

A GRAPH THEORETIC APPROACH TO DATA INCEST MANAGEMENT IN NETWORK CENTRIC WARFARE *

Samuel P. McLaughlin
Department of Electrical
& Electronic Engineering
Univ. of Melbourne, Aust.
s.mclaughlin@iee.org

Robin J. Evans
National ICT Australia
Department of Electrical
& Electronic Engineering
Univ. of Melbourne, Aust.
r.evans@ee.mu.oz.au

Vikram Krishnamurthy
Department of Electrical
& Computer Engineering
Univ. of British Columbia.
vikramk@ece.ubc.ca

Abstract – This paper considers the problem of data incest management in decentralised estimation networks. All nodes consist of both a measurement sensor and a local fusion centre. Global state information is exchanged across communications channels with bounded variable delays. Particular attention is devoted to the underlying delay structure that causes data incest. The investigation employs an efficient graph theoretic representation of the time varying nature of these delays. Applying this framework a data incest management algorithm is presented and is complimented with a tool for identifying the necessary and sufficient condition for its use.

Keywords: Decentralised estimation, full interconnection, data incest management, Kalman Filter, Information Filter, Communications Architecture Graph

1 Introduction

In network centric warfare [1], sensor nodes typically communicate state estimates over a network with random delays. Due to the uncertain nature of the network delay, the possibility of reusing past information arises - this is referred to as data incest and can result in a significant bias of the state estimates. The design of the networks over which the sensor nodes communicate requires the careful consideration of the configuration of sensor nodes and their interconnections. This becomes vitally important when common information is shared between nodes, as is the case in a *decentralized estimation network* (DEN) that communicates global information between nodes. Figure 1 illustrates an elementary network topology that can lead to data incest or equivalently requires *data incest management* (DIM) to maintain optimality. If node *C* treats the information (state estimate) from node *B* as independent from node *A*, information from *A*

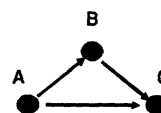


Fig. 1: Elementary network topology prone to data incest

will be duplicated in the fused information at node *C*. In this example, the detection and removal of incest is straightforward. As the size of network grows, and more information is shared, one needs a formal procedure to address the data incest problem.

Until recently, the area of target tracking was dominated by hierarchical structures [2]. In this setting, DIM amounts to little more than ensuring the prior information (common to all nodes) was not double counted. Even in recent decentralized architectures, which have obvious advantages in terms of robustness, the problem of data incest has been averted by only sharing local information between nodes [3].

There are number of arguments for sending global information in network centric warfare. Many legacy systems are designed to output global information directly onto the network. If the best information is available on the network, agents that rely on that information have immediate access to it without extensive preprocessing. This has lead to recent attempts to address the data incest problem [4], [5].

In this paper we focus on the problem of data incest in a fully interconnected DEN used to estimate a static state. Using a graph theoretic formulation of the DEN we are able to quantify the level of data incest between packets of information based exclusively on

*This work is supported by Defense Science Technology Organization (DSTO) through TDFL, the Center for Sensor Signal and Information Processing (CSSIP) and the University of Melbourne.

the network topology and delay characteristics. This allows for presentation of a successful DIM algorithm for a certain class of decentralised estimation problem and identifies necessary and sufficient conditions for its use. These results are presented from a global perspective. Implementing them at local nodes is beyond the scope here, but will follow directly from this work.

Section 2 formulates the decentralised estimation problem. Section 3 introduces a novel framework for the analysis of data incest. Section 4 synthesises the previous two sections and Section 5 applies the subsequent analysis to the problem of data incest management. The paper is concluded in Section 6.

2 Problem Formulation

The problem addressed in this paper is to analyse and overcome the problem of data incest in a DEN used to estimate the state of a static variable $\mathbf{x}_0 \in \mathbb{R}^r$. Any DEN formulation requires the identification of three key properties: network topology, communication protocol and requirements of the local fusion strategy at each node.

2.1 Network Topology

Each node in the DEN is defined as a *sensor fusion node* (SFN) as it has the capacity to combine local information with information received from other remote SFNs. Let $N \in \mathbb{N}^+$ be the number of SFNs in a DEN, let $\mathcal{N} = \{1, \dots, N\}$ and let $K = \{1, \dots, k\}$ be the time sequence history up to k . The DEN is to operate in an environment plagued by variable delays between SFNs; similar in principle to a packet switched network [6].

Assumption 2.1 Let $\mathbf{d}_{n,m}(k) \in \mathbb{N}^+$ be the size of the communication delay from n to m in a DEN. The bounds on the size of the delay are $1 \leq \mathbf{d}_{n,m}(k) \leq \mathcal{D}$.

An efficient referencing technique that distinguishes different SFNs at different times is achieved by a *reference point* (RP), whereby $[n, j]$ refers to the time j of an event at SFN n .

2.2 Communication Protocol

All inputs and outputs related to the fusion centre at $[n, k]$ are outlined in Figure 2. The same integer time reference k applies to the local measurement, the processing of this and other inputs immediately following and the subsequent output.

2.2.1 Outputs

The *output packet* from the fusion centre at $[n, k]$, denoted $\theta(n, k)$, provides the input to a *main operations centre* (MOC) which services local weapons systems etc [7]. The output data should resemble an optimal posterior density of $\mathbf{x}_0$ which encapsulates the local static state estimate.

Assumption 2.2 The output packet $\theta(n, k)$ is broadcast to all other SFNs, subject to Assumption 2.1.

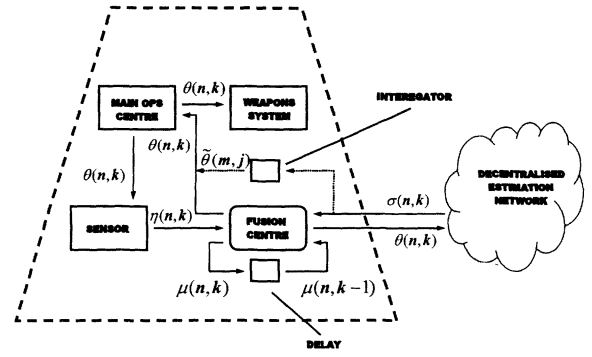


Fig. 2: Relationship between Inputs and Outputs to the Sensor Fusion Node at $[n, k]$.

2.2.2 Inputs

Denoted $\eta(n, k)$, the primary local input is the *measurement packet*. The timing of measurements is synchronised across all SFNs. The remaining local input $\mu(n, k)$, is known as the *storage information*.

Remark 2.1 The local availability of the previous output $\theta(n, k-1)$ dictates the elements of the storage.

The only external input at $[n, k]$ is the set of all remote output packets arriving at SFN n in the interval $k-1$ to k . For $\mathbf{d}_{m,n}(j) = k-j$, we define $[m, j]$ a *corresponding RP* (CRP) of $[n, k]$.

Definition 2.1 Let Ψ be the set of all CRPs of $[n, k]$. The remote set, denoted

$$\sigma(n, k) := \{\theta(m, j) : \forall m, n \in \mathcal{N}, m \neq n, j \in K, [m, j] \in \Psi\}, \quad (1)$$

is the set of remote output packets arriving at $[n, k]$.

The complete *input set* is summarised

$$\phi(n, k) = \{\mu(n, k), \sigma(n, k), \eta(n, k)\}. \quad (2)$$

2.3 Local Fusion Strategy

Each local fusion centre $[n, k]$ provides

$$\theta(n, k) = f(\phi(n, k)), \quad (3)$$

where $f(\cdot)$ is referred to as the *data incest management* (DIM) algorithm. The direct measurement¹ of the static state $\mathbf{x}_0 \in \mathbb{R}^r$ at $[n, k]$ provides

$$\mathbf{y}_n(k) = \mathbf{x}_0 + \mathbf{v}_n(k) \quad (4)$$

¹The results presented here generalise to the use of dynamic measurement matrices $\mathbf{H}_n(k) \in \mathbb{R}^{p \times r}$ for $\mathbf{y}_n(k) \in \mathbb{R}^p$

where the independence of the noise vector $\mathbf{v}_n(k) \sim \mathcal{N}(0, \mathbf{R})$ yields $\mathbf{E}\{\mathbf{v}_n(j)\mathbf{v}_n^T(k)\} = \mathbf{R}\delta_{jk}$ and $\mathbf{E}\{\mathbf{v}_n(k)\mathbf{v}_m^T(k)\} = \mathbf{R}\delta_{nm}$ respectively. The prior density $p(\mathbf{x}_0) \sim \mathcal{N}(\mathbf{x}_0; \hat{\mathbf{x}}_0, \mathbf{P}_0)$ is assumed known at all SFNs.

We define the *static optimal decentralised posterior density* (SODPD), denoted

$$p(\mathbf{x}_0|\mathbf{Z}_n^k) \sim \mathcal{N}(\mathbf{x}_0; \hat{\mathbf{x}}_{n,S}(k), \mathbf{P}_{n,S}(k)), \quad (5)$$

to be the posterior density of $\mathbf{x}_0$ conditioned on the *optimal decentralised set* (ODS) $\mathbf{Z}_n^k \subset \mathbf{Y}^k$ where $\mathbf{Y}^k := \{\mathbf{y}_m(j) : \forall m \in \mathcal{N}, \forall j \in K\}$ is the complete global measurement history. The ODS itself is based on both the delay characteristics and the success of the fusion strategy. The estimate $\hat{\mathbf{x}}_{n,S}(k) = \mathbf{E}\{\mathbf{x}_0|\mathbf{Z}_n^k\}$ is considered optimal if it is equivalent to the minimum variance estimate that would be computed at $[n, k]$, if all measurement packets $\mathbf{y}_m(j) \in \mathbf{Z}_n^k$ are sent directly to a standard Kalman filter [8] at $[n, k]$.

The following key assignments:

$$\mathbf{B}_{n,S}(k) := [\mathbf{P}_{n,S}(k)]^{-1} \quad (6)$$

$$\hat{\mathbf{a}}_{n,S}(k) := [\mathbf{P}_{n,S}(k)]^{-1}\hat{\mathbf{x}}_{n,S}(k). \quad (7)$$

constitute the two elements of the output packet

$$\theta(n, k) = (\hat{\mathbf{a}}_{n,S}(k), \mathbf{B}_{n,S}(k)). \quad (8)$$

Remark 2.2 It is the parameters of the SODPD which are ultimately used by the local MOC. However, the output packet comprises slight variations of these given by (6) and (7). Switching between the two is straightforward.

Definition 2.2 A SFN which successfully fuses the input set $\phi(n, k)$ to generate (8) is defined as an ideal fusion node (IFN). If each SFN in a DEN operates as an IFN, we define this to be an ideal DEN (IDEN).

Although the following assumption is not realistic for implementations at local SFNs, it greatly simplifies the global result provided.

Assumption 2.3 Up-to-date knowledge of all delay elements is available at all SFNs. These delays are constrained to ensure an IDEN is achievable based on Assumption 2.2.

3 Communication Architecture Graph (CAG)

3.1 Formulation of the CAG

A graph [9] is defined as a pair $\mathcal{G} = (\mathbf{V}, \mathbf{E})$ comprising a set of vertices $\mathbf{V}$ and edges $\mathbf{E} \subset \mathbf{V} \times \mathbf{V}$. An edge $(\alpha, \beta) \in \mathbf{E}$ is called a *directed edge* if $(\beta, \alpha) \notin \mathbf{E}$. If all edges of a graph $\mathcal{G}$ are directed, we call $\mathcal{G}$ a *directed graph*. Constraints imposed on a directed graph provides a dedicated graph for investigating the problem of decentralized estimation. These include:

- Each vertex of the new graph represents a unique RP in the DEN
- Each edge in the new graph represents communication between RPs
- Causality of communication between distinct nodes must be reflected by the edges
- Each SFN has access to its own previous output
- The impact of the communication delay between distinct nodes must be reflected by the edges

Definition 3.1 A communications architecture graph (CAG) is defined as a directed graph $\mathcal{G}_k(N) = (\mathbf{V}, \mathbf{E})$ over the finite set of vertices

$$\mathbf{V} := \{[n, i] : \forall n \in \mathcal{N}, \forall i \in K\}, \quad (9)$$

where each vertex $v = [\bar{v}, \underline{v}] \in \mathbf{V}$ represents an RP. The set of directed edges $\mathbf{E} \subset \mathbf{V} \times \mathbf{V}$ satisfies

$$\mathbf{E} = \{(v_o, v_t) : v_o \in \mathbf{V}, v_t \in \mathbf{T}(v_o)\} \quad (10)$$

where $\mathbf{T}(v_o) \subset \mathbf{V}$ such that $v \in \mathbf{T}(v_o)$ if, and only if,

1. $\underline{v} > \underline{v}_o$
2. $\underline{v} = \begin{cases} \underline{v}_o + 1 & \text{if } \bar{v}_o = \bar{v} \\ \underline{v}_o + d & \text{if } \bar{v}_o \neq \bar{v} \end{cases}$

where $d = d_{\bar{v}_o, \bar{v}}(\underline{v}_o)$ subject to Assumption 2.1.

Figure 3 illustrates the matrix like arrangement of the vertex set $\mathbf{V}$ (dots) and the subset of three directed edges from the set $\mathbf{E}$ (arrows) of $\mathcal{G}_4(3)$. This structure is typical of $\mathcal{G}_k(N)$ where each row represents a node $n \in \mathcal{N}$ and each column represents a time $j \in K$. The number of columns traversed, or equivalently the arrow length, reflects the size of the delay.

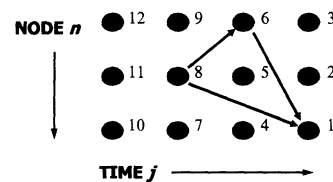


Fig. 3: Communications Architecture Graph

Definition 3.2 A path, denoted $\langle v_1, v_n \rangle = \{v_1, \dots, v_n\}$, is a sequence of adjacent vertices (joined by a directed edge) such that $(v_1, v_2), (v_2, v_3), \dots, (v_{n-1}, v_n)$ are all edges on $\mathcal{G}$. The path length, $n - 1$, equals the number of sequential edges.

A unique form of path is needed to delineate between paths that traverse different rows and those that don't.

Definition 3.3 A path $\langle v_i, v_j \rangle$ that traverses a sequence of vertices $V_p = \{v_i, \dots, v_j\}$ is defined a link, denoted $[v_i, v_j]$, if $\bar{v}_i \neq \bar{v}_j$ and $\bar{v}_j \neq \bar{v}_l$ for any $v_l \in V_p$.

3.2 Translator Matrix

Let $\mathcal{G}_k(N)$ be a CAG with $\mathcal{V} = |V|$ vertices. Assigning indices to vertices in a CAG is based on their relative location on $\mathcal{G}_k(N)$. The numbering in Figure 3 generalises for $\mathcal{G}_k(N)$ to

$$V := \{v_i : i = \mathcal{V} - (\bar{v}_i - 1) \times k - (\bar{v}_i - 1), \forall v_i \in \mathcal{V}\}, \quad (11)$$

where v_i represents $[\bar{v}_i, \underline{v}_i]$ in the corresponding DEN.

Definition 3.4 Let $\mathcal{G}_k(N) = (V, E)$ be a CAG with vertices $v_i \in V$ indexed by (11). We denote the strictly upper triangular translator matrix $M \in \mathbb{R}^{r\mathcal{V} \times r\mathcal{V}}$

$$M = \begin{bmatrix} m(1,1) & \cdots & m(1,\mathcal{V}) \\ \vdots & \ddots & \vdots \\ m(\mathcal{V},1) & \cdots & m(\mathcal{V},\mathcal{V}) \end{bmatrix}, \quad (12)$$

where $m(j,i)$ is either 1 or 0 depending on the adjacency between v_i and v_j .

3.3 Total Path Matrix

Based on M being the transpose of the well known adjacency matrix, standard results in graph theory [9] show that manipulation of M yields the number of paths between distinct vertices.

Lemma 3.1 Let $M \in \mathbb{R}^{r\mathcal{V} \times r\mathcal{V}}$ be the translator matrix for $\mathcal{G}_k(N)$. The total path matrix (TPM) $W \in \mathbb{R}^{r\mathcal{V} \times r\mathcal{V}}$, denoted

$$W := \sum_{q=1}^{k-1} M^q, \quad (13)$$

contains r -dimensional blocks $w(i,j) = \omega_{ji}I$. The total number of paths of any length $1 \leq q < k$ between distinct vertices v_j and v_i on $\mathcal{G}_k(N)$ is given by ω_{ji} .

4 Estimate Decomposition in Information Form

Before we can decompose the information form of the estimate we must first do so in Kalman form. We denote Z_m^{j*} the mask of Z_m^j such that

$$Z_m^j = Z_m^{j*} \cup y_m(j) \quad (14)$$

where $y_m(j)$ is referred to as the masking agent.

Lemma 4.1 The SODPD related to $[n,k]$ can be written as

$$p(x_0|Z_n^k) = \delta_2 p(x_0|Z_n^{k*}) p(y_n(k)|x_0) \quad (15)$$

where δ_2 is a normalising constant and $p(y_n(k)|x_0)$ is the likelihood density of the measurement $y_n(k) \in \mathbb{R}^p$ and $p(x_0|Z_n^{k*}) \sim \mathcal{N}(x_0; \hat{x}_n^*(k), P_n^*(k))$.

PROOF. Based on Bayes Theorem and the independence of each measurement we find

$$\begin{aligned} p(x_0|Z_n^k) &= p(x_0|Z_n^{k*}, y_n^k) \\ &= \delta_1 p(x_0, Z_n^{k*}, y_n^k) \\ &= \delta_1 p(y_n(k)|x_0, Z_n^{k*}) p(x_0, Z_n^{k*}) \\ &= \delta_2 p(x_0|Z_n^{k*}) p(y_n(k)|x_0) \end{aligned} \quad (16)$$

□

Based on (15) and track fusion principles [10]

$$\begin{aligned} P_{n,S}(k) &= P_{n,S}^*(k) \\ &\quad - P_{n,S}^*(k)(P_{n,S}^*(k) + R^{-1})^{-1} P_{n,S}^*(k) \\ \hat{x}_{n,S}(k) &= \hat{x}_{n,S}^*(k) + P_{n,S}(k) [y_n(k) - \hat{x}_{n,S}^*(k)] \end{aligned} \quad (17)$$

(18)

4.1 Information Filter Formulation

In line with (6) and (7) we make the assignments

$$\beta := R^{-1} \quad (19)$$

$$\alpha_m(j) := R^{-1} y_m(j) \quad (20)$$

and

$$B_0 := P_0^{-1} \quad (21)$$

$$\hat{\alpha}_0 := P_0^{-1} \hat{x}_0. \quad (22)$$

to facilitate $\eta(n,k) = (\alpha_m(j), \beta)$ and the prior information. These form the core of the following result.

Lemma 4.2 Let $Z = |Z_n^k|$ be the total number of elements in the ODS. We can be expressed

$$\hat{\alpha}_{n,S}(k) = \sum_{e=1}^Z \alpha_e + \hat{\alpha}_0 \quad (23)$$

and

$$B_{n,S}(k) = \sum_{e=1}^Z \beta + B_0 \quad (24)$$

where each α_e references a unique $\alpha_m(j)$ in (20) for $y_m(j) \in Z_n^k$ and likewise B given by (19).

PROOF. Repeated application of Lemma 4.1 and substitution of (19) and (21) into (17) gives

$$\begin{aligned} P_{n,S}(k) &= \left(\sum_{e=1}^Z R^{-1} + P_0^{-1} \right)^{-1} \\ &= \left(\sum_{e=1}^Z \beta + B_0 \right)^{-1}. \end{aligned} \quad (25)$$

Rewriting (18) and substituting (20) and (22) yields

$$\begin{aligned} \hat{x}_{n,S}(k) &= P_{n,S}(k) \left(\sum_{e=1}^Z R^{-1} y_e + P_0^{-1} \hat{x}_0 \right) \\ &= P_{n,S}(k) \left(\sum_{e=1}^Z \alpha_e + \hat{a}_0 \right) \end{aligned} \quad (26)$$

Substituting (6) into (25) and inverting both sides gives (24) as required. Left multiplying² both sides of (26) by $[P_{n,S}(k)]^{-1}$ and substituting (7) provides (23) as required.

□

The decoupling of related components and the similarity of equations (23) and (24) suggests that the ability to compute either equation implies the ability to compute the other. Given its close relationship to $\hat{x}_{n,S}(k)$, which is generally of greater importance than $P_{n,S}(k)$ in a tracking sense, we choose to focus solely on $\hat{a}_{n,S}(k)$ in the sequel.

4.2 Composition Matrix

In this section we establish a linear system to collectively decompose (7) for all RPs in the DEN.

Definition 4.1 Let $\mathcal{G}_k(N)$ be a CAG used to estimate x_0 in (4) with $\mathcal{V} = k \times N$ vertices, where the vertex v_i is indexed by (11). We define the block composition vector (BCV) $T_i \in \mathbb{R}^{r \times r\mathcal{V}}$ to be

$$T_i := [t(i, 1) \quad \dots \quad t(i, \mathcal{V})], \quad (27)$$

where each block $t(i, j) \in \mathbb{R}^{r \times r}$ represents

$$t(i, j) := \begin{cases} I & \text{if } y_{\bar{v}_j}(\underline{v}_j) \in \mathcal{Z}_{\bar{v}_i}^{v_i} \\ \mathbf{0} & \text{otherwise} \end{cases} \quad (28)$$

with v_j also indexed by (11).

In a IDEN, $t(i, j)$ is directly equivalent to the existence of the path $[v_j, v_i]$ on $\mathcal{G}_k(N)$. The BCV supports the following linear system.

²Left multiplying $A = BC$ by B^{-1} gives $B^{-1}A = C$.

Lemma 4.3 Consider $\mathcal{G}_k(N)$ with $\mathcal{V}$ vertices and $\hat{a}_0 \in \mathbb{R}^r$ given by (22). Let $\alpha_j := \alpha_{\bar{v}_j}(\underline{v}_j)$ be indexed by (11). Using $T_i \in \mathbb{R}^{r \times r\mathcal{V}}$ from (27), $\hat{a}_{i,S} := \hat{a}_{\bar{v}_i,S}(\underline{v}_i) \in \mathbb{R}^r$ in (23) can be rewritten

$$\hat{a}_{i,S} = T_i \alpha + \hat{a}_0 \quad (29)$$

where

$$\alpha = [\alpha_1^T \quad \dots \quad \alpha_{\mathcal{V}}^T]^T. \quad (30)$$

PROOF. Use (27) and (30) to expand (29) as

$$\begin{aligned} \hat{a}_{i,S} &= t(i, 1)\alpha_1 + \dots + t(i, \mathcal{V})\alpha_{\mathcal{V}} + \hat{a}_0 \\ &= \sum_{j=1}^{\mathcal{V}} t(i, j)\alpha_j + \hat{a}_0. \end{aligned} \quad (31)$$

According to (28), the number of non-zero values of $t(i, j)$ is equal to the number of elements in the conditioning set $\mathcal{Z}_n^k$. Therefore (31) is equivalent to (23). □

Letting $\hat{a}_S = [\hat{a}_{1,S}^T \quad \dots \quad \hat{a}_{\mathcal{V},S}^T]^T$ and creating the composition matrix

$$T = [T_1^T \quad \dots \quad T_{\mathcal{V}}^T]^T, \quad (32)$$

we can use (30) in conjunction with (32) and $\bar{I} = [I \quad \dots \quad I] \in \mathbb{R}^{r \times r\mathcal{V}}$ to yield

$$\hat{a} = T\alpha + \bar{I}\hat{a}_0 \quad (33)$$

Assuming an IDEN, we are able to relate the composition matrix of (32) to the TPM in (13).

Let v_i and v_j be two vertices for $\bar{v}_i \neq \bar{v}_j$ on $\mathcal{G}_k(N)$ and let $G \in \mathbb{R}^{r\mathcal{V} \times r\mathcal{V}}$ be a block matrix where each block $g(i, j) \in \mathbb{R}^{r \times r}$ is a function of $W \in \mathbb{R}^{r\mathcal{V} \times r\mathcal{V}}$ in (13) given by

$$g(i, j) = \begin{cases} I & \text{if } \omega_{ji} \geq 1 \\ \mathbf{0} & \text{otherwise.} \end{cases} \quad (34)$$

Under Assumption 2.3, the existence of the path $[v_j, v_i]$ of any length automatically implies $y_{\bar{v}_j}(\underline{v}_j) \in \mathcal{Z}_{\bar{v}_i}^{v_i}$. With $T \in \mathbb{R}^{r\mathcal{V} \times r\mathcal{V}}$ a composition matrix given by (32) then

$$T = G + I, \quad (35)$$

where I accounts for the discrepancy between the absence of the loop $[v_j, v_j]$ in M , which forms the basis of W , and acknowledgement of the presence of the current measurement in T .

4.3 Incest Matrix

We begin by describing a sub-optimal DEN.

Definition 4.2 Where the fusion centre of a SFN fuses the input set as if all elements are independent is defined as a naive fusion node (NFN). If all SFNs in a DEN operate as a NFN we call the network a corrupted DEN (CDEN).

The following assumption, adopted for this section only, directly opposes Assumption 2.3 but assists the overall theoretical development. □

Assumption 4.1 All SFNs in a DEN operate as a NFN.

Definition 4.3 Let $\bar{Z}_n^k$ be the conditioning set at the output of $[n, k]$ in a CDEN such that

$$p(x_0 | \bar{Z}_n^k) \sim \mathcal{N}(x_0; \hat{x}_{n,C}(k), P_{n,C}(k)), \quad (36)$$

Based on (36) and the indexing of (11) we define

$$\hat{a}_{i,C} = P_{n,C}(k)^{-1} \hat{x}_{n,C}(k) \quad (37)$$

to be the fully corrupted vector (FCV). Augmenting this for all $v_i \in V$ gives

$$\hat{a}_C = [\hat{a}_{1,C}^T \quad \dots \quad \hat{a}_{V,C}^T]^T. \quad (38)$$

Under Assumption 4.1, the value of ω_{ji} from (13) identifies the number of times that the measurement packet relating to vertex $v_i \in V$ has been incorporated in an incestuous output at another vertex $v_j \neq v_i$. We thus assign

$$B = W + I. \quad (39)$$

where again the presence of I stems from M not accounting for loops. Each row of $B \in \mathbb{R}^{rV \times rV}$ indicates the frequency that each measurement packet is incorporated in the conditioning of the FCV at v_j .

Definition 4.4 Let $B \in \mathbb{R}^{rV \times rV}$ and $T \in \mathbb{R}^{rV \times rV}$ given by (39) and (32) respectively. We define the incest matrix (IM), denoted

$$S := BT^{-1}, \quad (40)$$

as representative of surplus paths between vertices.

Lemma 4.4 Let $\mathcal{G}_k(N)$ be a CAG with V vertices and let $\hat{a}_S \in \mathbb{R}^{rV}$ be the augmenting of vectors in (7) relating to all vertices $v_i \in V$. The block vector $\hat{a}_C \in \mathbb{R}^{rV}$ given by (38) can be written as

$$\hat{a}_C = S\bar{a}_S + \bar{I}\hat{a}_0 \quad (41)$$

where

$$\bar{a}_S = \hat{a}_S - \bar{I}\hat{a}_0. \quad (42)$$

and S is the IM given by (40).

PROOF. Using (40), (42) and (33) we find

$$\begin{aligned} S\bar{a}_S + \bar{I}\hat{a}_0 &= BT^{-1}(\hat{a}_S - \bar{I}\hat{a}_0) + \bar{I}\hat{a}_0 \\ &= BT^{-1}\hat{a}_S - BT^{-1}\bar{I}\hat{a}_0 + \bar{I}\hat{a}_0 \\ &= BT^{-1}(T\alpha + \bar{I}\hat{a}_0) - BT^{-1}\bar{I}\hat{a}_0 + \bar{I}\hat{a}_0 \\ &= B\alpha + \bar{I}\hat{a}_0 \end{aligned} \quad (43)$$

From (39) and the fact W represents all paths between vertices, Assumptions 2.2 and 4.1 imply that $\hat{a}_C$ is given by (41) as required.

Conversely to the extension of (29) to (33), we can also partition (41) into separate row blocks. Initially

$$S = [S_1^T \quad \dots \quad S_V^T]^T \quad (44)$$

where each row block $S_i \in \mathbb{R}^{r \times rV}$ in (40) is given by

$$S_i = [s(i, 1) \quad \dots \quad s(i, V)] \quad (45)$$

and each block comprises

$$s(i, j) = \eta_{ji} I \quad (46)$$

with $\eta_{ji} \in \mathbb{N}^+$ equal to the number of paths between v_j and v_i in excess of one. Substituting (38) and (44) into (41) and extracting the i^{th} row block we have

$$\hat{a}_{i,C} = S_i \bar{a}_S + \hat{a}_0. \quad (47)$$

5 Applications of the CAG

The following two applications have important implications for decentralised estimation systems. However, as they are based on Assumption 2.3 their practical implementation at local SFNs in a DEN is beyond the scope of this work.

5.1 DIM Algorithm

Theorem 5.1 Suppose $\mathcal{G}_k(N)$, with translator matrix $M \in \mathbb{R}^{rV \times rV}$, characterises a DEN subject to Assumption 2.3. Let $\hat{a}_S \in \mathbb{R}^r$ and $\alpha \in \mathbb{R}^r$ be column vectors given by (33) and (30) respectively and $T \in \mathbb{R}^{rV \times rV}$ be the composition matrix in (32). The vector $\hat{a}_i \in \mathbb{R}^r$ in (29) can be rewritten

$$\hat{a}_{i,S} = M_i \hat{a}_S + E_i \alpha - (S_i - E_i) T \alpha - M_i \bar{I} \hat{a}_0 + \hat{a}_0 \quad (48)$$

where M_i is the i^{th} row block of M and $E_i \in \mathbb{R}^{r \times rV}$ comprises all zeros except $I \in \mathbb{R}^{r \times r}$ in column block i .

PROOF. Consider $M_i \hat{a}_S$ from (48). Substituting (33) and block multiplication yields to

$$\begin{aligned} M_i \hat{a}_S &= M_i T \alpha + M_i \bar{I} \hat{a}_0 \\ &= \left[\sum_{j=1}^V m(i, j) t(j, 1) \right. \\ &\quad \left. \dots \sum_{j=1}^V m(i, j) t(j, V) \right] \alpha + M_i \bar{I} \hat{a}_0. \end{aligned} \quad (49)$$

A similar matrix expansion yields

$$\begin{aligned} (S_i - E_i) T \alpha &= \left[\sum_{j=1}^V (s(i, j) - E_i(j)) t(j, 1) \right. \\ &\quad \left. \dots \sum_{j=1}^V (s(i, j) - E_i(j)) t(j, V) \right] \alpha, \end{aligned} \quad (50)$$

where $E_i(l)$ is the l^{th} block of E_i . Assigning

$$C_m = [\bar{m}(1) \ \cdots \ \bar{m}(\mathcal{V})] \quad (51)$$

$$C_s = [\bar{s}(1) \ \cdots \ \bar{s}(\mathcal{V})] \quad (52)$$

with

$$\bar{m}(l) := \sum_{j=1}^{\mathcal{V}} m(i, j) t(j, l) \quad (53)$$

$$\bar{s}(l) := \sum_{j=1}^{\mathcal{V}} (s(i, j) - E_i(j)) t(j, l), \quad (54)$$

we can rewrite (48) as

$$\begin{aligned} \hat{a}_{i,S} &= M_i \hat{a}_S + E_i \alpha - (S_i - E_i) T \alpha - M_i \bar{I} \hat{a}_0 + \hat{a}_0 \\ &= (C_m - C_s + E_i) \alpha + \hat{a}_0. \end{aligned} \quad (55)$$

Partitioning the bracketed component of (55) provides

$$C_m - C_s + E_i = [Q(1) \ \cdots \ Q(\mathcal{V})] \quad (56)$$

where each block $Q(l) \in \mathbb{R}^{r \times r}$ is denoted

$$\begin{aligned} Q(l) &= \bar{m}(l) - \bar{s}(l) + E_i(l) \\ &= \sum_{j=1}^{\mathcal{V}} (m(i, j) - (s(i, j) - E_i(l))) t(j, l) + E_i(l). \end{aligned} \quad (57)$$

With $E_i(l) = 0$ for all $l \neq i$, (57) decomposes to

$$\begin{aligned} Q(l) &= \sum_{j=1}^{i-1} (m(i, j) - s(i, j)) t(j, l) \\ &\quad + (m(i, i) - s(i, i) + I) t(i, l) + E_i(l) \\ &\quad + \sum_{j=i+1}^{\mathcal{V}} (m(i, j) - s(i, j)) t(j, l). \end{aligned} \quad (58)$$

Although limited space prevents it here, using the components in (58) it is possible to show that $Q(l) = t(i, l)$. This suggests

$$[Q(1) \ \cdots \ Q(\mathcal{V})] = T_i, \quad (59)$$

and then combining (59) with (56) yields

$$T_i = C_m - C_s + E_i. \quad (60)$$

Substituting (60) into (55) proves (48) is equivalent to (29) as required. $\square$

5.2 Requirement for DIM Algorithm

Data incest occurs when fusing two dependent probability densities as if they were independent of one another.

Remark 5.1 Assuming the same translator matrix in both instances, the need for a DIM algorithm to ensure an IFN is equivalent to the occurrence of data incest at an NFN.

Using this parallelism we can show the following result.

Lemma 5.1 Let $W \in \mathbb{R}^{\mathcal{V} \times r \times \mathcal{V}}$ be the TPM, given by (13), of $\mathcal{G}_k(N)$ for an IDEN. Suppose $w(i, j) \in \mathbb{R}^{r \times r}$ is a block element of W based on the integer ω_{ji} and $\hat{a}_i \in \mathbb{R}$ is the output vector from v_i . For $v_i > 1$ the equality

$$\max_j \omega_{ji} > 1, \quad (61)$$

for all j such that $v_j \in \mathcal{V}$, is a necessary and sufficient condition to require a DIM algorithm to guarantee $\hat{a}_i = \hat{a}_{i,S}$ where $\hat{a}_{i,S} \in \mathbb{R}^r$ is given by (29).

PROOF. By definition $\omega_{ji} \geq 0$. A proof by contradiction on the assumption that DIM is necessary, must consider two cases $\max_j \omega_{ji} = 1$ and $\max_j \omega_{ji} = 0$.

Based on Remark 2.1, the guaranteed path $\langle v_l, v_i \rangle$ for $\bar{v}_l = \bar{v}_i$ and $ulv_i = v_l + 1$, makes it impossible for $\max_j \omega_{ji} = 0$. As W is strictly upper triangular, $\omega_{ji} = 0$ for all $j \leq i$. For $j > i$ and $v_j \in \mathcal{V}$, the condition $\max_j \omega_{ji} = 1$ implies the maximum number of paths from any v_j to v_i is one. In this case, all inputs to v_i are guaranteed to be independent and a DIM algorithm is not required. These contradictions prove the necessary case.

The sufficient criteria is equally straightforward. The inequality $\max_j \omega_{ji} > 1$ indicates that information from at least one vertex on $\mathcal{G}_k(N)$ has arrived at v_i via two separate paths. Given this amounts to incest in a NFN, Remark 5.1 implies this same delay topology requires a DIM algorithm to provide an optimal output at the SFN referenced by vertex v_i . $\square$

Mindful of the parameters in (48), the following result provides a simple method using S for determining the need for a DIM algorithm.

Theorem 5.2 Let $S \in \mathbb{R}^{r \times \mathcal{V} \times r \times \mathcal{V}}$ be the IM on $\mathcal{G}_k(N)$ given by (40). DIM is not required on any vertex on $\mathcal{G}_k(N)$ if the condition

$$S = I \quad (62)$$

is satisfied.

PROOF. Substituting $S = I$ into (40) and right multiplying by T gives

$$T = B. \quad (63)$$

Based on (63) and the description of $\mathbf{T}$ in (28), none of the elements of $\mathbf{B}$ will have a block value greater than I . Using $\mathbf{B} = \mathbf{W} + I$ from (39), it follows that

$$\max_j \omega_{ji} = 1. \quad (64)$$

The equality in (64) contradicts the necessary condition of (61) so a DIM algorithm is not required.

□

This result is intuitive from the perspective of (48). The term $(\mathbf{S}_i - \mathbf{E}_i)\mathbf{T}\boldsymbol{\alpha}$ represents the DIM component of the new estimate. With $\mathbf{S} = I$, the find $\mathbf{S}_i - \mathbf{E}_i = \mathbf{0}$ nullifies the need to remove any repetitious elements of the conditioning set of individual elements of the input set. The same intuition suggests $(\mathbf{S} - I) > \mathbf{0}$ implies an optimal estimate can not be created without DIM.

6 Conclusion

Data incest issues are pertinent to a wide variety of decentralised estimation problems. In this paper we have developed a quantitative analysis tool, using a graph theoretic approach, for resolving these issues under controlled conditions. This graph was used to provide a data incest management algorithm for the solution of constrained optimal static estimation. The constraints involved the study of a fully interconnected network subject to variable delays in the communication of global state information between nodes. Necessary and sufficient conditions for the use of the algorithm were also provided, based on properties of the network.

References

- [1] L. J. Sciacca and R. J. Evans. Cooperative sensor networks with bandwidth constraints. In *Proceedings of Aerosense SPIE Conference, Battlespace Digitisation and Network-Centric Warfare*, Orlando Florida, April 2002.
- [2] H. R. Hashemipour, R. Sumit, and A. J. Laub. Decentralized structures for parallel kalman filtering. *IEEE Transactions on Automatic Control*, Vol. 33, No.1:88 – 94, January 1988.
- [3] S. Utete and H. F. Durrant-Whyte. Routing for reliability in decentralised sensing networks. In *Proceedings of the American Control Conference*, pages 2268 – 2272, Baltimore, Maryland, June 1994.
- [4] S. McLaughlin, V. Krishnamurthy, and R. J. Evans. Bayesian network model for data incest in a distributed sensor network. In *Proceedings of Information Fusion Conference*, Stockholm, Sweden, June 2004.
- [5] C-Y. Chong and S. Mori. Graphical models for nonlinear distributed estimation. In *Proceedings of Fusion 2004*, Stockholm, Sweden, June 2004.
- [6] S. Poulton. *Packet Switching and X.25 Networks*. Pitman, London, 1989.
- [7] S. Blackman and R. Polpoli. *Design and Analysis of Modern Tracking Systems*. Artech House, 1999.
- [8] B. D. O. Anderson and J. B. Moore. *Optimal Filtering*. Prentice-Hall, 1979.
- [9] C. Berge. *Graphs and Hypergraphs*. American Elsevier Pub. Co., New York, 1973.
- [10] Y. Bar-Shalom and T. E. Fortmann. *Tracking and Data Association*. Academic Press, Inc., Orlando, Florida, 1988.