Robust identification and characterization of thin soil layers in cone penetration data by piecewise layer optimization

Jon Cooper, Eileen R. Martin, Kaleigh M. Yost, Alba Yerro, Russell A. Green

Virginia Tech, Blacksburg, VA, USA

Abstract

Cone penetration testing (CPT) is a preferred method for characterizing soil profiles for evaluating seismic liquefaction triggering potential. However, CPT has limitations in characterizing highly stratified profiles because the measured tip resistance (q_c) of the cone penetrometer is influenced by the properties of the soils above and below the tip. This results in measured q_c values that appear "blurred" at sediment layer boundaries, inhibiting our ability to characterize thinly layered strata that are potentially liquefiable. Removing this "blur" has been previously posed as a continuous optimization problem, but in some cases this methodology has been less efficacious than desired. Thus, we propose a new approach to determine the corrected q_c values (i.e. values that would be measured in a stratum absent of thin-layer effects) from measured values. This new numerical optimization algorithm searches for soil profiles with a finite number of layers which can automatically be added or removed as needed. This algorithm is provided as open-source MATLAB software. It yields corrected q_c values when applied to computer-simulated and calibration chamber CPT data. We compare two versions of the new algorithm that numerically optimize different functions, one of which uses a logarithm to refine fine-scale details, but which requires longer calculation times to yield improved corrected q_c profiles.

Keywords: cone penetration test, data quality, inverse problems

1. Introduction and Motivation

Cone penetration testing (CPT) is a preferred method to characterize soil profiles to evaluate seismic liquefaction triggering potential. The test consists of hydraulically pushing an instrumented cone-shaped penetrometer into the soil profile at a constant rate, with measurements typically taken every one to two centimeters as the cone advances. In its basic form, CPT sounding data include tip resistance (q_c) and sleeve friction (f_s) as a function of depth (Schmertmann, 1978). CPT q_c profiles are extensively used in geotechnical applications, in particular, they serve as a proxy for a soil's ability to resist liquefaction triggering due to ground shaking (Shibata and Teparaksa, 1988).

CPT q_c profiles do not provide truly depth-specific measurements, because they are influenced by soil materials several cone diameters away from the cone tip (Ahmadi and Robertson, 2005). Consequently, if soil properties vary with depth, the measured q_c are "blurred" compared to the actual depth-specific or "true" corrected q_c (i.e., the q_c value that would be measured at that depth in a uniform profile, absent of boundary or thin-layer effects). This "blurring" is asymmetrical, with soil below the cone tip affecting q_c more strongly than soil above, as illustrated in Figure 1 (Boulanger and DeJong, 2018). This figure shows a multi-layer "true" q_c profile along with the computationally simulated data we would generate by convolving the profile with an asymmetrical function. Notice how the peaks of the simulated data are shifted up relative to the true high q_c layers and how the true q_c in the thin layers is obscured. Because soil profiles are typically stratified, the location of the interfaces between layers and the true q_c of layers can become difficult to precisely identify from the measured q_c, even in relatively simple profiles. These phenomena are typically referred to as thin-layer, transition-zone, or multiple thin-layer effects, as discussed in Yost et al. (2021b). Herein, we will generally refer to all of these effects as "thin-layer effects". In this context, a soil "layer" or "stratum" is a depth increment in the profile over which the soil has relatively uniform geotechnical engineering properties (e.g., soil type and q_c). Furthermore, a "thin" layer or stratum is one that is too thin for the measured q_c to fully develop or reach values that would be measured in the stratum absent of thin-layer effects (i.e. true corrected q_c profile). This required thickness will vary as a function of soil stiffness, but typically 10 to 30 cone diameters is required (Ahmadi and Robertson, 2005), and thus a stratum thinner than this would be a "thin layer."

The majority of studies on thin-layer effects on measured CPT q_c data

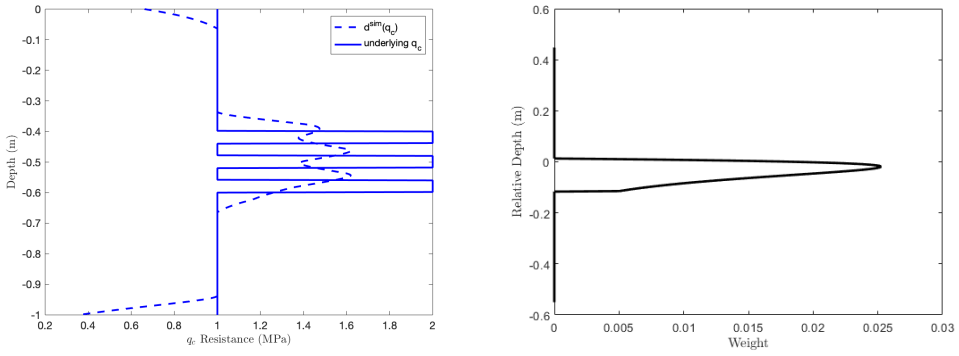

Figure 1: The q_c^{true} profile of a layered soil (left, solid) and its predicted blurred observation, $\tilde{q}_c^{sim}$ (left, dashed), generated by convolving the profile with a point spread function that is a truncated chi-squared distribution (right).

have focused on developing corrections to apply to measured q_c for specific layering sequences and geometries, as outlined in Boulanger and DeJong (2018). Some past efforts to correct the measured q_c for thin-layer effects in profiles consisting of a stiff layer embedded in a soft layer (e.g. sand and clay layers, respectively) assumed that the effect of the softer layer will be greater at the boundaries and lesser towards the middle of the stiff layer. This results in a V-shaped deblurring correction function to correct the CPT-measured q_c in the stiff layer so the corrected values more closely represent the true q_c that would be measured in the absence of thin-layer effects (Youd et al., 2001; de Greef and Lengkeek, 2018).

In comparison to corrections for specific layering scenarios, Boulanger and DeJong (2018) propose a potentially more flexible inverse problem approach that can be applied to measured q_c from a CPT performed in a profile containing any number of layers. However, the procedure is less efficacious than desired in some scenarios where layer thicknesses are less than 40 mm, even if there is significant contrast between the strengths and stiffnesses (i.e., the true q_c) of the thin layers and surrounding soil (Yost et al., 2021a). The inability to cost-effectively determine the true q_c from the measured q_c may be contributing to widespread over-prediction of the liquefaction hazard in highly interlayered soil deposits (e.g., as observed in Christchurch, New Zealand (Maurer et al., 2014, 2015) and the Hawk's Bay region of New Zealand (El Korthawi et al., 2019)). In this regard, a quantitative comparison of previous correction procedures is presented in Yost et al. (2021a).

The main objectives of this work are (i) to pose a new inverse problem to estimate true q_c from measured (or "blurred") q_c in highly stratified soil profiles, and (ii) to propose a new numerical optimization algorithm for efficiently correcting q_c in highly stratified soil profiles for thin-layer effects. This paper is organized as follows: In Section 2, we review the background of prior efforts to adjust or correct CPT data for thin-layer effects, including efforts to pose this correction as an inverse problem to be solved via numerical optimization. Further, we provide relevant background information on numerical optimization techniques. In Section 3, we propose a new approach to correct CPT data by removing the thin-layer effects via an inverse problem, posed in two different ways. We describe a new algorithm to solve both proposed versions of this inverse problem that incorporates global numerical optimization techniques, routines to automatically generate a reasonable initial guess, to adjust the number of layers, and to computationally simulate the blurring process. In Section 4, we show the results of this new algorithm for one version of the inverse problem applied to CPT data from calibration chamber tests, and the improvements typically achieved over simpler computational methods to automatically correct thin-layer effects in measured q_c profiles. In Section 5, we compare the results of the new algorithm for both formulations of the inverse problem, yielding a more precise, but more computationally expensive data correction method. In Section 6, we discuss some possible future extensions of this method, and in Section 7, we discuss limitations and benefits of the new algorithm.

2. Background

In computational science and engineering, the terms "forward problem" and "inverse problem" are often used to describe the problems we solve when different parts of a system are unknown. When we assume we know the subsurface soil characteristics and stratigraphy of a soil profile (i.e., we assume an unblurred q_c profile), and then perform a computational simulation to predict the response of the soil to an action (e.g., simulating a CPT in a "known" (unblurred) q_c profile to compute a simulated blurred tip resistance profile, $\tilde{q}_c^{sim}$, comparable to a measured q_c profile for a real CPT), we are solving the "forward problem." In contrast, we are solving the "inverse problem" when we only know the measured tip resistance profile, $\tilde{q}_c^{meas}$, but need to infer the true soil characteristics and stratigraphy (i.e., q_c^{true}) that would lead to computationally simulated data, $\tilde{q}_c^{sim}$, that most closely match the measured

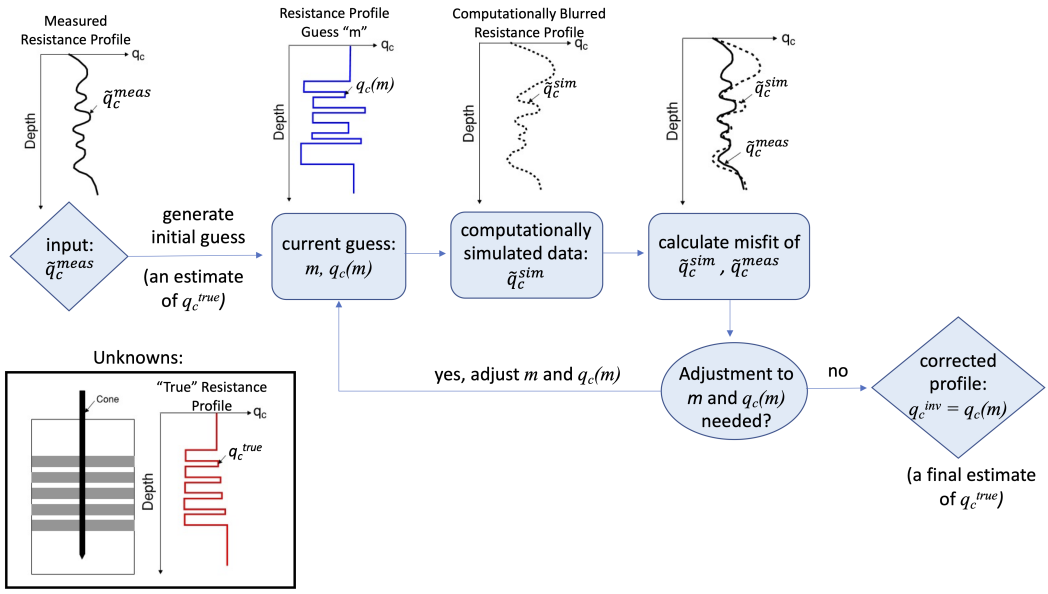

Figure 2: Diagram of the inverse problem approach to remove thin-layer effects.

tip resistance $\tilde{q}_c^{meas}$. In this inverse problem scenario, we begin with some initial guess at q_c and iteratively update the current guess to incrementally improve the match between $\tilde{q}_c^{meas}$ and $\tilde{q}_c^{sim}$ until reaching "the best" guess (i.e., the solution to the inverse problem), denoted by q_c^{inv}.

In this regard, the inverse problem is posed as an optimization problem (i.e., were the minimum difference between $\tilde{q}_c^{meas}$ and $\tilde{q}_c^{sim}$ is targeted) and solved using a variety of numerical optimization techniques. All numerical optimization techniques require calculating $\tilde{q}_c^{sim}$ for every q_c guess, so we solve the forward problem many times to solve the inverse problem once. We can predict $\tilde{q}_c^{sim}$ for any q_c either through i) numerical simulation of the soil being displaced by the cone penetrometer (e.g., with numerical methods such as the material point method (Yost et al., 2021b)), or ii) applying a simplified blurring function to q_c. This workflow, beginning with $\tilde{q}_c^{meas}$, iteratively updating the proposed q_c and its corresponding $\tilde{q}_c^{sim}$, and ultimately outputting the corrected q_c^{inv}, is diagrammed in Figure 2.

Approaches to solve inverse problems have previously been applied to other geotechnical engineering challenges. Notably, the multichannel analysis of surface waves (MASW) technique for seismic imaging of the near surface is an inverse problem (see Socco and Strobbia (2004) for details). However, the

development of thin-layer corrections for CPT data collected in interlayered soil profiles has only recently been posed as an inverse problem by Boulanger and DeJong (2018). In the following, we present some background on these methods.

2.1. Prior Methods and Limitations

Regardless of the method to calculate $\tilde{q}_c^{sim}$, this inverse problem can be generically posed as finding the assumed q_c that minimizes a function known as a *misfit function*:

$$q_c^{inv} := \arg\min_{q_c} \|\tilde{q}_c^{meas} - \tilde{q}_c^{sim}(q_c)\| \tag{1}$$

where $\tilde{q}_c^{meas}$, q_c, and $\tilde{q}_c^{sim}$ are all vectors with as many entries as there are depths of interest. The tilde indicates profiles that are "blurry" while the lack of a tilde indicates profiles that are constrained to be layered profiles with sharp transitions. The misfit function (Eq. 1) measures the difference between the actual measured profile and the simulated measured profile for any q_c guess. An engineers' physical understanding of the influence of thin-layer effects on measured CPT data can be incorporated by using a physically realistic computational simulation process (i.e., "blurring function") to map the current guess at the resistance profile, q_c, to $\tilde{q}_c^{sim}$. However, inverse problems are often ill-posed and data include noise, so it is possible that a small change in $\tilde{q}_c^{meas}$ could allow for significantly different q_c^{inv} profiles that both yield $\tilde{q}_c^{sim}$ equally close to $\tilde{q}_c^{meas}$. By modifying the way that we discretize and mathematically represent the soil profile, changing the form of the misfit function, adding physically realistic restrictions on q_c that are considered, or using different numerical optimization algorithms to iteratively improve the corrected q_c guesses, we may be able to improve the efficacy of our solution.

This inverse problem approach to correcting q_c for thin-layer effects was first used by Boulanger and DeJong (2018). Their key insight was representing thin-layer effects on $\tilde{q}_c^{meas}$ at a particular depth as a simple blurring filter, $w_c(z)$, applied to the true tip resistance profile q_c^{true}. They inherently assumed that the coefficients of the blurring function may vary with depth.

$$\tilde{q}_c^{sim}(q_c) := q_c * w_c(z) \tag{2}$$

where $*$ represents convolution. It is assumed that w_c is the discretization of a continuous function that represents the influence of soil above and be-

low the cone tip on $\tilde{q}_c^{meas}$ at a particular depth. An example of a w_c function that is a scaled and truncated chi-squared distribution is shown along with the layered soil q_c and $\tilde{q}_c^{sim}$ in Figure 1. This differs from the w_c used in Boulanger and DeJong (2018), but our numerical tests suggest this w_c more closely matches calibration chamber data. The numerical optimization method proposed in Boulanger and DeJong (2018) uses a common iterative splitting optimization technique, and smooths the results to keep them from becoming unstable. However, when applied to laboratory calibration chamber test data, we found that this method may still be unstable (i.e., it did not yield corrected q_c profiles that matched the stratigraphy of known soil profiles with thin layers) (Yost et al., 2021a). We have explored a variety of ways to pose this inverse problem as different optimization problems, methods to discretely represent the soil profiles, and numerical optimization techniques including both gradient-based methods and global optimization techniques. In this paper we i) propose two new representations of this inverse problem, ii) detail a new robust numerical optimization algorithm for each representation to find the best guess for the resistance profile, q_c^{inv}, iii) present the tradeoffs in accuracy and computational cost of these algorithms, and iv) provide open source MATLAB code for all algorithms and test cases.

3. New Method

We generally assume each soil layer is homogeneous and refer to the "corrected tip resistance" in this homogeneous layer as the tip resistance that would be measured in an entirely uniform profile of the same material (perhaps with some level of noise). Accordingly, our new method describes any proposed q_c profile as a piecewise constant function. Assuming there are N soil layers, each layer in the piecewise constant function is represented by just two degrees of freedom, rather than having as many degrees of freedom as number of depths where CPT data were collected. The two variables associated with each layer would be i) its depth, and ii) its characteristic tip resistance if the uniform material in the layer were measured without any thin-layer effects. This representation results in far fewer degrees of freedom compared to the formulation of Boulanger and DeJong (2018), and improves computational speed of the method. However, even simple soil profiles can have several dozen degrees of freedom (i.e., two degrees of freedom per layer), requiring a global optimization method that balances efficiency and precision.

We define the inverse problem to seek the q_c that results in $\tilde{q}_c^{sim}$ that

most closely matches $\tilde{q}_c^{meas}$. We restrict q_c to be a piecewise constant function defined by N layer depths paired with N q_c values. Therefore, each piecewise proposed q_c profile is described by a material property vector, m, that has $2N$ components. For any assumed m we can reconstruct the corresponding q_c at each depth where CPT data were measured by simply extracting the value of the piecewise function described by m at every depth of interest. The q_c profile resulting from this reconstruction process is denoted by $q_c(m)$. In what we refer to as the *new algorithm with the standard misfit function,* we quantify how closely the measured and simulated profiles match by calculating $q_c(m)$, calculating $\tilde{q}_c^{sim}$ from $q_c(m)$, then calculating the norm (the Euclidean or 2-norm) of the error between $\tilde{q}_c^{sim}$ and $\tilde{q}_c^{meas}$, scaled to be between 0 and 1. For reference, a good fit would have a score of less than 0.01 (i.e., less than 1% error). By convention, we consider the depth of the layer to be the depth to the top of the layer, and we also force the depth to the first layer to be zero. Written as an equation, this new algorithm with the standard misfit function optimizes:

$$m^{inv} = \arg\min_{m \in \mathbb{R}^{2N}} \|\tilde{q}_c^{meas} - \tilde{q}_c^{sim}(q_c(m))\|_2. \tag{3}$$

However, this is not the only way to quantify the misfit between $\tilde{q}_c^{meas}$ and $\tilde{q}_c^{sim}$. Some inverse problems that have data or material parameters that include both small-scale and large-scale values benefit from quantifying misfits with a logarithm applied. Thus we also propose *the new algorithm with the log misfit function*, which optimizes:

$$m^{inv} = \arg\min_{m \in \mathbb{R}^{2N}} \quad \log\left(\|\tilde{q}_c^{meas} - \tilde{q}_c^{sim}(q_c(m))\|_2\right). \tag{4}$$

Computationally optimizing either form of the misfit function based on measured and simulated data allows us to assess q_c guesses without direct knowledge of q_c^{true}, even when additional site characterization data are unavailable. The best assumed q_c profile we reconstruct, $q_c^{inv} = q_c(m^{inv})$, is likely to be close to the q_c^{true} profile with thin-layer effects removed, but practical numerical optimization algorithms may yield different answers depending on the choice of the misfit function.

In addition to designing the optimization problem, one must select a numerical optimization algorithm to iteratively update the q_c guesses. We chose the Particle Swarm Optimization (PSO) algorithm. PSO finds minima of the selected misfit function starting with many randomly generated trial

m values (i.e., the "particles"), each following its own path of new updated guesses of the q_c profile (i.e. guesses for m with a corresponding $q_c(m)$). Each particle explores the space of possible m vectors based on its most recent m guess, the best m guess it has tested, and the best m guess previously tested among all the particles. In this way, PSO does indeed have particles that swarm around local and global minima.

Since PSO does very well when the global minimum is surrounded by local minima or has a wide basin of attraction (i.e. a large region around the global minimum with no other local minima), and small adjustments to either the layer depths or resistances should only marginally affect $\tilde{q}_c^{meas} - \tilde{q}_c^{sim}$, we believe PSO is a practical choice. When $\tilde{q}_c^{meas}$ does not suggest a piecewise constant layer resistance profile (e.g., when there is a gradient in the q_c profile), we can still approximate the result well by adding several additional layers, each with constant q_c. In this case, there may be multiple local minima surrounding the global minimum, and so again PSO should be quite effective. The only drawback to PSO is that it can have low accuracy, i.e., different initial particle guesses may yield quite different q_c^{inv} values even if the average over all particles' q_c guesses is at the global minimum.

The pseudocode for this new algorithm is described in Algorithm 1 below. The following subsections step through the process to automatically compute good initial m guesses, followed by two methods used in tandem with PSO for adjusting the number of layers and re-fitting the q_c profile guess automatically. The pseudocode assumes that the user has already selected whether they wish to use the standard or log misfit function. Further, while the pseudocode is written assuming use of the recommended initialization methods outlined in Section 3.1, an engineer applying this algorithm may substitute their own initial guess of q_c based on their knowledge of local soils and geology.

3.1. Calculating Reasonable Initial m Guess

In order to combat the accuracy limitations of PSO, a standard technique in optimization is to focus on developing a good initial m guess (and its corresponding $q_c(m)$, referred to here as the initial q_c guess) which can then be further refined by PSO. Many PSO implementations allow for the specification of the initial m guess. Even if a single particle starts at a good initial m guess, by the nature of PSO, the other particles will quickly swarm the location and discover the global minimum. Here we propose a novel technique to automatically generate a good initial guess, which is specific to the

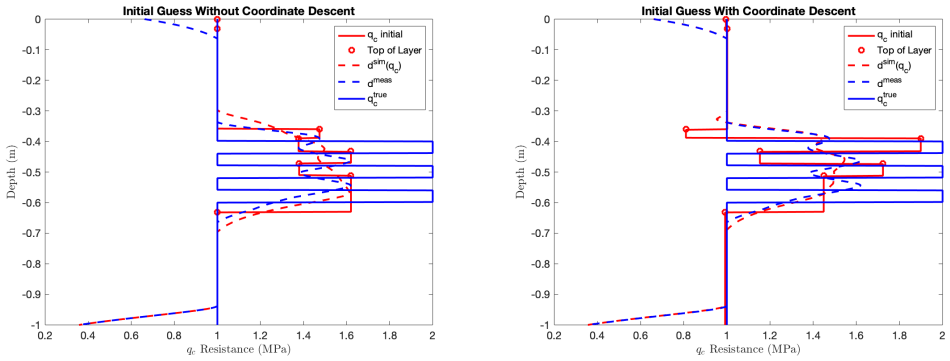

Figure 3: An initial guess for q_c (left, solid red) was automatically generated from derivative sign changes of the measured $\tilde{q}_c^{meas}$ resistance profile (dashed blue). Its computationally simulated $\tilde{q}_c^{sim}$ data is shown(left, dashed red). Coordinate descent optimization starting from that initial guess of q_c yielded a new initial guess (right, solid red) with an improved predicted blurred profile (right, dashed red).

thin soil layer problem.

The first step in constructing a good initial m guess (and its corresponding $q_c(m)$) is to automatically calculate an approximation of N, the number of layers, and the depths of each layer. This can be done by looking at the locations where the derivative of $\tilde{q}_c^{meas}$ changes signs. This will not capture features of q_c^{true} such as gradually increasing/decreasing resistances, but effectively this should identify most locations where there is a transition either from a layer having a low q_c^{true} value to a high q_c^{true} value or vice-versa. The q_c values of the N layers can simply be initialized as being equal to the measured resistances at a subset of the depths where $\tilde{q}_c^{meas}$ was measured.

Although this initialization works much better than a random initialization, there is a chance that the asymmetry of the influence of the soil above and below the cone tip on $\tilde{q}_c^{meas}$, and the number of thin layers might result in an initial m and $q_c(m)$ guess that are offset in depth from the true resistance profile, as seen in Figure 3. To fix this, the new proposed algorithm applies a simple coordinate descent optimization algorithm with the selected misfit function to improve the initial m guess further at low computational cost. Coordinate descent is a common numerical optimization technique (Wright, 2015). Applying coordinate descent optimization here helps to correct m when the locations of multiple layers are shifted from the true layer depths, and improves the estimated q_c values slightly.

The result of these steps is a reasonable initial guess for q_c which, in rare cases, might already be optimal. However, coordinate descent optimization is typically unable to refine the details of q_c^{inv}, so coordinate descent is only used for a quick update to the initial guess followed by application of PSO and computational procedures to add/remove layers for further improvement. This is because PSO explores many more minima than just the single local minimum that coordinate descent finds. By using a good initial guess, PSO takes far less time to converge than with a random initial guess. These steps for constructing a good initial m guess (and its corresponding initial q_c guess) have no way of adjusting the number of layers in the profile, usually resulting in initial guesses that are too simplistic, so our proposed new algorithm includes computational procedures to automatically add and remove layers.

3.2. Leave-One-Out (LOO)

We propose a new computational procedure to improve a q_c guess by removing any layers that would help reduce the misfit function (either the standard or log misfit, depending on the user's choice) up to some pre-defined tolerance, referred to as the Leave-One-Out (LOO) procedure. To accomplish this, LOO computes what the misfit would be if the i^{th} layer were removed from the profile for each $i = 1, 2, \ldots, N$, then removes whichever layer increases the misfit the least, up to the tolerance. This process is repeated until the removal of any single layer increases the misfit beyond the tolerance, or when the model contains only one layer. Algorithm 2 details the pseudocode of the LOO procedure.

LOO was designed to remove insignificant, if not detrimental, layers from any q_c guess that are not physically realistic and contribute to unnecessary additional degrees of freedom (which increase the run-time of) PSO. The provided software includes several options to set the tolerance automatically, most of which only rely on the misfit of the initial q_c guess and do not change between iterations.

3.3. Add-One-In (AOI)

In addition to removing unnecessary layers, we also developed a new computational procedure to automatically add missing layers, referred to as Add-One-In (AOI). AOI adds new layers between existing layers if the addition of that layer would reduce the misfit function at the proposed new profile, until the addition of another layer does not sufficiently decrease the misfit. This is a necessary condition to stop adding layers, since adding a

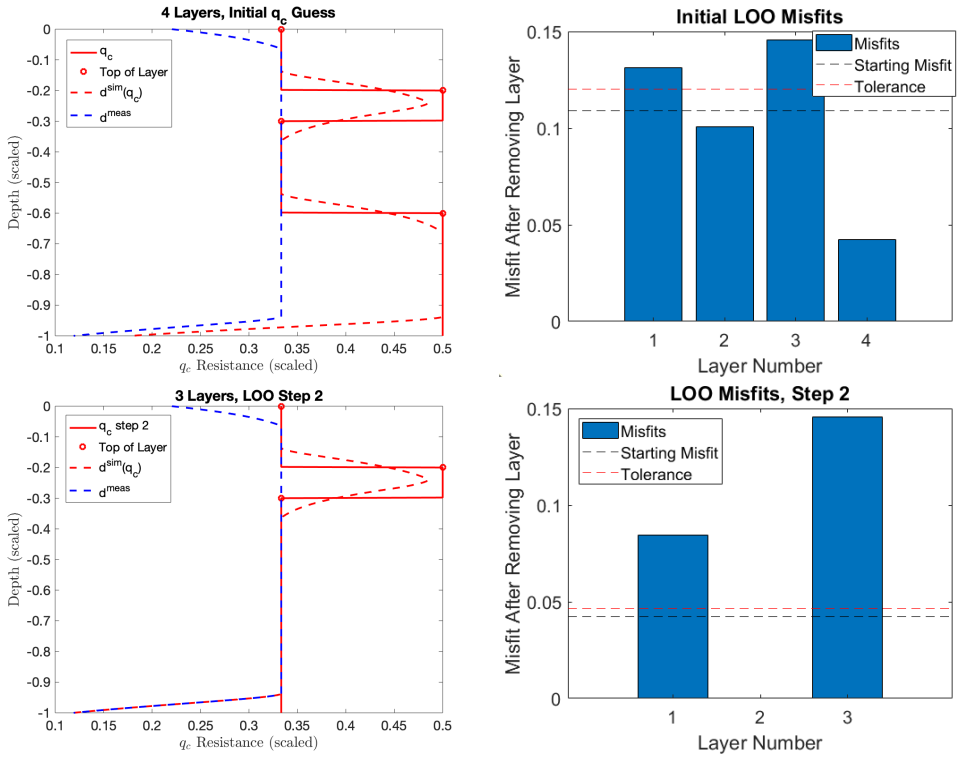

Figure 4: The first (top) and second (bottom) steps of the Leave-One-Out (LOO) process are demonstrated. At each step $\tilde{q}_c^{sim}$ (red dashed) is calculated from the current guess at q_c (red solid), and compared to $\tilde{q}_c^{meas}$ (blue dashed). The predicted misfit if each layer were to be removed is calculated (right). The first step has layers that can be removed to lower the misfit, but no layers are below the tolerance for removal in the second step.

layer is always guaranteed to at least keep the misfit the same, if not decrease it, which potentially allows the number of layers in the profile to grow to infinity without the stopping criteria. AOI accomplishes this by adding in a layer between every two consecutive pairs of layers, improving these layers' q_c values and thicknesses using PSO (a quick, 2-variable optimization for each layer), and computing which proposed additional layer decreases the misfit the most. Algorithm 3 details the pseudocode of the AOI procedure. AOI was designed to populate regions of high data misfit with more layers, assuming the next full application of PSO will be able to adjust these new layers appropriately. AOI struggles to add layers where multiple layers having a mix of high and low q_c values are missing. Unlike LOO, AOI tolerances must be updated each iteration to account for the potentially rapidly decreasing misfit.

Algorithm 1 Thin-Layer Correction Optimization

Require: measured data $\tilde{q}_c^{meas}$, and function to simulate data $blurfcn$
  $ndata \leftarrow$ normalize($\tilde{q}_c^{meas}$)
  create $misfitfcn$, a misfit function based on $ndata$ and $blurfcn$
  initialize m based on where deriv($ndata$)changes signs
  $m \leftarrow$ coordinateDescent(m,$misfitfcn$)
  while length(m) ! = length(m_O) **or** $\|m - m_O\| > \varepsilon$ **or** first iteration **do**
    $m_O \leftarrow m$
    $m \leftarrow$ LOO(m,$misfitfcn$)
    $m \leftarrow$ PSO(m,$misfitfcn$)
    $m \leftarrow$ AOI(m,$misfitfcn$)
    $m \leftarrow$ PSO(m,$misfitfcn$)
  end while
  $m \leftarrow$ LOO(m,$misfitfcn$)
  check for potential uncertainties, state warnings
  $m^{inv} \leftarrow$ rescale m to remove normalization
  $q_c^{inv} \leftarrow$ reconstruct depth profile $q_c(m^{inv})$
  return q_c^{inv}

3.4. Convolutional Blurring Procedure

A user of the proposed new algorithm can use any method to calculate $\tilde{q}_c^{sim}$ that they prefer. For the purposes of this study, we defined an artificial computational blurring function that, when applied to idealized q_c^{true},

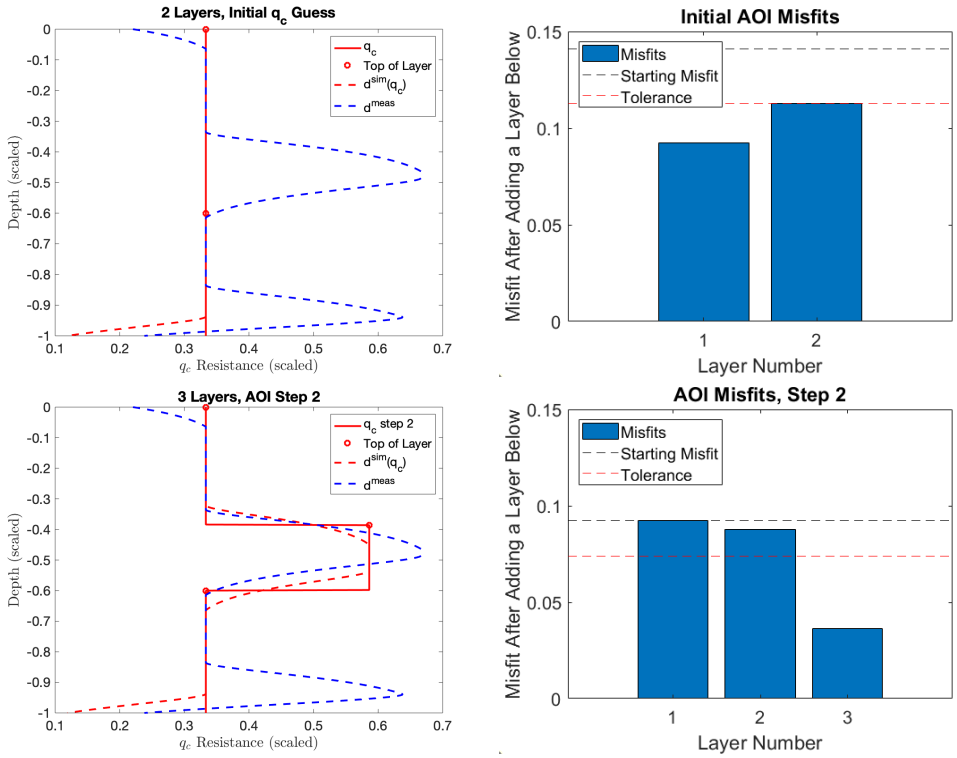

Figure 5: The first (top) and second (bottom) steps of the Add-One-In (AOI) process are demonstrated. At each step the predicted blurred data (red dashed) is calculated from the current guess at the resistance profile (red solid) and compared to the measured resistance profile (blue dashed). The predicted misfit assuming additional layers is calculated (right).

Algorithm 2 Leave-One-Out (LOO)

Require: profile guess m, function to evaluate misfit $misfitfcn$, tolerance TOL

```
misfits ← zeros(N,1)
while true do
  if N == 1 or all(misfits > TOL) then
    break
  end if
  i ← index of minimum entry of misfits
  m ← removeLayer(m, i)
  N ← N − 1
  for i = 1 : N do
    temp ← removeLayer(m, i)
    misfits(i) ← misfitfcn(temp)
  end for
end while
return m
```

replicates the blurring caused by the thin-layer effects. That is, it computationally simulates the $\tilde{q}_c^{sim}$ from q_c^{true}, and the resulting values should be close to the actual $\tilde{q}_c^{meas}$. Similar to Boulanger and DeJong (2018) we chose to use a blurring function defined by convolving the true resistance profile with a *point spread function* $p(z)$:

$$\tilde{q}_c^{sim}(z) := (\tilde{q}_c^{sim}(q_c))(z) = \int_{-\infty}^{\infty} q_c(\Delta z) p(z - \Delta z) \mathrm{d}\Delta z, \tag{5}$$

where $\int_{-\infty}^{\infty} p(z)dz = 1$ and $p(z) \geq 0$ for all z. In practice, this integral is only calculated on a finite interval. This blurring function was chosen because it is simple to implement and only requires the use of a matrix convolution function ("conv" in MATLAB). This blurring function results in a $\tilde{q}_c^{sim}$ value at each depth that is a weighted combinations of the surrounding soil layers' q_c values. Although this method can very quickly compute $\tilde{q}_c^{sim}$ for any q_c guess, it is a simplification of the true physics. For example, in regions with alternating thin layers of stiff and soft soils, q_c^{meas} in the layered zone is much closer to the lower of the two true resistances throughout the region (i.e., it is not a simple averaging process), as we see in Figure 6. Simplified physics models suggest that the layers below the tip of the cone affect the resistance

Algorithm 3 Add-One-In (AOI)

Require: profile guess m, function to evaluate misfit $misfitfcn$
  while true **do**
    recompute TOL
    if $misfitfcn(m) \leq TOL$ **then**
      break
    end if
    $misfits \leftarrow$ zeros($N-1$,1)
    for $i = 1 : N-1$ **do**
      initialize $extraLayer$ between layer i and layer $i+1$
      $temp \leftarrow$ addLayer$(m, extraLayer, i)$
      $misfits(i) \leftarrow misfitfcn(temp)$
    end for
    if all$(misfits) > TOL$ **then**
      break
    end if
    $i \leftarrow$ index of minimum entry of $misfits$
    $m \leftarrow$addLayer(m, $extraLayer, i$)
  end while
  return m

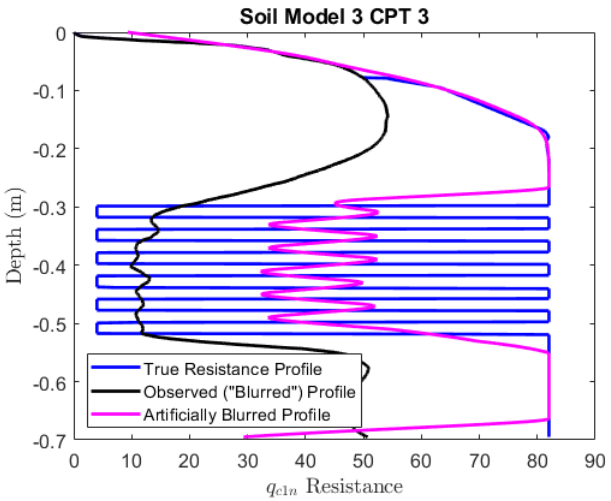

Figure 6: A CPT tip resistance profile from Soil Model 3 CPT 3 of the De Lange (2018) report, Section A. Here, the observed resistance profile $\tilde{q}_c^{meas}$ (black) is being contrasted with $\tilde{q}_c^{sim}$ (magenta) that is calculated from the true resistance profile q_c (blue) by simple convolution.

more than the layers above the cone tip (Boulanger and DeJong, 2018), so in our implementation, we performed the convolution in Equation 5, with a point spread function $p(z)$ derived from the Chi squared probability density function as the artificial blurring function for its smoothness and asymmetry, pictured in Figure 1. Our computational experiments to find the optimal $p(z)$ point spread function indicate that Equation 5 is likely too simple, and the existence of an efficient computational method to predict $\tilde{q}_c^{sim}$ for any q_c remains an open question.

4. Results of the New Algorithm with Standard Misfit

A suite of CPT tip resistance (q_c) data for known sand-clay layered profiles from calibration chamber tests performed by de Lange (2018) at Deltares (de Lange, 2018) were used to test the new algorithm with the standard misfit function. Details on how the digitized data and reported sample preparation were used to estimate q_c^{true} are provided in Appendix A. Before calculations were performed, $\tilde{q}_c^{meas}$ and q_c^{true} profiles were converted to normalized cone tip resistance (q_{c1n}) profiles, where q_{c1n} is computed as:

$$q_{c1n} = C_N \frac{q_c}{P_a} \tag{6}$$

where P_a is atmospheric pressure in the same units as q_c and C_N is a unitless overburden correction factor computed per the procedure in (Boulanger and Idriss, 2016). Furthermore, data in the upper 0.1 meters of the soil profiles was excluded from the analyses presented herein because it contained unintended artifacts caused by experimental testing limits and was considered to be unreliable.

It should be noted that all soil models in the de Lange (2018) report only have layered zones that contain layers of the same thicknesses, however soil models with varying layer thicknesses do not affect the algorithm's performance beyond what is discussed here. For the sake of designing and testing the optimization scheme described in Section 3, we replace our $\tilde{q}_c^{meas}$ data (which are "naturally blurred") with profiles that are "computationally blurred" (i.e., $\tilde{q}_c^{sim}$) by applying the convolution in Equation 5 to known soil resistance profiles. In the field of inverse problems, this is an example of an *inverse crime*, and it is done to test and verify algorithms in a more controlled setting by removing a source of error from these computational tests. This may mean that in practice, the new algorithm is less likely to yield an accurate estimate of q_c^{true}, or that it is more sensitive to the initial q_c profile guess used to begin particle swarm optimization. See the software in Section 8 and Appendix A for more implementation details.

The first soil profile we used to test the new algorithm with the standard misfit was Soil Model 4 CPT 2 from the start-up phase of the de Lange (2018) data (Section C of the report). This soil profile features 80-mm-thick layers of alternating stiffnesses, which are thin enough that existing algorithms struggle to correct for thin-layer effects (Yost et al., 2021a). The results are shown in Figure 7, comparing the automatically generated initial guess for q_c (red solid, left) to the final optimized q_c^{inv} resulting from the new algorithm with the standard misfit function (solid red, right), which is much closer to the true resistance profile, q_c^{true}, based on the known experimental soil layering (solid blue). The measured $\tilde{q}_c^{meas}$ (dashed blue) deviate noticeably from the $\tilde{q}_c^{sim}$ calculated using the initial resistance profile guess (dashed red, left), but $\tilde{q}_c^{sim}$ is extremely close to the computationally simulated $\tilde{q}_c^{sim}$ of the final optimized resistance profile q_c^{inv}. All computations were done in serial on an Intel i7 8th generation quad core processor with 8GB DRAM in MATLAB 2020a. The algorithm has a fairly good initial guess for $q_c$following

the procedure in Section 3, but the final best profile, q_c^{inv}, matches q_c^{true} extremely well. The only discrepancy is at the very top of the profile where the q_c^{true} profile shows a gradient rather than a constant value.

Although the new algorithm is only designed to fit piece-wise constant q_c profiles, it is still able to fit smooth transitions by approximating them with a stair step-like pattern, the granularity of which depends on the AOI and LOO parameters passed to the algorithm. Figure 8, which was calculated by the new algorithm with the standard misfit, shows an example of this. The default parameters result in an initial q_c profile guess with several unnecessary layers at the top of the profile (solid red, left), which were then removed in the final q_c^{inv} (solid red, right). This required changing the LOO procedure to include so-called "absolute thresholding." This means that, rather than just removing any layer that does not decrease the misfit much relative to the current misfit value, the LOO algorithm also removes any layer that only contributes to the misfit function in a very small region of the profile. Note that this increased the misfit score from 0.003560 to 0.014702, which can be interpreted to mean that the final q_c^{inv} profile has a larger misfit by a roughly a factor of four compared to the initial q_c guess, although the final profile is more physically realistic.

The algorithm performed very well on most of the suite of calibration chamber soil models, but in models with a cluster of very thin layers, the algorithm skips fitting the last several layers. Figure 9 provides a clear example (see also A.13 and A.14). Even though the difference between the output profile from the new algorithm with the standard misfit, q_c^{inv}, and q_c^{true} is large in these cases, the difference between $\tilde{q}_c^{meas}$ and $\tilde{q}_c^{sim}(q_c^{inv})$ (indicated by dashed lines) is quite small. The small misfit in this region is due to the choice of the blurring function, as we can see in the bottom layers in Figure 1. Considering this limitation, the algorithm performs rather well. Although this result might be improved by including more layers in this region in the initial q_c guess, it is unlikely that all of the layers will be recovered. A blurring function that is more in accord with the physical process might prevent missed layers in regions with many very thin layers, but this is not guaranteed. Another approach is to use the new algorithm with the log misfit function, which emphasizes small differences more than the standard misfit function when close to the global minimum. More results comparing the new algorithms with the standard and log misfit are presented in Section 5.

To alert software users when it appears there may be a similar scenario

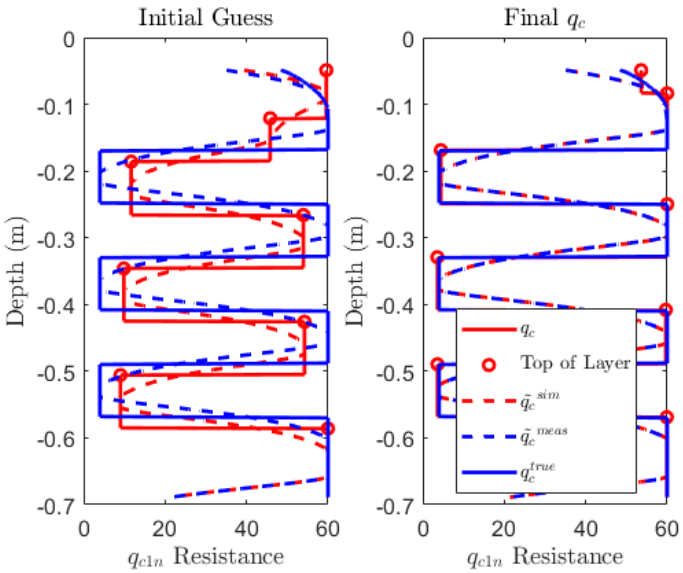

Figure 7: A known CPT tip resistance profile (solid blue), Soil Model 4 CPT 2 from the de Lange (2018) report, section C, compared with a measured q_c profile (dashed blue), an initial q_c guess (left, solid red), and the final q_c^{inv} after coordinate descent, LOO, AOI and particle swarm (right, solid red) show the performance of the method after the code ran for 30 seconds.

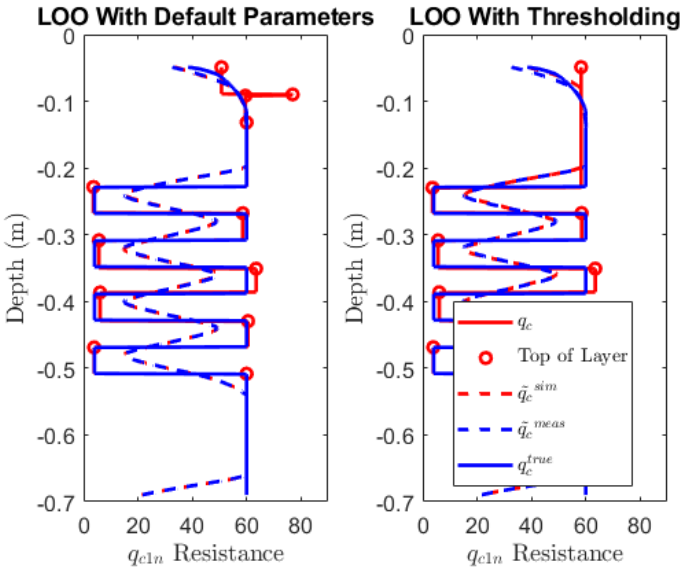

Figure 8: Soil Model 2 CPT 2 from the de Lange (2018) report, Section A. The new algorithm with run with default parameters yielded q_c^{inv} (left, solid red) that differs from q_c^{inv} resulting from the new algorithm run with a LOO parameter set to remove layers more aggressively (right, solid red). This example took 80 seconds to run.

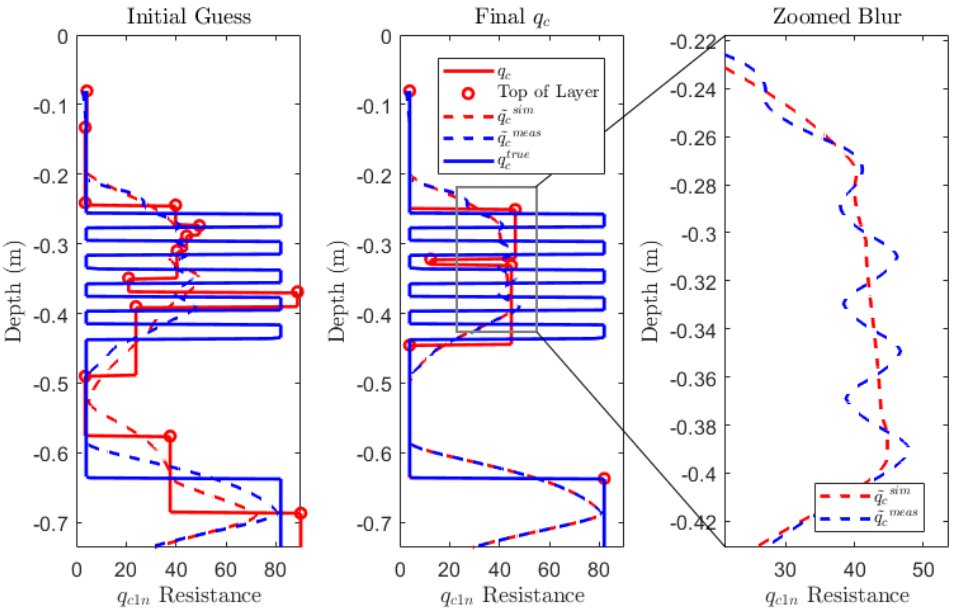

Figure 9: Soil Model 9 CPT 3 from de Lange (2018), Section A. The automatically generated initial q_c profile guess (left, red solid) and the final updated q_c^{inv} profile (center, red solid) both miss several layers that are in q_c^{true} (blue solid). A zoom in of this region (right) shows small-scale differences between $\tilde{q}_c^{meas}$ and $\tilde{q}_c^{sim}(q_c^{inv})$. This process took 40 seconds to run.

with many thin layers averaged together and missed, the provided software implementation of the new algorithm includes automatically generated warnings indicating where the algorithm is potentially leaving out a distinct soil layer. This is done by dividing up the entire depth profile into regions according to where $\tilde{q}_c^{sim}(q_c^{inv})$ overestimates or underestimates $\tilde{q}_c^{meas}$, and looking at the ratio of the misfit function to the signed difference. In practice, this seems to work very well even on the example shown in Figure 9. So, it could be incorporated into future improvements of the AOI method.

5. Results of the New Algorithm with Log Misfit

We found that the new algorithm with the log misfit function posed in Equation 4 was better suited to accurately refine small details of thinly layered profiles than the new algorithm with the standard misfit. Figure 10

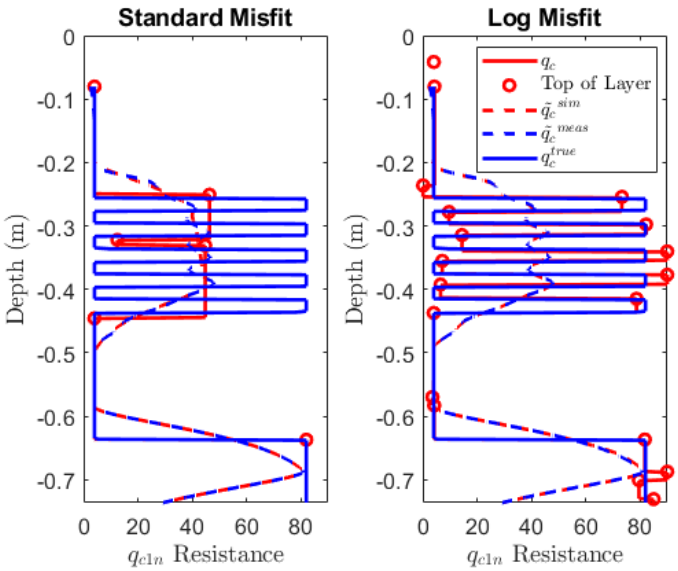

Figure 10: Soil Model 9 CPT 3 from de Lange (2018), Section A. The q_c^{inv} solution using the standard misfit (solid red, left) is much less accurate than the solution using the log misfit (solid red, right). The standard misfit algorithm took 40 seconds to run, while the log misfit algorithm took approximately 7 minutes.

shows the result of this log misfit function applied to the most difficult of our previous examples. Note how the q_c^{inv} resulting from the standard misfit (left, solid red) misses many layers in the true resistance profile (solid blue), while the q_c^{inv} resulting from the log misfit (right, solid red) detects every single thin layer. For both the standard misfit and the log misfit $\tilde{q}_c^{sim}(q_c)$ was extremely close to $\tilde{q}_c^{meas}$, showing that the detailed refinement done by the new algorithm with the log misfit was necessary to truly match the thin soil profile.

Note that minor modifications to LOO and AOI were implemented to accommodate the misfit function taking on negative logarithmic values. We found it necessary to limit the number of layers that could be added with each use of AOI to a small number (three worked well) since marginal improvements to $\tilde{q}_c^{sim}(q_c)$ can significantly impact the log misfit function. This means that, when close to the global minimum, it becomes increasingly difficult to add new layers as the algorithm progresses. The new algorithm with the log misfit takes between 5 and 10 minutes to run depending on how complex the actual resistance profile is, as well as how many layers are added

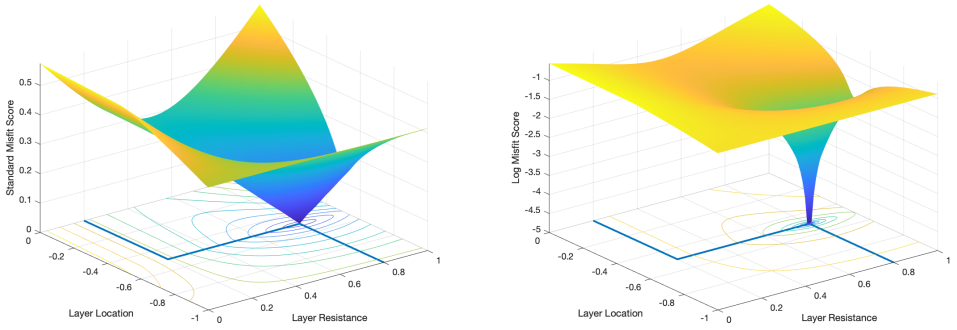

Figure 11: The standard misfit (left) and log misfit (right) are shown for all possible two-layer profiles with the q_c^{true} value of the top layer known.

each iteration. While it is more computationally expensive than the standard misfit, the log misfit is also more accurate and robust.

Why the log misfit outperforms the accuracy of the standard misfit can be understood as follows. Since the minimum of the standard misfit function is unique and equals zero, application of a log transform preserves the global minimizer (i.e. the best possible q_c^{inv}), which becomes the only location where the log misfit approaches negative infinity. This guarantees that as we approach the q_c^{true} profile, PSO should not terminate due to insufficient decrease in the objective function value (i.e. the new algorithm with the log misfit continues to refine q_c). This also has the effect of flattening out any local minima that are far from the global minimum, making it even easier for PSO to ignore those shallow minima and gravitate towards the global minimum. In Figure 11, we compare the contour plots of both misfit functions for a simple two-layer profile problem. Note that the original misfit function has very elongated contours around the minimum. Like the well-known Rosenbrock function, this elongated feature can cause numerical optimization algorithms to approach the minimum very slowly. Under the log transform, the region with elongated contours only has shallow decreases in the misfit function with more circular contours closer to the minimum.

6. Extensions of Method

Additional improvements in the computational efficiency of this method could be achieved through the use of parallel computing and GPU comput-

ing. While parallel computing in science and engineering has historically focused on massive problems running on large computer clusters, multi-core architectures and rapidly improving graphics cards are now widely available in laptops and desktops accessible to most engineers.

Further improvement in the accuracy of the method could be achieved through a more realistic representation of the measurement blurring process (i.e., the influence of thin-layer effects on $\tilde{q}_c^{meas}$). For our examples, we use a simple convolution with a smooth, asymmetric pointspread function (that is, $\tilde{q}_c^{meas}$ at a given depth is affected more by q_c^{true} below the depth than q_c^{true} above the depth). However, calibration chamber test data and high-fidelity material point method numerical simulations of soil displacement during penetrometer testing (Zambrano-Cruzatty and Yerro, 2020; Yost et al., 2021b) reveal much more complex physics, suggesting that a simple convolution with a single pointspread function is inadequate in some scenarios.

Moving forward, we aim to develop a method based on high-fidelity simulations to yield a more physically realistic computational model of this blurring process that is computationally fast to apply. For example, a neural network trained to mimic the numerical blurring of any q_c^{true} profile will also be computationally cheap to evaluate. Speed is an important feature in the layer optimization algorithm, since we expect to calculate $\tilde{q}_c^{sim}$ many times within each iteration. The primary drawback is the large amount of training data required, which is experimentally challenging to acquire and computationally taxing to generate via high-fidelity simulation.

7. Discussion and Conclusions

Thin-layer correction for CPT data can be posed as an inverse problem, similar to other signal deblurring problems. Our tests indicate that solving for a q_c profile mathematically represented by an independent tip resistance value at every depth (as done in Boulanger and DeJong (2018)) does not reliably yield improved data quality (Yost et al., 2021a), even when one adds regularization to enhance blocky layers of stratigraphy (a common strategy in image deblurring). We pose this inverse problem in a new way, searching for a finite number of subsurface layers, each having a thickness and uniform resistance that must be found such that $\tilde{q}_c^{sim}$ most closely matches $\tilde{q}_c^{meas}$. Our tests indicate this new formulation of the inverse problem is better able to identify thin interbedded layers in the soil profile compared to previous methods tested in Yost et al. (2021a).

We developed open-source software that takes as inputs a measured CPT tip resistance profile ($\tilde{q}_c^{meas}$) together with code to mimic the natural "blurring" of the true q_c profile due to the stress bulb that forms around the cone penetrometer tip. It estimates a piecewise constant q_c profile that is expected to result in data similar to the measured data, up to a desired tolerance. With appropriate settings, this method will correct for thin-layer effects during CPT soundings, although there may be other sources of measurement error that remain uncorrected. Our tests indicate that typical profiles characterized by a moderate number of depths at which q_c^{meas} is recorded (i.e., with several hundred values) can be corrected using the new algorithm with the standard misfit within 1-2 minutes on a standard laptop with one core.

This software has limitations. In many cases, the resulting q_c^{inv} profile will be simpler and smoother than what we might expect from the true resistance profile if we were to have other soil profile characterization data (e.g. core samples). However this tendency towards simplified models can easily be adjusted by the user's settings. The existence of a fast computational procedure that closely mimics this "natural blurring" of the true soil resistance profile remains an open question, so we use a simple convolution (5) to compute the "blurred" data, $\tilde{q}_c^{sim}$. Future research will focus on quantifying uncertainty in the corrected q_c^{inv} profiles, and improving the blurring functions that are used to represent a wider range of geotechnical scenarios.

For particularly complex soil stratigraphies, we suggest users take advantage of the new algorithm with the log misfit, which balances regions with large-scale and fine-scale features contributing to the misfit. Our tests reveal this yields more accurate and robust profiles, better reflecting subsurface stratigraphy, with more thin layers correctly identified. Our open-source software includes both the standard and log misfits. Note that the new algorithm with the log misfit requires a longer run time, typically 5-10 minutes for CPT soundings with several hundred depth points. By providing software for both formulations, users can decide which version to apply based on tradeoffs in computing time, accuracy, and assumed complexity of the soil stratigraphy.

8. Data Statement

Multiple examples in this paper were performed using data that is available through a technical report from Deltares (de Lange, 2018). Code for

every algorithm and example in this paper are publicly available at `https://github.com/jonc7/Soil-Layer-Optimization`.

9. Acknowledgements

This research was supported by the National Science Foundation (NSF) Grants CMMI-1825189 and CMMI-1937984. This support is gratefully acknowledged. We also thank Virginia Tech's Advanced Research Computing for computational resources. However, any opinions, findings, and conclusions or recommendations expressed in this material are those of the authors and do not necessarily reflect the views of the NSF.

Appendix A. Additional Soil Model Information

In the de Lange (2018) study, several different soil models consisting of layered sand-clay profiles were constructed and CPTs were performed in these models at various in-situ stress conditions. Sand layers in the soil models were prepared with target relative densities of either 30% (loose) or 60% (dense). Uniform or "reference" sand models were also prepared with the same target relative densities. CPTs performed in the reference sand models therefore provided a good estimate of the "true" resistance profile q_c for the sand layers in the corresponding layered soil models. Note that due to variation in experimental preparation of the soil models, the sand relative densities in the layered models did not always match well with that of the reference sand model. For the purposes of this study, only the layered soil models that were relatively good matches were considered [this included several layered models presented in the "Test Results" section (Section A) and Soil Model 4 from the "Start-Up Phase" section (Section C) of the de Lange (2018) report]. Furthermore, no uniform/reference clay soil models were considered by de Lange (2018), therefore, the "true" q_c in the clay was estimated based on the minimum q_c observed in the thickest of the clay layers in the de Lange (2018) experiments. With the estimated "true" q_c for both the clay and sand layers, and knowledge of the layer depths and thicknesses, q_c^{true} profiles for the layered soil models were constructed for comparison to the measured CPT tip resistance profiles, $\tilde{q}_c^{meas}$, and the corrected tip resistance profiles calculated by the new algorithms, q_c^{inv}.

Figures A.12, A.13 and A.14 show results of the simple procedure to pick an initial profile guess, the final q_c^{inv} for the standard objective function,

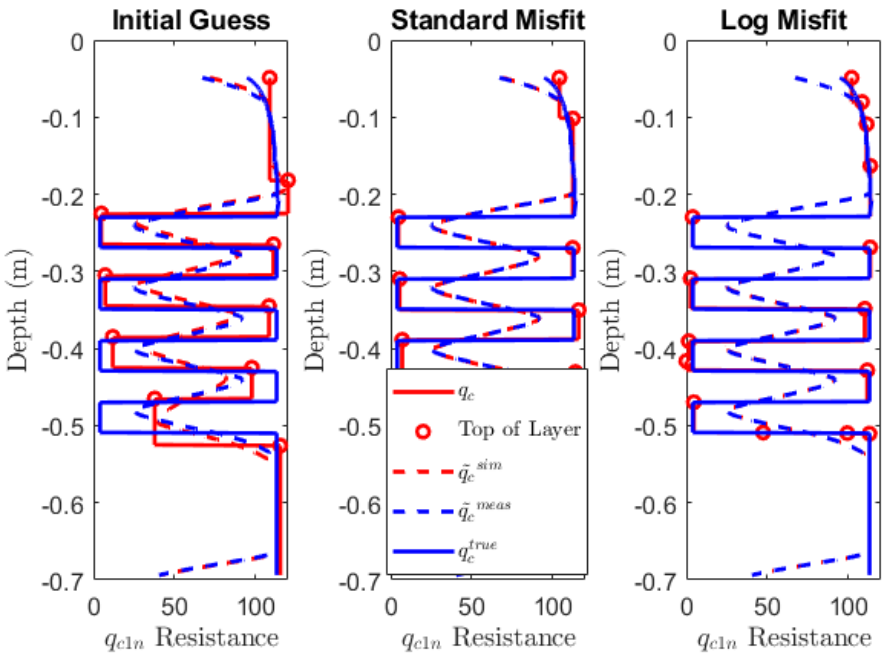

Figure A.12: Soil Model 4 CPT 3 from the de Lange (2018) report, Section A. The automatically generated initial q_c profile (left, solid red) is compared to the final q_c^{inv} profiles resulting from the new algorithm with standard misfit (center, solid red) and the log misfit (right, solid red). The algorithm with the standard and log misfits took 50 seconds and over 7 minutes, respectively, to run.

and the final q_c^{inv} for the log objective function for three laboratory datasets from the de Lange (2018) report.

References

Ahmadi, M., Robertson, P., 2005. Thin-layer effects on the cpt qc measurement. Canadian Geotechnical Journal 42, 1302–1317.

Boulanger, R., DeJong, J., 2018. Inverse filtering procedure to correct cone penetration data for thin-layer and transition effects, in: Cone Penetration Testing 2018: Proc. of the 4th Int. Symposium on Cone Penetration Testing. CRC Press, Delft, The Netherlands, 21-22 Jun., pp. 25–44.

Boulanger, R.W., Idriss, I., 2016. Cpt-based liquefaction triggering proce-

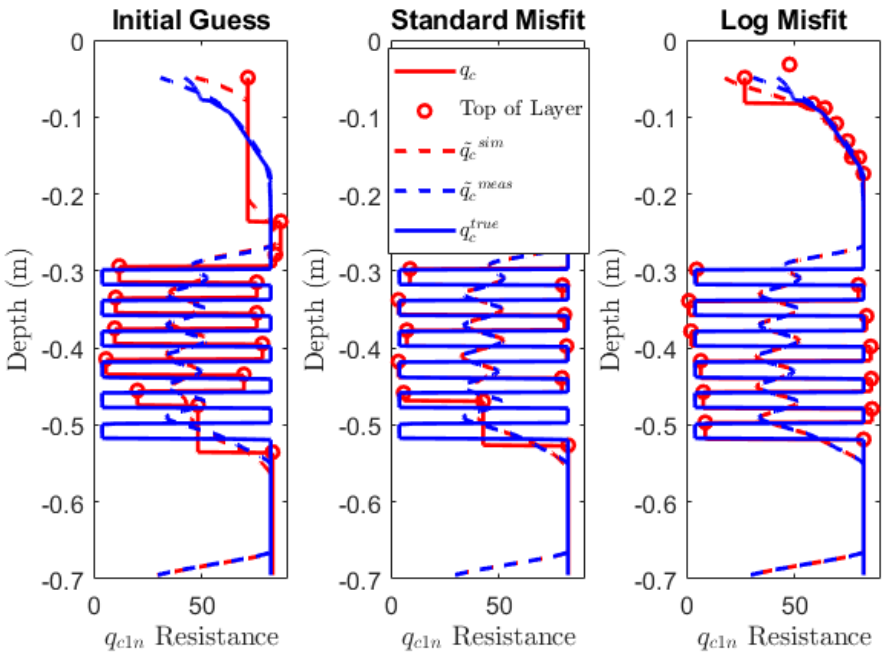

Figure A.13: Soil Model 3 CPT 3 from the de Lange (2018) report, Section A. The automatically generated initial profile (left, solid red), the final q_c^{inv} profiles from the new algorithm with the standard misfit (center, solid red) and the log misfit (right, solid red) are shown. The new algorithm with the standard misfit took 100 seconds, and 10 minutes with the log misfit.

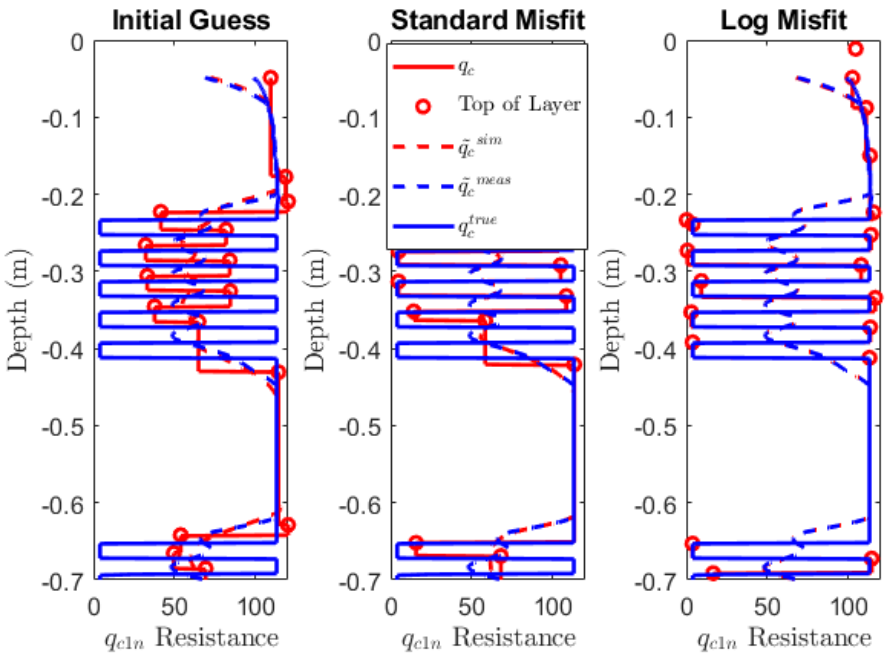

Figure A.14: Soil Model 8 CPT 3 from the de Lange (2018) report, Section A. The automatically generated initial q_c profile (left, solid red), the final q_c^{inv} profiles generated by the new algorithm with the standard misfit (center, solid red) and log misfit (right, solid red) are shown. The algorithm with the standard misfit took 70 seconds, and the log misfit took over 11 minutes.

dure. Journal of Geotechnical and Geoenvironmental Engineering 142, 04015065.

El Korthawi, M., Green, R., Wotherspoon, L., van Ballegony, S., 2019. Insights into the liquefaction hazard in Napier and Hastings based on the assessment of data from the 1931 Hawke's Bay, New Zealand, earthquake, in: Proc. 13th Australia New Zealand Conference on Geomechanics (13ANZCG). Perth, Australia, 1-3 Apr.

de Greef, J., Lengkeek, H., 2018. Transition- and thin layer corrections for cpt based liquefaction analysis, in: Cone Penetration Testing 2018: Proc. of the 4th Int. Symposium on Cone Penetration Testing. CRC Press, Delft, The Netherlands, 21-22 Jun., pp. 317–322.

de Lange, D., 2018. CPT in thinly layered soils: Validation tests and analysis for multi thin layer correction, in: van Elk, J., Doornhof, D. (Eds.), Technical Report. Deltares, Delft, Netherlands.

Maurer, B., Green, R., Cubrinovski, M., Bradley, B., 2014. Evaluation of the liquefaction potential index for assessing liquefaction hazard in Christchurch, New Zealand. J. Geotech. Geoenviron. Eng. 140, 04014032–1 – 04014032–11.

Maurer, B., Green, R., Cubrinovski, M., Bradley, B., 2015. Fines-content effects on liquefaction hazard evaluation for infrastructure in Christchurch, New Zealand. Soil Dynamics and Earthquake Engineering 76, 58–68.

Schmertmann, J., 1978. Guidelines for cone penetration test: performance and design. Technical Report. Federal Highway Administration. United Statees.

Shibata, T., Teparaksa, W., 1988. Evaluation of liquefaction potentials of soils using cone penetration tests. Soils and Foundations 28, 49–60.

Socco, L., Strobbia, C., 2004. Surface-wave method for near-surface characterization: a tutorial. Near Surface Geophysics 2, 165–185.

Wright, S., 2015. Coordinate descent algorithms. Mathematical Programming 151, 3–34.

Yost, K., Green, R., Upadhyaya, S., Maurer, B., Yerro, A., Martin, E., Cooper, J., 2021a. Assessment of the efficacies of correction procedures for multiple thin layer effects on cone penetration tests. Soil Dynamics and Earthquake Engineering 144, 106677.

Yost, K., Yerro, A., Green, R.A., Martin, E., Cooper, J., 2021b. Mpm modeling of cone penetrometer testing for multiple thin-layer effects in complex soil stratigraphy. Journal of Geotechnical and Geoenvironmental Engineering Under review, –.

Youd, T., Idriss, I., Andrus, R., Arango, I., Castro, G., Christian, J., Dobry, R., Finn, W., Harder, L., Hynes, M., Ishihara, K., Koester, J., Liao, S., Marcusson, W., Martin, G., Mitchell, J., Moriwaki, Y., Power, M., Robertson, P., Seed, R., Stokoe, K., 2001. Liquefaction resistance of soils: Summary report from the 1996 NCEER and 1998 NCEER/NSF workshops on evaluation of liquefaction resistance of soils. Journal of Geotechnical and Geoenvironmental Engineering 127, 817–833.

Zambrano-Cruzatty, L., Yerro, A., 2020. Numerical simulation of a free fall penetrometer deployment using the material point method. Soils and Foundations 60, 668–682.