiments, the result of sequential detection is better than that of synchronous detection in detecting a large number of patterns. The sequential detection also takes much less computation time than that of synchronous detection. In sequential detection, detection of one pattern at each step can have less computation time and good convergence in total detection than using two or more pattern detections. In simulated seismic data, SA is applied to detect the hyperbolas in the CDP gather. In real one-shot seismogram, SA is applied to detect lines of direct wave and hyperbolas of reflection wave. The results show that the proposed method is feasible. The computed hyperbolic parameters can be used in normal move-out (NMO) correction and velocity analysis for further seismic data processing [5], [6].

In the experiments, we set the parameters of SA that include the initial temperature T_{max} , parameter β , number of trials at one temperature N_t , and number of temperature cooling steps N_s , but we need to study a systematic method to determine those parameters.

ACKNOWLEDGMENT

The authors would like to thank the Colorado School of Mines for the use of Seismic Unix System. They thank the associate editor and the two reviewers for their helpful comments. They also thank K. J. Chen, Y. L. Chou, and Y. H. Hsieh for their discussion.

REFERENCES

- [1] P. V. C. Hough, "Method and means for recognizing complex patterns," U.S. Patent 3069654, 1962.
- [2] D. H. Ballard, "Generalizing the Hough transform to detect arbitrary shapes," *Pattern Recognit.*, vol. 13, pp. 111–122, 1981.
- [3] A. A. Kassim, T. Tan, and K. H. Tan, "A comparative study of efficient generalized Hough transform techniques," *Image Vis. Comput.*, vol. 17, pp. 737–748, 1999.
- [4] M. M. Slotnick, *Lessons in Seismic Computing*. Tulsa, OK, USA: The Society of Exploration Geophysicists, 1959.
- [5] O. Yilmaz, *Seismic Data Processing*. Tulsa, OK, USA: The Society of Exploration Geophysicists, 1987.
- [6] M. B. Dobrin and C. H. Savit, *Introduction to Geophysical Prospecting*. New York, NY, USA: McGraw-Hill, 1988.
- [7] K. Y. Huang, K. S. Fu, T. H. Sheen, and S. W. Cheng, "Image processing of seismograms: (A) Hough transformation for the detection of seismic patterns, (B) thinning processing in the seismogram," *Pattern Recognit.*, vol. 18, no. 6, pp. 429–440, 1985.
- [8] J. Basak and A. Das, "Hough transform networks: Learning conoidal structures in a connectionist framework," *IEEE Trans. Neural Netw.*, vol. 13, no. 2, pp. 381–392, Mar. 2002.
- [9] K. Y. Huang, K. J. Chen, J. D. You, and A. C. Tung, "Hough transform neural network for pattern detection and seismic applications," *Neurocomputing*, vol. 71, pp. 3264–3274, 2008.
- [10] S. Kirkpatrick, C. D. Gelatt, and M. P. Vecchi, "Optimization by simulated annealing," *Science*, vol. 220, no. 4598, pp. 671–680, May 1983.
- [11] N. Metropolis, A. Rosenbluth, M. Rosenbluth, A. Teller, and E. Teller, "Equation of state calculations by fast computing machines," *J. Chem. Phys.*, vol. 21, no. 6, pp. 1087–1092, Jun. 1953.
- [12] J. Schou and H. Skriver, "Restoration of polarimetric SAR images using simulated annealing," *IEEE Trans. Geosci. Remote Sens.*, vol. 39, no. 9, pp. 2005–2016, Sep. 2001.
- [13] R. Marion, R. Michel, and C. Faye, "Atmospheric correction of hyperspectral data over dark surfaces via simulated annealing," *IEEE Trans. Geosci. Remote Sens.*, vol. 44, no. 6, pp. 1566–1574, Jun. 2006.
- [14] P. Debba, E. J. M. Carranza, F. D. van der Meer, and A. P. Stein, "Abundance estimation of spectrally similar minerals by using derivative spectra in simulated annealing," *IEEE Trans. Geosci. Remote Sens.*, vol. 44, no. 12, pp. 3649–3658, Dec. 2006.
- [15] A. Tabatabaenejad and M. Moghaddam, "Inversion of subsurface properties of layered dielectric structures with random slightly rough interfaces using the method of simulated annealing," *IEEE Trans. Geosci. Remote Sens.*, vol. 47, no. 7, pp. 2035–2046, Jul. 2009.
- [16] D. H. Rothman, "Nonlinear inversion, statistical mechanics, and residual statics estimation," *Geophysics*, vol. 50, pp. 2784–2796, 1985.
- [17] K. Mosegaard and P. D. Vestergaard, "A simulated annealing approach to seismic model optimization with sparse prior information," *Geophys. Prospect.*, vol. 39, pp. 599–611, 1991.
- [18] E. Landa, W. Beydoun, and A. Tarantola, "Reference velocity model estimation from prestack waveforms: Coherency optimization by simulated annealing," *Geophysics*, vol. 54, p. 984, 1989.
- [19] S. K. Pullammanappallil and J. N. Louie, "Inversion of seismic reflection traveltimes using a nonlinear optimization scheme," *Geophysics*, vol. 58, pp. 1607–1620, 1993.
- [20] P. D. Vestergaard and K. Mosegaard, "Inversion of post-stack seismic data using simulated annealing," *Geophys. Prospect.*, vol. 39, pp. 611–624, 1991.
- [21] K. J. Chen and K. Y. Huang, "Simulated annealing for pattern detection and seismic application," in *Proc. Int. Joint Conf. Neural Netw. (IJCNN)*, Orlando, FL, USA, Aug. 12–17, 2007, pp. 477–482.
- [22] K. Y. Huang and K. J. Chen, "Simulated annealing for pattern detection and seismic applications," *J. Inf. Sci. Eng.*, vol. 25, no. 3, pp. 793–805, May 2009.
- [23] S. Theodoridis and K. Koutroumbas, *Pattern Recognition*, 4th ed. Amsterdam, The Netherlands: Elsevier, 2009.

- [24] C. H. Dix, "Seismic velocities from surface measurements," *Geophysics*, vol. 20, pp. 68–86, 1955.
- [25] P. Kearey, M. Brooks, and I. Hill, *An Introduction to Geophysical Exploration*. Hoboken, NJ, USA: Wiley, 2002.

Kou-Jen Huang, photograph and biography not available at the time of publication.

I-Chieh Chen, photograph and biography not available at the time of publication.

Luke K. Wang, photograph and biography not available at the time of publication.



Kou-Yuan Huang (S'81–M'83–SM'94) received the B.S. degree in physics and the M.S. degree in geophysics from the National Central University, Chungli, Taiwan, in 1973 and 1977, respectively, and the M.S.E.E. and Ph.D. degrees in electrical engineering from the Purdue University, West Lafayette, IN, USA, in 1980 and 1983, respectively.

Currently, he is a Professor with the Department of Computer Science, National Chiao Tung University, Hsinchu, Taiwan. He has authored two books: *Neural Networks and Pattern Recognition* (Weikeg Publishing Co., 2nd ed., 2003), and *Syntactic Pattern Recognition for Seismic Oil Exploration* (World Scientific Publishing Co., vol. 46, 2002). His research interests include geophysical pattern recognition using signal and image processing, statistical, syntactic, neural networks, fuzzy logic, and evolutionary computation methods, and well log data inversion using neural networks and evolutionary computation methods.