ACTUATOR NETWORKS: DISTRIBUTED EVALUATION OF CAUSAL EFFECT

Mark Coates and Garrick Ing

McGill University
Department of Electrical and Computer Engineering
3480 University St
Montreal, Quebec H3A 2A7 Canada.

ABSTRACT

We study sensor-actuator networks, extensions of sensor networks that consist of nodes that both monitor and interact with the environment. In particular, we focus on the evaluation of average causal effect within such networks. We describe a distributed algorithm that enables individual actuator nodes to determine the probable consequences of local action on the global environment and hence decide if such action is conducive to achieving the aims of the network. Our approach represents the relationship between actuation and sensor measurements using a causal graph, and applies a distributed expectation-maximization algorithm to estimate the average causal effect of actuation. We evaluate the effectiveness of our approach through simulations that examine the benefits of including side-information regarding possible event outcomes.

1. INTRODUCTION

Sensor networks have been successfully deployed for passively monitoring environments [1, 2], but there has been relatively little work in developing networks that interact with the environment. Wireless sensor and actuator networks (WSANs) represent an important extension, allowing nodes within the network to make autonomous decisions and perform actions (actuation) in response to sensor measurements and shared information. The potential applications of such WSANs are widespread, including agricultural maintenance and localized delivery of medication.

Causal assessment is an important step in the development of WSANs, enabling us to determine whether an actuation has an impact upon the monitored system, and whether it is positive or negative. The formulation of distributed treatment plans in the medical or agricultural context relies on an understanding of how the environment responds. We concern ourselves primarily with the problem of estimating the causal effect on system response when the actuation is effective (the treatment is delivered to the environment). Our focus is on developing a distributed algorithm that reduces communication requirements but maintains inferential accuracy.

Section 2 of the paper details the estimation problem we address and reviews the concept of average causal effect. Section 3 describes a graphical model of the causal system and a potential response formulation. Section 4 explains the distributed procedure for estimating the average causal effect. Section 5 examines communication requirements of the estimators, and Section 6 describes simulation performance. Finally, Section 7 makes concluding remarks and indicates future research directions.

2. PROBLEM STATEMENT

We consider a WSAN as depicted in Figure 1, where actuation at a given node has a localized effect. We model the system using the causal graph depicted in Figure 2. In this model, there is a global distribution on latent causal factors; at each sensor, there is an independent realization of these factors. For the j-th sensor, there is a binary variable Z_j, assumed known in our model, that indicates local actuation. We let $z \in \{z_0, z_1\}$ represent the value of Z, where z_1 indicates that actuation was performed, and z_0 that it was not. We interpret actuation as the application of some "treatment" to the system. The unobserved binary event D_j indicates reception of treatment, and Y_j indicates the response. The variables $d_j \in \{d_0, d_1\}$ and $y_j \in \{y_0, y_1\}$ represent, respectively, the values assumed by D_j and Y_j, with d_1 indicating that treatment was received, and y_1 indicating a positive observed response. The values d_0 and y_0 are the negations of their respective counterparts. The observed local measurements, X_j and W_j, are modeled as i.i.d., conditioned on D_j and Y_j, respectively. We leave the nature of these measurements, and the conditional probability distributions that relate them to D_j and Y_j unspecified, but assume that these distributions are known (or modeled) when our estimation techniques are applied. We denote by U_j all characteristics that externally influence the values of D_j and Y_j. In general, U_j will comprise several random variables, both discrete and continuous. These U_j are as-

0-7803-8622-1/04/$20.00 ©2004 IEEE

sumed to be local realizations of variables described by a global distribution g.

The graphical model depicted in Figure 1 represents independence assumptions about the joint probability distribution $p(\mathbf{z}, \mathbf{d}, \mathbf{y}, \mathbf{w}, \mathbf{x}, \mathbf{u})$. It specifies that the actuation Z does not *directly* affect the response Y, but only through the reception of treatment D. The model also asserts that actuation does not depend on the latent factors that determine treatment reception and system response.

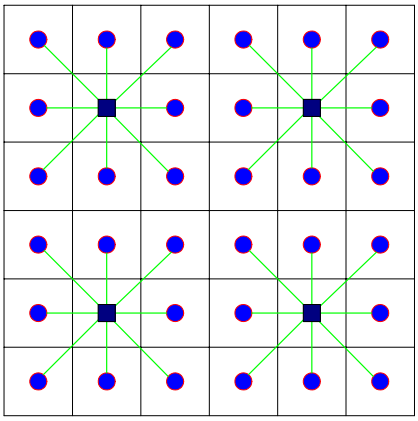

Fig. 1. The model for the sensor-actuator network. Solid squares represent actuation-capable nodes; circles represent sensor nodes. Lines between nodes describe the localized actuation model, i.e., actuation at a given node can only affect measurements at the sensors to which it is connected. Nodes are assumed to be capable of communicating only with nodes in neighbouring grid squares.

We address the problem of estimating the causal effect of the system. In particular, we focus on the average causal effect, as defined by Holland [3], of treatment reception upon response:

$$ACE(D \rightarrow Y) = E_u[p(y_1|d_1, u) - p(y_1|d_0, u)]. \quad (1)$$

The average causal effect is the expected difference between the probability of the response being positive when treatment is received and when it is not received (where the expectation is over the latent causal factors). Denoting our data across all sensors by $\mathcal{X}$, our goal in this paper is to estimate $ACE(D \rightarrow Y)$ by maximizing the likelihood function $\mathcal{L}(\mathcal{X}|ACE(D \rightarrow Y))$. A simple modification of the approach permits assessment of the indirect causal effect of actuation upon response.

Related work and Contribution

The estimation of causal effect using Bayesian graphical analysis techniques has been explored for many years [4–8]. In the context of clinical studies, the estimation of causal effect and average causal effect has been examined in [3, 9]. Graphical techniques and distributed EM algorithms have been applied recently in sensor networks for different estimation problems [10, 11]. In [12], we described an expectation-maximization algorithm for estimating average causal effect that operated by forming an overlay tree-topology network for aggregation. Therein we assumed that the events Z, D and Y could be directly observed. The major contribution of this paper is the identification of a distributed EM algorithm for performing distributed causal analysis across a sensor-actuator network that cannot directly observe the events D and Y.

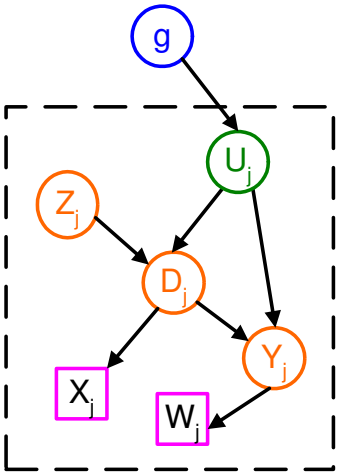

Fig. 2. Graphical model for the variables affecting measurement at an individual sensor node. g represents the global distribution of the latent causal factors; U_j represents a local realization of these factors; Z_j is a binary local actuation variable; D_j is a binary variable indicating whether treatment was received; Y_j is a binary variable indicating positive or negative (null) response. The observed measurements, X_j and W_j, depend only on the state variables D_j and Y_j respectively, according to a specified probability distribution.

3. POTENTIAL RESPONSE MODEL

Our estimation algorithm is based upon the observation in [6] that the latent factors U_j can be replaced by a single discrete and finite variable CR_j, resulting in an equivalent causal model for all manipulations of Z, D, and Y (and observations dependent upon them). The variables CR describe the impact of latent factors on the mappings from $Z \rightarrow D$ and $D \rightarrow Y$; this contrasts with the original U which describes the impact on the actual values of D and Y. Such a variable has been called a *response variable* [5]; its states correspond to the *potential response* vectors in [4].

Since Z, D and Y are binary variables, each mapping can be described using 4 states, meaning that the state vari-

able CR has 16 values. We can use two four-valued variables, C and R, which describe the individual mappings $Z \rightarrow D$ and $D \rightarrow Y$, respectively. The value of CR is then determined by the combination of these two values.

The variable C governs the mapping $Z \rightarrow D$ as follows.

$$d = F_D(z, c) = \begin{cases} d_0 & \text{if } c = 0 \\ d_0 & \text{if } c = 1 \text{ and } z = z_0 \\ d_1 & \text{if } c = 1 \text{ and } z = z_1 \\ d_1 & \text{if } c = 2 \text{ and } z = z_0 \\ d_0 & \text{if } c = 2 \text{ and } z = z_1 \\ d_1 & \text{if } c = 3 \end{cases} \quad (2)$$

The variable R determines the mapping from treatment to response.

$$y = F_Y(d, r) = \begin{cases} y_0 & \text{if } r = 0 \\ y_0 & \text{if } r = 1 \text{ and } d = d_0 \\ y_1 & \text{if } r = 1 \text{ and } d = d_1 \\ y_1 & \text{if } r = 2 \text{ and } d = d_0 \\ y_0 & \text{if } r = 2 \text{ and } d = d_1 \\ y_1 & \text{if } r = 3 \end{cases} \quad (3)$$

Denoting the states of CR using the notation $cr(s, t)$ with $0 \leq s, t \leq 3$, we can express the average causal effect as:

$$ACE(D \rightarrow Y) = \sum_i [p(cr(s, 1)) - p(cr(s, 2))]. \quad (4)$$

This is equivalent to the difference, for recipients of the treatment, between the probability of a positive effect and that of a negative effect. Note that the global distribution g is equivalent to $p(cr)$.

4. ESTIMATION ALGORITHM

We assume that there are N sensors and each sensor j makes a set of K measurements. We denote the i-th measurements of the j-th sensor w_{ij} and x_{ij}; these could be vectors and either discrete-valued or continuous. The associated events are d_{ij} and y_{ij}; we assume that z_{ij} is known by the sensor-actuator network. We estimate the average causal effect by applying an expectation-maximization (EM) algorithm [13] across the graphical model depicted in Figure 2. In this section, we begin by describing a partially-distributed version of the EM algorithm; we then describe an extension to a version that is more extensively distributed. The EM algorithm generates an estimate $\widehat{g}$ that (locally) maximizes the likelihood function $\mathcal{L}(\mathcal{X}|g)$. The estimate is used in equation 4 to determine $\widehat{ACE}(D \rightarrow Y)$.

The EM algorithm commences with an initial estimate g^0. It is an iterative algorithm with each iteration consisting of two steps: the expectation (E-) step and the maximization (M-) step. The E-step of the algorithm uses conventional message-passing techniques associated with Bayesian networks to determine the expected values of the state variables d, y, and cr [13]. At the m-th iteration of the algorithm, the distribution estimate is g^m. The E-step in this case involves the evaluation of the expected values of d_{ij} and y_{ij} and cr_{ij}. In fact, the critical step is the evaluation of cr_{ij}:

$$v_{ij}(s, t) = g^m(s, t) \sum_{d,y} \mathcal{I}(d, y|s, t, z_{ij}) p(x_{ij}|d) p(w_{ij}|y)$$

$$cr_{ij}^{m+1}(s, t) = \frac{v_{ij}(s, t)}{\sum_{s,t} v_{ij}(s, t)} \quad (5)$$

where $\mathcal{I}(d, y|s, t, z_{ij})$ is an indicator function that takes on the value 1 if (z_{ij}, d, y) is possible under the state $cr_{ij} = (s, t)$.

The maximization step at the m-th iteration involves the determination of the global distribution g^{m+1} which maximizes the likelihood of observing the cr_{ij}^{m+1} across the N sensor nodes. We model g as a multinomial distribution, so this maximization takes the form of an averaging of the expected cr_{ij} values. To calculate this average, we need to aggregate the cr_{ij}^{m+1} across the network (and divide by the total number of measurements). The resultant g^{m+1} is then distributed throughout the network. The algorithm is deemed to have converged when the change between g^m and g^{m+1} is sufficiently small. In this version of the algorithm, each calculation in the E-step is local to one of the sensor nodes (no inter-node communication is necessary). The M-step involves aggregation; this can be performed efficiently by constructing an overlay tree network. For this reason, we will call the algorithm we have just described the *tree-based EM* algorithm.

The rate of convergence of the algorithm can be improved (and the amount of communication reduced) if one employs a *distributed EM* algorithm, as described in [11]. In this algorithm, rather than constructing a tree, we construct a cyclical path through the network. We choose an initial estimate g_0 as before. At iteration $m + 1$ of the algorithm, one of the sensor nodes j receives an estimate g^m from the previous node in the path. It then performs the expectation as described above to calculate cr_{ij}^{m+1} for all of its measurements $i = 1, ..., K$. Finally, the node calculates

$$g^{m+1} = g^m + \frac{\sum_{i=1}^{K} \left[cr_{ij}^{m+1} - cr_{ij}^{m}\right]}{NK}, \quad (6)$$

and transmits this value on to the next sensor node in the path. In this algorithm, each node performs a local EM-step. The convergence properties and communication requirements of the two algorithms are discussed in Section 5.

5. COMMUNICATION REQUIREMENTS

In this section we briefly discuss the communication requirements of various approaches to the estimation of ACE in sensor-actuator networks. For concreteness, we focus on the specific case of the grid depicted in Figure 1 and measure communication cost in bit-hops. We assume Manhattan communication in the sense that a sensor can communicate in one hop to one of its 4 neighbours to the north, south, east or west. We consider a centralized estimation scheme where all data is relayed to a sink node at the top-left corner of the grid and compare its communication requirements to those of the decentralized schemes we outlined in the previous section.

Consider a grid of $N \times N$ sensors with K bits of measurements per sensor. In this case, the centralized algorithm will require that N^2 sensors transfer K bits to the sink node over a mean distance of $N-1$ hops. The communication requirements for the network are thus $\mathcal{O}(KN^3)$. Suppose that the tree-based EM algorithm takes M_t iterations to converge, that the distributed EM algorithm takes M_d iterations to converge, and that representation of the cr data-structure requires V bits. The tree-based EM algorithm requires that aggregation is performed across the network. The aggregation can be performed using a tree topology in which each node performs a partial aggregation of the transmissions from its children. At the source, the result is averaged and distributed throughout the tree. This process requires N^2 communications of V bits and is repeated M_t times, so the communication cost to the network is $\mathcal{O}(N^2 M_t V)$. In the distributed EM case, each sensor transmits its updated calculation one hop to the next sensor in the cycle. Thus N^2 sensors transmit V bits for M_d cycles, resulting in a communication cost of $\mathcal{O}(N^2 M_d V)$.

Compared to the centralized algorithm, both the tree-based and distributed EM algorithms have the advantage that communication is not concentrated anywhere in the sensor-actuator network (in the centralized case there is much heavier traffic near the sink node). In terms of average communication cost, there will be a reduction whenever $M_d V$ or $M_t V$ compares favourably with NK. In constructing this comparison, however, care must be taken to consider overhead. There is a communication overhead in the construction of an overlay network for transmitting data to the sink, for tree-based aggregation or for path-based communication in the distributed EM algorithm. These costs are likely to be comparable for the three schemes. However, the decentralized schemes require transmission of $\mathcal{O}(N^2 M)$ actual packets, whereas in the centralized scheme, the number of packets could range from $\mathcal{O}(N^2)$ to $\mathcal{O}(N^3)$ depending on packet size and K. There are overhead bits in each packet header, so this potential reduction can have a substantial impact.

6. SIMULATION RESULTS

In this section we describe the application of the algorithm to a simulated data set. We base our simulation on a distribution $p(z,d,y)$ which is known to give rise to an identifiable ACE ($= 0.55$); we use the distribution described in [9] and displayed in Table 1. In order to construct our data, we draw $[z,d,y]$ triples from this distribution for each sensor measurement. The observed measurements x_{ij} and w_{ij} are then drawn from Gaussian mixture distributions with component means dependent on the state values d_{ij} and y_{ij} respectively. In the example we discuss, these distributions are:

$$\begin{aligned} p(x|d=0) &= 0.8\mathcal{N}(0,1) + 0.2\mathcal{N}(0,3) \\ p(x|d=1) &= 0.8\mathcal{N}(2,1) + 0.2\mathcal{N}(2,3) \\ p(w|y=0) &= 0.7\mathcal{N}(0,2) + 0.3\mathcal{N}(0,5) \\ p(w|y=1) &= 0.7\mathcal{N}(4,2) + 0.3\mathcal{N}(4,5) \end{aligned} \quad (7)$$

where $\mathcal{N}(\mu,\sigma)$ indicates the normal distribution with mean μ and variance σ^2.

Table 1: Simulation population fractions (distribution) resulting in an identifiable ACE = 0.55.

z	d	y	$p(z,d,y)$
0	0	0	0.275
0	0	1	0.0
0	1	0	0.225
0	1	1	0.0
1	0	0	0.225
1	0	1	0.0
1	1	0	0.0
1	1	1	0.275

We considered a sensor-actuator network similar to that depicted in Figure 1, in which there were 9 actuators and 9 sensors associated with each actuator (there was one sensor collocated with the actuator). The simulation consisted of ten events (actuations or non-actuations), assumed sufficiently spaced in time so that system responses could be modelled as independent of one another. Each sensor thus recorded 10 independent measurements, for a total of 810 measurements across the entire network. The tree-based EM and distributed EM algorithms were applied to this data in 100 separate trials. In one experiment (labelled "no side"), we considered the case where the algorithms considered all $[z,d,y]$ states to be viable. In a second experiment (labelled "side") we examined the performance when the algorithms were provided with side information specifying exactly which $[z,d,y]$ states were possible.

Table 2 summarizes the results of our simulations, presenting empirical bias and standard error, and the mean convergence rate over the 100 trials. We deemed that the algo-

rithms had converged when the L1-distance between the estimate at iteration k and iteration $k-1$ was less than 0.005. We terminated the algorithms after 80 iterations if convergence had not been achieved; this event was very rare in simulations. Particularly when there is no side information, the algorithms were observed to converge to different local maxima (different initializations were used for the two algorithms and the search behaviour is different), but in terms of ACE estimation, the performance is reasonably consistent, with a mean absolute discrepancy of 0.03 in ACE estimates over the 100 trials. The distributed algorithm is seen to converge faster on average than the tree-based algorithm; this is expected because of its more aggressive updating procedure. When there is no side information, the ACE estimates are biased, because in the estimates positive weight is assigned to cr states that have no support in the generating model. The weights of the cr states with support in the generating model are thus primarily underestimated; those in $cr(\cdot, 2)$ are impacted more severely than $cr(\cdot, 3)$, resulting in underestimated ACEs. The inclusion of the side information results in a dramatic improvement in estimator performance. Figure 3 depicts histograms of the ACE estimates obtained with and without side information; inclusion eliminates almost all the bias and dramatically reduces variance.

Table 2: Simulation results for 100 trials; comparison between empirical bias and standard deviation and mean convergence rates of the two algorithms with and without side information.

Method	Bias	Std	Conv. (cycles)
Tree EM (no side)	-0.0837	0.0642	62
Dist. EM (no side)	-0.0633	0.0548	50.1
Tree EM (side)	0.0071	0.0290	31.2
Dist EM (side)	0.0050	0.0258	23.2

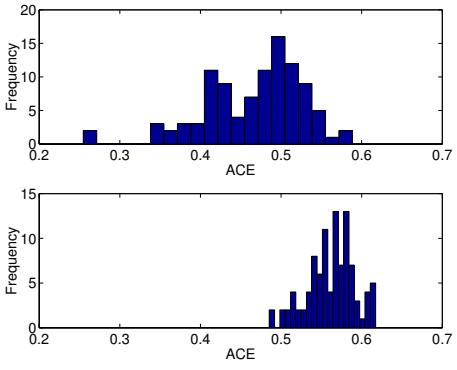

Fig. 3. Estimation results from 100 trials of the simulation of Section 6 using the distributed EM algorithm to estimate causal effect. Top panel: No side information. Bottom panel: Side information (the possible [Z,D,Y] states).

7. CONCLUSIONS

In agricultural and medical applications, the role of sensor networks can develop beyond monitoring to active control or localized treatment. This involves decision-making, an arena in which causal analysis plays a vital role. We have presented a distributed algorithm for a specific example of causal analysis using a sensor network, but there remains much to explore and develop.

8. REFERENCES

[1] K.A. Delin, R.P. Harvey, N.A. Chabot, S.P. Jackson, M. Adams, D.W. Johnson, and J.T. Britton, "Sensor web in Antarctica: developing an intelligent, autonomous platform for locating biological flourishes is cryogenic environments," in *Proc. 34th Lunar and Planetary Science Conf.*, Houston, TX, Mar. 2003.

[2] D. Myung, B. Duncan, D. Malan, M. Welsh, M. Gaynor, and S. Moulton, "Vital dust: Wireless sensors and a sensor network for real-time patient monitoring," in *Proc. 8th New England Reg. Trauma Conf.*, Burl., MA, Nov. 2003.

[3] P.W. Holland, *Sociological Methodology*, chapter Causal inference, path analysis, and recursive structural equations models, pp. 449–484, American Sociological Association, Washington D.C., 1988.

[4] D. Rubin, "Bayesian inference for causal effects: the role of randomization," *Annals of Statistics*, vol. 7, pp. 34–58, 1978.

[5] A. Balke and J. Pearl, "Counterfactual probabilities: computational methods, bounds and applications," in *Proc. Conf. Uncertainty in Artificial Intelligence*, Seattle, WA, 1994, pp. 46–54, Morgan-Kaufman.

[6] J. Pearl, "From Bayesian networks to causal networks," in *Proc. UNICOM Seminar Adaptive Computing and Information Processing*, London, England, 1994.

[7] G. Imbens and D. Rubin, "Bayesian inference for causal effects in randomized experiments with non-compliance," *Annals of Statistics*, vol. 25, pp. 305–327, 1997.

[8] J. Pearl, *Bayesian Networks and Probabilistic Reasoning*, chapter From Bayesian networks to causal networks, pp. 1–31, Alfred Walter Ltd., London, 1995.

[9] D. M. Chickering and J. Pearl, "A clinician's tool for analyzing non-compliance," in *Proc. 13th Nat. Conf. Artificial Intelligence*, Portland, OR, 1996, pp. 1269–1276.

[10] V. Delouille, R. Neelamani, V. Chandrasekaran, and R. Baraniuk, "The embedded triangles algorithm for distributed estimation in sensor networks," in *Proc. IEEE Workshop Stat. Signal Processing*, St. Louis, MO, Sept. 2003.

[11] R. Nowak, "Distributed EM algorithms for density estimation and clustering in sensor networks," *IEEE Trans. Signal Proc.*, vol. 51, no. 8, pp. 2245–2253, Aug. 2003.

[12] M.J. Coates and I. Psaromiligkos, "Evaluating average causal effect using wireless sensor networks," in *Proc. IEEE ICASSP*, Montreal, Canada, May 2004.

[13] M.I. Jordan, Ed., *Learning in Graphical Models*, MIT Press, Cambridge, MA, 1999.