

RECONSTRUCTION ALGORITHM FOR HIGH CONTRAST VELOCITY TRAVEL TIME TOMOGRAPHY

Yen-ting Lin and Antonio Ortega

Department of Electrical Engineering-Systems
University of Southern California
Los Angeles, California, 90089-2564, U.S.A

ABSTRACT

We consider travel time tomography problems involving detection of high contrast, discrete high velocity lines. This results in a discrete nonlinear inverse problem, for which traditional least-squares reconstruction algorithms are not suitable, as they tend to result in oscillations in the estimated values of the ray-path matrix. We propose a new algorithm that provides a more stable reconstruction for high contrast velocity scenarios. Our approach is based on using multiple candidate discrete high velocity lines along with a probabilistic mixture model that captures the likelihood of each of the lines. We propose an iterative algorithm based on a graphical model that successively updates the length and probability of these line structures. Preliminary simulation results show that exact reconstruction can be achieved in cases when the ground-truth lines are a subset of candidate structures.

Index Terms— Travel time tomography, Discrete tomography, High contrast velocity model.

1. INTRODUCTION

Travel time tomography aims to reconstruct an interior slowness (reciprocal of velocity) model based on measured first-arrival time between transmitters and receivers with known locations. The velocity model characterizes the physical properties of the region where propagation occurs. However, different from X-ray computed tomography, the assumption of a straight line wave propagation model is usually a poor approximation in acoustical or electromagnetic tomography, where the travel ray path bends according to the local propagation velocity. Without knowing the actual travel path, reconstruction of the slowness model becomes a nonlinear inverse problem. Many iterative algorithms have been proposed based on iteratively solving the linearized problem, and requires the slowness model is smooth [?]. However the reconstructed image resolution becomes worse for higher velocity contrast cases because linearized function is a poor approximation [?].

In our work, we focus on nonlinear travel time tomography with high contrast, discrete velocity values. Specifically we consider the case where velocity has two discrete values: very high velocity in line structures and low velocity in a homogeneous background. This work can be extended by increasing the number of allowable discrete values and by permitting discrete structures other than lines to be used. Example scenarios where this formulation can be encountered arise in different fields, e.g., finding a fault zone in geo-

physical acoustical tomography, discovering link failure in network tomography [?], etc. Our initial motivation for this work comes from problems in oil reservoir characterization. In this setting the permeability of open fractures can be orders of magnitude higher than that of surrounding tight rocks, providing fast pathways for fluid to flow. Thus, travel time through a fracture (which could be modeled as a line structure within the reservoir), is much faster than through surrounding areas. Understanding these fractures is critical in reservoir characterization [?].

We propose a method to estimate high velocity line (HVL) structures with discrete velocity. Based on Fermat's principle, we show that the actual travel path can be approximated by combining the effect of a few line segments, which significantly reduces the computation complexity to calculate the travel path. Traditional travel time tomography techniques estimate spatial velocity or slowness values for the area of interest by breaking it into cells and estimating local velocity under smoothness constraints. Instead we consider discrete structures (lines) on a homogeneous velocity background. We start by defining a list of possible HVLS, and associate a probability to each of them. Then, based on the travel time constraints, we iteratively refine the candidate HVL parameters (their length) and re-weight their probabilities. Similar to the expectation maximization (EM) algorithm, in every iteration we lower the bound for global mismatch function. The reconstruction results are presented as a probability map showing which of the candidate HVLS are more likely to represent the underlying structure of the system. Compared to matrix inversion methods, our algorithm avoids oscillations in the solution model. To the best of our knowledge most travel time tomography techniques focus on smooth low contrast cases and we are the first to propose a discrete line-based solution for the high contrast, discrete velocity case.

The rest of the paper is organized as follows. In Section 2 we formulate the problem. Section 3 gives an overview of our proposed approach for estimation. Simulation results and discussions are presented in Section 4. Section 5 concludes this paper.

2. PROBLEM FORMULATION

The travel time tomography problem can be modeled as a nonlinear inverse problem. Let \mathbf{t} be a m -vector of the measured travel time, $\mathbf{t} = (t_1, \dots, t_m)^T$, where t_i is the travel time between i -th transmitter-receiver pair and m is the total number of views (transmitter-receiver pairs). In standard approaches for travel time tomography, it is assumed that the model can be divided into cells (in 2D) and the travel velocity inside one cell is constant. Denote the slowness model model, $\mathbf{s} = (s_1, \dots, s_n)^T$ for the n cells, with s_j representing the slowness of the j -th cell. Then \mathbf{s} and \mathbf{t} are

This work is supported in part by Chevron Corp. under the joint project Center for Interactive Smart Oilfield Technologies (CiSoft), at the University of Southern California.

related by the travel path $m \times n$ matrix \mathbf{M} . The ray travel path matrix \mathbf{M} represents the cells through which the ray passes. \mathbf{M} is an $m \times n$ matrix where $M_{i,j}$ represents the length of the i th travel path through the j th cell:

$$\mathbf{M} \cdot \mathbf{s} = \mathbf{t} \quad (1)$$

Note that each solution \mathbf{s} leads to a different matrix \mathbf{M} and (??) simply states the total travel time along the i -th path is the sum of times spent in each cell along the path [?]. The inverse problem is how to find the corresponding slowness model \mathbf{s} and ray-path matrix \mathbf{M} with measured travel time \mathbf{t} . There are various of reconstruction algorithms for continuous slowness model \mathbf{s} . We refer to [?] for a good introduction to these techniques.

In this paper, we focus on scenarios where slowness cannot be assumed to be smooth. Instead our goal is detect discrete high velocity lines (HVLs) in a homogeneous background. The iterative linearized approximation method works poorly in this case [?]. As an alternative we propose a new approach where the goal is to identify HVLs among a set of candidates. Unlike previously proposed methods, continuous slowness values are assigned to 2D cells (under smoothness constraints), here we operate on discrete lines, whose length and probabilities are updated iteratively. We set travel time to 0 along HVLs and to 1 per unit distance along surrounding homogeneous areas.

According to Fermat's principle, the ray travel path corresponding to observed travel time between a transmitter and a receiver is the fastest one among all possible paths. Consider the example of Fig. ??, where only one HVL is considered. In this case the observed travel time should be of function of distance from transmitter to HVL and from HVL to receiver (assume infinite velocity within the HVL). As seen in Fig. ??, the fastest travel path "bends" to follow the HVL. Thus, if we considering an HVL of known orientation, near a transmitter and a receiver (as in Fig. ??), it is possible to find (using simple geometry) the optimal line length to match the measured travel time between the transmitter/receiver pair. Note that this geometric approach can be extended to cases where more than one HVL is considered.

As a starting point we define several HVL candidates between each transmitter and each receiver, as illustrated in Fig. 2. In this work, we assume the orientation and velocity associated to these lines remains fixed, but their length and probability is updated by the algorithm. Our goal is to estimate the length and probability for all lines, so that *all* travel times between pairs can be matched as well as possible.

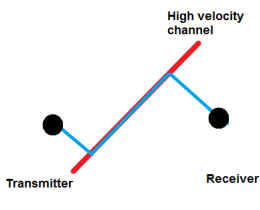


Fig. 1. Wave travel path with high velocity channel.

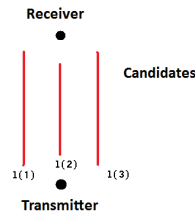


Fig. 2. Candidates of high velocity channels.

3. PROPOSED ALGORITHM

Given the location of transmitter and receiver, our goal is to find a solution of HVLs, L_λ to fit the measured travel time \mathbf{t} . However, as in many inverse problems, the solution \mathbf{s} may not be unique. We propose an algorithm to find solutions with different probability weights.

We start by defining the parameters. Assuming there are p transmitters, q receivers, we will have $m = p \cdot q$ views. We denote $T_i(L_1 \cup L_2)$ the travel time for i th transmitter-receiver pair given the presence of two high velocity lines L_1, L_2 . For each transmitter-receiver pair, we consider K possible HVL candidates with length $l_i(\lambda)$ and probability $P(l_i(\lambda))$, $i = 1, \dots, m$, $\lambda = 1, \dots, K$. with $\sum_{\lambda=1}^K P(l_i(\lambda)) = 1$ (see Figure ?? for an example). We initialize the candidates in each pair so that each candidate has initial length $l_i(\alpha)$ that satisfies the measured travel time t_i , i.e., $l_i(\alpha)$ are chosen so that $T_i(l_i(\alpha)) = t_i$, $\alpha = 1, \dots, K$, with all candidates having equal probability $P(l_i(\alpha)) = 1/K$. Given that there are m views, if we chose a candidate in each pair there would be K^m possible combinations of candidates to represent the global behavior, making the search for the correct combination impractical.

Thus, instead of searching among combinations of multiple HVLs our algorithm focuses on the interaction between two pairs of candidates. We propose a graphical model [?], shown in Fig. 3 and develop a message-passing algorithm that alternates between estimating the length $l_i(\lambda)$ and reweighing the probability $P(l_i(\lambda))$, for all the pairs $i = 1, \dots, m$ and candidates $\lambda = 1, \dots, K$. There are two types of nodes in the graphical model: the candidate variable nodes, $l_i(\alpha)$, which capture the length $l_i(\alpha)$ and probability $P(l_i(\alpha))$, the check nodes, T_k , which are used to compare estimated travel time T_k with the measured travel time t_k for k th travel path.

In first part of the iteration, check nodes send messages to variable nodes to inform them of the best estimated lengths based on the current information from candidate variable nodes. In the second part of the iteration, candidate variable nodes process the incoming messages and update their lengths and probabilities.

In order to quantify the interaction between the candidates in different pairs, we define a mismatch function $\phi(l_i(\alpha), l_j | t_k)$, that measures the expected value of the mismatch between estimated, T_k , and measured, t_k , travel times when considering the current length estimate $l_i(\alpha)$ for a given candidate and considering the effect of all existing candidates for another transmitter-receiver pair ($l_j(\beta)$):

$$\phi(l_i(\alpha), l_j | t_k) = \sum_{\beta=1}^K \|T_k(l_i(\alpha) \cup l_j(\beta)) - t_k\| * P(l_j(\beta)) \quad (2)$$

A key observation in our method is that the travel time function is monotonic, i.e., increasing the number or length of HVLs means the travel time T_k will strictly decrease:

$$\begin{aligned} T_k(l_i(\alpha) \cup \dots \cup l_m(\zeta)) &\leq T_k(l_i(\alpha) \cup l_j(\beta)) \\ &\leq t_k \end{aligned} \quad (3)$$

Therefore

$$\begin{aligned} \|T_k(l_i(\alpha) \cup \dots \cup l_m(\zeta)) - t_k\| \\ \geq \|T_k(l_i(\alpha) \cup l_j(\beta)) - t_k\| \end{aligned} \quad (4)$$

Thus, *the pairwise mismatch is a bound for the global mismatch*. At initialization time we have that $T_k \leq t_k$ for all k , and at every iteration we will decrease the length of the lines (increase travel time), in order to approach measured times. As in the EM algorithm, although we only consider partial structure (two transmitter-receiver pairs), in

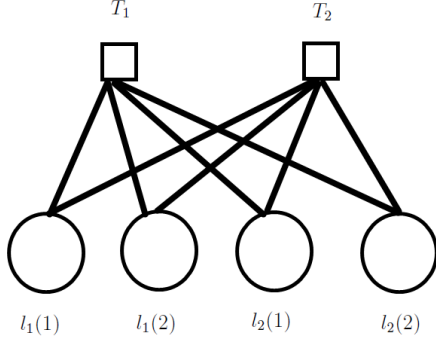


Fig. 3. Graphical model of the 2 candidate and 2 measured time case.

every iteration we will lower the matching error for the pairs, and therefore that for the global solution.

Now we introduce the message passing through the graphical model. $q(T_k \rightarrow l_i(\alpha))$ denotes the message from measured data node T_k to candidate variable node $l_i(\alpha)$. It gives the estimate of length $l_i(\alpha)$ based on the measured data t_k and pairwise interaction with candidates in other pairs. In order to find $q(T_k \rightarrow l_i(\alpha))$, first we calculate $l_i(\alpha|j, t_k)$, the length for $l_i(\alpha)$ which leads to the lowest average travel time error to match t_k when combining its effect with that of all the line candidates for view j , $l_j(\beta)$, for $\forall \beta$.

For each of the other pairs, $j = 1, \dots, m, j \neq i$, we solve the minimization problem

$$l_i(\alpha|j, t_k) = \arg \min_{l_i(\alpha)} \sum_{\beta=1}^K \|T_k(l_i(\alpha) \cup l_j(\beta)) - t_k\| \cdot P(l_j(\beta)) \quad (5)$$

with the constraint

$$T_k(l_i(\alpha) \cup l_j(\beta)) \leq t_k \quad (6)$$

Repeating this step, we will have $\{l_i(\alpha|j, t_k), j = 1, \dots, m, j \neq i\}$ that gives the estimated length when considering pairwise interaction with all other candidates $l_j(\beta), j = 1, \dots, m, j \neq i$. We have to choose the message $q(T_k \rightarrow l_i(\alpha))$ to send from $\{l_i(\alpha|j, t_k), j = 1, \dots, m, j \neq i\}$. In order to choose the message consider $l_i(\alpha|j, t_k)$ and $l_i(\alpha|q, t_k)$, and assume $l_i(\alpha|j, t_k) \leq l_i(\alpha|q, t_k)$. Due to the monotonicity property mentioned earlier if we compare two alternative global solutions that differ only in the term l_i , we will have that (shorter HVLs mean longer propagation times):

$$\begin{aligned} T_k(l_i(\alpha|q, t_k) \cup \dots \cup l_m(\zeta)) &\leq T_k(l_i(\alpha|j, t_k) \cup \dots \cup l_m(\zeta)) \\ &\leq T_k(l_i(\alpha|j, t_k) \cup l_j(\beta)) \\ &\leq t_k \end{aligned} \quad (7)$$

so that

$$\begin{aligned} &\|T_k(l_i(\alpha|q, t_k) \cup \dots \cup l_m(\zeta)) - t_k\| \\ &\geq \|T_k(l_i(\alpha|j, t_k) \cup \dots \cup l_m(\zeta)) - t_k\| \end{aligned} \quad (8)$$

i.e., $l_i(\alpha|j, t_k)$ leads to lower mismatch than $l_i(\alpha|q, t_k)$. Therefore, $l_i(\alpha|j, t_k)$ will be a better estimate for the length $l_i(\alpha)$ and we choose the message based on travel time t_k as:

$$q(T_k \rightarrow l_i(\alpha)) = \min_j l_i(\alpha|j, t_k).$$

Messages will flow from all check nodes $T_k, k = 1, \dots, m$ to the candidate variable node $l_i(\alpha)$. Next step will be to decide which one of the incoming messages $\{q(T_k \rightarrow l_i(\alpha)), k = 1, \dots, m\}$ is a better update for $l_i(\alpha)$. The $l_i(\alpha)$ needs to satisfy the constraint of (??) for all $k = 1, \dots, m$. Because shorter length corresponds to longer travel time, if we choose $l_i(\alpha) \geq q(T_k \rightarrow l_i(\alpha))$, this can guarantee the travel time is smaller than t_k . Therefore, we choose $l_i(\alpha) = \max_k q(T_k \rightarrow l_i(\alpha))$ as our new update.

After we update the length of the candidate variable node, we want to reweigh the probability of each candidate. We need to know which measured data t_k determines the update. Let

$$r = \arg \max_k q(T_k \rightarrow l_i(\alpha)) \quad (9)$$

r is the pair index that represents which travel time data t_r restricts the length of $l_i(\alpha)$. This pair should give the most information about $l_i(\alpha)$. We reweigh the probability of each candidate $P(l_i(\alpha))$, which is inversely proportional to the mismatch function.

$$P(l_i(\alpha)) \propto \frac{1}{\phi(l_i(\alpha), l_r|t_r) + \delta} \quad (10)$$

The δ term is added to help the stability.

After reweighing probability, we also need to check the validity of the solution. For all $k = 1, \dots, m$, if $T_k(l_i(\alpha)) < t_k$, set the probability

$$P(l_i(\alpha)) = 0.$$

The reason we set the probability to zero is if $T_k(l_i(\alpha)) < t_k$ is adding another line always decrease the travel time. The global structure $T_k(l_i(\alpha) \cup \dots \cup l_m(\zeta))$ will go further from the measured time t_k .

Finally, we normalize the probability

$$\sum_{\alpha=1}^K P(l_i(\alpha)) = 1$$

The algorithm is stated below:

- 1: Set the iteration number $I = 0$.
- 2: For every node of travel time T_k , calculate the message $q(T_k \rightarrow l_i(\alpha))$ send to node $l_i(\alpha)$. For each of the other pairs, $j = 1, \dots, m, j \neq i$, solve the minimization problem (??) to get $l_i(\alpha|j, t_k)$.
- 3: Set the message send from node T_k to $l_i(\alpha)$ as

$$q(T_k \rightarrow l_i(\alpha)) = \min_j l_i(\alpha|j, t_k) \quad (11)$$

- 4: In the candidate variable node, with all the messages send from every node $T_k, k = 1, \dots, m$, choose the length update by

$$l_i(\alpha) = \max_k q(T_k \rightarrow l_i(\alpha)) \quad (12)$$

And let $r = \arg \max_k q(T_k \rightarrow l_i(\alpha))$.

- 5: Check the validity: For all $k = 1, \dots, m$,
- 6: **if** $T_k(l_i(\alpha)) < t_k$ **then**
- 7: Set the probability $P(l_i(\alpha)) = 0$.
- 8: **else**
- 9: Reweight the probability of each candidate $P(l_i(\alpha))$ by (??), which is inversely proportional to the mismatch function.
- 10: Normalize the probability $\sum_{\alpha=1}^K P(l_i(\alpha)) = 1$.
- 11: Go to step 2 until reach the maximum number of iterations I_{max} .

Our algorithm is a heuristic search on graphical model. In step 2, we try to find the message $l_{i(\alpha)}(k \rightarrow i(\alpha))$, sending from node T_k to $l_{i(\alpha)}$. This gives the 'belief' of how $l_i(\alpha)$ should be depending on the data t_k . If we look at all of possible combinations, it will be $T_k(l_i(\alpha) \cup \dots \cup l_m(\zeta))$. Since it will not be practical to consider all the K^{m-1} combinations, instead, we consider only the pairwise interaction $T_k(l_i(\alpha) \cup l_j(\beta))$, $j = 1, \dots, m$ and update partial structure. Similar to the EM algorithm, every iteration we choose new estimated length and probability of candidates to decrease the local mismatch function. From (??), we can easily see the local mismatch is a lower bound for global mismatch. Therefore, we expect the global mismatch function will decrease through iterations.

4. SIMULATION RESULTS

In our simulation, we set the 2D testing region with $64 * 32$ blocks of homogeneous background. The travel time data has 25 ray paths, including $5 * 5$ from bottom to top. The transmitters and receivers are uniformly located in the lower and upper boundary. Each travel path has 3 possible high velocity line candidates.

In the first experiment, the candidates contain the ground truth, but not for second experiment.

The result should be read as a probability map. The lighter colored regions correspond to a higher probability that a high velocity region is present. For the first experiment, the result converges to exactly the true model. Our algorithm successfully picks the right candidate in iterations. As we expect, the mean square error decreases in each iteration even though we only consider pairwise interactions. In the second experiment, the true model does not belong to the set of candidates. Therefore, our algorithm will try to approximate the solution by the given set of candidates. The result should be viewed as the combination of different possible solutions. We list two possible solutions from estimated results. Because we only consider the interaction between two high velocity lines, the reconstruction performance is affected by several factors, such as number of candidates, complexity of underlying structure, number of transmitter and receivers, and that we can do well in some of these cases.

Adding more number of candidates will give a larger set of basis to approximate the ground truth. The price we pay is the computational complexity. Our algorithm has a cost of $O(m^3 K^2)$, significantly lower than that of the combinatorial approach, $O(K^m)$. While ours is a heuristic approach it does provide encouraging results in simple tests. We will consider more complex situations in future work.

5. CONCLUSION

The main purpose of this paper is to develop a new algorithm to reconstruct high contrast discrete velocity model in travel time tomography. We take advantage of high velocity lines and show the ray paths are piecewise linear by Fermat's principle. This significantly reduces the computation for determining the travel path. An iterative message-passing reconstruction algorithm is developed and used to give probability map of different possible scenario. Parallel processing technique can be used in each node to boost the computation. We show in simulation that if the ground truth belong to the set of candidates, we can successfully reconstruct it. Future work will be studying the multi-lines case and exploring how to generate the set of candidates to achieve better approximation of true model.

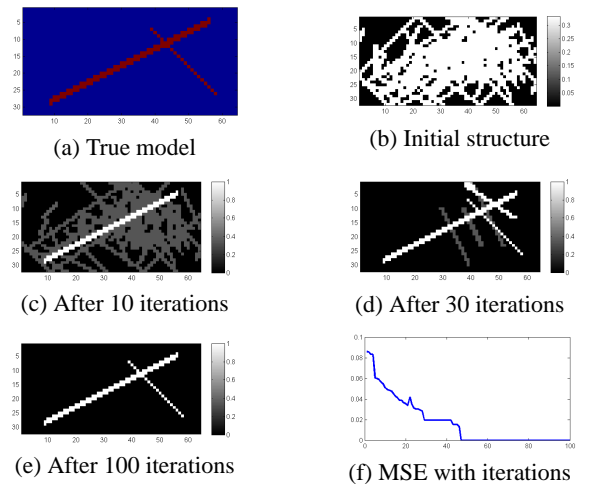


Fig. 4. Simulation with one high velocity channel

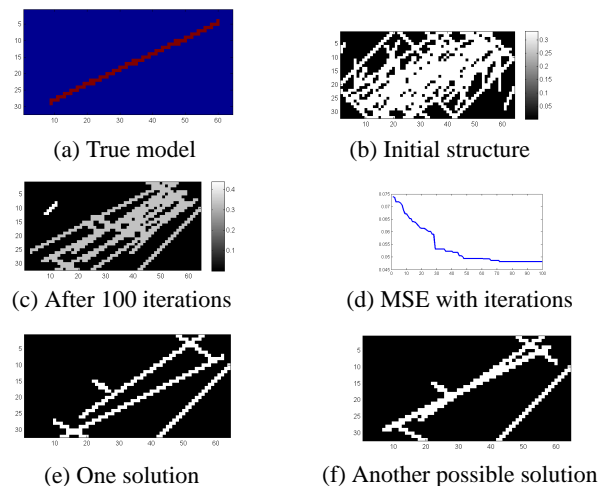


Fig. 5. Simulation with high velocity channel not align with the transmitter-receiver