DISTRIBUTED SPECTRUM ESTIMATION IN SENSOR NETWORKS

Omid Jahromi and Parham Aarabi

Artificial Perception Laboratory,
Department of Electrical and Computer Engineering, University of Toronto
Toronto, ON, Canada, M5S 3G4
Email: omidj@control.toronto.edu, parham@ecf.toronto.edu

ABSTRACT

This paper considers the problem of fusing the statistical information gained by a distributed network of sensors. We formulate the problem as a *convex feasibility problem*, that is, finding a point in the intersection of finitely many closed convex sets. We then present a distributed optimization algorithm to solve the problem.

1. INTRODUCTION

Widespread availability of cheap embedded processors combined with easily accessible wireless networks allow for a wide variety of devices and machines to be connected. We can now build a large-scale yet affordable information system using low-cost sensors interconnected through a wireless communication network. A distributed network of sensors can be highly scalable, cost effective, and robust with respect to individual node's failure. However, it creates new challenges in terms of how the information provided by the sensors in the network is to be processed.

The development of signal processing algorithms designed expressly for sensor network applications is an emerging interdisciplinary research area with deep connections to information theory, networking and distributed algorithms. In this paper, motivated by the developments mentioned above, we study the problem of estimating the power spectrum of a physical signal when the signal is observed by a network of distributed sensors. This problem has numerous potential applications in such areas as acoustical engineering, oceanography and geophysics.

Example 1 Consider the simple scenario where a sound source (a speaker) is listened to by a network of microphones put at various known locations in a room (Fig. 1). Because of reverberation and other artifacts, the signal arriving at each microphone is different. The microphones (which constitute the *sensor nodes* in our network) are equipped with sampling devices, sufficient signal processing hardware and some communication means. Each sensor node can process its observed data, come up with some statistical inference about it and share the result with other nodes in the network. However, to save energy and communication bandwidth, the network nodes cannot share their raw data (i.e. the local observed signals) with each other.

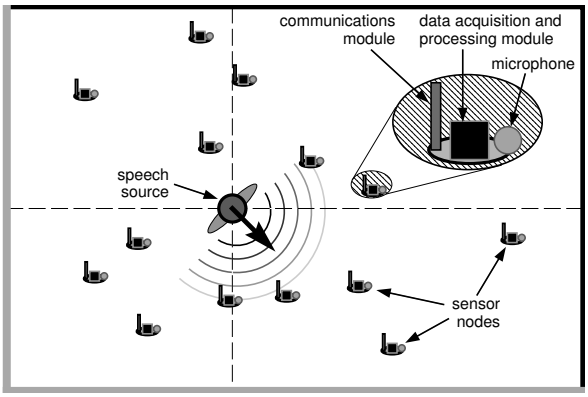

Figure 1: A simple sensor network setup

Now, how should the network operate so that we get an estimate of the power spectrum of the sound source reflecting the observations made by all the nodes? ♢

The problem posed in Example 1 is more challenging than classical spectrum estimation problems considered in the literature. This is because, here, the observed data must be processed in a distributed way at network nodes. In the next section, we will build a mathematical model for the problem posed in Example 1. Our formulation, nevertheless, is quite general and can be applied to other application areas as well.

Notation: Vectors will be denoted by capital letters. Script capital letters are used to denote sets. The spaces of Lebesgue-measurable functions are represented

by $\mathbf{L}^1(a,b)$, $\mathbf{L}^2(a,b)$, etc. The end of an example is indicated using the symbol $\diamondsuit$.

2. PROBLEM FORMULATION

Let $x(n)$ denote a discrete version of the signal produced by the source and assume that it is a zero-mean Gaussian wide-sense stationary (WSS) random process. The sampling frequency f_s associated with $x(n)$ is arbitrary and depends on the frequency resolution desired in the spectrum estimation process. It is well-known that a complete statistical description of a WSS process is provided by its autocorrelation sequence

$$R_x(k) \stackrel{\triangle}{=} E\{x(n)x(n+k)\}$$

or, equivalently, by its power spectrum

$$P_x(e^{j\omega}) = \sum_{k=-\infty}^{\infty} R_x(k)e^{-j\omega k}.$$

The autocorrelation sequence is a time-domain description of the second order statistics of a random process. The power spectrum provides a frequency domain description of the same statistics. The reader is referred to [1] for an overview of modern spectrum estimation techniques and their applications. We denote by $v_i(n)$

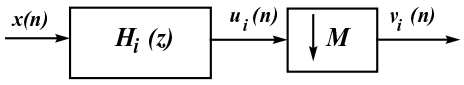

Figure 2: The relation between the signal $v_i(n)$ observed by the ith sensor and the original source signal $x(n)$.

the signal produced at the front end of the ith sensor node. We assume that $v_i(n)$ are related to the original source signal $x(n)$ by the model shown in Fig. 2. The linear filter $H_i(z)$ in this figure models the combined effect of room reverberations, the microphone's frequency response and an additional filter which the system designer might want to include. The decimator block which follows the filter represents the (potential) difference between the sampling frequency f_s assumed for the original signal $x(n)$ and the actual sampling frequency of the sensors' sampling device. Thus, the sampling frequency associated with $v_i(n)$ is f_s/M where M is a fixed natural number.

The signals $v_i(n)$ in Fig. 2 are also WSS processes. It is straightforward to show that the autocorrelation coefficients $R_{v_i}(k)$ associated with $v_i(n)$ are given by

$$R_{v_i}(k) = \frac{1}{2\pi}\int_{-\pi}^{\pi} P_x(e^{j\omega})|H_i(e^{j\omega})|^2 e^{jN_i k\omega}d\omega. \quad (1)$$

The above formula shows that $P_x(e^{j\omega})$ uniquely specifies $R_{v_i}(k)$ for all values of k. However, the reverse is not true. That is, in general, knowing $R_{v_i}(k)$ for some or all values of k is not sufficient for characterizing $P_x(e^{j\omega})$ uniquely.

Recall that $v_i(n)$ is a WSS signal so all the statistical information that can be gained about it is confined in its autocorrelation coefficients. One might use the signal processing hardware available at each sensor node and estimate the autocorrelation coefficients $R_{v_i}(k)$ for some k, say $0 \le k \le L-1$. Now, we may pose the sensor network spectrum estimation problem as follows:

Problem 1 *Let $\mathcal{Q}_{i,k}$ denote the set of all power spectra which are consistent with the kth autocorrelation coefficient $R_{v_i}(k)$ estimated at the ith sensor node. That is, $P_x(e^{j\omega}) \in \mathcal{Q}_i$ if*

$$\begin{aligned} \frac{1}{2\pi}\int_{-\pi}^{\pi} P_x(e^{j\omega})|H_i(e^{j\omega})|^2 e^{jMk\omega}d\omega &= R_{v_i}(k), \\ P_x(e^{j\omega}) &\ge 0, \\ P_x(e^{j\omega}) &= P_x(e^{-j\omega}), \\ P_x(e^{j\omega}) &\in \mathbf{L}^1(-\pi,\pi). \end{aligned}$$

Define $\mathcal{Q} \stackrel{\triangle}{=} \bigcap_{i=1}^{N}\bigcap_{k=0}^{L-1}\mathcal{Q}_{i,k}$ where N is the number of nodes and L is the number of autocorrelation coefficients estimated at each node. Find a $P_x(e^{j\omega})$ in $\mathcal{Q}$.

It is possible to show that the sets $\mathcal{Q}_{i,k}$ are closed and convex [2]. The problem of finding a point in the intersection of finitely many closed convex sets is known as the *convex feasibility problem* and is an active area of research in applied mathematics.

If we ignore measurement imperfections and assume that the observed autocorrelation coefficients $R_{v_i}(k)$ are exact, then the sets $\mathcal{Q}_{i,k}$ are non-empty and admit a non-empty intersection $\mathcal{Q}$ as well. In this case $\mathcal{Q}$ contains infinitely many $P_x(e^{j\omega})$. When the measurements $v_i(n)$ are contaminated by noise or $R_{v_i}(k)$ are estimated based on finite-length data records, the intersection set $\mathcal{Q}$ might be empty due to the potential inconsistency of the autocorrelation coefficients estimated by different sensors. Thus, Problem 1 has either no solution or infinitely many solutions. In either case, it is ill-posed[1].

[1] A problem is called well-posed (more precisely, well-posed in the sense of Hadamard) if i) for all admissible data it has a solution, ii) the solution is unique and iii) the solution depends on the data continuously. Otherwise, the problem is considered ill-posed.

3. THE METHOD OF GENERALIZED PROJECTIONS

An elegant way to approach the convex feasibility problem is to employ *generalized projections* [3]. A projection of a given point onto a convex set is defined as another point which has two properties. First, it belongs to the set onto which the projection operation is performed and, second, it renders a minimal value to the distance between the given point and any point of the set (Fig. 3).

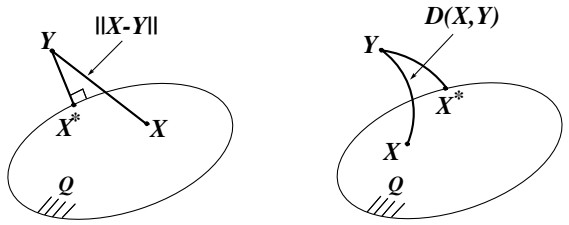

Figure 3: Symbolic depiction of metric projection (left) and generalized projection (right) of a vector Y into a closed convex set $\mathcal{Q}$. In the left figure, the projection X^* is selected by minimizing the metric $||X-Y||$ over all $X \in \mathcal{Q}$ while in the right X^* is found by minimizing the generalized distance $D(X,Y)$ over the same set.

If the Euclidean distance $||X-Y||$ is used in this context then the projection is called a metric projection . In some applications, such as the one considered here, it turns out to be very useful to introduce more general means to measure the "distance" between two vectors[2]. A *generalized distance* is a real-valued non-negative function of two vector variable $D(X,Y)$ defined in a specific way such that it's value may represent the distance between X and Y in some generalized sense. When defining generalized distances, it is customary not to require the symmetry condition. Thus, $D(X,Y)$ may not be the same as $D(Y,X)$. Moreover, we do not insist on the triangle inequality that a traditional metric must obey either.

Example 2 Let $P_1(e^{j\omega}) > 0$ and $P_2(e^{j\omega}) > 0$ be two power spectra in $\mathbf{L}^1(-\pi,\pi)$. The functions

$$\begin{aligned} D_1(P_1,P_2) &= \int_{-\pi}^{\pi} (P_1-P_2)^2\, d\omega, \\ D_2(P_1,P_2) &= \int_{-\pi}^{\pi} \left(P_1 \ln \frac{P_1}{P_2} + P_2 - P_1 \right) d\omega, \\ D_3(P_1,P_2) &= \int_{-\pi}^{\pi} \left(\frac{P_1}{P_2} - \ln \frac{P_1}{P_2} - 1 \right) d\omega, \end{aligned}$$

can be used to measure the generalized distance between $P_1(e^{j\omega})$ and $P_2(e^{j\omega})$. These functions[3] are non-negative and become zero if and only if $P_1 = P_2$. $\diamondsuit$

Using a generalized distance such as the functions mentioned in the above example, we can convert our convex feasibility problem (Problem 1) into a minimization problem:

Problem 2 *Let $\mathcal{Q}$ be defined as in Problem 1. Find $P_x^*(e^{j\omega})$ in $\mathcal{Q}$ such that*

$$P^* = \arg \min_{P \in \mathcal{Q}} D(P, P_0), \tag{2}$$

where $P_0(e^{j\omega})$ is an arbitrary power spectrum, say $P_0(e^{j\omega}) = 1, -\pi \le \omega < \pi$.

When a unique P^* exists, it is called the generalized projection of P_0 onto $\mathcal{Q}$. For our application here we do not require a unique solution. Nonetheless, it can be shown that the generalized distances D_1 and D_2 in Example 2 lead to a unique solution. The choice D_3 will lead to a unique solution too but ceratin singular power spectra must be excluded from the space of valid solutions [4].

4. DISTRIBUTED CALCULATION OF GENERALIZED PROJECTIONS

A very interesting aspect of the generalized projections formulation is that the solution $P^* \in \mathcal{Q}$ can be found using a series of projections onto the intermediate sets $\mathcal{Q}_{i,k}$. These intermediate projections can be computed locally at each sensor node thus allowing the computations to be done simultaneously and in a highly distributed fashion.

In this section, we first calculate the generalized projection of a given power spectrum onto the sets $\mathcal{Q}_{i,k}$ for the sample distance functions introduced in Example 2. Then, we propose a distributed algorithm for

[2]The functional form of the solution will depend on the choice of the generalized distance used in the projection. Obtaining a particular functional form which is easy to manipulate or interpret is often the main reason for using a generalized distance instead of the conventional Euclidean metric.

[3]Note that D_1 is simply the Euclidean distance between P_1 and P_2. The functions D_2 and D_3 have roots in information theory and statistics. They are known as the Kullback-Leibler divergence and Burg cross entropy, respectively.

calculating the final solution P^* from these intermediate projections.

Let $P_{[P_1 \mapsto \mathcal{Q}_{i,k}; D_j]}$ denote the power spectrum resulting from projecting a given power spectrum P_1 onto the set $\mathcal{Q}_{i,k}$ using a given distance functions D_j. That is,

$$P_{[P_1 \mapsto \mathcal{Q}_{i,k}; D_j]} \stackrel{\triangle}{=} \arg \min_{P \in \mathcal{Q}_{i,k}} D_j(P, P_1). \qquad (3)$$

Using standard techniques from calculus of variations we can show that the generalized distances D_1, D_2 and D_3 introduced in Example 2 result in projections of the form

$$\begin{aligned} P_{[P_1 \mapsto \mathcal{Q}_{i,k}; D_1]} &= P_1(e^{j\omega}) - \alpha |H_i(e^{j\omega})|^2 \cos(Mk\omega), \\ P_{[P_1 \mapsto \mathcal{Q}_{i,k}; D_2]} &= P_1(e^{j\omega}) \exp\left(-\beta |H_i(e^{j\omega})|^2 \cos(Mk\omega)\right), \\ P_{[P_1 \mapsto \mathcal{Q}_{i,k}; D_3]} &= \left(P_1(e^{j\omega})^{-1} + \gamma |H_i(e^{j\omega})|^2 \cos(Mk\omega)\right)^{-1}, \end{aligned}$$

where α, β and γ are parameters (Lagrange multipliers). These parameter should be chosen such that in each case $P_{[P_1 \mapsto \mathcal{Q}_{i,k}; D_j]} \in \mathcal{Q}_{i,k}$. That is,

$$\int_{-\pi}^{\pi} P_{[P_1 \mapsto \mathcal{Q}_{i,k}; D_j]} |H_i(e^{j\omega})|^2 e^{jMk\omega} d\omega = 2\pi R_{v_i}(k). \quad (4)$$

The reader may observe that the above equation leads to a closed-form formula for α but in general finding β and γ requires numerical methods.

The projection formulae developed above can be employed in a variety of iterative algorithms to find a solution in the intersection of $\mathcal{Q}_{i,k}$ [5]. Here, we opt for a particular algorithm that combines successive projections onto $\mathcal{Q}_{i,k}$ with a kind of generalized averaging to generate a sequence of solutions $P^{(m)}$ which will eventually converge to a $P^* \in \bigcap_{i,k} \mathcal{Q}_{i,k}$. We call this algorithm the *Star Algorithm* since its data flow is similar to a star pattern. The main steps of the Star Algorithm are outlined in the box at the top right of this page.

5. CONCLUDING REMARKS

When the autocorrelation estimates $R_{v_i}(k)$ are accurate, the Star Algorithm will generate a solution in $\mathcal{Q}$ but when $R_{v_i}(k)$ are not estimated accurately, the intersection set $\mathcal{Q}$ might become empty. In this case, the Star Algorithm converges to a solution whose average generalized distance from the feasible sets $\mathcal{Q}_{i,k}$ is minimum [6].

Lack of space prohibits a thorough discussion of the practical computational issues associated with the Star Algorithm here. For the same reason, illustrative simulation examples are not provided either. We invite the reader to visit the website `www.multirate.org` where further material supplementing this paper will be maintained.

The Star Algorithm

Input: A distance function $D(P_1, P_2)$, an initial power spectrum $P_0(e^{j\omega})$, the sensor frequency responses $H_i(e^{j\omega})$, the autocorrelation estimates $R_{v_i}(k)$ and a tolerance parameter $\epsilon > 0$.
Output: A power spectrum $P_*(e^{j\omega}) \in \mathcal{Q}$.
Procedure:

1. Let $m = 0$ and $P^{(0)} = P_0$.
2. Send $P^{(m)}$ to all sensor nodes.

   $\Rightarrow$ at the ith sensor node:

   (i) Let $n = 0$ and define $\tilde{P}^{(n)} = P^{(m)}$.

   (ii) Calculate $\tilde{P}_k = P_{[\tilde{P}^{(n)} \mapsto \mathcal{Q}_{i,k}; D_j]}$ for all k.

   (iii) Increase n by 1 and calculate $\tilde{P}^{(n)} = \arg\min \sum_k D(P, \tilde{P}_k)$.

   (iv) If $D(\tilde{P}^{(n)}, \tilde{P}^{(n-1)}) > \epsilon$ go to Step (ii) and repeat. Otherwise, define $P_i^{(m)} = \tilde{P}^{(n)}$ and send it to the central unit.
3. Receive $P_i^{(m)}$ from all sensor nodes.
4. Increase m by 1 and calculate $P^{(m)} = \arg\min \sum_i D(P, P_i^{(m-1)})$.
5. If $D(P^{(m)}, P^{(m-1)}) > \epsilon$, go to Step 2 and repeat. Otherwise stop and output $P^* = P^{(m)}$.

6. REFERENCES

[1] S. M. Kay, *Modern Spectrum Estimation: Theory and Applications*, Prentice Hall, Upper Saddle River, NJ, 1988.

[2] O. S. Jahromi, B. A. Francis, and R. H. Kwong, "Spectrum estimation using multirate observations," *IEEE Trans. Signal Process.*, (to appear), preprint available from www.multirate.org.

[3] H. H. Bauschke and J. M. Borwein, "On projection algorithms for solving convex feasibility problems," *SIAM Review*, vol. 38, pp. 367–426, 1996.

[4] J. M. Borwein and A. S. Lewis, "Partially-finite programming in $\mathbf{L}_1$ and the existence of maximum entropy estimates," *SIAM Journal of Optimization*, vol. 3, pp. 248–267, 1993.

[5] Y. Censor and S. A. Zenios, *Parallel Optimization: Theory, Algorithms, and Applications*, Oxford University Press, 1997.

[6] C. Byrne and Y. Censor, "Proximity function minimization using multiple Bregman projections, with applications to split feasibility and Kullback-Leibler distance minimization," *Annals of Operations Research*, vol. 105, pp. 77–98, 2001.