

A BAYESIAN APPROACH FOR GAS CONCENTRATION RECONSTRUCTION BASED ON TUNABLE DIODE LASER ABSORPTION SPECTROSCOPY

Chao Yuan

Siemens Corporate Research, 755 College Road East, Princeton, NJ, 08540

ABSTRACT

We address the problem of 2D reconstruction of the gas concentration map from its 1D path averages based on Tunable Diode Laser Absorption Spectroscopy (TDLAS). This problem is challenging because typically only a very small number of paths are available. In addition, a path may not be set up to our desirable configuration. These challenges result in a large percentage of unobserved pixels in the map that do not have a single crossing path. We propose a Bayesian approach for this reconstruction problem by modeling the 2D map as a Gaussian process. The correlation among pixels is used to propagate information from observed pixels to unobserved pixels.

Index Terms— Gaussian process, TDLAS

1. INTRODUCTION

Tunable diode laser absorption spectroscopy (TDLAS) is a proven technique for measuring gas concentrations (e.g., of O_2 , CO) and temperatures simultaneously in combustion systems such as a boiler [1]. A single TDLAS setup consists of a laser transmitter (the dark-colored box on the left side of Figure 1), sending laser beams over a path through the combustion region (the grids or pixels in the middle of Figure 1), to a laser receiver (the light-colored box on the right side of Figure 1). Each TDLAS path measures an average value of the gas concentrations along the path. One active research area in TDLAS is to reconstruct the 2D gas concentration map based on multiple path averages. The reconstructed map is useful in many applications such as combustion monitoring, diagnosis and optimization.

This reconstruction task is essentially a problem of 2D reconstruction from 1D projections, which is similar to the concept of computed tomography (CT) (Chapter 3 in [2]). However, most widely used CT algorithms such as filtered back-projection require a lot of projections (multiple views and dense projections per view) to achieve a good resolution. In contrast, only a very small number of paths are typically set up on a boiler; for example, it is not uncommon that we only have five to ten paths (projections). In addition, a path may not be arranged at a location or a direction (view) that we

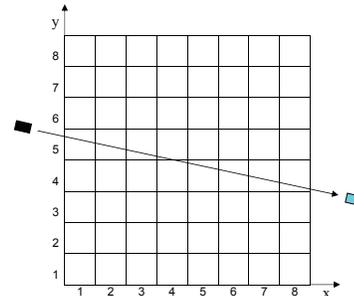


Fig. 1. Illustration of gas concentration reconstruction

desire because of certain restrictions or mounting difficulties of a boiler.

Algebraic reconstruction technique (ART) seems to be a natural fit here because it can handle the issue of a small number of projections (Chapter 7 in [2]). However, for this extremely under-constrained problem (the number of unknown variables, gas concentrations in a 2D map, is far more than the number of available equations), ART's solution may not be a realistic one. Smoothness constraints among neighboring pixels are introduced as prior information to help addressing this under-constrained problem. Smoothness can be incorporated via smooth basis functions [3, 4, 5] or by bi-cubic spline interpolation as a post-processing step [5]. However, such smoothness constraints are local and cannot capture long-range correlation among pixels. Moreover, [3, 4] also apply 1D smooth basis functions to *multiple* path averages under the *same* view. This requirement is difficult to meet due to the challenges noted above.

In this paper, we propose a Bayesian approach for gas concentration reconstruction. The 2D gas concentration is assumed to be a Gaussian process (GP), a special type of multivariate Gaussian distribution. GP models the global correlation among all pixels in the 2D map vs. the local correlation modeled by prior work [3, 4, 5]. Therefore, information from path averages can be propagated to all pixels instead of just local pixels. This allows us to reconstruct the whole map with as few as one path and with arbitrary path configurations. The rest of this paper is organized as follows. In Section 2, we formally define this reconstruction problem. In Section 3, the proposed algorithm is described. We present simulation results in Section 4 and conclude this paper in Section 5.

2. PROBLEM DEFINITION

Our goal is to reconstruct an $M \times N$ gas concentration map \mathbf{v} from I path averages (projections), denoted by an I -dimensional vector \mathbf{b} . In the example shown in Figure 1, we have $M = N = 8$ and we want to estimate the gas concentration at each of the 64 grid pixels. Note that M doesn't need to be equal to N . Typically, we represent the map \mathbf{v} as a J -dimensional vector for easy algebraic manipulation, where $J = M \times N$. There are two ways to denote a pixel. One is to use its coordinates (x, y) in the 2D map; the other is to use its corresponding index j in \mathbf{v} .

Figure 1 shows one of the I paths. The i -th projection b_i represents the average gas concentration along the i -th path, where $i = 1, 2, \dots, I$. The projections \mathbf{b} and map \mathbf{v} have the following relation

$$\mathbf{b} = \mathbf{A}\mathbf{v}. \quad (1)$$

\mathbf{A} is an $I \times J$ projection matrix. Once the paths are configured, \mathbf{A} becomes a constant matrix. The sum of the i -th row and the sum of the j -th column of \mathbf{A} are respectively defined by

$$A_{i,+} = \sum_{j=1}^J A_{i,j} \quad \text{and} \quad A_{+,j} = \sum_{i=1}^I A_{i,j}. \quad (2)$$

We now describe how to determine a projection weight $A_{i,j}$ in the projection matrix \mathbf{A} . The basic idea is that the contribution of a pixel to the projection is proportional to the length of intersection between the path and the pixel area. In Figure 1, the path intersects with both pixel (1, 5) and (4, 5). However, because the intersection length of pixel (1, 5) is longer than that of pixel (4, 5). Therefore, pixel (1, 5) has a larger weight than pixel (4, 5). In this example, only 9 out of 64 pixels have non-zero weights in the corresponding row of \mathbf{A} . We normalize every row to make the projection a path average such that

$$A_{i,+} = 1. \quad (3)$$

A j -th pixel is termed as *observed* if there is at least one path crossing it or $A_{+,j} > 0$. Otherwise, it is *unobserved*.

Our reconstruction problem now becomes solving a linear equation (1) with both \mathbf{A} and \mathbf{b} known. If \mathbf{A} is an invertible square matrix, where $I = J$, then we have $\mathbf{v} = \mathbf{A}^{-1}\mathbf{b}$. If $I > J$ and \mathbf{A} has full rank, based on linear least square we have $\mathbf{v} = (\mathbf{A}^T\mathbf{A})^{-1}\mathbf{A}^T\mathbf{b}$. However, our problem is very under-constrained because of $I \ll J$. In such a case, we have multiple solutions satisfying the same equation (1).

Simultaneous algebraic reconstruction technique (SART) is one of the most widely used ART algorithms [2]. It has the following iterative procedure

$$v_j^{(k+1)} = v_j^{(k)} + \frac{\lambda}{A_{+,j}} \sum_{i=1}^I A_{i,j} \frac{b_i - (\mathbf{A}\mathbf{v}^{(k)})_i}{A_{i,+}}. \quad (4)$$

$v_j^{(k)}$ denotes the current reconstructed map after the k -th iteration. $(\mathbf{A}\mathbf{v}^{(k)})_i$ indicates the estimated i -th projection from

the current map. Note that in our problem $A_{i,+} = 1$ and can be omitted from (4), but in general it has a non-zero value. (4) can be understood as follows. The residual between the true projection b_i and the estimated projection $(\mathbf{A}\mathbf{v}^{(k)})_i$ indicates in which direction $v_j^{(k+1)}$ should move from $v_j^{(k)}$ to reduce this residual. The final direction is determined by the weighted average of all I projections, each with a weight of $A_{i,j}$ normalized by $A_{+,j}$. The magnitude of this movement is adjusted by a parameter λ .

The SART algorithm is proven to converge to a solution of (1) if $A_{+,j} \neq 0$ [6]. However, there is no guarantee whether the found solution, out of all possible solutions, is realistic or not. In addition, in our application, the path configuration can be arbitrary. It often happens that $A_{+,j} = 0$ for some columns so the j -th pixel is unobserved. In such cases, division-by-zero problem will happen and (4) will stop working. We will compare our algorithm with SART in our tests.

3. ALGORITHM DESCRIPTION

We model \mathbf{v} as a multivariate Gaussian distribution

$$P(\mathbf{v}) = \mathcal{N}(\mathbf{v}|\mathbf{m}, \mathbf{C}), \quad (5)$$

where \mathbf{m} and \mathbf{C} are the J -dimensional mean vector and $J \times J$ covariance matrix, respectively. Because \mathbf{v} represents a 2D map, its distribution is also referred to as a Gaussian process (GP). We assume that this GP is homogeneous since we don't have a priori knowledge about the differences between different pixels.

The cross-covariance C_{j_1, j_2} between v_{j_1} and v_{j_2} in \mathbf{v} , where $1 \leq j_1, j_2 \leq J$, is defined as

$$C_{j_1, j_2} = f \exp\left(-\frac{(x_{j_1} - x_{j_2})^2 + (y_{j_1} - y_{j_2})^2}{r^2}\right). \quad (6)$$

(x_{j_1}, y_{j_1}) and (x_{j_2}, y_{j_2}) are the coordinates for pixels j_1 and j_2 , respectively. f and r are two parameters. (6) indicates that if pixels j_1 and j_2 are close to each other, their distance is small and thus the corresponding v_{j_1} and v_{j_2} will have high correlation. On the other hand, far-apart v_{j_1} and v_{j_2} will be less correlated. If $j_1 = j_2$, $C_{j_1, j_2} = f$ which is the variance of v_{j_1} . Similar GP models have been successfully applied in image processing. For example, under this assumption, the original high-resolution image can be recovered from one or more low-resolution images even if a lot of information is missing [7].

To make the projection process probabilistic, we introduce a projection noise model as an extension from (1)

$$\mathbf{b} = \mathbf{A}\mathbf{v} + \mathbf{e}, \quad (7)$$

where \mathbf{e} is the projection noise vector and has a Gaussian distribution $P(\mathbf{e}) = \mathcal{N}(\mathbf{e}|\mathbf{0}, \sigma^2\mathbf{I})$. σ^2 is the noise variance and \mathbf{I} is a $I \times I$ identity matrix. The projections \mathbf{b} can thus be

modeled as another multivariate Gaussian distribution given the 2D map \mathbf{v}

$$P(\mathbf{b}|\mathbf{v}) = \mathcal{N}(\mathbf{b}|\mathbf{A}\mathbf{m}, \sigma^2\mathbf{I}). \quad (8)$$

We can now write the joint probability $P(\mathbf{v}, \mathbf{b}) = P(\mathbf{v})P(\mathbf{b}|\mathbf{v})$

$$P(\mathbf{v}, \mathbf{b}) = \mathcal{N}\left(\begin{bmatrix} \mathbf{v} \\ \mathbf{b} \end{bmatrix} \mid \begin{bmatrix} \mathbf{m} \\ \mathbf{A}\mathbf{m} \end{bmatrix}, \begin{bmatrix} \mathbf{C} & \mathbf{C}\mathbf{A}^T \\ \mathbf{A}\mathbf{C} & \mathbf{A}\mathbf{C}\mathbf{A}^T + \sigma^2\mathbf{I} \end{bmatrix}\right). \quad (9)$$

Our task is to infer map \mathbf{v} from projections \mathbf{b} , or computing the conditional probability of $P(\mathbf{v}|\mathbf{b})$, which is another multivariate Gaussian distribution and can be easily derived from (9). Finally the mean of $P(\mathbf{v}|\mathbf{b})$ is output as the reconstructed 2D map

$$\tilde{\mathbf{v}} = \mathbf{C}\mathbf{A}^T(\mathbf{A}\mathbf{C}\mathbf{A}^T + \sigma^2\mathbf{I})^{-1}(\mathbf{b} - \mathbf{A}\mathbf{m}) + \mathbf{m}. \quad (10)$$

Parameter settings. We fix $f = 1$ and $\sigma^2 = 0.001$ such that the variance ratio between signal and noise is a large enough $f/\sigma^2 = 1000$. This ensures that we can faithfully reconstruct the map without much distortion due to noise. At the same time, $\sigma^2 = 0.001$ is also large enough to avoid a singular matrix before the matrix inverse in (10). One may ask what if the actual signal variance is much larger than 1. In that case, we can multiply both $f = 1$ and σ^2 by the same factor. This will not change the result of (10). Once we collect enough path projections, we calculate the average \bar{b} from all of them. We then set every element $m_j = \bar{b}$ in the mean vector \mathbf{m} , where $j = 1, 2, \dots, J$, due to our homogeneous assumption of the GP model. In practice, we find that when the performance of our algorithm doesn't vary appreciably when r is between 0.5 and 1.5. Therefore, we fix $r = 1.0$. Alternatively, we can learn all above parameters by cross validation. However, this is not done in this work.

Complexity. We can rewrite (10) as a simpler linear form of $\tilde{\mathbf{v}} = \mathbf{G}\mathbf{b} + \mathbf{u}$. Both \mathbf{G} and \mathbf{u} are computed once off-line and then become constant during on-line testing. Even so, calculation of \mathbf{G} and \mathbf{u} can be a problem. First, the covariance matrix \mathbf{C} has a size of $J \times J$. Recall that $J = M \times N$. Let's assume that $M = N$ from now on and $J \gg I$. Calculation and storage of \mathbf{C} take a complexity of $\mathcal{O}(N^4)$ if we want to compute each element of \mathbf{C} only once. Second, after we have \mathbf{C} , computing \mathbf{G} and \mathbf{u} has a complexity of $\mathcal{O}(J^2I)$ or $\mathcal{O}(N^4I)$. For a medium-size map where $M = N = 200$, this can pose a serious computational burden.

We adopt the following strategies to mitigate these issues. First, note that each element C_{j_1, j_2} in \mathbf{C} is a function of $|x_{j_1} - x_{j_2}|$ and $|y_{j_1} - y_{j_2}|$ as in (6). There are only N different values for all possible $|x_{j_1} - x_{j_2}|$ or $|y_{j_1} - y_{j_2}|$. Therefore, instead of indexing each element in \mathbf{C} using j_1 and j_2 , we index them using $|x_{j_1} - x_{j_2}|$ and $|y_{j_1} - y_{j_2}|$. We thus bring down the handling cost for \mathbf{C} from $\mathcal{O}(N^4)$ to $\mathcal{O}(N^2)$. Also note that the projection matrix \mathbf{A} is a sparse matrix: each row has only $\mathcal{O}(N)$ non-zero entries. Using this fact, we can bring down the computational complexity for \mathbf{G} and \mathbf{u} from $\mathcal{O}(N^4I)$ to $\mathcal{O}(N^3I)$. Testing is fast because the complexity is just $\mathcal{O}(N^2I)$ once \mathbf{G} and \mathbf{u} are known.

4. SIMULATION RESULTS

In order to evaluate our algorithms, we need to compare the reconstructed map with the ground-truth map. However, it is difficult to directly measure the gas concentration inside the combustion region. Thus, we resort to simulated tests. In all our tests, the size of a map is $M = N = 200$. The coordinates of the map is set to be between 0 and 1. In other words, all pixel coordinates satisfy $0 \leq x_j, y_j \leq 1$ and each pixel has a dimension of 0.005×0.005 .

A ground-truth map is generated by summing L 2D smooth functions (similar to the covariance function in (6)). In particular, the j -th pixel value v_j is produced as follows

$$v_j = \sum_{l=1}^L h_l \exp\left(-\frac{(x_j - \mu_{xl})^2 + (y_j - \mu_{yl})^2}{s_l^2}\right). \quad (11)$$

(x_j, y_j) is the corresponding 2D coordinates of v_j . The l -th smooth function is defined by $h_l \exp\left(-\frac{(x - \mu_{xl})^2 + (y - \mu_{yl})^2}{s_l^2}\right)$, where $l = 1, 2, \dots, L$. There are four parameters for each function, h_l , s_l , μ_{xl} and μ_{yl} , representing the peak, width and center locations of the function, respectively.

We fix $L = 10$ for all our tests. To produce a 2D map, we randomly create each of the $L = 10$ smooth functions by randomly selecting h_l between 0 and 1, s_l between 0.1 and 0.4, μ_{xl} and μ_{yl} between 0 and 1, respectively. Then all the L functions are summed to form the final ground-truth map \mathbf{v} as in (11). The map created in this way is smooth, and usually multi-peaked and with an irregular shape, which resembles a realistic gas concentration map. We create a total of ten maps using this approach. One example is shown in Figure 2 (e and j).

Although our algorithm can handle any path setup and the locations of the transmitter and the receiver can be anywhere in the map, we configure the paths in a more realistic way. A path is defined by two end points (x_{e1}, y_{e1}) and (x_{e2}, y_{e2}) . We require that the end points must be on the boundary of the map because transmitter and receiver cannot be mounted inside the boiler. In addition, a path must either go from left to right or from bottom to top (Figure 1). This makes better use of a path than, for example making a path from left to bottom. However, the actual location of an end point on a side is random.

There are two steps to create a path. First, we randomly select which direction the path will go, from left-to-right or from bottom-to-top. Second, we randomly select the two end point locations between 0 and 1 on a selected side. We gradually increase the number of paths I from 1 until 500 for each map.

We run both our algorithm and SART [2] for each path configuration and for each of the ten test maps. We then compare the reconstructed map $\tilde{\mathbf{v}}$ with the ground-truth map \mathbf{v} using mean squared error $MSE = \sqrt{\frac{\sum_{j=1}^J (\tilde{v}_j - v_j)^2}{J}}$. The MSE

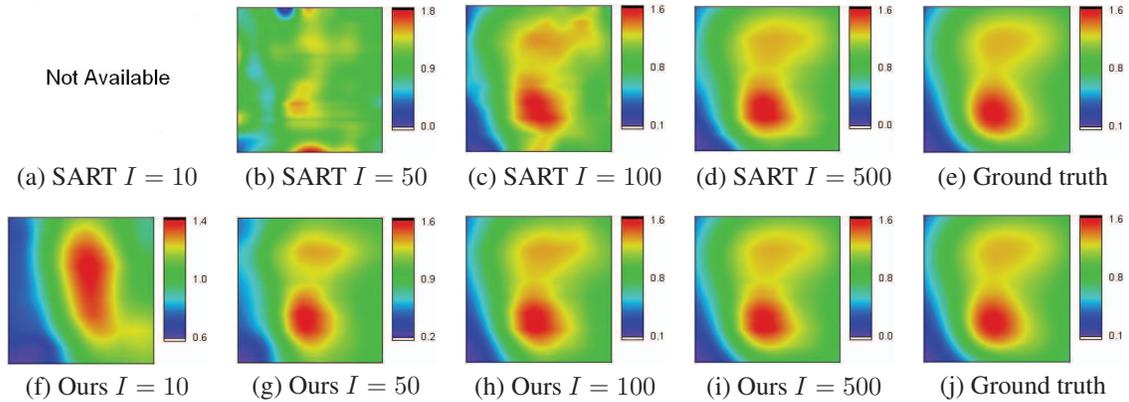


Fig. 2. Comparing the reconstructed results with the ground truth with different number of paths

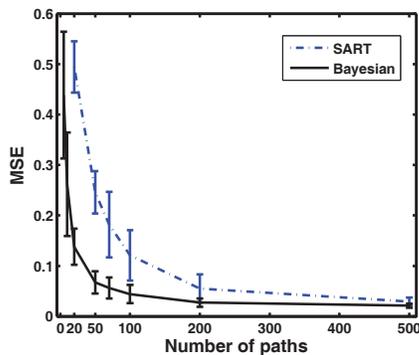


Fig. 3. MSE scores for SART and the proposed Bayesian algorithm when the number of paths varies from 5 to 500

scores are averaged from these ten results. Figure 3 shows the comparison results between our Bayesian algorithm and SART for number of paths $I = 5, 10, 20, 50, 70, 100, 200, 500$. One standard-deviation error bar is also shown. For SART, the parameter λ in (4) is set to 1.0, which produces the lowest overall MSE errors.

Our Bayesian algorithm outperforms SART by producing lower MSE errors for all path settings. Due to the division-by-zero error, SART is unable to produce reconstructed map when I is very small at 5 and 10. In contrast, our algorithm can produce results for all $I \geq 1$. Note that when $I = 0$, our algorithm just outputs \mathbf{m} , the mean of the Gaussian process (GP) because no extra information is available. Figure 2 also compares the reconstructed maps from both SART and our algorithm for one of the ten test cases. Our Bayesian algorithm is able to reconstruct the rough structure of the map when $I = 10$ (Figure 2 (f)). When $I = 50$, the reconstructed results already show a lot of details (Figure 2 (g)). However, SART can only produce reasonably good results when $I = 100$ (Figure 2 (c)). At $I = 500$, both algorithms produce results close to the ground truth. At such a large I , we have adequate information for reconstruction and thus it will be less useful to use our GP model.

5. SUMMARY

We present a Bayesian approach to tackle the problem of 2D reconstruction of the gas concentration map from its 1D path averages. The prior probability of the 2D map is modeled as a Gaussian process (GP). Using GP, the correlation among pixels is used to propagate information from observed pixels to unobserved pixels. Our approach is demonstrated to outperform previous algebraic reconstruction technique in our simulated tests.

6. REFERENCES

- [1] R. K. Hanson, P. A. Kuntz, and C. H. Kruger, “High-resolution spectroscopy of combustion gases using a tunable IR diode laser,” *Applied Optics*, vol. 16, pp. 2045–2048, August 1977.
- [2] A. C. Kak and M. Slaney, Eds., *Principles of Computerized Tomographic Imaging*, SIAM Press, 2001.
- [3] M. Ravichandran and F. C. Gouldin, “Reconstruction of smooth distributions from a limited number of projections,” *Applied Optics*, vol. 27, pp. 4084–4097, October 1988.
- [4] K. B. Chung, F. C. Gouldin, and G. J. Wolga, “Experimental reconstruction of the spatial density distribution of a nonreacting flow with a small number of absorption measurements,” *Applied Optics*, vol. 34, pp. 5492–5500, August 1995.
- [5] “Zoloboss tomography algorithm an overview,” *White papers*, Zolo Technologies, Inc.
- [6] M. Jiang and G. Wang, “Convergence of the simultaneous algebraic reconstruction technique (SART),” *IEEE Trans. on Image Processing*, vol. 12, pp. 957–961, August 2003.
- [7] M. E. Tipping and C. M. Bishop, “Bayesian image super-resolution,” in *Advances in Neural Information Processing Systems*, 2002.