

Wiring Diagnostics via ℓ_1 -Regularized Least Squares

Stefan Schuet, *Member, IEEE*

Abstract—A new method for detecting and locating wiring damage using time domain reflectometry with arbitrary input interrogation signals is presented. This method employs existing ℓ_1 regularization techniques from convex optimization and compressed sensing to exploit sparsity in the distribution of faults along the length of a wire, while further generalizing and improving commonly used fault detection techniques based on sliding correlation and peak detection. The method’s effectiveness is demonstrated using a simulated example, and it is shown how Monte Carlo techniques are used to tune it to achieve specific detection goals, like a certain false positive error rate. Furthermore, the method is easily implemented by adapting readily available optimization algorithms to quickly solve large, high resolution, versions of this estimation problem. Finally, the technique is applied to a real data set, which reveals its impressive ability to identify a subtle type of chafing damage on real wire.

Index Terms—diagnostics, fault detection, inverse scattering, lossless media, sparsity, time domain reflectometry (TDR), wiring.

I. INTRODUCTION

THIS paper considers the specific problem of detecting faults in wiring systems using time domain reflectometry. Generally, this is performed by launching a known signal into a wire, and examining the signal reflected back for potential issues (Figure 1 below). An important aspect of this technique is that one can detect and locate wiring problems well before hard short or open conditions occur. With this application in mind we are particularly concerned with the detection of small faults such as chafing damage to shielded wire. One specific application is to aircraft wiring systems that are hard to inspect visually, and where it is critical to identify problems before components start to fail.

The setup is presented in Figure 1. A Time Domain Reflectometer (TDR) is connected to the transmission line we want to check, and is used to send a signal down the wire. The reflected signal is then measured, and checked for anomalies that might indicate possible wiring problems along the line. For example, consider a simple case where the original transmission line is perfect (and has matched source and load impedance). In this case, we will see the incident signal pass right through the line without receiving any reflected signal back. Now imagine that during the course of its lifetime, the outer shielding along a section of the wire is damaged, a common problem with aging aircraft caused by decades of wires rubbing together, among other things. This sort of damage will cause the incident voltage wave to reflect and travel back along the line where it will be measured by the TDR.

This work was supported by NASA’s Aviation Safety Program, as part of Challenge Problem 8 within the Aircraft Aging and Durability Program.

The author is with NASA Ames Research Center, Intelligent Systems Division, Moffett Field, CA 94035. (e-mail: stefan.r.schuet@nasa.gov).

Final draft published in the IEEE Sensors Journal, Vol. 10, No. 7, July 2010.

It is common practice for trained technicians to manually examine the raw TDR response by eye in search of reflections indicative of wiring faults. While this is perhaps the simplest approach, it often does not work well for small faults with subtle reflection signatures. Furthermore, the current state of the art uses sequence time domain reflectometry (STDR), or spread spectrum time domain reflectometry (SSTDR) to detect wiring problems on live wires [1]. In essence these technologies spread the input interrogation signal energy out over time by using a pseudo noise sequence with a small amplitude so that the regular data signals on the wire appear unaffected. With this type of interrogation signal direct manual inspection of the reflected signal is near impossible, so some sort of processing is required to detect fault severity and location. By far the dominant processing algorithm for all these systems is to simply compute the correlation between the input signal and the reflected response and then to use a peak detection algorithm (or just a threshold) to locate the faults [1]–[5].

Obviously, it is beneficial to process the reflected response in order to automate the detection process, or perhaps just to make manual inspection easier. In this paper, we develop an improvement to the traditional sliding correlator and peak detection method, by incorporating the additional prior information that small wiring faults are generally sparsely populated along the line. The approach is motivated by some recent theoretical and software developments in compressed sensing for sparse signal recovery using ℓ_1 -norm regularization [6], [7]. This method appears to be effective and new to the field of time domain reflectometry. The presentation here is meant to be clear, practical, and immediately applicable to any existing TDR hardware system. It is thus critical that improvements are both computationally efficient and effective for arbitrary interrogation signals.

The fundamental problem presented here is to reconstruct the properties of a transmission line from the measurement of its response to an input interrogation signal. This problem has been studied for at least half a century, but seldom under the time domain reflectometry heading, which seems more focused on hardware development. Some early work dating back to at least 1957 appears in the geophysics community where researchers focused on identifying systems of layered earth by applying the TDR principle to the ground. A survey of this work is provided by J. Mendel in [8]. The electrical engineering community has of course tackled the problem as well, sometimes referring to it as *inverse scattering* or *layer peeling*. The literature here is extensive and many different models have been proposed. Two representative examples might be the lossless discrete piecewise constant impedance model described in [9], or the more general continuous RCLG transmission line model presented in [10].

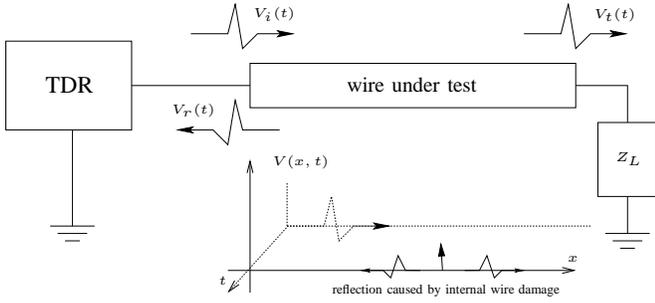


Fig. 1. Basic TDR Setup. The TDR interrogates the wire with input signal $V_i(t)$, which propagates along the wire and reflects off of impedance discontinuities caused by damage. The reflected signal, $V_r(t)$, is measured at the input of the wire and used to determine the location and severity of the damage.

The application of this prior research to the detection of wiring faults usually falls into one of two categories. The first category contains techniques that solve the transmission line partial differential equations, using discrete or continuous methods, and then directly invert the solution process without incorporating the effects of measurement noise, such as those presented in [9], [10]. The second category consists of methods that use simple linear models, which account for noise, and apply various least-squares based techniques to the inversion process, such as Kalman filtering [8]. The sliding correlation and peak detection method, as well as the method we pursue in this paper also fall into this category. Although these models are not as general or precise as those falling into the first category, they are practical to work with, lead to robust inversion algorithms where measurement noise is a consideration, and are *effective for the detection of small faults on near lossless wire*. In many ways they are analogous to the small-signal models, which are used with great success throughout electrical engineering and physics in general.

This paper is organized as follows. First we present a linear model for the TDR setup and measurement process just described. Next, the problem of detecting the location and severity of wiring damage is posed as an estimation problem, and a heuristic is introduced to find effective solutions to the original problem, by solving a convex optimization problem. Finally, we will show how the Fast Fourier Transform (FFT) makes it possible to efficiently solve large-scale problems. Numerical examples are presented along the way, and the improved technique is verified using real TDR data.

II. A LINEAR TDR MODEL

We assume the transmission line or just wire is lossless (and hence also distortionless), that any voltage wave traveling through it moves at constant velocity, and that the line is initially quiescent.

We consider the following discrete convolution model for the TDR measurement process:

$$V_r(k) = \sum_{j=0}^{n-1} \mu(j)V_i(k-j) + \eta(k), \quad (1)$$

where for $k = 0, 1, 2, \dots, n-1$, $V_r(k)$ is the measured response, $\mu(k)$ is a series of impulse response *reflection coefficients* that characterize the damaged wire, $V_i(k)$ is the known incident wave launched into the transmission line, and $\eta(k)$ is random measurement noise. This model has a simple interpretation: the measured signal is the sum of time shifted and scaled replicas of the input signal $V_i(k)$, plus noise. For an unfaulted line $\mu(k) = 0$, for all k (except perhaps at the source or load end of the wire). Thus, fault severity and location are indicated by the magnitude and position of each nonzero reflection coefficient.

The model can represent either causal or circular (periodic) convolution. For circular convolution we put $V_i(-k) = V_i(n-k)$. For causal convolution, we simply define $V_i(k) = 0$ for all $k < 0$. Obviously, for either case $V_r(k)$ must get the same treatment.

It is both instructive and notationally convenient to rewrite (1) in an equivalent matrix vector form:

$$v_r = V\mu + \eta \quad (2)$$

where,

$$\begin{aligned} v_r &= [V_r(0), \dots, V_r(n-1)]^T \\ \mu &= [\mu(0), \dots, \mu(n-1)]^T \\ \eta &= [\eta(0), \dots, \eta(n-1)]^T \end{aligned}$$

and,

$$V = \begin{bmatrix} V_i(0) & V_i(-1) & \dots & V_i(1-n) \\ V_i(1) & V_i(0) & \ddots & \vdots \\ \vdots & \vdots & \ddots & V_i(-1) \\ V_i(n-1) & V_i(n-2) & \dots & V_i(0) \end{bmatrix}.$$

So, v_r , μ , and $\eta \in \mathbf{R}^n$. V is a Toeplitz matrix in $\mathbf{R}^{n \times n}$ entirely determined by the input signal $V_i(k)$.

Finally, it is important to note the linear model presented here is motivated by the lossless discrete piecewise constant impedance transmission line model shown in figure 2. A clear, detailed study of this model is presented in [9]. The linear model is actually an approximation to the piecewise constant model that follows by assuming only the primary reflection from each impedance discontinuity is significant, and that all additional reflections are negligible. In cases where the impedance discontinuities are both sparse and small, the reflection coefficients $\mu(k)$ are *approximately* related to impedances discontinuities in the traditional sense (according to the equation shown on figure 2). See §5.10 of [11] for more discussion. The validity of these assumptions will also be supported by the practical examples using real TDR data presented in §VI.

III. REFLECTION COEFFICIENT ESTIMATION

Nonzero values of $\mu(k)$ indicate the location and severity of faults along the wire. Given the prior information that wires are typically undamaged for most of their length, except perhaps at a few locations, the reflection coefficient vector μ should contain only a few nonzero values. In other words, we

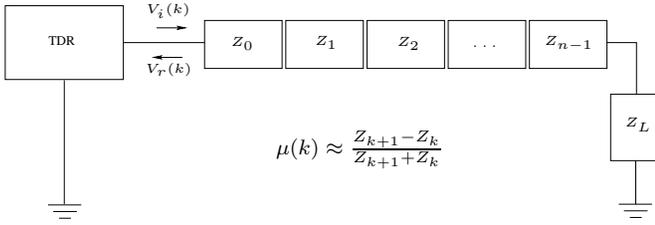


Fig. 2. The discrete piecewise constant impedance model. Small faults are modeled as impedance discontinuities, which are equivalently specified as reflection coefficients.

expect μ to be sparse. Thus, we are interested in solving the optimization problem:

$$\begin{aligned} & \text{minimize} && f_0(\mu) \\ & \text{subject to} && \mu \text{ sparse,} \end{aligned} \quad (3)$$

where,

$$f_0(\mu) = \frac{1}{2\sigma^2} \|V\mu - v_r\|^2 \quad (4)$$

is the objective representing the negative log-likelihood of observing the signal V_r given μ , under the assumption that the noise $\eta(k)$ is IID $N(0, \sigma^2)$.

One heuristic to handle the rather vague sparsity constraint in (3), is to add an ℓ_1 -norm penalty to the objective. This regularization technique is well known to produce sparse solutions (see [6], [7], [12], [13] and [14] §6.3.2). To this end, we consider solving the convex ℓ_1 -regularized least squares problem (LSP):

$$\text{minimize } f_0(\mu) + \lambda \|\mu\|_1, \quad (5)$$

with ℓ_1 -norm defined as $\|\mu\|_1 = \sum_{j=0}^{n-1} |\mu(j)|$. Intuitively, the solution is sparse because in the process of finding an optimal solution, the solver will routinely reduce a small coefficient identically to zero at the cost of increasing the associated squared error $f_0(\mu)$ by a smaller amount. The key observation is perhaps that square error measured by f_0 stays relatively flat near a minimum, while absolute error measured by $|\mu(k)|$ decreases to zero at a constant rate and does not level off (it is also not differentiable at $\mu(k) = 0$ for each k). Please see the references just cited for more examples and discussion.

The parameter $\lambda \geq 0$ adjusts the trade-off between sub-optimality in the likelihood of the measured response, and the sparsity of μ . Since effective values of λ for a given problem depend on the measurement noise variance σ^2 , we will frequently specify the product $\lambda\sigma^2$ (rather than just λ) to highlight the interdependence between these two constants.

The fact that (5) is a convex optimization problem is an important feature for practical applications. Primarily, it means the optimal solution can be computed *globally*, in a robust and efficient manner [14].

A. Relation to Least-Squares, Correlation Detectors and Optimal Input Signals

To see that (5) is a generalization of the least-squares problem we need only set $\lambda = 0$. In this case, the optimal solution is well known: $\mu^* = (V^T V)^{-1} V^T v_r$, assuming $(V^T V)^{-1}$ exists. From here we will make a few observations.

The first is that if $V^T V = I$, then the least-squares estimate reduces to $\mu^* = V^T v_r$, which when written out becomes the familiar discrete equation for the correlation between the input and output signals:

$$\mu^*(k) = \sum_{j=0}^{n-1} V_i(j-k) V_r(j), \quad (6)$$

The condition $V^T V = I$ implies the input signal is shift orthogonal, a condition that is met only when the frequency spectrum given by the DFT of $V_i(k)$ has uniform magnitude across all frequencies. Furthermore, we can show this type of signal minimizes the mean square error between the actual reflection coefficient profile and its estimate under the assumption that the only interference source is Gaussian noise. Thus, shift orthogonal input signals are in this sense the *optimal* input signals. There are at least 2 important examples. The first is of course $V_i(k) = \delta(k)$, where $\delta(k)$ is the well known discrete impulse function. The second is a pseudo noise sequence with uniform magnitude and random phase in the frequency domain. These signals are technically only circular-shift orthogonal, but are sometimes also considered shift orthogonal in a statistical sense (e.g., $\mathbf{E}[V_i(j)V_i(j-k)] = \delta(k)$). From these considerations we can see that the optimality of the sliding correlator detection method relies heavily on the assumption that the input interrogation signal is shift orthogonal.

However, in practice these optimal input signals are not typically found because to avoid aliasing effects it is good practice to sample considerably faster than the highest frequency in the input signal. Thus the frequency spectrum of the input signal is never uniform across all frequencies. Of course one could measure the system response at a high sampling rate, and then process (filter and subsample) the input signal and measurements to get the desired result, but that approach usually destroys information. For example, consider Figure 3, which shows the measured input signal of the 3M 900AST handheld TDR device. This signal is sampled at 0.32 ns, and would be approximately shift orthogonal for shifts of 50 ns. Subsampling to obtain the desired “ideal” discrete input would ruin the additional information in the side lobes of the time domain signal, information that might improve fault detectability and resolution.

Despite our best efforts, in practice all input signals are non-ideal. In general, over sampling leads to effective zeros in the DFT of the input signal. That reduces the effective rank of $V^T V$, which leads to ambiguity in the best least-squares estimate. The method presented in this article attempts to resolve that ambiguity with the prior information that the faults are sparsely populated along the length of the wire. It is in this way that an improvement over existing correlation based TDR fault detection algorithms is obtained.

B. Example

To simulate the TDR measurement process, we begin by generating a sparse vector of reflection coefficients $\mu \in \mathbf{R}^n$ as follows:

- 1) Randomly pick an integer N between 0 and 10 (with equal probability). N is the number of faults on the wire.

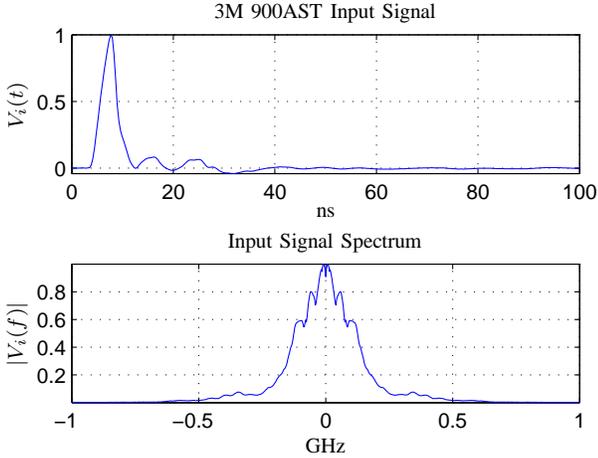


Fig. 3. Normalized input signal and frequency spectrum from a commercially available 3M 900AST handheld TDR unit.

- 2) Draw N random reflection coefficients from a uniform distribution on $[-0.5, 0.5]$.
- 3) Assign the coefficients to N randomly chosen (equally probable) locations in μ , and set all other elements to zero.

Next, n measurements of the reflected signal $V_r(k)$, for $k = 0, 1, 2, \dots, n-1$, are obtained by using the TDR measurement model (1), with some specified input signal $V_i(k)$. This method is used to generate simulated TDR data for the rest of the paper.

Consider an example with $n = 200$. The above procedure was used to generate a sparse reflection coefficient vector $\mu \in \mathbf{R}^{200}$, and a measurement of the reflected signal $V_r(k)$, from a unit pulse input signal $V_i(k)$ (a traditional TDR input signal), and measurement noise $\sigma = 0.02$. The ℓ_1 -regularized LSP (5) was then solved for several different values of λ using CVX, a package for specifying and solving convex programs [15], [16]. The input signal, reflected signal, and detection results are plotted in Figure 4. This figure also shows the normalized correlation signal between the input pulse and the reflected response (*i.e.*, $V^T v_r / \|v_i\|^2$). While the ℓ_1 method successfully detects all the faults in this example, the traditional correlation based detection method, which looks for peaks in the correlation signal, would fail to detect the first fault due to its close proximity to the second larger fault.

C. Polishing

The previous example shows that for larger values of λ , the estimated reflection coefficients appear in the correct location, but typically have reduced amplitude (see Figure 4). This can be viewed as an artifact of the ℓ_1 -norm penalty function, since it favors smaller elements in μ .

A simple technique called *polishing* alleviates this problem, simply by solving the original problem (3) with the sparsity pattern obtained from the solution to the ℓ_1 -regularized heuristic (5). Of course, when problem (3) has a fixed sparsity pattern, it becomes a simple least-squares problem.

Figure 5 shows the effect of polishing on the previous example (for the second largest value of λ considered). Note, at least in this case, the technique almost always does the right thing: $\mu_{est}(k)$ is brought closer to the actual value.

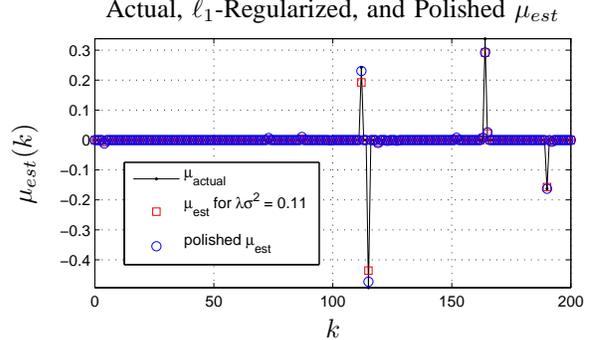


Fig. 5. Polishing example. The plot shows how the polished reflection coefficients μ_{est} , are closer to the actual values than the original set of estimated coefficients.

IV. SELECTING λ

Intuitively, larger values of λ lead to sparser solutions, and sparser solutions lead to the missed detection of small faults (false negatives). On the other hand, a detector that trips too easily will find faults that are not really there (false positives), which is also impractical especially when a technician is constantly sent to find and repair specious faults. This section explores how the selection of λ affects performance results in two important ways. First, we establish a theoretical result that determines the effective range of values λ can take along with the condition under which the sensor detects no faults. Second, we show how Monte Carlo methods are used to quantify the trade-offs between differing values of λ for a specific example.

A. The No-Fault Condition

This section presents how the selection of λ determines the no-fault condition (*i.e.*, all estimated reflection coefficients are zero).

We begin by defining the correlation signal $y(k)$ as,

$$y(k) = \sum_{j=0}^{n-1} V_i(j-k)V_r(j), \quad (7)$$

For each value of k , this signal measures the correlation between the measured response, and the input signal shifted k units in time.

Using subgradient calculus it is readily shown the optimal solution to (5) is $\mu = 0$, if and only if

$$\|\nabla f_0(0)\|_\infty = \max_{k=0, \dots, n-1} \left\{ \left| \frac{\partial f_0(0)}{\partial \mu_k} \right| \right\} \leq \lambda. \quad (8)$$

For our problem this implies:

$$\left| \frac{y(k)}{\sigma^2} \right| \leq \lambda \quad \text{for all } k = 0, 1, \dots, n-1. \quad (9)$$

This sensitivity condition simply states that if the best case correlation $y(k)$ to noise ratio is less than λ , then the optimal

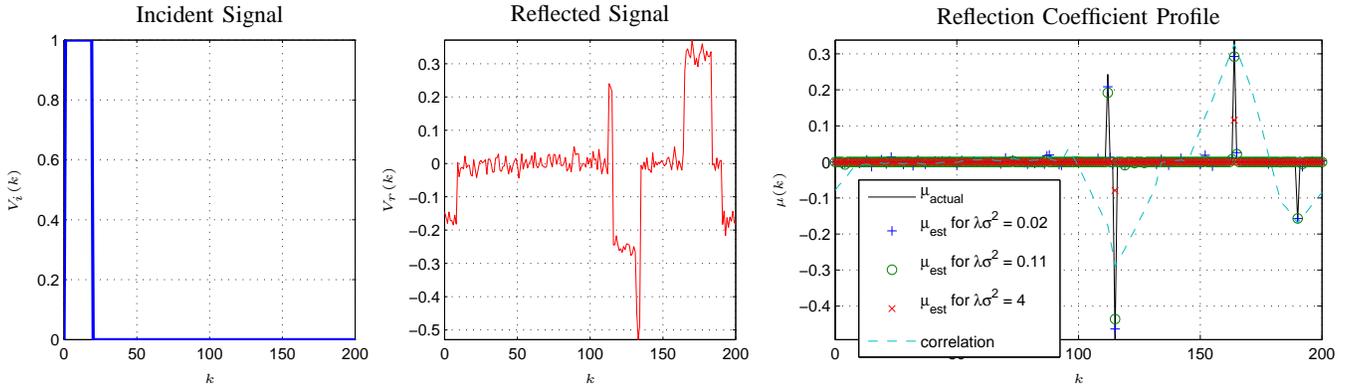


Fig. 4. A reflection coefficient estimation example. The estimation results for different values of $\lambda\sigma^2$ are shown on the plot to the right. Note, that in all cases most of the reflection coefficients $\mu(k)$ are zero as desired. The normalized correlation between the input and output signals is also shown. Peak detection applied to the correlation signal would clearly fail to resolve the first two faults in close proximity to each other.

solution to (5) will indicate no faults on the line (of course in reality faults may still be present). There are two ways to view this condition. First, given a particular reflected signal it determines a finite maximum value λ can take before the sensor ceases to detect any faults. Second, and more importantly, given a particular fixed setting for λ it determines the signal to noise ratio needed before the sensor trips. Thus, the condition is met each time the sensor clears a wire as unfaulted. With these two view points in mind, the condition might be used to select λ to create a fault sensor that is less prone to accidental tripping, but only if one can afford decreased sensitivity to smaller faults. This is further explored in the next section.

B. Estimation Performance vs. λ

In this section, Monte Carlo simulation is used to investigate how the selection of λ is used to trade off detection performance goals to meet application specific requirements.

To do this we will continue to build on the previous example with $n = 200$. First, a set of 100 random reflection coefficient profiles, and corresponding TDR response data, were generated using the same process described earlier in §III-B (again with fixed noise standard deviation $\sigma = 0.02$, and the same input pulse voltage wave). For each measured response, the estimation problem (5) was solved for a series of values λ . The number of false positives and false negatives were counted. Figure 6 shows the Receiver Operating Characteristic (ROC) curve for these results. As one might expect, larger values of λ lead to fewer false positives (because we are encouraging sparsity) and as a consequence, more false negatives. The plot also shows the true negative vs. false negative ROC curve for a simple correlation based detector, over a range of threshold values. Clearly, the ℓ_1 method significantly outperforms this simple detector over nearly all values of λ .

Figure 6 provides us with a way to make decisions about which value of λ we want to use. For example, if we require a false negative rate less than 10%, we might select $\lambda\sigma^2 = 0.1$. With this setting now fixed, we evaluated the estimation performance on a new *test set* of 400 more random coefficient profiles and TDR response data. For this set of 80000 test

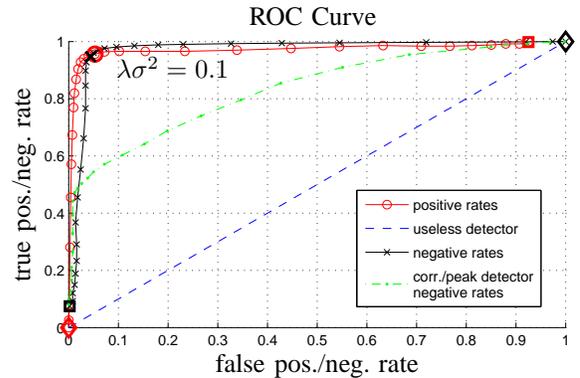


Fig. 6. ROC performance curve. Each point on the curve corresponds to a different value of $\lambda\sigma^2$. For the positive (negative) rates $\lambda\sigma^2$ increases from right (left) to left (right). The \square s corresponds to $\lambda\sigma^2 = 0$, and the \diamond s corresponds to $\lambda\sigma^2 \simeq 22$, which is the value that causes the method to report no faults along the wire (see §IV-A). The true negative vs. false negative curve for a correlation based detection method over a range of threshold values is shown for comparison.

points, the false positive rate was 5.93%, and the corresponding false negative rate was 4.84%. Furthermore, since false negatives might be of particular concern, Figure 7 compares the overall distribution of nonzero fault amplitudes to the distribution of actual fault amplitudes contributing to the false negatives in the test set. This plot clearly shows false negatives are more likely for smaller reflection coefficients, as one might expect, and that this particular detector was able to identify all faults with amplitudes greater than about 0.1. Finally, note that Figure 5 already presented an example comparing the actual, estimated, and polished reflection coefficients achieved with this value of $\lambda\sigma^2$.

Before moving on we wish to make clear that the results presented here are dependent on the particular input signal used and the system noise. This section was intended to highlight the potential performance achievable for this type of detector, and more importantly to provide a clear example demonstrating how Monte Carlo techniques are used to explore the full trade-offs between performance goals. Ultimately, the actual selection of λ must lie in the hands of the end users and the specific requirements of their applications.

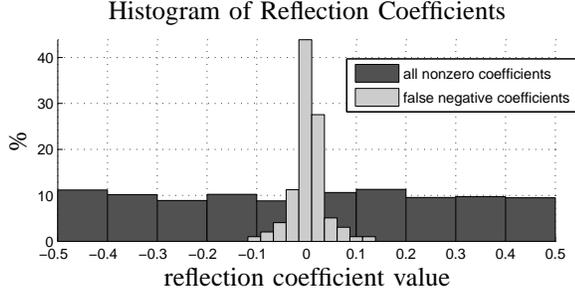


Fig. 7. A comparison between the actual distribution of all nonzero reflection coefficients, and the distribution of coefficients corresponding to the false negatives in the test set.

V. SOLVING LARGE SCALE PROBLEMS

The ℓ_1 -regularized LSP (5) is readily solved for small to medium sized problems through any one of a variety of existing solvers (some of which are available online under the GNU Public License): CVX [15], [16], MOSEK [17], `l1-magic` [12], and LASSO [18], [19] to name a few. For example CVX can handle problems with up to a few thousand reflection coefficients.

Here we consider using yet another solver, `l1_ls` [7]. This Matlab based solver uses a truncated Newton interior-point method that computes search directions with a preconditioned conjugate gradient algorithm [6]. Through these techniques, `l1_ls` allows us to solve our particular estimation problem for a large number of reflection coefficients ($n = 100000$ or more) by taking advantage of algorithms that efficiently compute convolution.

A. Implementation

The `l1_ls` algorithm solves the general ℓ_1 -regularized LSP problem:

$$\text{minimize } \|Ax - y\|_2^2 + \hat{\lambda}\|x\|_1, \quad (10)$$

with variable $x \in \mathbf{R}^n$, given the observations $y \in \mathbf{R}^m$, and data matrix $A \in \mathbf{R}^{m \times n}$. Clearly, this handles the estimation problem (5) we are interested in with $x = \mu$, $y = V_r$, $\hat{\lambda} = 2\lambda\sigma^2$, and $A = V$. Note that A is an $n \times n$ convolution matrix entirely determined by the input interrogation signal $V_i(k)$.

Conveniently, the `l1_ls` routine allows one to overload matrix multiplication by A and A^T (by creating a new Matlab object), when there is a more efficient way of performing the calculation. This is important because the cost of solving (10), via `l1_ls`, is dominated by the cost of performing matrix vector multiplies by A and A^T , which is up to order n^2 floating point operations (flops). However, it is often possible to achieve a substantial improvement by exploiting the structure in A . For our estimation problem, multiplication by A computes convolution, and multiplication by A^T computes correlation. As we will review in the next section, both of these operations are performed efficiently with the FFT in order $n \log(n)$ flops.

B. Fast Convolution

This section reviews how the FFT algorithm is used to efficiently compute the convolution needed for our problem.

We start by defining the *circulant Toeplitz* matrix $C(r)$ as:

$$C(r) = \begin{bmatrix} r_0 & r_{-1} & r_{-2} & \dots & r_{1-n} \\ r_1 & r_0 & r_{-1} & \dots & r_{2-n} \\ r_2 & r_1 & r_0 & \dots & r_{3-n} \\ \vdots & \vdots & \vdots & \ddots & \vdots \\ r_{n-1} & r_{n-2} & r_{n-3} & \dots & r_0 \end{bmatrix}, \quad (11)$$

where $r_{-k} = r_{n-k}$. With this definition it is not hard to see that Cx computes the circular convolution between $r \in \mathbf{R}^n$ (the first column of C) and a vector $x \in \mathbf{R}^n$ in order n^2 flops.

We can, however, use the FFT to compute the same product in order $3n \log(n)$ flops, which is significantly less than n^2 for any appreciable value of n . Let $F \in \mathbf{C}^{n \times n}$ be the matrix that computes the discrete Fourier transform of a vector in \mathbf{R}^n , with inverse F^H , the complex conjugate transpose of F . Using the fact that the Fourier transform converts convolution in the time domain into multiplication in the frequency domain, we have:

$$y = Cx = F^H \text{diag}(Fr)Fx. \quad (12)$$

Thus, circular convolution is efficiently calculated via the following steps:

- 1) Use the FFT to compute $u = Fx$ and $v = Fr$ (order $2n \log(n)$).
- 2) Perform an element by element multiply between u and v (order n).
- 3) Compute y by taking the inverse FFT of the result from step 2 (order $n \log(n)$).

Note that we never actually form the matrices F or $\text{diag}(Fr)$ in this process. Furthermore, we also get an efficient method for computing $C^T x$, by simply noting that from equation (12) we have $C = F^H \text{diag}(Fr)F$. Thus,

$$C^H = C^T = F^H \text{diag}(\overline{Fr})F. \quad (13)$$

Therefore, to compute $C^T x$, the same process enumerated above is used, except in step 2 we multiply by the complex conjugate of v .

To implement the causal (rather than circular) convolution version of our problem we simply use zero padding. Specifically, we construct $C(r) \in \mathbf{R}^{2n \times 2n}$ by setting $r = (V_i, \mathbf{0})$, where $\mathbf{0} \in \mathbf{R}^n$ is a vector with all zero elements. Thus, the causal part of the convolution (this is Ax with respect to the `l1_ls` algorithm) is just the first n elements of $C\hat{x}$, where $\hat{x} = (x, \mathbf{0})$. The same idea holds for $C^T \hat{x}$.

Finally, we note for some specific input signals V_i , it is even possible to implement faster convolution than with the FFT. A trivial example is $V_i(k) = \delta(k)$, where $\delta(k)$ is the discrete delta function. In this case, we do not have to perform a convolution at all. Another example is $V_i(k) = u(k)$, where $u(k)$ is a discrete step function. It is not hard to see that convolution with this function can be computed in order n flops.

C. Speed Performance Example

In this section we compare the performance between CVX and `l1_ls` (using efficient FFT convolution). To do this we solved our estimation problem (5) for increasing values of the

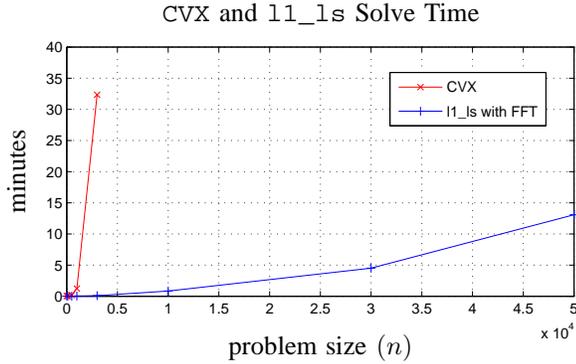


Fig. 8. Comparison between CVX and `l1_ls` for solving large-scale reflection coefficient estimation problems.

problem size n , and clocked the time taken to find the optimal solution (on a 1.8GHz Intel Core Duo processor under 64-bit Linux).

The measurement data was generated by the same method presented in §III-B. This data was then used to estimate the reflection coefficients μ_i , with $\lambda\sigma^2 = 1/2$, for the different solvers. Figure 8 shows the dramatic improvement obtained by the `l1_ls` solver for increasing values of n . We note that the solution time for the `l1_ls` solver tends to vary, depending on V_i , and the actual number of nonzero reflection coefficients (this behavior is not expected of CVX). For the test cases we tried, this variance was on the order of minutes for the larger values of n . However, in general, the `l1_ls` method always performed much better than CVX.

VI. REAL TDR DATA EXAMPLE

In the above development it may have seemed a large number of assumptions were made which do not strictly hold in practice. In particular, we assumed a lossless transmission line (and lossless faults), and that the load and source impedances were matched. While these assumptions were needed to make a logical derivation of the linear model, it turns out they do not have to strictly hold in practice and that many wiring systems, by their very nature, get close enough that the method presented here remains effective. In this section an example using real TDR data collected from a faulted wire is used to demonstrate that this method can remain successful in at least one extreme case where those assumptions are challenged in every way.

The experiment was conducted on a one meter section of *twisted shielded pair* (TSP), a common type of shielded aircraft wire that is very lossy. To simulate one form of chafing fault, a diamond coated abrasive rod was used to file away a small 10×2 mm section of the shield as shown in Figure 9. The cable was then connected to an Agilent 54754A digital TDR unit with the fault located about 66 cm down the line. This particular TDR unit uses an input 30 ps rise time step voltage, which was *applied across one of the wires inner conductors and the shield*. A measurement of the actual input signal along with the reflected response for both the unfaulted and faulted cable is shown in Figure 10. The measured data was linearly interpolated onto a grid of 1024 evenly spaced time samples, but *no* other preprocessing was performed. The



Fig. 9. Image of a chafing fault breaching the shield of a common type of aircraft wire (twisted shielded pair). The internal twisted pair that carries the signal is largely unaffected by this type of fault. However, methods that can detect it enable technicians to make repairs before serious problems occur.

reflection coefficients were computed in 2.4 seconds, and the results are also shown in Figure 10. As noted in the figure caption, there are now nonzero reflection coefficients caused by an impedance mismatch between the TDR port and the wire, the fault, and the hard reflection from the open ended wire. Although small, the set of reflection coefficients caused by the fault are well localized, easy to spot, and clearly singled out from the other nonzero coefficients. Note the detection succeeds without using any additional processing or baseline information (*e.g.*, like subtracting the TDR response of the undamaged wire). That is an important result because in many applications baseline information is inconsistent.

VII. CONCLUSION

In this paper we have described a method for detecting and locating wiring damage using TDR measurement data. Unlike some other methods, this one uniquely takes advantage of the fact that faults are often sparsely located along the length of the wire. We demonstrated the effectiveness of our method on a simulated example, and showed how Monte Carlo simulation might be used to tune it (by selecting λ) to achieve specific detection goals (like a certain false positive error rate). In addition, we saw that preexisting algorithms, like `l1_ls`, can be adapted to efficiently solve large-scale (high resolution) versions of our estimation problem. Finally, we applied the method to actual TDR data and revealed its impressive ability to identify a very subtle type of damage. It is hoped the fault detection method presented here will serve as a straightforward improvement to existing techniques that is readily put into practice.

ACKNOWLEDGMENT

The author would like to thank Prof. S. Boyd for his guidance in developing many of the ideas presented in this paper as a project component of his EE364b Convex Optimization II course, at Stanford University. Many thanks to Dr. K. Wheeler, Dr. D. Timucin, and Dr. M. Kowalski at NASA Ames Research Center for their leadership and help through many enlightening discussions (and also for getting the author involved with this project).

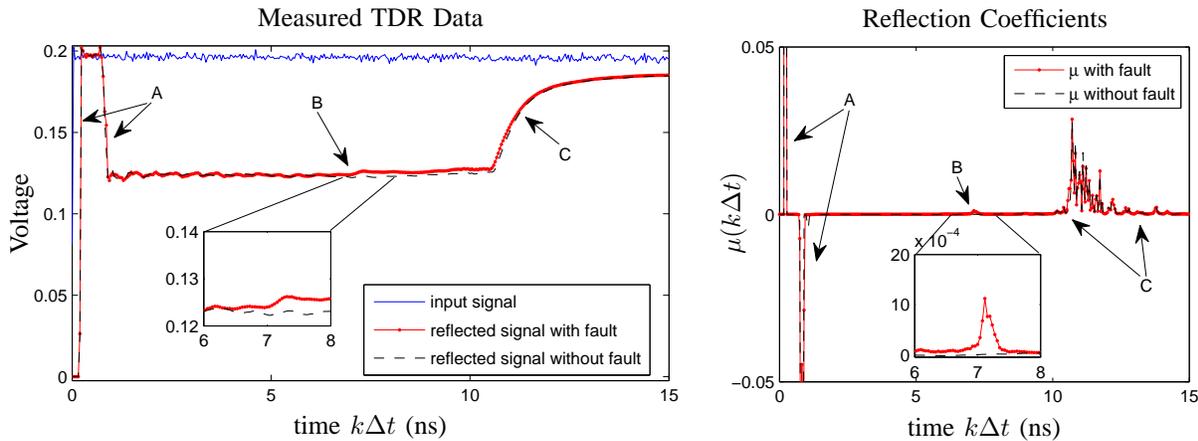


Fig. 10. (Left) Measured input, and reflected voltage waves for both the faulted and unfaulted wire, recorded with a digital TDR unit (Agilent 54754A). There are three main noticeable effects: the reflections caused by a mismatched impedance between the TDR port and wire (A), the reflection caused by the chafe (B), and a lossy reflection caused by the open load at the end of the wire (C). (Right) Reflection coefficient $\mu(k)$ estimation results for both the faulted and unfaulted case, using ℓ_1 -regularized least squares, with $n = 1024$, $\lambda\sigma^2 = 0.005$, and $\Delta t = 0.04$ ns (the entire recorded signal is not shown). The result shows the reflection coefficients caused by the port impedance mismatch (A), the chafing fault (B), and a distribution of coefficients caused by the lossy reflection from the end of the line (C). The faulted region is well localized and much easier to discern (without looking to the baseline signal for comparison) in the reflection coefficient plot.

REFERENCES

- [1] P. Smith, C. Furse, and J. Gunther, "Analysis of spread spectrum time domain reflectometry for wire fault location," *IEEE Sensors J.*, vol. 5, no. 6, pp. 1469–1478, December 2005.
- [2] C. Furse, P. Smith, C. Lo, Y. Chung, P. Pendayala, and K. Nagoti, "Spread spectrum sensors for critical fault location on live wire networks," in *Structural Control and Health Monitoring*, June 2005, pp. 257–267.
- [3] C. Furse, S. Wu, M. Diamond, D. Mih, C. Lo, and P. Smith, "The potential of prognostics for preventative maintenance of electrical wiring," in *Conference on Aging Aircraft*. FAA and DoD and NASA, April 2007.
- [4] C. Furse, Y. Chung, C. Lo, and P. Pendayala, "A critical comparison of reflectometry methods for location of wiring faults," in *Smart Structures and Systems*, 2006, pp. 25–46.
- [5] L. Griffiths, R. Parakh, C. Furse, and B. Baker, "The invisible fray, a critical analysis of the use of reflectometry for fray location," *IEEE Sensors J.*, vol. 6, no. 3, pp. 697–706, June 2006.
- [6] S. Kim *et al.*, "An interior-point method for large-scale ℓ_1 -regularized least squares," *Journal of Machine Learning Research*, vol. 8, pp. 1519–1555, July 2007.
- [7] K. Koh, S. Kim, and S. Boyd. (2008) ℓ_1 _ls: Simple matlab solver for ℓ_1 -regularized least squares problems (web page and software). [Online]. Available: http://stanford.edu/~boyd/l1_ls
- [8] J. M. Mendel and F. Habibi-Ashrafi, "A survey of approaches to solving inverse problems for lossless layered media systems," *IEEE Trans. Geosci. Remote Sens.*, vol. GE-18, no. 4, pp. 320–330, October 1980.
- [9] A. Bruckstein and T. Kailath, "Inverse scattering for discrete transmission-line models," *SIAM Review*, vol. 29, no. 3, pp. 359–389, September 1987.
- [10] J. Lundstedt and S. Strom, "Simultaneous reconstruction of two parameters from the transient response of a nonuniform LCRG transmission line," *Journal of Electromagnetic Waves and Applications*, vol. 10, no. 1, pp. 19–50, 1996.
- [11] R. Collin, *Foundations for Microwave Engineering*, 2nd ed. New York: McGraw-Hill, Inc., 1992.
- [12] E. Candes and J. Romberg. (2006) ℓ_1 -magic: A collection of MATLAB routines for solving the convex optimization programs central to compressive sampling. [Online]. Available: www.acm.caltech.edu/l1magic
- [13] E. Candes, M. Wakin, and S. Boyd, "Enhancing sparsity by reweighted ℓ_1 minimization," *Journal of Fourier Analysis and Applications*, vol. 14, no. 5, pp. 877–905, December 2007.
- [14] S. Boyd and L. Vandenberghe, *Convex Optimization*. Cambridge, UK: Cambridge University Press, 2004.
- [15] M. Grant and S. Boyd. (2008) CVX: Matlab software for disciplined convex programming (web page and software). [Online]. Available: <http://stanford.edu/~boyd/cvx>
- [16] M. Grant and S. Boyd, "Graph implementations for nonsmooth convex programs," in *Recent Advances in Learning and Control*, V. Blondel, S. Boyd, and H. Kimura, Eds. New York, NY: Springer, 2008, pp. 95–110. [Online]. Available: http://stanford.edu/~boyd/graph_dcp.html
- [17] MOSEK ApS. (2008) The MOSEK optimization tools manual version 5.0. [Online]. Available: www.mosek.com
- [18] R. Tibshirani, "Regression shrinkage and selection via the lasso," *J. Royal. Statist. Soc. B.*, vol. 58, no. 1, pp. 267–288, 1996.
- [19] B. Efron, I. Johnstone, T. Hastie, and R. Tibshirani. (2003) LARS software for Splus and R. [Online]. Available: <http://www-stat.stanford.edu/~tibs/lasso.html>



Stefan Schuet (M'09) received the B.S. and M.S. degrees in electrical engineering from Santa Clara University, Santa Clara, CA, in 2001 and 2004, respectively.

He is a Research and Development Engineer at NASA Ames Research Center, Moffett Field, CA. Over the last decade he has worked in the field of electronic sensors and instruments to create novel hardware and software signal processing systems. Projects include the 3D imaging of CEV/Orion thermal protection materials, automated Space Shuttle tile defect inspection, and Mars rover instrument placement. He is presently working in the Intelligent Systems Division to research and develop advanced applications for optimization, statistical signal processing, and Bayesian inference.

Mr. Schuet received the 2008 NASA Invention of the Year Award and the *R&D Magazine* Top 100 Award in 2009.