On the Interdependence of Routing and Data Compression in Multi-Hop Sensor Networks *

Anna Scaglione
School of Electrical and Computer Engineering
Cornell University
http://people.ece.cornell.edu/scaglione/

Sergio D. Servetto
School of Electrical and Computer Engineering
Cornell University
http://people.ece.cornell.edu/servetto/

ABSTRACT

We consider a problem of broadcast communication in a multi-hop sensor network, in which samples of a random field are collected at each node of the network, and the goal is for all nodes to obtain an estimate of the entire field within a prescribed distortion value. The main idea we explore in this paper is that of jointly compressing the data generated by different nodes as this information travels over multiple hops, to eliminate correlations in the representation of the sampled field. Our main contributions are: (a) we obtain, using simple network flow concepts, conditions on the rate/distortion function of the random field, so as to guarantee that any node can obtain the measurements collected at every other node in the network, quantized to within any prescribed distortion value; and (b), we construct a large class of physically-motivated stochastic models for sensor data, for which we are able to prove that the joint rate/distortion function of all the data generated by the whole network grows slower than the bounds found in (a). A truly novel aspect of our work is the tight coupling between routing and source coding, explicitly formulated in a simple and analytically tractable model—to the best of our knowledge, this connection had not been studied before.

Keywords

Multi-hop networks, sensor networks, source coding, routing, cross-layer interactions

1. INTRODUCTION

1.1 Problem Setup

Consider the following data transmission problem. N nodes v_i are placed on the closed set $[0,1]\times[0,1]$, at locations (x_i, y_i). Each v_i observes a sample S_i of some spatial stochastic process with rate/distortion function $R_S(D)$ (see Appendix B), having the property that the correlation between samples increases as the distance between them in the grid decreases. Each v_i wants to communicate an approximation of its S_i to every other node in the network. Each v_i can only send messages to and receive messages from nodes within distance C_N, where this connectivity radius is chosen so as to ensure that the network will be connected with high probability [8]. Links have a fixed finite capacity L, independent of network size. Given these assumptions, we are interested in determining under what conditions it is possible for each node v_i to obtain an estimate of the sample observed at every network location v_k, $k = 1, ..., N$, such that v_i has an estimate of the entire field of samples whose total distortion $\mathtt{E}(d(S,\hat{S})) < D$, for any prescribed value $D \geq 0$, and for an appropriate distortion measure $d(\cdot,\cdot)$. (Note: if $d(S,\hat{S}) = \sum_{i=1}^{N} d(S_i,\hat{S}_i)$ and if the distribution of the samples is identical, each sample will have average distortion $\mathtt{E}(d(S_i,\hat{S}_i)) < D/N$.)

Independent Encoders

To provide the sought conditions, the first thing we need to do is find out how much traffic does this network generate. Very clearly this question depends on the statistics of the sensed data. For illustration purposes and without loss of generality, consider a simple example: suppose that S_i is uniform in the range $[0,1]$, that each node uses a scalar quantizer with B bits of resolution (i.e., the quantization step is 2^{-B}), and that distortion is measured in the mean-square sense. A well known result from basic quantization theory states that the average distortion achieved by such a quantizer on this particular source is $\frac{1}{12}2^{-2B}$ [6]. Adding distortions over all nodes in the network we then have that $\sum_{i=1}^{N} d(S_{ij},\hat{S}_i) = \frac{1}{12}N2^{-2B}$, and hence, solving for B in $D = \frac{1}{12}N2^{-2B}$, we get that to maintain a total distortion over the entire network of D each sample requires $B = \lceil \frac{1}{2}\log_2(\frac{N}{12D}) \rceil$ bits. As a result, the total amount of traffic generated by the whole network scales like $O(N\log N)$ in network size. Furthermore, it is important to emphasize that although with more sophisticated quantization strategies than the simple uniform quantizer considered in this motivational example one could certainly reduce the number of bits generated for a fixed distortion level D, this reduction would only affect constants hidden by the big-oh notation, but the $O(N\log N)$ scaling behavior of network traffic would remain unchanged [3] (examples of analyses of this type can be found in [12, Sec. 3] and [14, Sec. 7]).

Once we have determined how much data our particular coding strategy (independent quantizers at each node) produces, we need to know if the network has enough capacity to transport all that data. And the answer is *no* [9]. To see how this is so, consider a partition of the network as shown in Fig. 1.

---

*Work supported in part by the National Science Foundation under grant CCR-0227676 and ONR Contract N00014-00-1-0564.

MOBICOM'02, September 23–26, 2002, Atlanta, Georgia, USA.

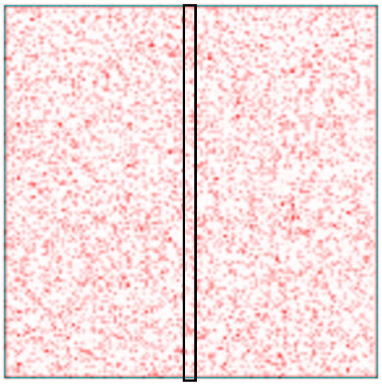

Figure 1: N **nodes are spread uniformly over the unit square** $[0,1] \times [0,1]$ **(**N **large). Take a differential volume of size** $\Delta \times \Delta$ **(**Δ **small). With high probability, the number of nodes in a differential volume is** $\approx N\Delta^2$**, and so the number of nodes in a strip as shown in the figure is** $\approx N\Delta$**. Since the total number of nodes is** N**, we must have** $N = N\Delta \times N\Delta$ **(because the total area of the unit square is the product of the areas of two strips as shown in the figure, one horizontal and one vertical), and hence we must have that the number of nodes in a strip as shown is** $N\Delta = O(\sqrt{N})$**.**

In the sensor broadcast problem of interest in this work, all the nodes in the network must receive information about the measurements collected by all other nodes. As a result, all the traffic generated on the left portion of the network must be carried to the right, and all the traffic generated on the right portion of the network must be carried to the left. That is, according to our calculation above, the nodes within the strip marked in Fig. 1 must share the load of moving $O(N \log N)$ bits across this network cut. But since links have finite capacity L, the capacity of this cut cannot be larger than $O(L\sqrt{N})$. But from the max-flow/min-cut theorem [4, Ch. 27], we know that the value of *any* flow in this network is upper bounded by the capacity of *any* network cut, and therefore, the total transport capacity of this network cannot be higher than $O(L\sqrt{N})$.

And now we see what the problem is: $O(N \log N)$ bits must go across a cut of capacity $O(L\sqrt{N})$. That is, even optimal vector quantization strategies cannot compress the data enough so that the network can carry it—the network does not scale up.[1]

Correlated Samples

The scaling analysis for independent encoders presented above ignores one fundamental aspect: increasing correlations in sensor data as the density of nodes increases. And indeed, if the data is so highly correlated that all sensors observe essentially the same value, at least intuitively it seems clear that almost no exchange of information at all is needed for each node to know all other values: knowledge of the local sample and of the global statistics already provide a fair amount of information about remote samples. And this naturally raises the question: are there *other* coding strategies that can compress sensor data enough?

1.2 Routing and Data Compression

We consider two approaches to the problem of data compression for multi-hop sensor networks. One of these is the idea of using *distributed* source coding techniques, that has attracted some attention recently [11, 12, 15], as multi-hop sensor networks appear to be the "killer application" for some of these well-established theoretical results. The second one is a combination of classical source coding methods and routing algorithms which, to the best of our knowledge, has not been explored before.

Distributed Source Coding

Under some mild technical assumptions, recent work has established the possibility of using *distributed* source coding techniques to overcome the problem of vanishing throughputs of [9], that renders estimation impossible in the case of independent sources [12]. This result is very interesting from a theoretical point of view, since it proves that it is possible for each node v_i to get rid of all correlation in the data *without* requiring nodes to exchange a single bit of information.

Whereas certainly interesting in their own right, from the point of view of an engineer engaged in the design of a practical sensor network, the results of [12] are not very relevant. This is because the proof developed in [12] to show that arbitrarily accurate estimation of the remote values is possible—even without any form of cooperation among sensing nodes—involves the use of codes for the problem of rate/distortion with side information [5, Ch. 14.9]. Efficient codes for this problem are characterized by the fact that the probability of all codewords is nearly uniform, irrespective of the statistics of the source and of the length of the blocks used. As a result, for practical, short codes (e.g., scalar quantizers), gains due to entropy coding are negligible. From a theoretical point of view this is not a problem: by considering ergodic sources and large blocks of data, the Asymptotic Equipartition Property guarantees that these blocks will be uniformly distributed anyway, and hence no entropy coding is needed [5, Ch. 3]. But from a practical point of view, high-dimensional vector quantizers are not always feasible, and so in practical, short-block settings, it is unclear how much of the gains obtained by these codes by means of exploiting correlations among samples are lost due to the inability of these codes to take full advantage of entropy coding gains.

Routing and Source Coding

Whereas requiring that nodes compress all their data without exchanging any information at all would certainly be a valid way to go about solving this problem, we submit that this approach may be needlessly restrictive. After all, in multi-hop networks, a message does visit multiple nodes before it reaches its destination, and therefore one may wonder why not use classical source codes, and then re-encode the data as it hops around the network to remove correlations among samples. Two examples (among many more possibilities) of such a scheme are shown in Fig. 2.

We see from the examples in Fig. 2 that, at the expense of increased transmission delays, we can communicate all samples to all nodes generating an amount of traffic which is essentially the same as if one node had collected all the samples and encoded them jointly, and then this information had been broadcast to all other nodes. Alternatively, by sacrificing some compression efficiency, it is also possible to incur in lower transmission delays. That is, *there is an inherent tradeoff between bandwidth use and decoding delay, and these two quantities are linked together by the routing strategy employed.*

Contrary to the complicated vector quantizers required by distributed source coding, when routing and source coding are combined the processing at each node can be done using any of the standard compression technique which are used normally to compress sequences from analog sources (DPCM, $\Sigma - \Delta$, subband-coding

[1] Techniques based on flows and cuts to analyze information theoretic capacity problems in networks have been proposed in [1], [5, Ch. 14.10], In this context, those techniques provide an alternative interpretation for the Gupta/Kumar results [9].

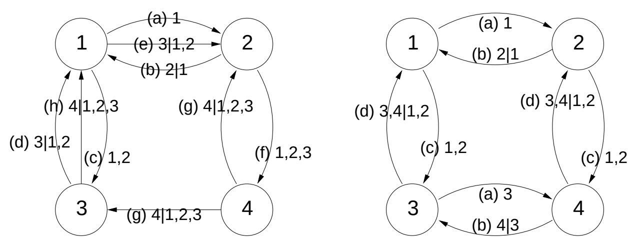

Figure 2: Consider a network with 4 nodes, each of which observes a variable X_i ($i = 1...4$), with joint entropy $\mathcal{H}(X_1, X_2, X_3, X_4)$. Two possible ways of scheduling transmissions are shown. The notation used in the figure is that (a) is the first transmission, (b) the second, (c) the third, and so on; if two transmissions have the same letter label, they can be performed in parallel; $i|j$ means that the sample of node i is encoded when knowledge of the sample of node j is available. Using chain rules for entropies, we see that in the transmission schedule of the left figure we generate a total traffic of $3\mathcal{H}(X_1, X_2, X_3, X_4)$, and it takes 8 time slots to complete. In the schedule of the right figure, we generate more traffic ($2\mathcal{H}(X_1, X_2, X_3, X_4) + \mathcal{H}(X_1, X_2) + \mathcal{H}(X_3, X_4)$ bits), but now we only require 4 time slots to complete all transmissions.

such as JPEG, etc.). An interesting alternative is using a scalar quantizer locally and then forward the data in files compressed using universal source coding algorithms, such as Lempel-Ziv. The latter option would be effective in more general contexts than the one of sensors networks: indeed, *we believe that the idea of combining routing with data compression is key to the most general multi-hop networks* not only composed of sensors. In fact, oftentimes distributed communication and routing protocols themselves generate correlated and redundant data that are broadcasted through the network and that can easily cause congestion. Clearly the compression ratio will be higher as more and more redundant data are pooled together at each node, but this mechanism would precisely be the one that prevents congestion.

Our goal in this first paper on the subject is to provide a solid theoretical framework based on rate distortion theory (see Appendix B) which supports these intuitions.

1.3 Main Contributions and Organization of the Paper

In the framework defined above, we see that there are two fundamental questions that need answers:

1. Under what conditions on the statistics of the source can a network of the type considered in this work transport all the data generated by the sources?

2. What are the tradeoffs between bandwidth requirements and transmission delays?

This paper focuses on the first question, and our main results are:

(a) Proof of the existence of routing algorithms and source codes that require no more than $O\left(R_{S_1 \ldots S_N}(D)\right)$ bits in $O(\sqrt{N})$ transmissions, where $R_{S_1 \ldots S_N}(D)$ is the joint rate/distortion function of all samples in the field. This would prove that, even under decentralization constraints, *classical* source codes can still achieve optimal compression efficiency. And furthermore, attaining that optimal performance requires a number of transmissions that is *sub-linear* in the number of nodes in the network.

(b) Proof that $R_{S_1 \ldots S_N}(D) \leq O(\log(N/D))$, under some extremely mild regularity conditions on the random field. That is, if the average distortion per sample D/N is kept constant, the field generates a *bounded* amount of information, independently of its size. And if the total distortion is kept constant, the growth of R is only logarithmic in N, well below the total transport capacity of the network, which grows like $O(L\sqrt{N})$ (as argued in Section 1.1 and [9]).

The second question however is outside the scope intended for this paper, and will be dealt with elsewhere.

Since the sensors are continuous and not discrete sources the theoretical tool to analyze the problem is Rate Distortion Theory [5, Ch. 13], which allows us to determine the minimum number of bits required to represent the source samples to achieve the desired level of average distortion. The number of bits is obtained as a function of the joint statistics of the process S only, without making reference to a specific quantization algorithm and using the notion of joint differential entropy rather than joint entropy. To keep this paper self-contained the reader not familiar with Rate Distortion Theory can find a succinct description of it in Appendix B.

The rest of this paper is organized as follows. In Section 2 we compute bounds on the transport capacity of our network, and we use these bounds to impose constraints on the rates at which the random field sensed can generate information. Then, in Section 3, we propose a model for the generation of sensor data, based on which we prove that indeed, the amount of data generated by the network is well below network capacity. Concluding remarks are presented in Section 4.

2. TRANSPORT CAPACITY

Using simple network flow concepts, in Section 1.1 we argued that an upper bound on the transport capacity of a network of size N is $O(L\sqrt{N})$. Our goal in this section is to construct one particular flow: from the amount of data that this flow needs to push across the network, and from the upper bound on the capacity of the network, we derive a constraint on the amount of data that the source can generate if it is to be broadcast over the whole network.

2.1 A Network Flow on a Regular Grid

We consider first the case of a regular grid, as it naturally precedes the construction for a general random grid.

Suppose that we want to achieve an average distortion D in the reproductions of S_u. Let $q(S_i)$ denote a *quantized* version of S_i, where the quantizer q is such that $\mathtt{E}\left(d(S_i, q(S_i)\right) \leq D$. The entropy induced on the codewords of q by quantization of the source S is denoted by $\mathcal{H}_q(S)$.

We construct a flow recursively. For simplicity, assume $N = 2^{2k}$, for some integer $k \geq 0$. When $k = 0$, we have the trivial case of a network of size $N = 1$, in which all nodes (i.e., v_1) know the value of all samples (i.e., S_1) without any transfer of information.

Consider now a partition of nodes v_i into 4 groups: P_{UL} containing all the nodes in the upper-left corner of size $\frac{1}{2} \times \frac{1}{2}$, P_{UR} nodes in the upper-right corner, P_{LL} lower-left, and P_{LR} lower-right. S_P denotes the set of variables observed by nodes in a set P. This partition is illustrated in Fig. 3.

We have that $P_{UL}, P_{UR}, P_{LL}, P_{LR}$ are all subnets of size $N = 2^{2(k-1)}$, and so from our recursion we get that all nodes within each subnet know the values of all variables within their subnet to within distortion D. But now, we have reduced our problem to the problem with four nodes considered in Fig. 2, and we know that exchanging a total of $3\mathcal{H}_q(S_{P_{UL}}, S_{P_{UR}}, S_{P_{LL}}, S_{P_{LR}})$ bits, in a total of 8 transmissions across cuts (plus $O(\sqrt{N})$ transmissions to

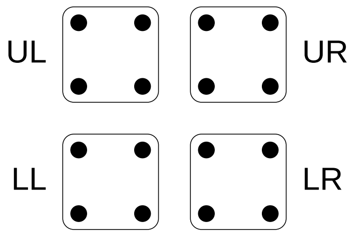

Figure 3: Partition of a network of size 4×4 into four subnets.

spread data within cuts), is enough to ensure that every node in the network of size $N = 2^{2k}$ knows every value to within distortion D. This construction is illustrated in Fig. 4.

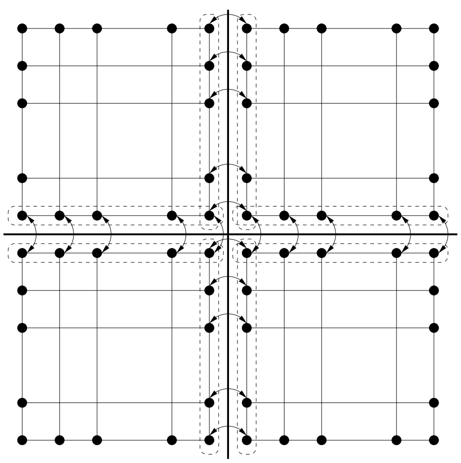

Figure 4: Since each node on the boundary of a cut has knowledge of all samples within its subnet, each one of them can encode all these samples jointly and send $\frac{1}{2\sqrt{N}}$ -th of this data across the cut. Then, in $O(\sqrt{N})$ more transmissions, all these pieces can be spread throughout the subnet to reach all nodes.

For reference, compare the performance of this flow against an ideal one, in which each v_i holds $\frac{1}{N}$-th of a *joint* encoding of $S_1...S_N$ computed by some "genie": the task in this case is to broadcast the piece held by each node to all other nodes, so that all nodes can obtain the values of all measurements. That idealized system still must generate a total of $3R_{S_1...S_N}(D)$ bits of traffic, and still requires $O(\sqrt{N})$ transmissions to complete the broadcast (both follow from straightforward calculations). Therefore, we see that the only penalty that the flow constructed above takes with respect to the genie-aided flow is the need to re-encode messages at intermediate nodes.

2.2 The Case of a Random Grid

The only difference between the case of a random grid and the case of a regular grid as considered above is the fact that, in the random case there is a non-zero probability that, in the recursive definition of the flow above, we may encounter an empty subnet. But in that case, only trivial modifications can take care of this problem. For example, in the left schedule of Fig. 2, if node 3 was not there, an appropriate schedule of transmissions would be: (a) X_1 sent from 1 to 2, (b) $X_2|X_1$ sent from 2 to 1, (c) X_1, X_2 sent from 2 to 3, (d) $X_3|X_1, X_2$ sent from 3 to 2, (e) $X_3|X_1, X_2$ sent from 2 to 1. In this case, again using the chain rules for entropies we get that the total number of bits sent is $2\mathcal{H}(X_1, X_2, X_3)$. So with simple modifications like this one, the case of a random grid is easily reduced to the case of a regular grid described above.

2.3 General Constraints

We know the following facts:

- Since $\{S_{P_{UL}}, S_{P_{UR}}, S_{P_{LL}}, S_{P_{LR}}\} = \{S_1...S_N\}$, $O\left(\mathcal{H}_q(S_1...S_N)\right)$ bits must go across the 4-way cut defined by $S_{UL}, S_{UR}, S_{LL}, S_{LR}$.
- The capacity of the 4-way cut is $O(L\sqrt{N})$.
- From rate/distortion theory [5, Ch. 13], we know that $\mathcal{H}_q(S_1 \ldots S_N) \geq R_{S_1 \ldots S_N}(D)$, since q is a quantizer with mean distortion D.

Therefore, as long as

$$O\left(R_{S_1 \ldots S_N}(D)\right) \leq O\left(L\sqrt{N}\right), \qquad (1)$$

there exist codes and routes such that every node can have enough information to form an estimate of the sample available at every other node within a prespecified distortion value D.

3. A MODEL FOR SENSOR DATA

In Section 2 we saw that, by appropriate routing and re-encoding along routes, we can compress all the data generated by the entire network down to $O\left(R_{S_1 \ldots S_N}(D)\right)$. Our goal in this section is to verify that, for reasonable models of sensor data, we have that eqn. (1) is satisfied, so that the broadcast problem can be effectively solved.

To be able to talk about "the rate/distortion function of the data generated by the entire network" we need a model for this data. The main idea that we would like to capture in our models for sensor data is that, if this data corresponds to measurements of a random process with some kind of regularity conditions (as would be the case for almost any physical process one could conceive), then these measurements have to become increasingly correlated as the density of nodes becomes large. We propose to this end a fairly general class of such models, under two assumptions: (a) the data are Gaussian random variables (b) the correlation among samples is an arbitrary spatially homogeneous function; and (c), as we let the number of nodes in the network grow, the correlation matrix converges to a *smooth* two-dimensional function.

Assumption (a) is a worse case scenario as far as compression is concerned, as a consequence of Theorem 4 in Appendix B. Spatial stationarity, even though not totally general is a technical assumption common to many statistical analysis and captures well local properties of random processes.

3.1 Source Model

This section establishes the basic model upon which we will base our asymptotic analysis. Let $S(t)$ denote the random vector $\{S(t)\}_i = S_i(t)$ of the samples collected by the sensors at time t. Our first assumption is:

(a) $S(t)$ is a spatially correlated random Gaussian vector $\sim \mathcal{N}(0, \boldsymbol{R})$. The samples are temporally uncorrelated.

The temporal independence can be obtained if we assume that the power spectrum of the data collected in time is band-limited and the data are sampled at the Nyquist rate. In any case, it is not a restrictive assumption and further gains in terms of compression could be obtained exploiting the temporal dependence of the samples. Because of the temporal independence, we will focus on one vector of samples only S, and from now on we will drop the time index.

Considering as distortion measure the mean square error (MSE), i.e. $d(S,\hat{S}) = \|S-\hat{S}\|^2$, and putting the constraint

$$E(\|S-\hat{S}\|^2) < D,$$

under assumption (a1) we can calculate the rate/distortion function of the network using the well known *reverse water-filling* result [5]. Denoting by $\lambda_1 \geq \lambda_2 \ldots \geq \lambda_N$ the eigenvalues of $\boldsymbol{R}$, the rate/distortion function is

$$R(D) = \sum_{n=1}^{N} \frac{1}{2} \log \quad \frac{\lambda_n}{D_n} \tag{2}$$

where

$$D_n = \begin{cases} K & \text{if } K < \lambda_n, \\ \lambda_n & \text{otherwise.} \end{cases} \tag{3}$$

and K is such that

$$\sum_{n=1}^{N} D_n = D. \tag{4}$$

For $\sum_{n=1}^{N} \lambda_n \geq D$, there exists an $\bar{n} \leq N$ such that $K < \lambda_{\bar{n}}$ and $K \geq \lambda_{\bar{n}+1}$, therefore:

$$K = \frac{D - \sum_{n=\bar{n}+1}^{N} \lambda_n}{\bar{n}} \tag{5}$$

and

$$R(D) = \sum_{n=1}^{\bar{n}} \frac{1}{2} \log \quad \frac{\lambda_n}{K} \quad . \tag{6}$$

Assuming a given correlation structure among samples, computation of the rate/distortion function just requires calculating numerically the eigenvalues of $\boldsymbol{R}$. The covariance matrix is formed with the samples of the continuous multivariate function that represents the correlation between the samples taken two arbitrary points in the network:

$$\{\boldsymbol{R}\}_{i,k} \triangleq E\{S_i S_k\} = R(x_i, x_k, y_i, y_k). \tag{7}$$

The eigenvalues $\boldsymbol{u}$, with entries $\{\boldsymbol{u}\}_i = u(x_i, y_i)$ satisfy the following equation:

$$\lambda^{(N)} u^{(N)}(x_i, y_i) = \sum_{k=1}^{N} R(x_i, x_k, y_i, y_k) u^{(N)}(x_k, y_k) \tag{8}$$

The challenges are: 1) proving that the eigenvalue of the correlation matrix formed with this samples tend to the eigenvalues of the continuous integral equation:

$$\lambda^{(\infty)} u^{(\infty)}(x,y) = \iint R(x,\xi,y,\nu) u^{(\infty)}(\xi,\nu) d\xi d\nu; \tag{9}$$

2) providing a reasonable model for the continuous integral equation that allows to obtain the asymptotic rate distortion bound. To this end, we postulate that

(b) the correlation between points in (7) is spatially homogeneous:

$$R(x,\xi,y,\nu) = R(x-\xi, y-\nu), \tag{10}$$

The consequent structure of $\boldsymbol{R}$ on a regular grid is also known as *doubly Toeplitz*, i.e. $\boldsymbol{R}$ is a block Toeplitz matrix with blocks that are Toeplitz themselves.

Assumption (b) implies that

$$\lambda^{(N)} u^{(N)}(x_i, y_i) = \sum_{k=1}^{N} R(x_i - x_k, y_i - y_k) u^{(N)}(x_k, y_k) \tag{11}$$

$$\lambda^{(\infty)} u^{(\infty)}(x,y) = \iint R(x-\xi, y-\nu) u^{(\infty)}(\xi,\nu) d\xi d\nu. \tag{12}$$

The advantage of this model is that the empirical distribution of the eigenvalues of $\boldsymbol{R}$ converges under mild assumptions to the 2-D Fourier Spectrum of (10), as we see next.

3.2 Asymptotic Distribution of the Eigenvalues

To address the first issue we have first to rewrite (8) as a quadrature formula which approximate the integral equation (9). For a general integral, there will be quadrature coefficients α_i such that the approximation of (9) holds:

$$\begin{aligned} &\sum_{k=1}^{N} R(x_i, x_k, y_i, y_k) u^{(N)}(x_k, y_k) \alpha_k \\ &\approx \iint R(x,\xi,y,\nu) u^{(\infty)}(\xi,\nu) d\xi d\nu. \end{aligned} \tag{13}$$

In our case, since we want to explore the convergence of the eigenvalues of (8) rather than having an accurate numerical integration, we can set $\alpha_i = 1/N$. In fact, normalizing by N both sides of (8) one can note that when (13) is valid also the following approximation holds

$$\lambda^{(N)}/N \approx \lambda^{(\infty)}. \tag{14}$$

The assumptions for the approximation (13) that set up the limits of the approximation in (14) are specified in the following theorem derived from [10, Sec. 5.4]:

LEMMA 1. *Denoting by $\lambda^{(\infty)}$ an arbitrary eigenvalue of (9) and by $u^{(\infty)}(\xi,\nu)$ the corresponding normalized eigenvector, for N sufficiently large there exist an eigenvalue of $\boldsymbol{R}$ such that:*

$$|\lambda^{(N)} - N\lambda^{(\infty)}|^2 \leq \frac{\frac{1}{N}\sum_{i=1}^{N} \mathcal{E}_N\{R(x_i, y_i, \xi, \nu) u(\xi,\nu)\}}{1 + \mathcal{E}_N\{u^2(\xi,\nu)\}}, \tag{15}$$

where $\mathcal{E}_N\{f(x,y)\}$ denotes the quadrature error, i.e.:

$$\iint f(x,y) dx dy = \sum_{i=1}^{N} f(x_i, y_i)\alpha_i + \mathcal{E}_N\{f(x,y)\}.$$

Assuming that:

(c.I) *For any continuous $f(x,y)$ the grid (x_i, y_i) is such that with $\alpha_i = 1/N$ the quadrature error $\mathcal{E}_N\{f(x,y)\} \to 0$;*

then (15) implies that:

$$\lim_{N\to\infty} \lambda^{(N)}/N = \lambda^{(\infty)}.$$

Beside providing the necessary assumption on the grid for the asymptotic analysis to hold, using the triangular inequality the previous Lemma allows us to easily prove the following:

COROLLARY 1. *The eigenvalues of $\boldsymbol{R}$ and $\boldsymbol{R}'$ corresponding to two different grids are such that*

$$\begin{aligned} |\lambda^{(N)} - \lambda'^{(N)}|^2 \leq &\frac{\frac{1}{N}\sum_{i=1}^{N} \mathcal{E}_N\{R(x_i, y_i, \xi, \nu) u(\xi,\nu)\}}{1 + \mathcal{E}_N\{u^2(\xi,\nu)\}} \\ &+ \frac{\frac{1}{N}\sum_{i=1}^{N} \mathcal{E}_N\{R(x'_i, y'_i, \xi, \nu) u(\xi,\nu)\}}{1 + \mathcal{E}_N\{u^2(\xi,\nu)\}}, \end{aligned} \tag{16}$$

Corollary 1 implies that we can rely on any grid that has the same asymptotic behavior, such as for example a regular lattice, and extrapolate the asymptotic behavior of the eigenvalues from the latter.

Under the assumption of e can define:

$$\psi(\xi_{i,k}, \nu_{i,k}) \triangleq R(x_i - x_k, y_i - y_k) = \{\boldsymbol{R}\}_{i,k}, \qquad (17)$$

Adopting a regular grid covering the square of area 1, the spacing between them is $x_i - x_k = \xi_{i,k} = (i-k)/\sqrt{N}$ and like-wise $\nu_{i,k}$. Szegö's theorem [7] establishes that asymptotically the eigenvalues of a Toeplitz matrix converge to the spectrum of the correlation function. Essentially, Szegö's theorem establishes that the eigenfunctions of an homogeneous kernel tend to be the Fourier basis of complex exponentials. The result can be generalized to the two dimensional case when the matrix is doubly Toeplitz, i.e.:

$$\lim_{N\to\infty} \lambda_n(\boldsymbol{R}) = \iint \psi(\xi/\sqrt{N}, \nu/\sqrt{N}) e^{-j2\pi(f\xi + f'\nu)} d\xi d\nu \qquad (18)$$

$$= N\Psi(\sqrt{N}f, \sqrt{N}f). \qquad (19)$$

Before proceeding, a second modeling assumption therefore is:

(c.II) $\Psi(f, f')$id bandlimited with respect to f, f' bandwidth $\leq W$, i.e. $\Psi(f, f') = 0$ for $|f, f'| > W$.

With this assumption, we capture the notion that the limit covariance function varies smoothly in space.

3.3 Asymptotic Rate Distortion Function of the Network

The asymptotic rate/distortion function is obtained replacing the summations in (5) and (6) with integrals over (f_x, f_y). Specifically, because the eigenvalues become asymptotically a continuous function, there will be points (f'_x, f'_y) where $K = N\Psi(\sqrt{N}f'_x, \sqrt{N}f'_y)$.

Let us denote the sets of points f_x and f_y where $0 \leq N\Psi(\sqrt{N}f_x, \sqrt{N}f_y) \leq K$ as $\mathcal{I}$, i.e.:

$$\mathcal{I} \triangleq \{(f_x, f_y) : 0 \leq N\Psi(\sqrt{N}f_x, \sqrt{N}f_y) \leq K\} \qquad (20)$$

and similarly for $\mathcal{I}$. Let us also introduce the set $\bar{\mathcal{I}}$

$$\bar{\mathcal{I}} \triangleq \{(f_x, f_y) : N\Psi(\sqrt{N}f_x, \sqrt{N}f_y) > K\} \qquad (21)$$

$$K \approx \frac{\frac{D}{N} - \iint_{\mathcal{I}} N\Psi(\sqrt{N}f_x, \sqrt{N}f_y) df_x df_y}{\iint_{\bar{\mathcal{I}}} df_x df_y}. \qquad (22)$$

The rate/distortion function is:

$$R(D) = \frac{N}{2} \iint_{\bar{\mathcal{I}}} \log \frac{N\Psi(\sqrt{N}f_x, \sqrt{N}f_y)}{K} df_x df_y. \qquad (23)$$

Because of (c.II) the areas of both $\mathcal{I}$ and $\bar{\mathcal{I}}$ are smaller than $4W^2/N$. Thus, we have the following lower bound on K (also illustrated in Fig. 5):

$$K \geq \frac{\frac{D}{N} - K\frac{4W^2}{N}}{\frac{4W^2}{N}} \quad \Rightarrow \quad K \geq \frac{D}{8W^2}, \qquad (24)$$

Together with (c.II), this justifies the following upper-bound on the rate/distortion function:

$$R(D) \leq \frac{N}{2} \iint_{\bar{\mathcal{I}}} \log \frac{N\Psi(\sqrt{N}f_x, \sqrt{N}f_y)}{D/8W^2} df_x df_y.$$

$$\leq 2W^2 \log(ND^{-1}\sup(\Psi(f_x, f_y))8W^2). \qquad (25)$$

So, we see that the total rate/distortion function over the entire network is $O(\log(N/D))$. This is very significant. D/N is the average distortion per sample. If kept fixed, then the total amount of traffic generated by the network is upper bounded by a constant, irrespective of network size. Alternatively, if we keep the total distortion D fixed, by considering increasingly large N we let the average distortion $D/N \to 0$. And the amount of traffic generated by doing this grows only *logarithmically* in N, well below the capacity bound $O(L\sqrt{N})$ proved in eqn. (1).

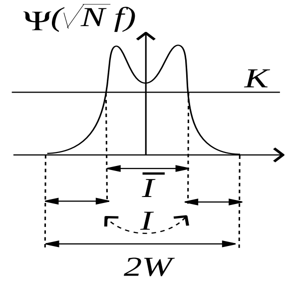

Figure 5: One dimensional illustration of how the lower bound on K from eqn. (24) is obtained by bounding the integrals that define K in eqn. (22).

3.4 Compression Efficiency vs. Complexity at the Nodes

Motivated by how much below capacity is $R(D)$ in (25), we are prompted to consider a special case: joint compression only along routes consisting of straight lines, instead of joint compression over the entire network (as illustrated in Fig. 6). The amount of traffic generated will be clearly higher, since now we are ignoring all correlations across lines. But the scheme is inherently less complex, and therefore of interest as well.

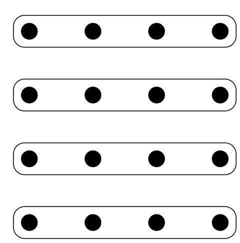

Figure 6: Partition of the network into independent rows. Nodes re-encode their messages only if they are in the same row—across rows, data is routed without any form of processing. Note that the rows are represented on a regular grid for simplicity [cf. Section 2.2].

Let $R^{IR}(D)$ denote the rate/distortion function of the data generated by the whole network, but under the constraint that joint compression is performed only across rows, ignoring dependencies along columns. Clearly we must have

$$R^{IR}(D) \leq \sqrt{N}\ R(D) = O(\sqrt{N}),$$

since in this (necessarily loose) upper bound we pretend that each row produces as much information as the entire network does, and assuming the average per-sample distortion is kept constant. This is interesting because we still get the growth rate of $R^{IR}(D)$ to match that in the capacity expression of eqn. (1). Therefore, we see that we can sacrifice some compression efficiency to gain a potentially big reduction in complexity, by having each node process at most $\sqrt{N}$ samples instead of N. Note also that besides compression efficiency, what we lose as well is the ability to reduce the average per-sample distortion by keeping D fixed and increasing N, since this results in an extra logarithmic term multiplying $\sqrt{N}$ in the amount of traffic generated, thus violating the capacity constraint.

4. CONCLUSIONS

In recent work on the transport capacity of large-scale wireless networks, it has been established that the per-node throughput of these networks vanishes as the number of nodes becomes large. This result poses a serious challenge for the design of such networks—some have even argued that large networks are *not* feasible, precisely because of this reason [9]. Previous work however pointed out that, in the context of sensor networks, the amount of information generated by each node is not a constant, but instead decays as the density of sensing nodes increases—this was illustrated with an example based on the transmission of samples of a Brownian process, with arbitrarily low distortion (even with vanishing per-node throughput), by means of using distributed source coding techniques [12].

In this work we have shown an alternative approach to work around the vanishing per-node throughput constraints of [9]. This new approach is not based on distributed coding techniques, but instead is based on the use of classical source codes combined with suitable routing algorithms and re-encoding of data at intermediate relay nodes. To the best of our knowledge, these are the first results in which interdependencies between routing and data compression problems are captured in a system model that is also analytically tractable. And a key (and enabling) step in our derivation was the construction of a family of spatial processes satisfying some fairly mild (and easily justifiable from a physical point of view) regularity conditions, for which we were able to show that the amount of data generated by the sensor network is well below its transport capacity. This provides further evidence that large-scale multi-hop sensor networks are perfectly feasible, even under the network model considered in [9].

APPENDIX

A. ENTROPY AND CODING

The entropy of a random variable X with probability mass function $P(x)$ is defined as:

$$\mathcal{H}(X) = E\{-\log_2 P(x)\} = -\sum_x P(x)\log_2 P(x). \qquad (26)$$

For multivariate random variables the generalization is straightforward:

$$\mathcal{H}(X_1,\ldots,X_n) = E\{-\log_2 P(x_1,\ldots,x_n)\}. \qquad (27)$$

Note also that, the entropy of a vector can be decomposed according to the so called chain rule, resulting from the iterated application of the chain rule for probability $P(x,y) = P(x|y)P(y)$:

$$\begin{aligned}\mathcal{H}(X_1,\ldots,X_n) &= \mathcal{H}(X_1) + \mathcal{H}(X_2|X_1) + \ldots \qquad (28)\\ &+ \mathcal{H}(X_n|X_1,\ldots,X_{n-1}) \le \sum_i \mathcal{H}(X_i)\end{aligned}$$

which was used in Figure 2. The importance of the definition of entropy lies in the fact that it provides a very accurate answer to the following question regarding discrete data sources (i.e. sources producing symbols drawn form a discrete alphabet):

> What is the minimum number of bits necessary to represent the data from a discrete source so that they can be reconstructed without distortion?

The answer is given by the the following theorem [5, Ch.5]:

THEOREM 1. *The expected length L of any instantaneous D-ary code for a random variable X is:*

$$\mathcal{H}(X) \le L < \mathcal{H}(X) + 1 \qquad (29)$$

The proof of the theorem is based on the existence of a coding technique, Huffman coding, is known to achieve the entropy within one bit if one symbol is encoded and, if multiple symbols are encoded together, the efficiency of Huffman coding tends to be 100% [5]. This theory cannot be directly generalized to handle the case of *analog sources* because their entropy is infinite even if the signal is a discrete-time sequence. In fact, the source signal can take any real value so that each sample still requires infinite precision to be represented exactly. However, once a certain level of distortion is accepted, the minimum number of bits necessary to represent the source can be calculated just as rigorously as in the case of discrete sources, resorting to the parallel theory for analog sources which is called *Rate Distortion Theory*.

B. RATE DISTORTION THEORY IN A NUTSHELL

Rate Distortion is based on the seminal contribution of Claude Shannon who tried to provided a theoretical framework for the representation of a continuous source through discrete symbols [13]. Suppose a memoryless continuous source produces a random sample S with density $p(S)$: the quantization problem boils down to representing S through discrete values $\hat{S}$ so that, if our measure of the distortion is $d(S,\hat{S})$, our mapping $S \to \hat{S}$ is such that $E\{d(S,\hat{S})\} \le D$.

To quantify how many bits are needed to represent $\hat{S}$, in his 1959 paper Shannon defined the so called *rate distortion function*:

$$R(D) = \min_{p(\hat{S}|S):E\{d(S,\hat{S})\}\le D} I(S,\hat{S}) \qquad (30)$$

where:

$$I(S,\hat{S}) = \sum_{\hat{S}} \int p(S,\hat{S}) \log \frac{p(S,\hat{S})}{p(S)P(\hat{S})}\, dS \qquad (31)$$

is the average mutual information between S and $\hat{S}$. In the same paper he proved the following two fundamental theorems:

THEOREM 2. *The minimum information rate necessary to represent the output of a discrete-time, continuous-amplitude memoryless Gaussian source based on a mean-square distortion measure per symbol (single letter distortion measure) is:*

$$R_g(D) = \begin{cases} \frac{1}{2}\log(\sigma_S^2/D) & \textit{if } 0 \le D \le \sigma_S^2, \\ 0 & D > \sigma_S^2. \end{cases} \qquad (32)$$

where σ_S^2 is the variance of the Gaussian source output.

THEOREM 3. *There exists an encoding scheme that maps the source output into code words such that for any given distortion D, the minimum rate $R(D)$ bits per symbol (sample) is sufficient to reconstruct the source output with an average distortion that is arbitrarily close to D.*

The implication of the two theorems above is that $R(D)$ is the minimum number of bits that can represent S within the prescribed mean-square error if the source is Gaussian. In other words any discrete code $\hat{S}$ that represents a Gaussian source S with a meanquare distortion $\leq D$ has

$$\mathcal{H}(\hat{S}) \geq R_S(D), \tag{33}$$

and the lower bound is asymptotically achievable. The following important theorem, proven by Berger in 1971 [2], generalizes the results by Shannon:

THEOREM 4. *The rate-distortion function of a memoryless, continuous -amplitude source with zero mean and finite variance σ_S^2 with respect to the mean-square-error distortion measure is upper-bounded as:*

$$R(D) \leq \frac{1}{2}\log(\sigma_S^2/D) \qquad \text{if } \ 0 \leq D \leq \sigma_S^2. \tag{34}$$

Berger's theorem implies that the Gaussian source is the one the requires the maximum encoding rate, if the distortion function is the MSE. Hence, if our distortion metric is the mean-square error, the case of a Gaussian source has to be seen as a worse case scenario.

The theorems above have been extended to multivariate sources which are analogous to the ones we consider in this paper. In particular, the so called *inverse water-filling* result used in Section 3 is the direct generalization of Theorem 2 to the multivariate Gaussian source.

3. REFERENCES

[1] R. Ahlswede, N. Cai, S.-Y. R. Li, and R. W. Yeung. Network Information Flow. *IEEE Trans. Inform. Theory*, 46(4):1204–1216, 2000.

[2] T. Berger. *Rate distortion Theory*, Prentice Hall, Englewood Cliffs, NJ.

[3] J. H. Conway and N. J. A. Sloane. *Sphere Packings, Lattices and Groups*. Springer Verlag, 3rd edition, 1998.

[4] T. Cormen, C. Leiserson, and R. Rivest. *Introduction to Algorithms*. MIT Press, 1990.

[5] T. M. Cover and J. Thomas. *Elements of Information Theory*. John Wiley and Sons, Inc., 1991.

[6] A. Gersho and R. M. Gray. *Vector Quantization and Signal Compression*. Kluwer Academic Publishers, 1992.

[7] U. Grenander and G. Szegő. *Toeplitz Forms and their Applications*. University of California Press, 1958.

[8] P. Gupta and P. R. Kumar. Critical Power for Asymptotic Connectivity in Wireless Networks. In W. M. McEneany, G. Yin, and Q. Zhang, editors, *Stochastic Analysis, Control, Optimization and Applications: A Volume in Honor of W. H. Fleming*. Birkhauser, 1998.

[9] P. Gupta and P. R. Kumar. The Capacity of Wireless Networks. *IEEE Trans. Inform. Theory*, 46(2):388–404, 2000.

[10] H. B. Keller Numerical Methods For Two-Point Boundary-Value Problems. *Dover Publications Inc.*, 1992.

[11] S. S. Pradhan and K. Ramchandran. Distributed Source Coding: Symmetric Rates and Applications to Sensor Networks. In *Proc. IEEE Data Compression Conf. (DCC)*, Snowbird, UT, 2000.

[12] S. D. Servetto. Lattice Quantization with Side Information. In *Proc. IEEE Data Compression Conf. (DCC)*, Snowbird, UT, 2000. An extended journal version, with title *Quantization with Side Information: Lattice Codes, Asymptotics, and Applications in Wireless Networks*, has been submitted for publication, and is available from `http://people.ece.cornell.edu/servetto/`.

[13] C. E. Shannon, Coding Theorems for a Discrete Source With a Fidelity Criterion *Institute of Radio Engineers, International Convention Record*, Vol. 7 (Part 4), pp. 142–163, 1959.

[14] V. A. Vaishampayan, N. J. A. Sloane, and S. D. Servetto. Multiple Description Vector Quantization with Lattice Codebooks: Design and Analysis. *IEEE Trans. Inform. Theory*, 47(5):1718–1734, 2001.

[15] Q. Zhao and M. Effros. Optimal Code Design for Lossless and Near Lossless Source Coding in Multiple Access Networks. In *Proc. Data Compression Conf. (DCC)*, Snowbird, UT, 2001.