Mathematical Statistics
Stockholm University

Fully Decentralized Sequential Detection in an Underwater Setting

Tommy Nyberg

Examensarbete 2008:20

Postal address:

Mathematical Statistics
Dept. of Mathematics
Stockholm University
SE-106 91 Stockholm
Sweden

Internet:

http://www.math.su.se/matstat

Mathematical Statistics
Stockholm University
Examensarbete **2008:20**,
http://www.math.su.se/matstat

Fully Decentralized Sequential Detection in an Underwater Setting

Tommy Nyberg*

December 2008

Abstract

In this thesis we present a model for a detector network of acoustic sensor nodes to be used in the military underwater setting. The model accounts for well-known properties of acoustic sensing and communication equipment, and to this end we employ the traditional decentralized detection framework combined with ideas from the field of team decision theory, thus arriving at a network model which operates without a centralized fusion center. The model instead gives the decision to sound the alarm to individual detector nodes, which classify the presence of a target using the classic Bayes' detector framework, applied for time-dependent sequential hypothesis testing. The detector network employs an internal message-passing scheme, thus reducing outside communication to a minimum. Reduction of internal network communication is achieved through a sensor censoring scheme where uninformative messages are withheld, and the detector nodes adjust for these missing data. Indicative simulation results are presented.

KEY WORDS: Decentralized Detection, Team Decision Theory, Bayesian Detection, Acoustic Sensors, Message-passing, Sensor Censoring.

*Postal address: Mathematical Statistics, Stockholm University, SE-106 91, Sweden.

Acknowledgments

This work is the first in a project commissioned by the Swedish Defence Research Agency and has been written as a 30 credits Master's thesis in Mathematical Statistics.

First of all I would like to thank my supervisor, Prof. Ola Hössjer at the Dept. of Mathematical Statistics at Stockholm University for all his help, for his commitment to the subject at hand, for his ability to bring a set of disparate ideas into a complete model, and for all his help and suggestions when writing this report.

Further thanks goes to Dr. Tommy Öberg at the Dept. of Signal Processing at the Swedish Defence Research Agency for his ability to make unknown fields of research understandable, for all his help in literature gathering, for his ideas in modeling, and for trusting me with a great deal of freedom.

Others to be thanked in the project work group include Stefan Petrovic for all his help and input during the implementation in Matlab, and Ron Lennartsson and David Lindgren for their invaluable comments and suggestions at our work group meetings and through e-mail. I also give thanks to all the others at the Research Agency who from time to time have supplied me with literature, programming help, and method suggestions.

Finally, I wish to thank my friend Robert Sehlberg for some suggestions and feedback.

Contents

1 Introduction

Underwater environments are among the most difficult environments to monitor anything but the smallest of areas over extended periods of time. This remains true whether it be for intruders, chemical spills, or changes in a wildlife population, to name a few examples. In the military framework — which will be the focus of this thesis — one is faced with a difficult and costly task when trying to efficiently defend a harbor or other installation against submarines, unmanned underwater vehicles, scuba divers and other potential threats which cannot be anticipated. To remedy these problems, the *Swedish Defence Research Agency* (Swedish: *Totalförsvarets Forskningsinstitut*, *FOI*) wish to develop a cost-efficient **wireless underwater sensor network**. The sensor nodes to be used in this case, described in detail in the FOI report of Grönkvist et al. (2007), are *acoustic* which means that communication and data sampling is conducted by means of acoustic microphone and speaker. For a brief survey of the pros and cons for this kind of communication equipment as compared to other plausible alternatives for the underwater environment, see the survey article of Liu et al. (2008).

Our idea is to create a model for a detector network which utilizes a message-passing scheme, thus combining the structure of the traditional **decentralized detector network** with the more general field of **team decision theory** which sprung from the pioneering work of Borkar and Varaiya (1982). This approach was suggested for further research in the excellent survey article on wireless sensor networks of Chamberland and Veeravalli (2007, p. 23).

In this thesis, we aim to give the reader a short survey on the existing theory in the field of decentralized detection and the wireless sensor network structure, as well as team decision theory, and will present a basic model for such an underwater network according to the specific setting in this case. We are only interested in *sequential detection* problems, i.e. problems of the type where we wish to test the time-dependent two-way hypothesis

$$H_0(t) : \text{No target present at time } t,$$
$$\text{vs.}$$
$$H_1(t) : \text{Target present at time } t.$$

This will be defined more stringently in the next chapter. It should however be stressed that although this model is thought to have a military application, it could with some generalization apply to more general underwater detection problems — as well as estimation problems.

The traditional approach for decentralized detection networks is to have a dedicated so-called **fusion center**, to which the detectors transmit their data. However, due to the costliness of underwater communication, we wish to create a network model without a fusion center.

This thesis is broken into four main areas. In the remainder of this introductory chapter we will present the problem and will as mentioned conduct a brief survey on existing theory in the relevant fields of research. In the next chapter we will propose a model to be used in this setting, and will go into detail on the rules for each of the four relevant phases of the detector network: observation, communication, hypothesis classification, and alarming, as well as listing some possible performance measures. In the third chapter we give simulation results, and in the final chapter we discuss our finds and point at a number of issues for further research.

1.1 Setting

In the general sensor network framework, detection is carried out by a network formed by a number of so-called **detector nodes** — detectors which are deployed across the area of interest to collect observations from their immediate surroundings. Before going into further detail on more specific network architectures, let us review our specific setting.

In the wireless sensor network (of which the wireless underwater sensor network can be considered a special case), the detector nodes can generally be assumed to consist of the following components (Chamberland and Veeravalli, 2007, p. 24):

- a sampling unit,
- a data processing unit,
- a communication unit, and
- a power supply.

Our detector nodes follow this conventional component scheme with a few notable exceptions, stemming from the acoustic communication and sampling equipment. While each node's speaker is strictly a communication device, its microphone is used both for data sampling and incoming communication. This, coupled with general underwater constraints, gives a rather specific setting. Let us list these points in order to address them later:

- The detector nodes will carry high initial assembly and deployment costs, stemming from the high costs of underwater acoustic components. Thus, the network size must be limited. We assume that a network will contain no more than 30 nodes.

- Data processing will be the cheapest component, in monetary as well as power resources. We thus allow a great deal of processing to be done at each node.

- Communication will be the most costly action in terms of power consumption and should be limited.

- The communication range (i.e. the signal strength) of sent messages for each node is variable and can be adjusted before deployment. However, power consumption increases with range. Thus, our model should let us keep the communication range low.

- Consisting of sound waves, messages sent from a node will travel *isotropically*, i.e. equally fast in all directions, essentially in an expanding sphere around the node.

- Since acoustic communication is limited to the sound of speed in water, there will generally be a time delay before a sent message reaches its target according to the distance between sender and receiver.

- Communication will never be fully reliable; there is always a substantial risk of lost messages.

- The cost of false alarms will be very high — both in terms of power consumption since it involves communication to an external receiver, and in monetary terms because of the likely expense of actions taken at

a military alarm. The false alarm rate should be kept as low as possible while still retaining an acceptable detection probability.

The reader should be aware that not much is known about the specific range of these listed restrictions at this time, and that they are specified to serve as guidelines rather than formal assumptions when developing our network model. Further formal studies are required in order to find optimal network structures, node configurations, etc.

1.2 Decentralized Detection

Let us now give a short review of the rather broad research field of *decentralized* detection. As described in the previously mentioned survey article of Chamberland and Veeravalli (2007), contemporary decentralized (or *distributed*, as it is called by some authors) sensor networks consist of a number of detector nodes which are distributed across the area of interest to collect data from their surroundings. In addition to these nodes the network traditionally holds a dedicated so-called *fusion center*, located in some key position or possibly outside the area of interest entirely. The nodes transmit data — reliably or otherwise — to the fusion center, which has the task of making an informed decision on whether to set off an alarm based on this received information. See Figure 1. However, in order to limit the amount of communication, it is generally desirable to have the nodes not send their full observation data. Hence, some form of data processing (or *aggregation*) is done locally at each node in order to transmit only a summary statistic to the fusion center. The network is therefore *decentralized*, as the local detector nodes themselves are involved in the processing of detector data. With this approach, as opposed to that of a *centralized* detection network where all available raw data is transmitted to the fusion center, the amount of network communication can usually be significantly reduced, with an acceptably small decrease in detection performance.

Numerous proposals have been made on which summary statistic to be used, and for some specific problems and settings it is possible to find an optimal solution. A few common choices of local data aggregations include having each node transmit a real-valued statistic of its latest observations

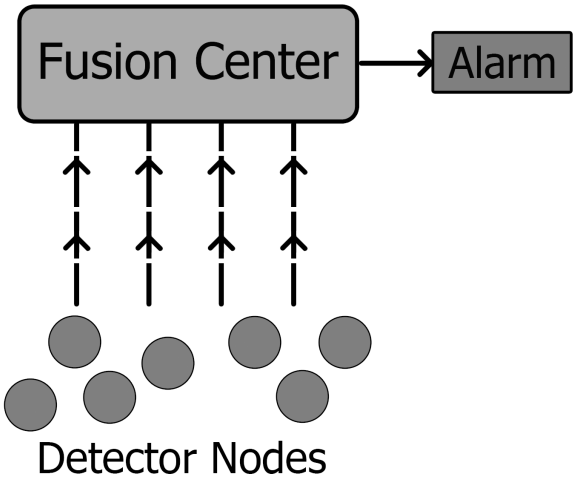

Figure 1: Outline of the typical decentralized sensor network. A number of detector nodes are distributed across the area of interest, where they collect data. The detector nodes transmit summary data to the fusion center, which in turn decides whether to set off the alarm.

(sample mean, variance, etc.) (e.g. Appadwedula et al., 2008), or a message chosen among a finite set of possible messages (e.g. Veeravalli, 2001; Chamberland and Veeravalli, 2003) — a special case being the nodes' local binary H_0/H_1 classification (e.g. Niu et al., 2006; Crow and Schwartz, 1996). A common (and intuitive) result — although it is possible to construct examples of network structures where this does not hold (Chamberland and Veeravalli, 2003) — is that the complexity of sent messages can be reduced as the number of nodes grow, a conclusion reached by Tsitsiklis (1988) and Chamberland and Veeravalli (2003, 2004) to name a few.

One method of reducing communication is the so-called censoring sensor scheme (see e.g. Appadwedula et al., 2008; Rago et al., 1996), in which each node may decide whether to send its latest statistic based on if it is deemed relevant to the hypothesis testing problem at hand. Thus, a node may for instance choose to withhold its latest message if the value of the contents or the corresponding observations fail to exceed a certain threshold.

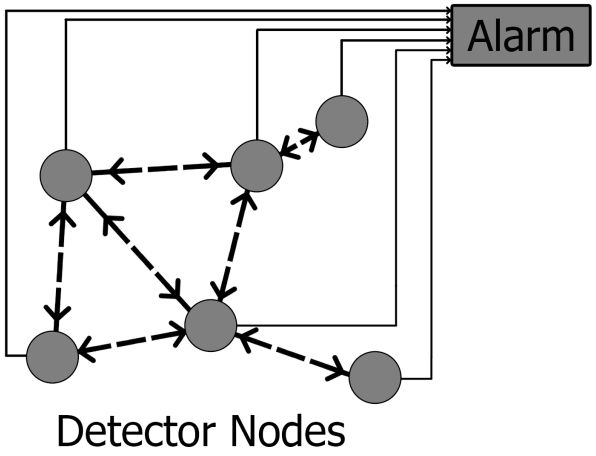

Figure 2: Outline of the typical fully decentralized sensor network. A number of detector nodes are distributed across the area of interest, where they collect data. The detector nodes communicate with each other, and work together in determining whether to set off the alarm.

1.2.1 *Fully* Decentralized Detection

We wish to take the decentralized nature of the decentralized sensor network one step further, in creating a sensor network which does **not** employ a designated fusion center, thus creating what we choose to call a *fully* decentralized sensor network. There are numerous reasons to strive for such a setup, an important one being the reduced vulnerability that can be achieved if the network performance is made not to rely on communication reaching a potentially far-off fusion center, which might itself be prone to breakdowns. In our setting, this characteristic should also serve to reduce communication costs as the nodes may be configured to send messages no further than to their closest neighbors. Further specific reasons in our setting include how both observation sampling and communication are performed by acoustic means, thus allowing each node to pick up messages sent from neighboring nodes as part of their regular observation sampling. Finally, in a military setting, a network with limited external communication is likely to be easily concealed from intruders, thus indirectly increasing intruder detection.

As briefly mentioned above, we have decided to implement a message-

passing scheme in which the communication takes place *between* the detector nodes within the network. We will allow a greater subset (possibly all) of the sensor nodes to make decisions on the presence of a target (in a sense, each of these sensor nodes will function as a fusion center by themselves), thus limiting outside communication until a rough network consensus on setting off the alarm has been reached. We might however want to impose some restriction on when a node may set off the alarm, as each detector node under most communication schemes can be assumed to have less information than a designated fusion center would have. See Figure 2.

1.3 Team Decision Theory

To our knowledge, not much research has been done on the fully decentralized sensor network, but much has been written in the more general field of communicating decision makers that seek to find a consensus, so-called team decision theory. We will make a brief survey of this field of research, in order to arrive at a suitable network model.

An often quoted early work in team decision theory is that of Borkar and Varaiya (1982), in which the authors present a model and convergence criteria for a network of communicating decision makers (so-called *agents*) which seek to reach a consensus by passing messages to each other. Under this model — even allowing for new observations to be taken, random transmission times, and allowing which messages reach a certain agent to be random — the network is shown to reach an almost sure asymptotic consensus on estimation and detection problems by letting the agents send to the others nothing more than their latest point estimate or likelihood ratio estimate, respectively, if only each agent in the network communicates with each other agent at least indirectly infinitely often. Numerous generalizations and expansions have been made for this model. Tsitsiklis and Athans (1984) generalized to allow for situations where less than all available information is remembered, and also to allow for more general problems where a cost function is to be minimized — a class of problems of which those addressed by Borkar and Varaiya (1982) are special cases. Furthermore, they show that similar results hold even when the agents send messages chosen among

a finite set (such as their latest H_0/H_1 classification). However, they do assume that no new observations are collected once network communication has started, thus weakening the results. In the very general work of Washburn and Teneketzis (1984), these two models are treated as special cases, and conditions are presented under which also networks using finite message models reach consensus. Finally, in the survey article of Varaiya (1985), a general framework for team decision models is presented, with the mentioned models as special cases.

The ability of a network to reach a common consensus through such a limited amount of communication, that would result if all information which a certain node has received could be summed up in a single statistic, certainly seems appealing. However, one soon finds that these and similar models come with a number of practical concerns. Let us address the model presented by Borkar and Varaiya (1982), as the problems faced illustrates well the type of problems which can be found in all of these mentioned models.

First of all, the agents' latest estimate to be sent is defined rather theoretically; as the conditional expectation *given* the information (observations and messages) available to the agent at a given time. For instance,

$$L_t^m := \mathrm{E}\left\{L \,|\mathcal{F}_t^m\right\}$$

is the current *estimate* of the likelihood ratio L for agent m at time t, where $\mathcal{F}_t^m$ is the sigma algebra generated by the observations *and* messages available to the agent up to time t (Borkar and Varaiya, 1982, p. 650). Besides the fact that an exact expected value might be calculationally intractable even without further obstacles, as pointed out by Tsitsiklis and Athans (1984, p. 43) these calculations implicitly rely on the requirement that information on which agents each sent message has reached are either available to the entire network at all times, or that the agents make probabilistic inferences on the history of communication for all other agents in the network.

In the network setting considered in this thesis, neither of these two possible requirements for the history of communication are satisfactory. Due to isotropic nature of transmitted messages, the amount of possible receptor nodes for each message will generally be great, thus making the probabilistic approach intractable for all but the smallest of networks. It might, in

principle, be possible to have every node in the network know the history of communication with *good enough* certainty, but likely not without wasting a good amount of communication and time — two of our most precious resources.

While a traditional team decision model thus is out of the question, in order to continue our research on a fully decentralized detection model, we identify two possibilities for further investigation. One plausible method would be to keep the notion of a communicating network in which the nodes aggregate all their available information into one estimate, but where this aggregation is done in a more ad hoc fashion according to some simple and intuitive — although not necessarily mathematically stringent — rule. For instance, a rather simple model in which incoming decisions are added together using constant weights, which are arbitrarily assigned to the different decision makers according to their *perceived* reliability was proposed by DeGroot (1974). It might be possible to create a model somewhat akin to this, and to find *good* (in some sense) ad hoc aggregation weights by extensive simulations. However, while such a sensor network would certainly keep the amount of communication constantly low, we deem it likely to be both too unstable (and thus error-prone) and too specific (in the sense that it might be difficult to find general rules that apply in more than one setting). Further research in this field would certainly be of interest, but for this thesis we will not consider this ad hoc model approach.

The other plausible method — which we have chosen to follow — simply involves sending more data, while retaining the idea of a network seeking consensus. In the next section we present this model, which is based on the traditional Bayesian maximum a posteriori (MAP) detector (see Hippenstiel, 2002, Chap. 4, esp. 4.2–3).

2 Model[1]

Suppose we have a network consisting of N detector nodes, labeled $1, \ldots, N$, each of which sequentially receives external observation samples in discrete time. Based on these samples and network communication, the nodes' objective is to form a sequential decision. Sequential means that detection goes on simultaneously with network communication and observation sampling, and thus that the current hypothesis classification is updated continually in time.

An outline of a typical detector node's activities are given in Figure 3. Note that this is only a simple sketch given to increase understanding, and that it is fully possible to change the order of these activities. Also, we will see that the model is significantly more complex than shown here. For instance, some nodes may have a different configuration, messages from other nodes are actually received through observations sampling and observations sampling actually goes on simultaneously with the other activities.

Since we deal with networks with a small number of detector nodes we will not follow the large-network convention of letting the nodes send as simple statistics as possible (usually binary H_0/H_1 messages), and will instead use real-valued statistics conveying more information. For a number of reasons we choose to take a Bayesian view on the detection problem (as opposed to a Neyman-Pearson frequentist view). First of all, one will generally have some idea on the behavior of a possible target to be detected. For instance, when looking for a moving intruder it would be unreasonable not to take into account the topology of the network and surrounding environment — in essence, the nodes should be aware that no ship will travel through land. Secondly, the Bayesian approach gives the model a somewhat increased flexibility in its deployment. Returning to the moving intruder example, a detector network employed to look for submarines likely needs a significantly different configuration than a network looking for divers.

We assume that the nodes operate according to a common, synchronized clock. In practice, the network will need some form of synchronization pro-

[1]Parts of this chapter, as well as the appendices referenced herein, are based on the working paper of Hössjer et al. (to appear), in which this model is presented.

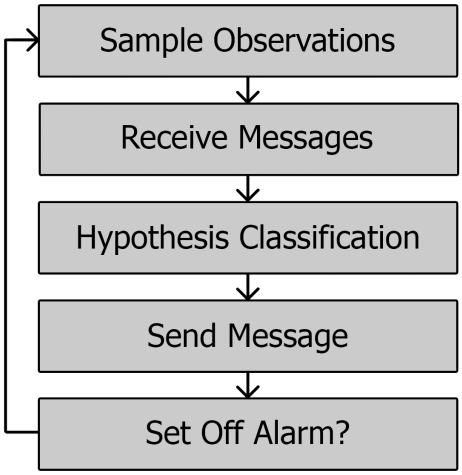

Figure 3: Outline of the activities of a typical detector node. The node observes its surroundings and creates observations, it receives messages from the other nodes in the network, it uses its observations and received messages to perform a hypothesis test, it communicates these finds to the others, and finally it decides whether to set off the alarm.

tocol, but this lies outside the scope of this thesis. We will use three different time scales for measuring time. These are given here, from shortest to longest:

- A **time point** corresponds to the time needed for a node to receive an observation sample from the environment, and is thus generally a very short amount of time.
- A **time block** consists of r time points and corresponds to the time between which each node aggregates its latest samples into a summary statistic, and possibly reevaluates its current hypothesis classification.
- A **time window** consists of the w latest time blocks and corresponds to the amount of time a hypothesis classification should be based upon. Thus, it also corresponds to the length of time a new observation must be remembered before being discarded.

In our military framework, for reasons which will become clear below, we assume that the length of a time block corresponds to 1 second, and that

the nodes sample their surroundings at a rate of at least 1 kHz (i.e. $r \geq 1000$). Further, as a military detector deals with detecting abrupt changes, the typical length of a time window could vary between 10 and 120 seconds (i.e. $10 \leq w \leq 120$) depending on application.

2.1 Observations Sampling

In compliance with the acoustic nature of the observation sampling equipment, we assume that the detector nodes' incoming data have the shape of a simply modeled sound wave. Let

$$Z_{ti} = s_i(t - \tau; \boldsymbol{\theta}) \cos(\omega t + \phi) + \epsilon_{ti}$$

denote an **observation sample** available to node i at time $t \in \mathbb{N}$, where $\mathbb{N} = \{1, 2, \ldots\}$ is the set of natural numbers. It consists of a **target signal** $s_i(t - \tau; \boldsymbol{\theta})$, which is the time varying non-negative amplitude of a sinusoidal bandpass signal, corrupted by additive noise. The center frequency ω of the sinusoidal signal is assumed known, whereas the phase ϕ is unknown. The time point τ when the target passes the network is unknown, as is the parameter vector $\boldsymbol{\theta} = (\theta_1, \ldots, \theta_p) \in \Theta$ that indexes the form of the target signal. The noise terms ϵ_{ti} are assumed to have a Gaussian distribution and to be independent of each other and the target signal, but they may vary with time and between nodes (i.e. node positions); $\epsilon_{ti} \in N(0, \sigma_i^2(t))$.

Typically, $s_i(t - \tau; \boldsymbol{\theta})$ and $\sigma_i(t)$ vary much more slowly than the sinusoidal signal, and will be regarded as constant over short time intervals. We generally assume that the nodes receive different signals but allow for the possibility that all nodes receive the same signal, in which case we write $s_i(t - \tau; \boldsymbol{\theta}) = s(t - \tau; \boldsymbol{\theta})$.

2.1.1 Hypothesis Testing and the Target Signal

Leading up to the formal definition of the hypothesis testing problem, let us first consider the disjoint decomposition $\Theta = \Theta_0 \cup \Theta_1$ of the parameter space, where

$$\begin{aligned} s_i(t - \tau; \boldsymbol{\theta}) &= 0, \quad \boldsymbol{\theta} \in \Theta_0, \\ s_i(t - \tau; \boldsymbol{\theta}) &\geq 0, \quad \boldsymbol{\theta} \in \Theta_1, \end{aligned} \tag{1}$$

holds for all t and i, with strict inequality in the second equation for at least one t. Hence, $\boldsymbol{\theta} \in \Theta_1$, and $\boldsymbol{\theta} \in \Theta_0$, correspond to existence, and nonexistence, of a target respectively. However, this formulation is insufficient for use in a hypothesis test, as it is impossible at a given time to distinguish between the situations where no target exists, and where a target exists but is simply currently outside the nodes' field of vision.

For each $t \in \mathbb{N}$, we therefore define the hypothesis testing problem

$$\begin{aligned} H_0 : &\quad \boldsymbol{\theta} \in \Theta_0, \\ H_1(t) : &\quad \boldsymbol{\theta} \in \Theta_1, \tau = t \end{aligned} \tag{2}$$

where τ — as mentioned — denotes the unknown time point when the target is present in the vicinity of the network. Note that τ is assumed to be a fixed time point, as the hypothesis testing problem is defined for a single time point t at which the test is performed — in most applications however, a target will obviously be present for more than a single time point if it is indeed present. Under the null hypothesis H_0, the received observations are just noise, whereas under the alternative $H_1(t)$, a target exists and is present at the network at time t.

We also assume that

$$\begin{aligned} \Theta_0 &= \{\boldsymbol{\theta};\, \boldsymbol{\theta} \cdot \boldsymbol{u} = 0\}, \\ \Theta_1 &= \{\boldsymbol{\theta};\, \boldsymbol{\theta} \cdot \boldsymbol{u} > 0\}, \end{aligned} \tag{3}$$

where $\boldsymbol{u}$ is a fixed unit vector in $\mathbb{R}^p$. That is, the null hypothesis corresponds to a $(p-1)$-dimensional subspace of Θ and the alternative hypothesis to a p-dimensional half plane. This is a rather general formulation which allows for a large class of possible hypothesis tests, but we will focus on the set of hypotheses where the null hypothesis corresponds to one or more parameters in $\boldsymbol{\theta}$, specified according to $\boldsymbol{u}$, being zero.

Let us now look at some examples of possible target signal models.

Example 1 (Change point) The nodes in the area all receive the same signal, and are to detect if and when the signal makes a sudden jump in magnitude. This is a class of problems which has received a great deal of attention in the past (cf. Basseville, 1988; Crow and Schwartz, 1996; Veeravalli,

2001). Formally, let $p = 1$ and put

$$s(t - \tau; \boldsymbol{\theta}) = \theta 1_{\{t-\tau \geq 0\}},$$

where θ is the magnitude of the change point and $1_{\{t-\tau\geq 0\}}$ equals one if $t - \tau \geq 0$ and zero otherwise. We put $\Theta_0 = \{0\}$, $\Theta_1 = (0, \infty)$ and $\boldsymbol{u} = 1$. Analogously, a change point with linear trend has

$$s(t - \tau; \boldsymbol{\theta}) = \theta(t - \tau + 1)_+,$$

with Θ_0, Θ_1, $\boldsymbol{u}$ and τ the same and $x_+ = \max(0, x)$.

□

Example 2 (Moving Target) For this model the nodes' objective is to detect possible intruders. A similar target signal model was proposed by Choi et al. (2008), although the objective in that article was to estimate the parameter values of an already known target. We put $p = 7$ and let $(\theta_1, \theta_2, \theta_3)$ denote the position in Euclidean space at time τ of a (possible) target. This target is assumed to be moving along a straight line with velocity vector $(\theta_4, \theta_5, \theta_6)$. Further, let (x_i, y_i, z_i) denote the position of detector node i and

$$d_i(t - \tau; \boldsymbol{\theta}) = |(x_i, y_i, z_i) - (\theta_1, \theta_2, \theta_3) - r^{-1}(t - \tau)(\theta_4, \theta_5, \theta_6)|$$

its Euclidean distance to the target at time t. The term r^{-1} above simply stems from the convenience of specifying the velocity per time block, as opposed to per time point. Put

$$s_i(t - \tau; \boldsymbol{\theta}) = \theta_7 / g(d_i(t - \tau; \boldsymbol{\theta}))$$

where θ_7 is proportional to the (constant) power radiating from the target and g is some known function, e.g. $g(d) = 1 + (d/\delta)^2$ in shallow water and $g(d) = 1 + (d/\delta)^\nu$ for some $2 < \nu \leq 3$ in deep water.[2] The parameter δ is assumed to be a known constant corresponding to the approximate size of the targets of interest, e.g. $\delta = 2$ if looking for divers or $\delta = 20$ for larger vessels

[2] This is simply a formulation of the well-known physical property that the strength of an acoustic signal fades roughly quadratic in shallow water and slightly faster in deep water.

assuming the scale is meters. We have $\Theta_0 = \mathbb{R}^6 \times \{0\}$, $\Theta_1 = \mathbb{R}^6 \times (0, \infty)$ and $\boldsymbol{u} = (0, \ldots, 0, 1)$, where $\mathbb{R}$ is the set of real numbers. The reader should note that the target is assumed to have constant velocity and direction, and also that the model allows for only one target to be present at each time. These weaknesses will be discussed in Chapter 4.

□

2.1.2 Aggregation

At fixed and commonly known time points the nodes aggregate their sampled data into a summary statistic to be used in classification and shared with the others. As briefly mentioned above, it will be convenient to partition the received external observation samples into disjoint time blocks of length r time units, so that

$$\boldsymbol{Z}_{ki} = \{Z_{ti};\ (k-1)r + 1 \leq t \leq kr\}$$

is node i's vector of observations during the k^{th} time block for any $k \in \mathbb{N}$. In this section, we will rely on two strong assumptions regarding the center frequency. First, we assume that ω is small enough that the value of $\cos(\omega t + \phi)$ changes negligibly between two consecutive time points, i.e. $\omega \ll 2\pi$. Second, we assume that ω is big enough that $\cos(\omega t + \phi)$ oscillates a great number of times during the r time points of the time block, i.e. we require that if we write the frequency as $\omega = 2\pi G/r$, then $G \propto r$. Note that these assumptions can always be made to hold by adjusting the length of the time point and the time block. In the military setting where targets generally appear fast we deem the proposals given at the introduction of Chapter 2 satisfactory.

To begin with, i extracts the feature vector $\boldsymbol{Y}_{ki} = (Y_{1ki}, Y_{2ki})$, where

$$\begin{array}{rcl} Y_{1ki} & = & (2/r) \sum_{t=(k-1)r+1}^{kr} Z_{ti} \cos(\omega t), \\ Y_{2ki} & = & (2/r) \sum_{t=(k-1)r+1}^{kr} Z_{ti} \sin(\omega t). \end{array} \tag{4}$$

Since noise variance and target signals vary slowly, we assume that

$$\begin{array}{l} s_i(t - \tau; \boldsymbol{\theta}) \text{ is constant with value } s_{k-l,i}(\boldsymbol{\theta}) = \sum_{t=(k-1)r+1}^{kr} s_i(t - \tau; \boldsymbol{\theta})/r, \\ \sigma_i^2(t) \text{ is constant with value } r\sigma_{ki}^2/2 \end{array} \tag{5}$$

during the k^{th} time block when $\tau = lr$. It then follows that

$$[\boldsymbol{Y}_{ki}|\tau = lr] \tilde{\in} N\left(s_{k-l,i}(\boldsymbol{\theta})(\cos(\phi), -\sin(\phi)), \sigma_{ki}^2 \begin{pmatrix} 1 & 0 \\ 0 & 1 \end{pmatrix}\right) \tag{6}$$

has an asymptotic bivariate Gaussian distribution, with amplitude $s_{k-l,i}(\boldsymbol{\theta})$ and noise variance σ_{ki}^2 (see Appendix A-1 for a proof). The phase ϕ is regarded as a nuisance parameter. Finally, we get amplitude and noise standard deviation estimates as

$$\begin{aligned} A_{ki} &= \sqrt{Y_{1ki}^2 + Y_{2ki}^2}, \\ \hat{\sigma}_{ki} &= \sqrt{(r\lfloor r/2 \rfloor)^{-1} \sum_{t=1}^{\lfloor r/2 \rfloor} (Z_{(k-1)r+2t,j} - Z_{(k-1)r+2t-1,j})^2}, \end{aligned} \tag{7}$$

where $\lfloor x \rfloor$ is the largest integer smaller or equal to x. In the second part of (7) we have used that the value of $\cos(\omega t + \phi)$ changes negligibly between two consecutive time points. This implies that $\hat{\sigma}_{ki}$ can be regarded as a consistent estimator of σ_{ki} as $r \to \infty$. It can further be shown that A_{ki} follows a Rice distribution (cf. Appendix A-2) — a property we will use in the hypothesis tests below.

For notational simplicity, we will call each pair $(A_{ki}, \hat{\sigma}_{ki})$ the **observation** of node i for time block k. The *observation* — which actually is a summary statistic — should thus not be confused with the corresponding *observation samples*.

2.2 Communication Model

The nodes are linked into a communication network. In this section we will give a model for this communication, which specifies when the nodes send messages or forward messages received from other nodes, the contents of these messages and to which other nodes messages are sent.

2.2.1 Simple Messages

After completion of a time block (and thus creation of a new observation), each node may send a message to be received by the other nodes. The contents of each message is a collection of a number of **simple messages** (SMs), each one organized according to a **message protocol** (MP) with the following four parts:

MP1 The **node** j from which the information is originally sent (if $j \neq i$, this is a forwarded SM).

MP2 The **time block** q after which the SM was originally created.

MP3 The **observation** $(A_{qj}, \hat{\sigma}_{qj})$ based on j's received observation samples $\boldsymbol{Z}_{qj}$ during time block q.

MP4 The **hypothesis classification** $\hat{m}_{qj}$ (0 or 1) made by j after time block q.

Let SM_{ki} denote the simple message created by node i after completion of time block k.

In order to save energy resources, we impose a censoring condition where the node determines whether to send its own SM according to the real-valued parameter λ, as

$$\begin{array}{lcl} A_{ki}/\hat{\sigma}_{ki} > \lambda & \Rightarrow & \text{send } \mathrm{SM}_{ki}, \\ A_{ki}/\hat{\sigma}_{ki} \leq \lambda & \Rightarrow & \text{do not send } \mathrm{SM}_{ki}. \end{array} \tag{8}$$

The reader may note that $A_{ki}/\hat{\sigma}_{ki}$ is a monotone function of the **observation signal-to-noise ratio** $\mathrm{OSNR}_{kli}(\boldsymbol{\theta}) = (s_{k-l,i}(\boldsymbol{\theta})/\sigma_{ki})^2$ of node i during the k^{th} time block when $\tau = lr$.[3] Thus, (8) implies that the probability of a SM being sent increases with the OSNR. If $\lambda = 0$, nodes always send their own SMs after completion of a time block. Otherwise, when $\lambda > 0$, only a fraction

$$P_{H_0}(\text{send a SM}) = P_{H_0}(\text{send } \mathrm{SM}_{ki}) \approx P_{H_0}(A_{ki}/\sigma_{ki} > \lambda) = \exp(-\lambda^2/2),$$

of SMs are sent under H_0, since then A_{ki}/σ_{ki} has a Rayleigh distribution.[4] Under H_1, the amount of communication can be expected to increase as the target signal increases.

[3] We have $\mathrm{E}[(A_{ki}/\sigma_{ki})^2] = \mathrm{OSNR}_{kli}(\boldsymbol{\theta}) + 2$. For $\mathrm{E}[(A_{ki}/\hat{\sigma}_{ki})^2]$ the result holds asymptotically.

[4] A random variable R has a Rayleigh distribution if $R = \sqrt{G_1^2 + G_2^2}$ where G_1 and G_2 are independent standard Gaussian random variables.

2.2.2 Reception Probabilities, Forwarding and Decision Active Nodes

Let the **reception probability** P_{ji} denote the probability that a message sent from node j is received by node i (without any forwarding by other nodes). We require that all networks considered are **connected**, in the sense that all nodes can communicate with each other at least indirectly through other nodes. We will make the rather strong assumption that the entire **reception probability matrix** $\boldsymbol{P} = (P_{ji})$ is commonly known by all nodes. In application, this can be achieved to a good certainty by estimation through test communication during a certain predetermined time period, but the specifics on how to achieve this will be left for further research.

However, as we have some knowledge of the acoustic communication to be used for our network model, we will here present a plausible model which satisfies the above assumption. Since we have assumed that the angular distribution of sent messages is isotropic, it is reasonable to model the reception probabilities as some function h of the Euclidean distance $d_{ji} = \sqrt{(x_i - x_j)^2 + (y_i - y_j)^2 + (z_i - z_j)^2}$ between the two nodes,

$$P_{ji} = h(d_{ji}).$$

We let $h(0) = 1$, so that $P_{ii} = 1$, which simply indicates that i knows the contents of its own messages. For $d > 0$, we will use the two parameter step function

$$h(d) = P1_{\{d \leq d_{\max}\}}, \quad d > 0, \tag{9}$$

so that sent messages are received by other nodes in a sphere of radius $d_{\max}$ around the sending node, each with probability P, and are never received by those outside the sphere.

We may decide which simple messages to be forwarded by means of a binary $N \times N$ **routing matrix** $\boldsymbol{R} = (R_{ji})$ and the following **routing rules**:

R1 A necessary condition for i to forward a message originally from j is $R_{ji} = 1$.

R2 $R_{ii} = 0$, i.e. i does not forward any of its own messages.

R3 No SMs are forwarded more than once.

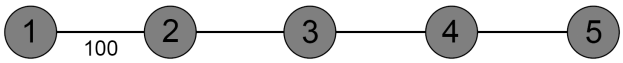

Figure 4: The network structure in Example 3. Lines are drawn between nodes within communication range.

R4 No SMs older than $w - 2$ time blocks are forwarded, where w is the number of time blocks of the time window on which each detector node operates.

We further allow for the possibility that only a subset $\mathcal{A} \subseteq \{1, \ldots, N\}$ of **decision active nodes** perform classification. The role of the remaining nodes is then simply to provide those in $\mathcal{A}$ with observations. In general, there is a trade-off between the size of $\mathcal{A}$ and the number of 1 entries in $\boldsymbol{R}$. When $\mathcal{A}$ consists of one single node the network may be seen as a special case of a decentralized network with a fusion center.

Example 3 (Five nodes on a line) The topology with $N = 5$ nodes is shown in Figure 4. If (9) holds with $d_{\max} = 100$, it follows that the reception probability matrix $\boldsymbol{P} = (P_{ji})$ has the form

$$\boldsymbol{P} = \begin{pmatrix} 1 & P & 0 & 0 & 0 \\ P & 1 & P & 0 & 0 \\ 0 & P & 1 & P & 0 \\ 0 & 0 & P & 1 & P \\ 0 & 0 & 0 & P & 1 \end{pmatrix}.$$

Here all simple messages have to be forwarded through a line of intermediate nodes in order to reach from one end of the network to the other. By symmetry, three different configurations are credible if every decision active node is to have a chance to receive SMs from all nodes. If all nodes are decision active, all nodes except the edge nodes (i.e. 1 and 5) should forward

everything they receive.

$$\mathcal{A} = \{1, 2, 3, 4, 5\}, \quad \boldsymbol{R} = \begin{pmatrix} 0 & 1 & 1 & 1 & 0 \\ 0 & 0 & 1 & 1 & 0 \\ 0 & 1 & 0 & 1 & 0 \\ 0 & 1 & 1 & 0 & 0 \\ 0 & 1 & 1 & 1 & 0 \end{pmatrix}$$

This next configuration serves as a middle ground between communication reduction and performance.

$$\mathcal{A} = \{2, 3, 4\}, \quad \boldsymbol{R} = \begin{pmatrix} 0 & 1 & 1 & 0 & 0 \\ 0 & 0 & 1 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 1 & 0 & 0 \\ 0 & 0 & 1 & 1 & 0 \end{pmatrix}$$

Finally, this configuration — similar to networks with a Fusion Center — keeps communication at a minimum.

$$\mathcal{A} = \{3\}, \quad \boldsymbol{R} = \begin{pmatrix} 0 & 1 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 1 & 0 \end{pmatrix}$$

□

Example 4 (Asymmetric network with five nodes) The topology of $N = 5$ nodes is shown in Figure 5. If (9) holds with $d_{\max} = 100$, it follows that $\boldsymbol{P} = (P_{ji})$ has the form

$$\boldsymbol{P} = \begin{pmatrix} 1 & P & P & P & 0 \\ P & 1 & P & P & 0 \\ P & P & 1 & P & 0 \\ P & P & P & 1 & P \\ 0 & 0 & 0 & P & 1 \end{pmatrix}. \tag{10}$$

In this case node 5 only communicates with 1,2 and 3 through 4. If $5 \in \mathcal{A}$, then 4 has to forward messages from 1,2,3 and 5, whereas $5 \notin \mathcal{A}$ only requires

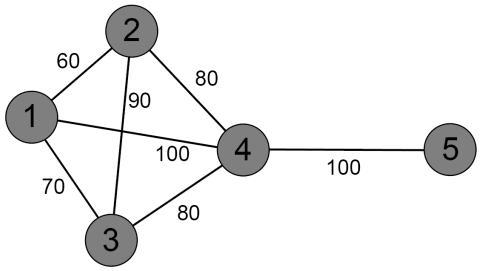

Figure 5: The network structure in Example 4. Lines are drawn between nodes within communication range.

that 4 forwards messages from 5. This gives two possibilities,

$$\mathcal{A} = \{1,2,3,4,5\}, \quad \boldsymbol{R} = \begin{pmatrix} 0 & 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 1 & 0 \end{pmatrix}$$

and

$$\mathcal{A} = \{1,2,3,4\}, \quad \boldsymbol{R} = \begin{pmatrix} 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 1 & 0 \end{pmatrix}.$$

□

2.2.3 Sending and Reception Times

Suppose v_{send} bits per time unit is the maximal sending capacity for the whole send spectrum. This sending speed will strongly depend on the data quantization alphabet used, and we can thus suspect a negative dependence

between v_{send} and $\boldsymbol{P}$ since a smaller alphabet requires less communication at a cost of increased communication vulnerability.

We have assumed that observation sampling as well as the sending and reception of messages can proceed simultaneously. In practice, this would mean that the available spectral bandwidth for sending messages is divided into N equally wide frequency bands, so that each node sends over a separate band. Then each node is able to send at a maximal speed of v_{send}/N bits per time unit. We assume that each node starts sending its (possible) message at the beginning of each new time block. If node i starts sending n_{ki} SMs after the end of the k^{th} time block (i.e. during time block $k+1$), and each SM consists of b bits, the whole message will be sent during the time interval[5] $\{kr+1,\ldots,kr+T_{ki}^{\text{send}}\}$, where

$$T_{ki}^{\text{send}} = \lceil n_{ki} bN/v_{\text{send}} \rceil \tag{11}$$

is the time duration of sending the message and $\lceil x \rceil$ the smallest integer larger or equal to x. The n_{ki} SMs consist of i's own SM from time block k, provided $A_{ki}/\hat{\sigma}_{ki}$ is large enough (cf. (8)), and all SMs that i has received during time block k and is to forward according to (R1)-(R4). Hence

$$n_{ki} \leq 1 + (N-1)(w-2),$$

although, depending on the form of $\boldsymbol{P}$ and $\boldsymbol{R}$, n_{ki} may be much lower than this upper bound.

The number of bits of a SM can be expanded as[6]

$$b = b_1 + b_2 + b_3 + b_4 = \log_2(N) + \log_2(K) + 2c + \log_2(2), \tag{12}$$

where b_a is the number of bits required for sending part MPa of the SM, K is the number of time blocks before the time block counter is reset to 0 and c is the number of bits used to transmit each time block's estimated amplitude and standard deviation (7) in the quantization used. We will assume that c is

[5] A natural generalization is $\{kr+\delta+1,\ldots,kr+\delta+T_{ki}^{\text{send}}\}$, where δ represents the time delay before all information to be sent has been computed and composed into a message.

[6] To be precise, a SM from a node $j \notin \mathcal{A}$ only consists of $b = b_1 + b_2 + b_3$ bits, since part MP4 of the SM need not be included.

large enough that no loss of precision takes place when an observation is sent to other nodes. Hence, in most cases, the third term of (12) is dominating.

The time for a message sent from i to reach j is

$$T_{ij}^{\text{rec}} = d_{ij}/v_{\text{water}},$$

where v_{water} is the speed of sound in water. Thus, the total **transmission time** from when i starts sending its message at time $kr+1$ until it is completely received by j is $T_{k,ij}^{\text{trans}} = T_{ki}^{\text{send}} + T_{ij}^{\text{rec}}$.

2.3 Classification

In this section, we shall lay the rules for how the nodes update their current hypothesis classification. We will assume that no node performs a classification other than when they have just created a new observation (thus a maximum of one classification is performed during each time block), then taking into account its new observation and the observations contained in any messages which may have arrived during the last time block. We may wish to impose further restrictions on when a classification is to be performed, which will be discussed below.

If a classification is to take place after the end of time block k by a decision active node $i \in \mathcal{A}$, the node tests H_0 versus $H_1 = H_1(lr)$ based on all observations created by i itself and received from other nodes over a **time window** of length $w = w_1 + w_2$ time blocks. The time window contains time blocks

$$\text{TW}(k) := \{k - w + 1, \ldots, l, \ldots, k\} \tag{13}$$

where $l = k - w_2$, the block which the hypothesis test is to be performed upon. The resulting classifier is $H_{\hat{m}_{ki}}$. Note that since classification is based solely on observations created during the time window, any observations older than w time blocks may be discarded.

There are a number of plausible choices of (w_1, w_2). A natural choice is $w_2 = 0$ (i.e. $l = k$), which means that the node will try to classify the existence of the target at the current time. However, a potential risk with this approach is that the node for obvious reasons may have access to few observations from other nodes created at or shortly before the latest time

block, thus weakening detection stability. For other choices of l (i.e. $w_2 > 0$) the node will generally have access to more relevant observations, but will not try to detect a target passing the network at a certain time until w_2 time blocks later, thus weakening detection speed.

Now, if communication was instant and perfect, all nodes would at the current time block k have access to SMs from the entire time window (13), with indices belong to the **block-node matrix**

$$\mathcal{BN}_k = \{(q, j);\, k - w + 1 \leq q \leq k, 1 \leq j \leq N\}.$$

However, at time block k, node i only has access to SMs corresponding to the subset

$$\begin{aligned} \mathcal{BN}_{ki} &= \{(q, j);\, k - w + 1 \leq q \leq k, \\ &\quad j = i \text{ or } i \text{ has received } \mathrm{SM}_{qj} \text{ by time } kr\} \end{aligned} \tag{14}$$

of $\mathcal{BN}_k$, and of course, this limits detector performance. In view of (14), we notice that SMs older than $w-2$ time blocks should not be forwarded, since they are too old to be part of detection when they reach other nodes. This justifies the fourth routing rule (R4) of the previous section.

In our the Bayesian framework, we let $\mathcal{P}_m$ denote the prior probability of H_m and $f(\boldsymbol{\theta})$ the prior density of $\boldsymbol{\theta}$ on Θ_1 given H_1. A prior of $\boldsymbol{\theta}$ under H_0 is not necessary, since then $\boldsymbol{\theta}$ is not identifiable (cf. (1)–(3)). We assume

$$[\boldsymbol{\theta}|\boldsymbol{\theta} \in \Theta_1] \stackrel{d}{=} [\boldsymbol{X}|\boldsymbol{X}\boldsymbol{u}' > 0], \tag{15}$$

where $\boldsymbol{X} \in N_p(\boldsymbol{\mu}, \boldsymbol{\Sigma})$ has a p-dimensional multivariate Gaussian distribution, $\stackrel{d}{=}$ means equality in distribution and $\boldsymbol{u}'$ is the transpose of $\boldsymbol{u}$. Equivalently, we may rewrite (15) as

$$\begin{aligned} f(\boldsymbol{\theta}) &= \Phi\left(\frac{\boldsymbol{\mu}\boldsymbol{u}'}{|\boldsymbol{\Sigma}^{1/2}\boldsymbol{u}|}\right)^{-1} (2\pi)^{-p/2} \det(\boldsymbol{\Sigma})^{-1/2} 1_{\{\boldsymbol{\theta}\boldsymbol{u}' \geq 0\}} \\ &\cdot \exp\left(-0.5(\boldsymbol{\theta} - \boldsymbol{\mu})\boldsymbol{\Sigma}^{-1}(\boldsymbol{\theta} - \boldsymbol{\mu})'\right), \end{aligned} \tag{16}$$

where $\Phi(x) = \int_{-\infty}^{x} \exp(y^2/2)dy/\sqrt{2\pi}$ is the standard Gaussian cumulative distribution function and $\det(\boldsymbol{\Sigma})$ is the determinant of $\boldsymbol{\Sigma}$.

Note that f does not depend on time, as it is defined *given* $H_1 = H_1(lr)$ (i.e. given $\tau = lr$). In words, f gives the distribution of the parameter vector

given that the target is present at the network during time block l, and thus holds the a priori most likely parameter values in mean vector $\boldsymbol{\mu}$ with uncertainty according to covariance matrix $\boldsymbol{\Sigma}$.

Example 5 (Example 1, contd.) We put $\boldsymbol{\mu} = 0$ and

$$\boldsymbol{\Sigma} = \Sigma.$$

The prior uncertainty of signal strength after the change point is determined by Σ.

□

Example 6 (Example 2, contd.) We put

$$\begin{aligned} \boldsymbol{\mu} &= (0, 0, 0, \mu_4, \mu_5, \mu_6, 0), \\ \boldsymbol{\Sigma} &= \mathrm{BD}(\boldsymbol{\Sigma}_{1:3}, \boldsymbol{\Sigma}_{4:6}, \Sigma_7), \end{aligned}$$

where BD means block diagonal. In this case $(0, 0, 0)$ is a reference position of the network, where the target is most likely positioned at time $\tau = lr$, with prior covariance matrix $\boldsymbol{\Sigma}_{1:3}$. The a priori most likely velocity vector is (μ_4, μ_5, μ_6), with covariance matrix $\boldsymbol{\Sigma}_{4:6}$. The prior parameters (μ_4, μ_5, μ_6), $\boldsymbol{\Sigma}_{1:3}$ and $\boldsymbol{\Sigma}_{4:6}$ might all depend on knowledge of the geometry of the network and the surrounding area. Finally, Σ_7 reflects prior uncertainty of signal strength.

□

Let

$$\begin{aligned} \mathcal{L}_{ki}(\boldsymbol{\theta}) &= P_{\boldsymbol{\theta}}\left(\mathcal{BN}_{ki}, \{A_{qj};\, (q,j) \in \mathcal{BN}_{ki}\} | \tau = lr\right) \\ &= \textstyle\prod_{(q,j)\in\mathcal{BN}_{ki}} P\left(A_{qj} \text{ known}, A_{qj} | \tau = lr\right) \\ &\cdot \textstyle\prod_{(q,j)\in\mathcal{BN}_k \setminus \mathcal{BN}_{ki}} P\left(A_{qj} \text{ unknown} | \tau = lr\right) \end{aligned} \tag{17}$$

denote the likelihood function of node i's data (observations — i.e. estimated amplitudes and noise variances — from itself and other nodes) during the time window TW(k). In (17), we used the fact that A_{qj} for different time blocks and sensors are independent, which is a consequence of that all error terms ϵ_{tj} are independent. In fact, the likelihood also depends on all noise variances σ^2_{qj} for time blocks within the time window. However, we will replace each σ^2_{qj} by its estimate $\hat{\sigma}^2_{qj}$ and regard the latter as the true values. Therefore, we drop dependence on noise variances in the notation.

Let C_{mn} be the (fixed) cost of selecting H_n when H_m is true, and set $\pi_m = \mathcal{P}_m(C_{m,1-m} - C_{mm})$. By Bayes' rule, the cost-weighted a posteriori probabilities of H_0 and H_1 are (Hippenstiel, 2002, Chap. 4.2)

$$\begin{array}{rcl} P_{ki0} & = & \pi_0 \mathcal{L}_{ki}(\mathbf{0}) / \left(\pi_0 \mathcal{L}_{ki}(\mathbf{0}) + \pi_1 \int_{\Theta_1} f(\boldsymbol{\theta}) \mathcal{L}_{ki}(\boldsymbol{\theta}) d\boldsymbol{\theta} \right) \\ & = & 1 / \left(1 + \int_{\Theta_1} \Lambda_{ki}(\boldsymbol{\theta}) d\boldsymbol{\theta} \right) \end{array} \tag{18}$$

and $P_{ki1} = 1 - P_{ki0}$, where $\mathbf{0} = (0, \ldots, 0)$ is a parameter vector with zero components and $\Lambda_{ki}(\boldsymbol{\theta}) = \pi_1 f(\boldsymbol{\theta}) \mathcal{L}_{ki}(\boldsymbol{\theta}) / (\pi_0 \mathcal{L}_{ki}(\mathbf{0}))$ can be regarded as a joint likelihood ratio, taking the decision costs, the prior and likelihood from data into account[7]. This gives the Bayes' detector

$$\hat{m}_{ki} = \arg \max_{m=0,1} P_{ikm}, \tag{19}$$

which can equivalently be formulated as the decision rule

$$\int_{\Theta_1} \Lambda_{ki}(\boldsymbol{\theta}) d\boldsymbol{\theta} \quad \begin{array}{c} H_1 \\ > \\ < \\ H_0 \end{array} \quad 1 \tag{20}$$

or

$$\int_{\Theta_1} f(\boldsymbol{\theta}) \mathcal{L}_{ki}(\boldsymbol{\theta}) / \mathcal{L}_{ki}(\mathbf{0}) d\boldsymbol{\theta} \quad \begin{array}{c} H_1 \\ > \\ < \\ H_0 \end{array} \quad \frac{\pi_0}{\pi_1} = \frac{\mathcal{P}_0(C_{01} - C_{00})}{\mathcal{P}_1(C_{10} - C_{11})}.$$

We may adjust the **tolerance level** π_0/π_1 (i.e. the classification costs) in order to achieve a certain network performance. A higher tolerance level decreases the probability of false alarms but also the probability to detect a target. Notice in particular that when $C_{00} = C_{11} = 0$ and $C_{01} = C_{10} = 1$, this reduces to a maximum a posteriori (MAP) detector (Hippenstiel, 2002, Chap. 4.3).

In order to give a more explicit expression of the likelihood (17), we notice that when $(q, j) \in \mathcal{BN}_{ki}$,

$$\begin{array}{l} P\left(A_{qj} \text{ known}, A_{qj} | \tau = lr\right) \\ \quad = P(A_{qj} \text{ known} | \tau = lr) \cdot P(A_{qj} | A_{qj} \text{ known}, \tau = lr) \\ \quad = Q_{qlj}(\boldsymbol{\theta})^{\{j \neq i\}} P_{qk,ji} \cdot L_{qlj}(\boldsymbol{\theta}) / Q_{qlj}(\boldsymbol{\theta})^{\{j \neq i\}} \\ \quad = P_{qk,ji} L_{qlj}(\boldsymbol{\theta}), \end{array} \tag{21}$$

[7] In view of (1)–(3), the likelihood $\mathcal{L}_{ki}(\boldsymbol{\theta})$ is constant over Θ_0. Hence we may choose an arbitrary $\boldsymbol{\theta} \in \Theta_0$ instead of $\mathbf{0}$ in (18).

where

$$\begin{aligned} L_{qlj}(\boldsymbol{\theta}) &= P_{\boldsymbol{\theta}}(A_{qj}|\tau = lr), \\ Q_{qlj}(\boldsymbol{\theta}) &= P_{\boldsymbol{\theta}}(\mathrm{SM}_{qj} \text{ sent}|\tau = lr) \end{aligned}$$

and

$$P_{qk,ji} = \begin{cases} 1, & j = i, \\ P(\mathrm{SM}_{qj} \text{ received by } i \text{ by time } kr|\mathrm{SM}_{qj} \text{ sent}), & j \neq i. \end{cases}$$

When $(q, j) \in \mathcal{BN}_k \setminus \mathcal{BN}_{ki}$, we have that

$$P\left(A_{qj} \text{ unknown}|\tau = lr\right) = 1 - Q_{qlj}(\boldsymbol{\theta})P_{qk,ji}. \tag{22}$$

In principle, the transmission probabilities $P_{qk,ji}$ may depend on $\boldsymbol{\theta}$, since more SMs are sent when the target signals $s_i(t - \tau; \boldsymbol{\theta})$ grow strong, and this may delay reception of these messages. However, if we assume that send times of SMs (11) are negligible, we have that

$$P_{qk,ji} = P_{k-q,ji} \tag{23}$$

is independent of $\boldsymbol{\theta}$, since the routing and reception probability matrices $\boldsymbol{R}$ and $\boldsymbol{P}$ do not depend on time. In Appendix A-3 we give an outline for Monte Carlo estimation of $P_{s,ji}$ under these assumptions.

Inserting (21)-(23) into (17), we get

$$\begin{aligned} \mathcal{L}_{ki}(\boldsymbol{\theta}) &= \textstyle\prod_{(q,j)\in\mathcal{BN}_{ki}} L_{qlj}(\boldsymbol{\theta})P_{k-q,ji} \\ &\cdot \textstyle\prod_{(q,j)\in\mathcal{BN}_k\setminus\mathcal{BN}_{ki}}(1 - Q_{qlj}(\boldsymbol{\theta})P_{k-q,ji}) \\ &\propto \textstyle\prod_{(q,j)\in\mathcal{BN}_{ki}} L_{qlj}(\boldsymbol{\theta}) \\ &\cdot \textstyle\prod_{(q,j)\in\mathcal{BN}_k\setminus\mathcal{BN}_{ki}}(1 - Q_{qlj}(\boldsymbol{\theta})P_{k-q,ji}). \end{aligned} \tag{24}$$

In the last step of (24) we dropped the product of all terms $P_{k-q,ji}$, $(q, j) \in \mathcal{BN}_{ki}$, as these are independent of $\boldsymbol{\theta}$ and hence do not influence the Bayes' detector (19), since any multiplicative constant of $\mathcal{L}_{ki}(\cdot)$ cancels in the definition of the a posteriori probabilities (18).

Notice that the likelihood (24) adjusts for missing data (Little and Rubin, 2002), which may result from either unsent SMs or SMs that are sent but not received by i. This is quite necessary for detection whenever the nodes receive different target signals, such as in Example 2 — a consequence of this adjustment is facilitated detection of targets which pass by just a few nodes. Let us look at an example.

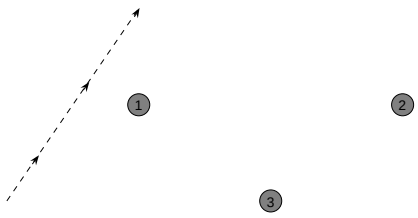

Figure 6: Target moving past a three node network, with a high received signal only at node 1.

Example 7 (Example 2, contd.) Consider the moving target situation given in Figure 6, where a target moves past a network of three nodes according to a given fixed $\breve{\boldsymbol{\theta}}$, and let us focus on the performance of node 3. The target moves close to node 1 but far from node 2, so $L_{ql1}(\breve{\boldsymbol{\theta}})$ can be expected to become large if SM_{q1} is received by 3, whereas $(1-Q_{ql2}(\breve{\boldsymbol{\theta}})P_{k-q,23})$ becomes large if SM_{q2} is not received by 3. On the contrary, $(1-Q_{ql1}(\breve{\boldsymbol{\theta}})P_{k-q,13})$ becomes small if SM_{q1} is not received by 3, as we would expect node 1 to send a message if there is a target in its vicinity. Hence, the overall likelihood $\mathcal{L}_{ki}(\breve{\boldsymbol{\theta}})$ would for this $\breve{\boldsymbol{\theta}}$ tend to get the largest when node 3 has received observations from node 1 but not from node 2.

□

Assuming that each $\hat{\sigma}_{qj}$ is the true value of σ_{qj}, in Appendix A-2 we derive the formulas

$$L_{qlj}(\boldsymbol{\theta})/L_{qlj}(\mathbf{0}) = \exp(-s^2_{q-l,j}(\boldsymbol{\theta})/(2\hat{\sigma}^2_{qj}))\sum_{n=0}^{\infty}(A_{qj}s_{q-l,j}(\boldsymbol{\theta})/(2\hat{\sigma}^2_{qj}))^{2n}(n!)^{-2} \tag{25}$$

and

$$\begin{aligned} Q_{qlj}(\boldsymbol{\theta}) &= \exp(-s^2_{q-l,j}(\boldsymbol{\theta})/(2\hat{\sigma}^2_{uj})) \\ &\cdot \sum_{n=0}^{\infty} 2^{-n}(s_{q-l,j}(\boldsymbol{\theta})/\hat{\sigma}_{uj})^{2n}(n!)^{-1}\Gamma^{\mathrm{inc}}(\lambda^2/2, n+1) \end{aligned} \tag{26}$$

where $\Gamma^{\mathrm{inc}}(x,a) = \int_x^{\infty} y^{a-1}\exp(-y)dy/(a-1)!$ is the (regularized) upper

incomplete gamma function. Notice that $\hat{\sigma}_{qj}$ is unknown to i at time kr when $(q,j) \notin \mathcal{BN}_{ki}$. Hence we replace $\hat{\sigma}_{qj}$ by $\hat{\sigma}_{uj}$ in (26), where $u = u(q,k,j,i)$ is the time block closest to q from which i has received a SM from j by time kr. The choice of λ in (8) guarantees that $\hat{\sigma}_{uj}$ is well defined with probability one[8].

Summarizing, in order to compute $\hat{m}_{ki}$, we must evaluate the integral $\int_{\Theta_1} \Lambda_{ki}(\boldsymbol{\theta}) d\boldsymbol{\theta}$ and insert into (18). Formulas (16) and (24) imply that

$$\begin{aligned} \Lambda_{ki}(\boldsymbol{\theta}) &= 1_{\{\boldsymbol{\theta}\boldsymbol{u}' \geq 0\}} (\pi_1/\pi_0) \Phi\left(\boldsymbol{\mu}\boldsymbol{u}'/|\boldsymbol{\Sigma}^{1/2}\boldsymbol{u}|\right)^{-1} \\ &\cdot (2\pi)^{-p/2} \det(\boldsymbol{\Sigma})^{-1/2} \exp\left(-0.5(\boldsymbol{\theta}-\boldsymbol{\mu})\boldsymbol{\Sigma}^{-1}(\boldsymbol{\theta}-\boldsymbol{\mu})'\right) \\ &\cdot \prod_{(q,j)\in\mathcal{BN}_{ki}} L_{qlj}(\boldsymbol{\theta})/L_{qlj}(\mathbf{0}) \\ &\cdot \prod_{(q,j)\in\mathcal{BN}_k\setminus\mathcal{BN}_{ki}} \left(1 - Q_{qlj}(\boldsymbol{\theta}) P_{k-q,ji}\right) / \left(1 - \exp(-\lambda^2/2) P_{k-q,ji}\right), \end{aligned} \tag{27}$$

and the last two rows on the right-hand side are obtained from (25), (26) and the Monte Carlo estimate of $P_{s,ji}$ outlined in Appendix A-3.

The integral $\int_{\Theta_1} \Lambda_{ki}(\boldsymbol{\theta}) d\boldsymbol{\theta}$ may be approximated by various numerical methods, and in Appendix A-4 we describe one such method based on a quadratic approximation of Λ_{ki}. For the moving target example, this method is very similar to the maximum likelihood (ML) estimation of target trajectories proposed by Choi et al. (2008), although that model was based on a Neyman-Pearson (rather than a Bayesian) formulation.

A word of warning should however be raised about the quadratic approximation method, as it involves creating a maximum a posteriori (MAP) estimate of the parameter vector and therefore suffers from the same kind of problems as the ML estimate of the mentioned work (Choi et al., 2008, p. 1274f) — if a node has too few observations available, Λ_{ki} may lack a unique maximum. To see this, consider the moving target example with $N = 1$. Since the node only has access to its own observations and these observations only measure incoming signal strength, the node will be unable to determine from which direction a possible target passes, and also whether its observations correspond to a remote and loud target, or a close and quiet target.

[8] According to (8), the time between two sent SMs has a geometric distribution with success probability $\exp(-\lambda^2/2)$ and expected value $\exp(\lambda^2/2)$ under H_0. Under H_1, the expected value is smaller. Hence, λ should not be too large, to guarantee that $\hat{\sigma}_{uj}$ does not differ substantially from $\hat{\sigma}_{qj}$.

Hopefully, the prior distribution f will dominate and thus serve to remedy these problems, but even for large networks the same class of problems may arise given little communication. For this reason, when using this method one may wish to take one or more of the following steps to increase the observations available at each node at the time of classification: [1] decrease λ, [2] increase w_2, and/or [3] have the nodes only do a new classification after they have received new observations from others.

2.4 Alarms

In this section, we give the rules for how and when the network may set off the alarm. The idea is that this will happen whenever one of the decision active nodes realizes that a sufficiently large portion of decision active nodes have reached consensus in rejecting the null hypothesis.

In more detail, each decision active node $i \in \mathcal{A}$ keeps track of the most recent hypothesis classification of all decision active nodes (including i itself) which it has received through incoming messages. Let N_{ki}^{alarm} be the number of $j \in \mathcal{A}$ from which the latest classifier $\hat{m}_{qj}$ sent to i at the end of time block k is 1 (i.e. H_1). In case i has not received any SM from j within the time window TW(k), i assumes that j's current classifier is 0 (i.e. H_0). Let

$$P_{ki}^{\text{cons}} = N_{ki}^{\text{alarm}}/|\mathcal{A}|$$

be the fraction of decision active nodes that, by time block k, have reported a classifier H_1 to i. Notice that P_{ki}^{cons} quantifies the amount of consensus among the decision active nodes, with a large degree of consensus when P_{ki}^{cons} is close to 0 or 1.

Assume we observe the network during the time interval

$$t \in \mathcal{T} = \{1, \ldots, t^{\max}\},$$

for some $t^{\max} = k^{\max} r \leq \infty$. The time of alarm, $t^{\text{alarm}} \in \mathcal{T} \cup \{\infty\}$, is the first time point at which some decision active node sets off the alarm. We have

$$t^{\text{alarm}} = k^{\text{alarm}} r,$$

where

$$k^{\text{alarm}} = \min\{k; \text{TW}(k) \subset \{1, \ldots, k^{\max}\} \text{ and } P_{ki}^{\text{cons}} \geq \alpha \text{ for at least one } i\}, \tag{28}$$

where $0 < \alpha \leq 1$ is a prespecified number and $\min \emptyset = \infty$. In words, an alarm is sent when some node notices that a fraction α or more of all decision active nodes have detected a target. The larger α is, the larger is the requirement for setting off the alarm.

The reader should notice that the consensus requirement presented here is an ad hoc requirement, as each decision active node actually creates its classification using observations from all nodes in the network — which means that given enough communication we should expect every classifying node to reach the same classification. However, we have chosen to include this possibility to require a rough consensus before the alarm is set off, as there are a number of practical concerns for which this may be convenient. In practice, there is always the possibility of corrupted messages (i.e. messages which are received but misinterpreted rather than completely lost) and numerous other *local* errors which may produce faulty classifications at only one or a small number of nodes. Even though our model does not take these into account specifically, it may be of interest to study the drop in detector performance (most notably detection speed) for various values of α if it will indeed be used for increased robustness.

If $\alpha = 1/|\mathcal{A}|$, that is the alarm is set off immediately as a node rejects H_0, part MP4 of the message protocol may be dropped since consensus checks are unimportant under this scheme.

2.5 Performance Measures

In this section we will look at some statistics of interest which measure the performance of a given network. Since the time blocks is generally a more interesting time scale than the time points, we will for convenience look at the time block rather than the time point in which alarms are set off. This is reasonable since a maximum of one classification is performed at each time block for each node.

For target models where a target shows up at the network and then

disappears again if not detected, such as in the moving target case (Example 2), detection performance is our chief interest. For a given target vector $\boldsymbol{\theta} \in \Theta_1$ and time τ, we define the **probability of detection**

$$P_{\mathrm{D}}(\boldsymbol{\theta}) = P_{\boldsymbol{\theta}}(k^{\mathrm{alarm}} \leq k^{\max} | \tau)$$

and the **probability of false alarm**

$$P_{\mathrm{FA}} = P_{H_0}(k^{\mathrm{alarm}} \leq k^{\max})$$

which lets us create the **receiver operating characteristic** (ROC) curve in which $P_{\mathrm{D}}(\boldsymbol{\theta})$ is plotted as a function of P_{FA} as the tolerance π_0/π_1 is varied (Hippenstiel, 2002, Chap. 4.10). Notice that $P_{\mathrm{D}}(\boldsymbol{\theta})$ and P_{FA} are both increasing functions of $k^{\max}$. This is to some extent inconvenient, since the choice of $k^{\max}$ may seem arbitrary. In general, it quantifies the maximum possible acceptable delay for detection of the target and is often related to w. For instance, in Example 2, both $k^{\max}$ and w could be related to the time it typically takes for the target to pass the network.

The reader may note that the above definition for the probability of detection differs from the traditional,

$$P_{\mathrm{D}} = P_{H_1(\tau)}(k^{\mathrm{alarm}} \leq k^{\max}) = \int_{\Theta_1} P_{\mathrm{D}}(\boldsymbol{\theta}) f(\boldsymbol{\theta}) d\boldsymbol{\theta},$$

in which the target vector is assumed to be drawn according to the prior distribution f. With this formulation, when $C_{00} = C_{11} = 0$ the **average cost of misclassification** equals

$$C = \pi_0 P_{\mathrm{FA}} + \pi_1 (1 - P_{\mathrm{D}}).$$

Complementary performance criteria, used when a target shows up at the network and stays there indefinitely such as in the change point case (Example 1), are the **average maximum delays**

$$\begin{aligned} D_1 &= \mathrm{E}\left\{\min(k^{\mathrm{alarm}}, k^{\max}) | H_1(\tau)\right\}, \ 1 \leq \tau \leq k^{\max} r, \\ D_0 &= \mathrm{E}\left\{\min(k^{\mathrm{alarm}}, k^{\max}) | H_0\right\} \end{aligned} \tag{29}$$

which measures detection speed and the time between false alarms respectively. It is reasonable to make τ small and $k^{\max}$ very large in (29), e.g. taking the limit $k^{\max} \to \infty$.

For constant noise variance ($\sigma_{qj}^2 = \sigma^2$) we can also define the **target signal-to-noise ratio** (TSNR). For instance,

$$\text{TSNR} = \theta^2/\sigma^2 = 20\log_{10}(\theta/\sigma) \text{ dB}$$

in Example 1 and

$$\text{TSNR} = \theta_7^2/\sigma^2 = 20\log_{10}(\theta_7/\sigma) \text{ dB}$$

in Example 2, with the TSNR given in unitless and decibel scale respectively.

3 Simulation Results

Due to time constraints, a small number of simulations has been performed in order to present indicative results. In all of these, we have used the common parameters from Table 1 unless specifying otherwise. We use the network architecture from Example 3 with all nodes decision active, but with the simplifying assumption that transmission times are negligible (i.e. all messages sent during a time block have arrived at the receiving nodes at the end of the time block). We assume constant noise variance and thus set $\sigma_{qj}^2 = \sigma^2 = 1$. The network is to detect a moving target (Example 2) — here assumed to be a large watercraft — and because of prior knowledge of the surrounding area the nodes' have been configured to expect targets moving in from the "north" (i.e. along the y-axis) through the middle of the network at a speed of around 5 meters per second (1 second = 1 time block). We place the Cartesian coordinate system so that the origin (0,0,0) is at the middle node (node 3) and specify $\boldsymbol{\mu}$ accordingly with prior uncertainty according to the covariance matrix $\boldsymbol{\Sigma}$. Since a target with such a high speed can be expected to have passed the system in less than 30 seconds we put $w = 30$, and since a fast-moving target needs to be detected quickly we set $w_2 = 0.9$[9]

After a "burn-in" period to achieve stationarity, the network starts to actively detect a possible target and keeps doing so for 120 time blocks per simulation iteration. The transmission probability matrix $\boldsymbol{P}_s$ is estimated according to the Monte Carlo method described in Appendix A-3 with 100,000 iterations, and further we use the approximation method described in Appendix A-4 during the classification phase. Because this network has shown symptoms of the problems associated with MAP estimation discussed at the very end of Section 2.3, we impose the restriction that no classification is to be carried out unless a node has received at least one new observation from another node during the last time block.

We perform a number of simulations under the null hypothesis (no target)

[9]The reader should be aware that the choices of parameter values corresponding to node and network configurations are in no way found to be optimal, but are specified according to intuition. Further extensive simulation studies are needed to find good (but likely sub-optimal) configurations.

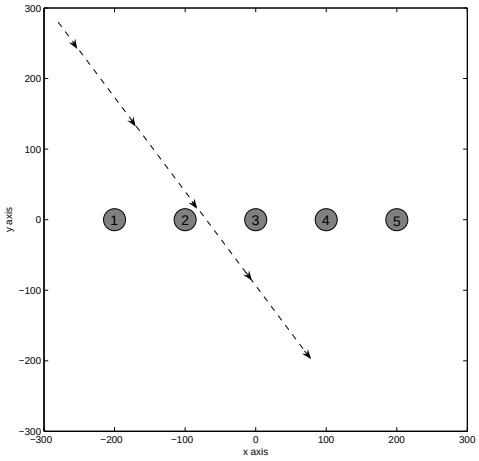

Figure 7: Trajectory of the target used in the simulations relative to the detector nodes, in the x-y-plane.

and the alternative hypothesis (target), and when we simulate under the alternative we use the same target trajectory, which corresponds to a target traveling through the network at a speed of 5 meters per second roughly "south-east" in the x-y-plane. See Figure 7.

For all of these simulations we will present only ROC plots, which are produced by varying the tolerance level π_0/π_1 from 0 to 400 in small steps for the same simulated data. For each of these tolerance levels the probability of detection ($P_{\mathrm{D}}(\boldsymbol{\theta})$) and that of false alarm (P_{FA}) is estimated from simulation data, and the resulting points in the $P_{\mathrm{FA}} \times P_{\mathrm{D}}(\boldsymbol{\theta})$-plane are plotted.

3.1 Simulation 1: Target Signal-to-Noise Ratio

Let us start with a simulation where the performance of the detector model is studied for varying degrees of the target signal-to-noise ratio. We vary the

$$\mathrm{TSNR} = 20 \log_{10}(\theta_7/\sigma) \text{ dB}$$

according to $\mathrm{TSNR} = \{10, 12, 14, 16\}$ dB, which for $\sigma = 1$ produces target signal strengths $\theta_7 = \{3.162, 3.981, 5.012, 6.310\}$. Further we allow each node to set off the alarm individually ($\alpha = 1/N = 0.2$). For comparison, we also

Network topology:	No. of nodes	N	5
	Node positions	$\boldsymbol{x}$	$\begin{bmatrix} -200 & 0 & 0 \\ -100 & 0 & 0 \\ 0 & 0 & 0 \\ 100 & 0 & 0 \\ 200 & 0 & 0 \end{bmatrix}$
Observations sampling:	Frequency	ω	$\pi/24$
	Phase	ϕ	$\pi/2$
	Noise variance	σ^2	1
	Sampling rate	r	10000
	Assumed target size	δ	20
Communication:	Communication range	$d_{\max}$	100
	Reception probability	P	0.9
	Send threshold	λ	$\sqrt{-2\log(0.1)} \approx 2.15$
	Decision active nodes	$\mathcal{A}$	$\{1, 2, 3, 4, 5\}$
	Consensus threshold	α	0.2
	Routing matrix	$\boldsymbol{R}$	$\begin{pmatrix} 0 & 1 & 1 & 1 & 0 \\ 0 & 0 & 1 & 1 & 0 \\ 0 & 1 & 0 & 1 & 0 \\ 0 & 1 & 1 & 0 & 0 \\ 0 & 1 & 1 & 1 & 0 \end{pmatrix}$
Prior:	A priori mean	$\boldsymbol{\mu}$	$(0, 0, 0,\ 0, -5, 0,\ 0)$
	A priori covariance		
		$\boldsymbol{\Sigma}_{1:3}$	$\begin{pmatrix} 200^2 & 0 & 0 \\ 0 & 200^2 & 0 \\ 0 & 0 & 5^2 \end{pmatrix}$
		$\boldsymbol{\Sigma}_{4:6}$	$\begin{pmatrix} 5^2 & 0 & 0 \\ 0 & 5^2 & 0 \\ 0 & 0 & 1 \end{pmatrix}$
		Σ_7	10^2
Time parameters:	Time window	w_1	30
		w_2	0
	Simulation length	$k^{\max}$	120
Target:	True target vector	$\boldsymbol{\theta}$	$(-70, 0, 0,\ 3, -4, 0,\ \theta_7)$
	Reference time	τ	$70r$

Table 1: Common simulation parameters

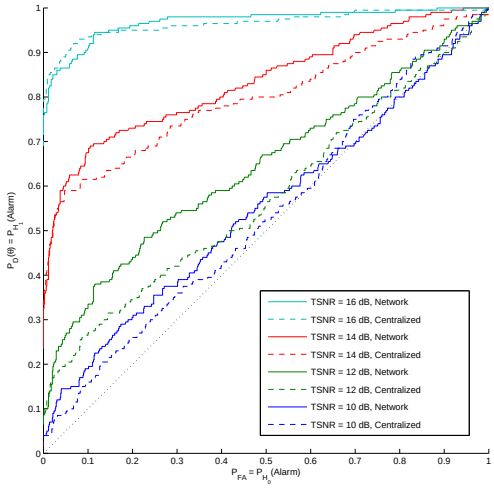

Figure 8: ROC curves for various target signal-to-noise ratios (TSNRs), for the network and the centralized detector.

check the performance of the corresponding detector which without delay has access to all observations from all nodes at all times, and which performs a new classification at the end of every time block (but in every other respect functions as a regular node). Somewhat inaccurately we choose to call this the **centralized detector** of a particular setting.[10] For each of the four TSNR levels we perform 200 simulations under the alternative hypothesis, and in addition we perform 800 simulations under the null hypothesis. From these $P_{\mathrm{D}}(\boldsymbol{\theta})$ and P_{FA} is estimated as the fraction of alarms. The resulting ROC curves are shown in Figure 8.

As expected, target detection becomes easier with increasing TSNR. However, neither of the curves are satisfactory (except maybe the one corresponding to the highest TSNR) since they all show that in order to get a network with an acceptable detection probability $P_{\mathrm{D}}(\boldsymbol{\theta})$, we must accept a very high false alarm rate P_{FA}; recall that we have simulated only during 120 seconds, so even a seemingly low false alarm rate of e.g. $P_{\mathrm{FA}} = 0.01$ corresponds to an expected time between false alarms of $120/0.01 = 12000$ seconds, or 3.33

[10] Inaccurate because this detector actually operates on the aggregated observations and not the underlying raw observation samples, and so might perform slightly worse than the true centralized detector.

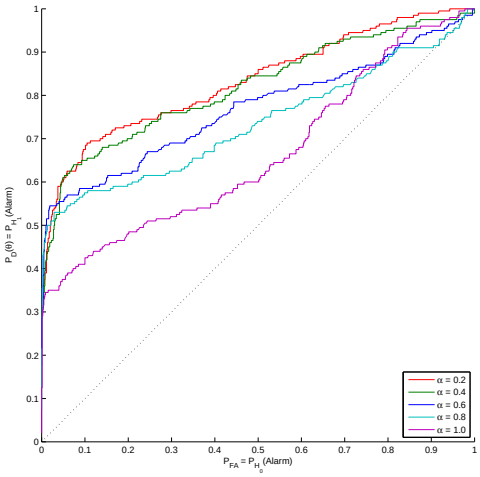

Figure 9: ROC curves for networks operating with consensus level α.

hours. Whether this high false alarm rate stems from something specific to the model, from the instability issues mentioned or some as of now unknown error source remains to be determined.

Further, it also seems that the network consistently performs better than the centralized detector which has access to all observations. While it is currently unknown what causes this rather unintuitive behavior, a plausible theory is that it is simply an effect of that the centralized detector performs a classification after *every* time block as opposed to the individual network nodes which only classify after reception of messages from other nodes, as mentioned at the beginning of this chapter. Thus, the centralized detector can be expected to have a higher rate of false alarms as an effect of classifications made on weak premises.

3.2 Simulation 2: Consensus

For the same data as above, we look only on the case TSNR = 14 dB and instead vary the consensus limit $\alpha = \{0.2, 0.4, 0.6, 0.8, 1.0\}$ in order to see if any clear difference can be discerned. The result is found in Figure 9.

At first glance, little to no gain seems to be achieved by requiring the nodes to wait for network consensus. Most likely, the reason is that the

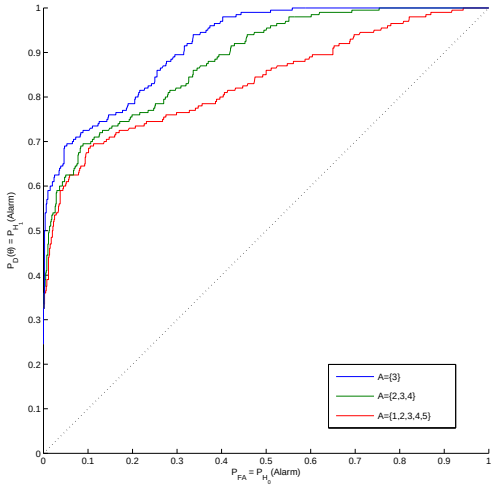

Figure 10: ROC curves for networks with $(\mathcal{A}, \boldsymbol{R})$ settings.

increased robustness of networks with $\alpha > 0.2$ (as compared to the networks with $\alpha = 0.2$) lets us decrease the tolerance level π_0/π_1 in order to achieve the same false alarm rate, which however does not increase the detection probability above the original level. However, the reader should also notice that the figure shows an increased detection probability in some of the curves for the very lowest values of P_{FA}. If this result can be shown to hold in more extensive simulation studies, the ad hoc consensus method should be seriously considered since it as mentioned before also can be expected to bring more robustness into the detector network when not all model assumptions hold.

3.3 Simulation 3: Decision Active Nodes

Yet again for the same data as in Simulation 1 and the case TSNR = 14 dB, we compare detection performance between the three proposed pairs of settings for decision active nodes and the routing matrix $(\mathcal{A}, \boldsymbol{R})$ in Example 3. The results are shown in Figure 10.

Somewhat surprisingly, the plot shows that the setting with the fewest number of decision active nodes has the best performance. Then again, the target passes the network between nodes 2 and 3, and the fact that the network performs best when nodes which we know to be far from the target

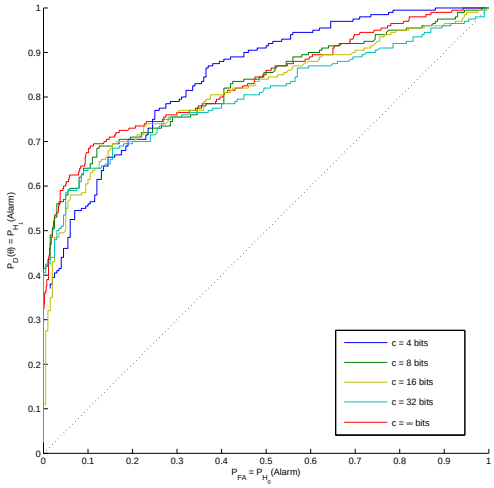

Figure 11: ROC curves for networks using c bits as quantization levels.

are exempt from making classifications is rather intuitive. Nevertheless this, as well as the results of Simulation 1, does show that the network model may suffer from problems associated with mass significance. That is, there may be a positive correlation between the number of classifications performed, i.e. the number of decision active nodes, and the rate of false alarms. Then again, we have only looked at the results of one network structure and one target setting, so it is too early to make strong conclusions.

3.4 Simulation 4: Data Quantization

As mentioned in Section 2.2.3, each A_{qj} and $\hat{\sigma}_{qj}$ is quantized to c bits as they are sent in a SM. We have so far assumed that the loss of precision during quantization is negligible, so an important issue is to find under what circumstances this assumption holds. For this reason, we have performed a number of simulations in which the amplitude and noise standard deviation is quantized to $c = \{4, 8, 16, 32\}$ bits according to the uniform quantization method described in Appendix A-6, with $A^{\max} = 100$ and $\hat{\sigma}^{\max} = 10$. For each of the quantization levels, 200 iterations are simulated under H_1, again with TSNR = 14 dB, and 200 iterations are simulated under H_0. The resulting ROC curves are compared to the corresponding curve of Simulation

1 (which uses the unquantized observations), and are plotted in Figure 11.

While the curve corresponding to $c = 4$ bits seems to perform significantly worse (at least for low false alarm rates), it seems that for this network structure as little as 8 bits may be enough to achieve the same performance as the infinite precision case. Then again the curve corresponding to $c = 16$ bits might serve as warning that in order to have good detection performance while having a low false alarm rate, a higher quantization may be needed. Once again, the conclusion which must be drawn is that further studies are needed in order to ascertain these finds.

4 Concluding Remarks

We have presented an model for a wireless underwater detector network to be used for sequential detection problems. The model utilizes the classic Bayes' detector framework applied for a decentralized detection network with a message-passing algorithm which censors for low observation signal-to-noise ratios. Nevertheless, this is a first model in this project, and it does present us with a number of problems which will have to be addressed in further research.

A general problem with the classification method used in our model is that maximum a posteriori methods (as well as related likelihood-based methods) is non-robust, in the sense that performance may be impaired if not all assumptions hold which the model rests upon. This problem becomes strikingly clear for our model, because a failure in a single node (e.g. breakdowns in the communication equipment, corrupted observations, etc.) is likely to affect the performance of the entire network via the message-passing scheme used. Steps will need to be taken to make the model more robust.

Another problem stems from the computationally intense classification phase. Because of the relative component inexpense we have so far assumed that the nodes have unlimited computational capacity, but one quickly realizes that this assumption may fail to hold. Each node may be required to produce as much as one approximation of a multidimensional integral *per second*, which may be difficult even for the fastest of microprocessors. As discussed above the rather simple approximation method proposed in Appendix A-4 comes with a number of problems which makes it unstable, so in order to use our model one needs to either find a better approximation method, or come up with a simplified classification scheme.

Further, we should address the model assumed for observation sampling: a deterministic signal with one center frequency corrupted by external white noise. This may prove to be an oversimplified model for sampled sound, and one should try modeling the target signal by means of a stochastic process involving a whole spectrum of frequencies. Furthermore, the error terms are likely to be correlated for high sampling rates, in particular when other noise sources are present within the network.

Further research is also needed for the moving target detection problem, as it (as has been noted) can currently only handle one target at each time. The field of research known as multiple target tracking might serve as a guideline (see e.g. Stone et al., 1999). Whether constant target velocity and direction is a good enough approximation of the actual target trajectory may be determined for each application. If not, it may be further parametrized, but this is outside the scope of this thesis.

In order to reduce disturbance sources we may consider to perform Simulations 2–4 (among other possible simulations) using the same target signal at all nodes (resulting in identical distribution of A_{qj} and $\hat{\sigma}_{qj}$ for all $j = 1, \ldots, N$). Further, since the most important portions of the ROC curves are those which correspond to low false alarm rates, extensive simulations are required to get clear estimates of these.

An interesting energy savings paradigm to explore besides *sensor censoring* is the proposal to have a number of the nodes go into *sleeping mode* at times when little activity is to be expected. In this mode the node may turn off functions vital to detection either for a prespecified period of time or until a *wake-up call* is received from some other active node. A brief outline is given by Chamberland and Veeravalli (2007, p. 24), and Appadwedula et al. (2005) study the performance of a Neyman-Pearson model utilizing censoring and sleeping for a simple decentralized detection problem.

It should be stressed that for good performance it is necessary to find good configuration settings for the detector network and its nodes. Since the model is complex this will likely need to be done through extensive simulations. A number of additional performance measures needs to be defined, in order to quantify the model's communication requirements, etc.

Finally, and perhaps most important, the greater promise of a self-configuring fully decentralized detector network should be mentioned. If it became possible to create a scalable architecture in which new nodes, with little to no preconfiguration, could be easily and quickly deployed into an existing network and there establish themselves as new network nodes without outside help, the reduction of deployment costs would be tremendous. The achievement of something to this effect will likely require extensive research combining a multitude of academic disciplines.

Appendices

A-1 Distribution of Feature Vector

We will only give the proof when $k = 1$ as the proof for the general case is analogous. For simplicity of notation we write $Y_a = Y_{a1i}$, $s = s_{1-l,i}(\boldsymbol{\theta})$, $\epsilon_t = \epsilon_{ti}$ and $\sigma = \sigma_{1i}$. According to (4) we have

$$\begin{aligned} Y_1 &= (2/r)\left\{s\left[\cos(\phi)\textstyle\sum_{t=1}^{r}\cos^2(\omega t) - \sin(\phi)\sum_{t=1}^{r}\sin(\omega t)\cos(\omega t)\right]\right. \\ &+ \left.\textstyle\sum_{t=1}^{r}\epsilon_t\cos(\omega t)\right\}, \end{aligned} \tag{A.1}$$

and from (5), we get

$$(2/r)\textstyle\sum_{t=1}^{r}\epsilon_t\cos(\omega t) \in N(0, \sigma^2(2/r)\sum_{t=1}^{r}\cos^2(\omega t)). \tag{A.2}$$

Further, relying on the assumption that $G \propto r$ and treating the above sums as Riemann sums, we get

$$\begin{aligned} (2/r)\textstyle\sum_{t=1}^{r}\cos^2(\omega t) &= (2/r)\int_0^r \cos^2(\omega t)dt + O(r^{-2}) \\ &= 1 + O(r^{-1}), \\ (2/r)\textstyle\sum_{t=1}^{r}\sin(\omega t)\cos(\omega t) &= O(r^{-1}). \end{aligned} \tag{A.3}$$

This and the Lindeberg-Feller Central Limit Theorem imply that $Y_1 \xrightarrow{\text{d}} N\left(s\cos(\phi), \sigma^2\right)$ as $r \to \infty$, and it can be similarly shown that $Y_2 \xrightarrow{\text{d}} N\left(-s\sin(\phi), \sigma^2\right)$ as $r \to \infty$. Asymptotic bivariate normality of $\boldsymbol{Y}$ follows in a similar manner using the Cramér-Wold device, i.e. showing that all linear combinations of Y_1 and Y_2 are asymptotically normal. To see asymptotic independence of Y_1 and Y_2, notice that

$$\mathrm{Cov}(Y_1, Y_2) = \mathrm{E}(Y_1Y_2) - \mathrm{E}(Y_1)\mathrm{E}(Y_2) = \sigma^2(2/r)\sum_{t=1}^{r}\sin(\omega t)\cos(\omega t) = O(r^{-1}). \tag{A.4}$$

We have thus shown that (6) holds when r is large.

A-2 Distribution of Estimated Amplitudes

Let

$$f(x; m) = \frac{x^{m/2-1}\exp(-x/2)}{2^{m/2}(m/2-1)!}, \quad x > 0,$$

be the density of a $\chi^2(m)$-distributed random variable with m degrees of freedom. A noncentral $\chi^2(m,\eta)$-distributed random variable with m degrees of freedom and noncentrality parameter η has density (Abramowitz and Stegun, 1972, p. 942)

$$f(x;m,\eta) = \sum_{n=0}^{\infty} \frac{\exp(-\eta/2)(\eta/2)^n}{n!} f(x;m+2n), \quad x>0. \tag{A.5}$$

It follows from (6) that $\sigma^{-2}A^2 \tilde{\in} \chi^2(2, s^2/\sigma^2)$, where, for simplicity of notation, we treat the asymptotic (non-approximate) case and write $A = A_{qj}$, $s = s_{q-l,j}(\boldsymbol{\theta})$ and $\sigma^2 = \sigma^2_{qj}$. Hence $f_A(x;s,\sigma^2)$, the density of A, has a Rice distribution

$$f_A(x;s,\sigma^2) = \frac{2x}{\sigma^2} f(x^2/\sigma^2; 2, s^2/\sigma^2)$$

and the likelihood ratio

$$\begin{aligned} f_A(x;s,\sigma^2)/f_A(x;0,\sigma^2) &= f(x^2/\sigma^2;2,s^2/\sigma^2)/f(x^2/\sigma^2;2,0) \\ &= \textstyle\sum_{n=0}^{\infty} \exp(-s^2/(2\sigma^2))(s^2/(2\sigma^2))^n (n!)^{-1} f(x^2/\sigma^2; 2+2n)/f(x^2/\sigma^2;2) \\ &= \exp(-s^2/(2\sigma^2)) \textstyle\sum_{n=0}^{\infty} (xs/(2\sigma^2))^{2n}(n!)^{-2} \end{aligned}$$

implies (25), since

$$L_{qlj}(\boldsymbol{\theta})/L_{qlj}(\mathbf{0}) = f_A(A_{qj}; s_{q-l,j}(\boldsymbol{\theta}), \hat{\sigma}^2_{qj})/f_A(A_{qj};0,\hat{\sigma}^2_{qj}).$$

Introduce the distribution functions $F(x;m) = \int_0^x f(y;m)dy$, $F(x;m,\eta) = \int_0^x f(y;m,\eta)dy$ and $F_A(x;s,\sigma^2) = \int_0^x f_A(y;s,\sigma^2)dy$. Then, integrating (A.5), we find that

$$\begin{aligned} 1 - F_A(x\sigma;s,\sigma^2) &= 1 - F(x^2;2,s^2/\sigma^2) \\ &= \exp(-s^2/(2\sigma^2)) \textstyle\sum_{n=0}^{\infty} (s^2/(2\sigma^2))^n (n!)^{-1}(1 - F(x^2;2+2n)). \end{aligned}$$

The last equation implies (26), noticing that

$$Q_{qlj}(\boldsymbol{\theta}) = 1 - F_A(\lambda\hat{\sigma}_{uj}; s_{q-l,j}(\boldsymbol{\theta}), \hat{\sigma}^2_{uj}).$$

A-3 Approximation of Transmission Probabilities

In this appendix, we derive an approximation for $P_{s,ji}$ assuming negligible send times. These probabilities are needed for computing $\Lambda_{ki}(\boldsymbol{\theta})$ in (27).

Let us start by introducing $\boldsymbol{I}_{qk} = (I_{qk,ji})_{j,i=1}^{N}$, where

$$I_{qk,ji} = \begin{cases} 1, & j = i, \\ 1_{\{\mathrm{SM}_{qj} \text{ received by } i \text{ during blocks } (q,k] | \mathrm{SM}_{qj} \text{ sent}\}}, & j \neq i. \end{cases}$$

We assume that $T_{ki}^{\mathrm{send}} \ll T_{ij}^{\mathrm{rec}}$ for all k, i and j, so that the delays due to finite send rate are negligible in comparison to delays due to the speed of sound in water. Under this assumption we notice that the distribution of $I_{qk,ji}$ is stationary in the sense that $I_{qk,ji} \stackrel{d}{=} I_{q+c,k+c,ji}$ for all integers c. Thus, without loss of generality we put $c = -q$ and use the identity

$$\mathrm{E}(\boldsymbol{I}_{0,k-q}) = \boldsymbol{P}_{k-q} \tag{A.6}$$

in order to compute $\boldsymbol{P}_s = (P_{s,ji})_{j,i=1}^{N}$ for $s = 0, \ldots, w-1$.

We introduce the reception indicator matrices $\boldsymbol{I}_q = (I_{q,ji})_{j,i=1}^{N}$, where

$$I_{q,ji} = \begin{cases} 0, & j = i, \\ 1_{\{\mathrm{SM}_{qj} \text{ received (without routing) by } i \text{ at some time} > qr | \mathrm{SM}_{qj} \text{ sent}\}}, & j \neq i, \end{cases}$$

and notice that all $I_{q,ji}$ are independent, with

$$P(I_{q,ji} = 1) = \begin{cases} 0, & j = i, \\ P_{ji}, & j \neq i. \end{cases} \tag{A.7}$$

We also need the $N \times N$ delay matrices $\boldsymbol{D}_v = (D_{v,ji})$, defined by

$$D_{v,ji} = 1_{\{(v-1)r < T_{ji}^{\mathrm{rec}} \leq vr\}}$$

for $v = 0, 1, 2, \ldots$. Notice in particular that $\boldsymbol{D}_0$ is the identity matrix of order N.

Having generated all matrices $\boldsymbol{I}_0, \ldots, \boldsymbol{I}_{w-1}$ from (A.7), we put $\boldsymbol{I}_{00} = \boldsymbol{D}_0$ and then recursively compute

$$\boldsymbol{I}_{0s} = \min\left(\boldsymbol{1}_{N\times N}, \boldsymbol{I}_{0,s-1} + \boldsymbol{I}_0 \cdot \boldsymbol{D}_s + \sum_{\nu=1}^{s-1} \left((\boldsymbol{I}_{0\nu} - \boldsymbol{I}_{0,\nu-1}) \cdot \boldsymbol{R}\right)(\boldsymbol{I}_\nu \cdot \boldsymbol{D}_{s-\nu})\right) \tag{A.8}$$

for $s = 1, \ldots, w-1$, where $\boldsymbol{1}_{N\times N}$ is an $N \times N$-matrix with 1 entries and $\min(\boldsymbol{A}, \boldsymbol{B})$ refers to elementwise minimum of matrices $\boldsymbol{A}$ and $\boldsymbol{B}$.

It remains to estimate $\boldsymbol{P}_s$ by Monte Carlo from (A.6) by repeatedly generating $\{\boldsymbol{I}_{0s}\}_{s=0}^{w-1}$ according to (A.7) and then using (A.8). After generating K matrices $\boldsymbol{I}_{0s}^{(a)}$ we get

$$\hat{\boldsymbol{P}}_s = K^{-1} \sum_{a=1}^{K} \boldsymbol{I}_{0s}^{(a)}. \tag{A.9}$$

When no routing occurs ($\boldsymbol{R} = \boldsymbol{0}_{N\times N}$), (A.8) simplifies to the explicit expression

$$\boldsymbol{I}_{0s} = \boldsymbol{D}_0 + \boldsymbol{I}_0 \cdot \sum_{\nu=1}^{s} \boldsymbol{D}_\nu,$$

in which case

$$P_{s,ji} \;=\; P_{ji} 1_{\{T_{ji}^{\mathrm{rec}} \le sr\}}.$$

A-4 Approximation of A Posteriori Integral

In this section we present an approximation of $\int_{\Theta_1} \Lambda_{ki}(\boldsymbol{\theta}) d\boldsymbol{\theta}$ to insert into (18), with Λ_{ki} as in (27). We use the quadratic approximation

$$\Lambda_{ki}(\boldsymbol{\theta}) \approx \Lambda_{ki}(\hat{\boldsymbol{\theta}}_{ki}) \exp\left(-0.5(\boldsymbol{\theta} - \hat{\boldsymbol{\theta}}_{ki})\boldsymbol{J}_{ki}(\boldsymbol{\theta} - \hat{\boldsymbol{\theta}}_{ki})'\right),$$

of (27), where

$$\hat{\boldsymbol{\theta}}_{ki} = \arg\max_{\boldsymbol{\theta}\in\mathbb{R}^p} \Lambda_{ki}(\boldsymbol{\theta})$$

and the $p \times p$ matrix

$$\boldsymbol{J}_{ki} = -(\log \Lambda_{ki})''(\hat{\boldsymbol{\theta}}_{ki})$$

gives the curvature of $-\log \Lambda_{ki}$ at $\hat{\boldsymbol{\theta}}_{ki}$. For formulas, see Appendix A-5. Changing variables $\boldsymbol{\xi} = (\boldsymbol{\theta} - \hat{\boldsymbol{\theta}}_{ki})\boldsymbol{J}_{ki}^{1/2}$, we get

$$\begin{aligned}
\textstyle\int_{\Theta_1} \Lambda_{ki}(\boldsymbol{\theta}) d\boldsymbol{\theta} \;&\approx\; \Lambda_{ki}(\hat{\boldsymbol{\theta}}_{ki}) \textstyle\int_{\Theta_1} \exp\left(-0.5(\boldsymbol{\theta} - \hat{\boldsymbol{\theta}}_{ki})\boldsymbol{J}_{ki}(\boldsymbol{\theta} - \hat{\boldsymbol{\theta}}_{ki})'\right) d\boldsymbol{\theta} \\
&=\; |\boldsymbol{J}_{ki}|^{-1/2} \Lambda_{ki}(\hat{\boldsymbol{\theta}}_{ki}) \textstyle\int_{\Xi_1} \exp\left(-0.5\boldsymbol{\xi}\boldsymbol{\xi}'\right) d\boldsymbol{\xi} \\
&=\; (2\pi)^{p/2} |\boldsymbol{J}_{ki}|^{-1/2} \Lambda_{ki}(\hat{\boldsymbol{\theta}}_{ki}) \Phi\left(\frac{\hat{\boldsymbol{\theta}}_{ki}\cdot\boldsymbol{u}}{|\boldsymbol{u}\boldsymbol{J}_{ki}^{-1/2}|}\right),
\end{aligned}$$

where

$$\begin{aligned}
\Xi_1 \;&=\; (\Theta_1 - \hat{\boldsymbol{\theta}}_{ki})\boldsymbol{J}_{ki}^{1/2} \\
&=\; \{(\boldsymbol{\theta} - \hat{\boldsymbol{\theta}}_{ki})\boldsymbol{J}_{ki}^{1/2};\, \boldsymbol{\theta}\boldsymbol{u}' > 0\} \\
&=\; \{\boldsymbol{\xi};\, (\boldsymbol{\xi}\boldsymbol{J}_{ki}^{-1/2} + \hat{\boldsymbol{\theta}}_{ki})\boldsymbol{u}' > 0\} \\
&=\; \{\boldsymbol{\xi};\, \boldsymbol{\xi}\boldsymbol{v}' > -\hat{\boldsymbol{\theta}}_{ki}\boldsymbol{u}'/|\boldsymbol{u}\boldsymbol{J}_{ki}^{-1/2}|\}
\end{aligned}$$

and $\boldsymbol{v} = \boldsymbol{u}\boldsymbol{J}_{ki}^{-1/2}/|\boldsymbol{u}\boldsymbol{J}_{ki}^{-1/2}|$ is a unit vector in $\mathbb{R}^p$.

A-5 Derivatives

For completeness, we here give the derivatives of $\log \Lambda_{ki}$ to be used in the quadratic approximation of $\int_{\Theta_1} \Lambda_{ki}(\boldsymbol{\theta}) d\boldsymbol{\theta}$. We write

$$\begin{aligned} M(\boldsymbol{\theta}) &= -0.5(\boldsymbol{\theta}-\boldsymbol{\mu})\boldsymbol{\Sigma}^{-1}(\boldsymbol{\theta}-\boldsymbol{\mu})', \\ LR(\boldsymbol{\theta}) &= L_{qlj}(\boldsymbol{\theta})/L_{qlj}(\mathbf{0}), \\ Q(\boldsymbol{\theta}) &= Q_{qlj}(\boldsymbol{\theta}) \text{ and} \\ P &= P_{k-q,ji}. \end{aligned}$$

Then, according to (27), we have

$$\begin{aligned} \log \Lambda_{ki}(\boldsymbol{\theta}) &= \text{constant} + M(\boldsymbol{\theta}) \\ &+ \textstyle\sum_{(q,j)\in \mathcal{BN}_{ki}} \log LR(\boldsymbol{\theta}) \\ &+ \textstyle\sum_{(q,j)\in \mathcal{BN}_k \setminus \mathcal{BN}_{ki}} \log\left(1 - Q(\boldsymbol{\theta})P\right). \end{aligned} \tag{A.10}$$

We put $s(\boldsymbol{\theta}) = s_{q-l,j}(\boldsymbol{\theta})$ and further

$$\begin{aligned} \mathrm{D}_a(f) &= \tfrac{\partial f}{\partial \theta_a}, & \mathrm{D}^2_{ab}(f) &= \tfrac{\partial^2 f}{\partial \theta_a \partial \theta_b}, \\ s_a &= \mathrm{D}_a(s), \text{ and} & s_{ab} &= \mathrm{D}^2_{ab}(s) \end{aligned}$$

and see that

$$\begin{aligned} \mathrm{D}_a(\log \Lambda_{ki}(\boldsymbol{\theta})) &= \mathrm{D}_a(M) \\ &+ \textstyle\sum_{(q,j)\in \mathcal{BN}_{ki}} \mathrm{D}_a(\log LR) \\ &+ \textstyle\sum_{(q,j)\in \mathcal{BN}_k \setminus \mathcal{BN}_{ki}} \mathrm{D}_a(\log(1-QP)), \\ \mathrm{D}^2_{ab}(\log \Lambda_{ki}(\boldsymbol{\theta})) &= \mathrm{D}^2_{ab}(M) \\ &+ \textstyle\sum_{(q,j)\in \mathcal{BN}_{ki}} \mathrm{D}^2_{ab}(\log LR) \\ &+ \textstyle\sum_{(q,j)\in \mathcal{BN}_k \setminus \mathcal{BN}_{ki}} \mathrm{D}^2_{ab}(\log(1-QP)). \end{aligned} \tag{A.11}$$

We get

$$\begin{aligned} \mathrm{D}_a(M) &= -\textstyle\sum_{b=1}^{p} (\theta_b - \mu_b)\Sigma^{-1}_{ab}, \\ \mathrm{D}^2_{ab}(M) &= \Sigma^{-1}_{ab} \end{aligned}$$

where we have used that $\boldsymbol{\Sigma}^{-1}$ is a symmetric matrix. Further, letting $A = A_{qj}$, $\hat{\sigma} = \hat{\sigma}_{qj}$ and $E = \exp(-s^2/(2\hat{\sigma}^2))$, from (25) we get

$$\begin{aligned} \mathrm{D}_a(\log LR) &= \mathrm{D}_a(LR)/LR, \\ \mathrm{D}^2_{ab}(\log LR) &= (\mathrm{D}_{ab}(LR)LR - \mathrm{D}_a(LR)\mathrm{D}_b(LR))/LR^2, \\ \mathrm{D}_a(LR) &= E s_a S_1^{LR}, \\ \mathrm{D}^2_{ab}(LR) &= E((-\hat{\sigma}^{-2} s s_a s_b + s_{ab})S_1^{LR} + s_a s_b S_2^{LR}), \end{aligned}$$

where

$$\begin{aligned} S_1^{LR} &= -\hat{\sigma}^{-2}s + \textstyle\sum_{n=1}^{\infty} \Omega_n^{LR} s^{2n-1}(-\hat{\sigma}^{-2}s^2 + 2n), \\ S_2^{LR} &= -\hat{\sigma}^{-2} + \textstyle\sum_{n=1}^{\infty} \Omega_n^{LR} s^{2n-2}\left(-\hat{\sigma}^{-2}(2n+1)s^2 + 2n(2n-1)\right), \\ \Omega_n^{LR} &= (A/(2\hat{\sigma}^2))^{2n}(n!)^{-2}. \end{aligned}$$

Further, to keep notation simple we redefine $\hat{\sigma} = \hat{\sigma}_{uj}$ and $E = \exp(-s^2/(2\hat{\sigma}^2))$, and get, from (26),

$$\begin{aligned} \mathrm{D}_a(\log(1-QP)) &= -P\mathrm{D}_a(Q)/(1-PQ), \\ \mathrm{D}^2_{ab}(\log(1-QP)) &= -P(\mathrm{D}^2_{ab}(Q)(1-PQ) + P\mathrm{D}_a(Q)\mathrm{D}_b(Q))/(1-PQ)^2 \\ \mathrm{D}_a(Q) &= Es_aS_1^Q, \\ \mathrm{D}^2_{ab}(Q) &= E((-\hat{\sigma}^{-2}ss_as_b + s_{ab})S_1^Q + s_as_bS_2^Q), \end{aligned}$$

where

$$\begin{aligned} S_1^Q &= -\hat{\sigma}^{-2}\Omega_0^Q s + \textstyle\sum_{n=1}^{\infty} \Omega_n^Q s^{2n-1}(-\hat{\sigma}^{-2}s^2 + 2n), \\ S_2^Q &= -\hat{\sigma}^{-2}\Omega_0^Q + \textstyle\sum_{n=1}^{\infty} \Omega_n^Q s^{2n-2}(-\hat{\sigma}^{-2}(2n+1)s^2 + 2n(2n-1)), \\ \Omega_n^Q &= 2^{-n}\hat{\sigma}^{-2n}(n!)^{-1}\Gamma^{\mathrm{inc}}(\lambda^2/2, n+1). \end{aligned}$$

A-6 Uniform Quantization

In a uniform quantization of variable x to c bits on the interval $[0, x^{\max})$, in which the variable is expected to lie, we divide this interval into 2^c steps of length $S = x^{\max}/2^c$. The quantized variable is set to

$$x^q := \begin{cases} \frac{1}{2}S & \text{if } x < 0, \\ (n+\frac{1}{2})S & \text{if } nS \le x < (n+1)S, \\ & \qquad n = 0, \ldots, 2^c - 1, \\ (2^c - \frac{1}{2})S & \text{if } x^{\max} \le x, \end{cases}$$

and we see that, for $x \in [0, x^{\max})$,

$$n = \left\lfloor \frac{x}{S} \right\rfloor.$$

Thus, the above simplifies to

$$x^q := \begin{cases} \frac{1}{2}S & \text{if } x < 0, \\ \left(\left\lfloor \frac{x}{S} \right\rfloor + \frac{1}{2}\right) S & \text{if } 0 \le x < x^{\max}, \\ (2^c - \frac{1}{2})S & \text{if } x^{\max} \le x. \end{cases}$$

References

M. Abramowitz and I.A. Stegun, editors. *Handbook of mathematical functions with formulas, graphs, and mathematical tables. National Bureau of Standards Applied Mathematics Series 55. Tenth printing.* Dover, New York, USA, 1972.

S. Appadwedula, V.V. Veeravalli, and D.L. Jones. Energy-efficient detection in sensor networks. *IEEE Journal on Selected Areas in Communications*, 23(4):693–702, 2005.

S. Appadwedula, V.V. Veeravalli, and D.L. Jones. Decentralized detection with censoring sensors. *IEEE Transactions on Signal Processing*, 56(4): 1362–1373, 2008.

M. Basseville. Detecting changes in signals and systems – a survey. *Automatica*, 24(3):309–326, 1988.

V. Borkar and P. Varaiya. Asymptotic agreement in distributed estimation. *IEEE Transactions on Automatic Control*, 27(3):650–655, 1982.

J.F. Chamberland and V.V. Veeravalli. Decentralized detection in sensor networks. *IEEE Transactions on Signal Processing*, 51(2):407–416, 2003.

J.F. Chamberland and V.V. Veeravalli. Asymptotic results for decentralized detection in power constrained wireless sensor networks. *IEEE Journal on Selected Areas in Communications*, 22(6):1007–1015, 2004.

J.F. Chamberland and V.V. Veeravalli. Wireless sensors in distributed detection applications. *IEEE Signal Processing Magazine*, 1053(5888/07), 2007.

S. Choi, C.R. Berger, S. Zhou, and P. Willett. Estimation of target trajectories based on distributed channel energy measurements. In *Proceedings of the 2008 11th International Conference on Information Fusion*, Cologne, Germany, June, 30 - July, 03 2008.

R.W. Crow and S.C. Schwartz. Quickest detection for sequential decentralized decision systems. *IEEE Transactions on Aerospace and Electronic Systems*, 32(1):267–283, 1996.

M.H. DeGroot. Reaching a consensus. *Journal of the American Statistical Association*, 69(345):118–121, 1974.

J. Grönkvist, J. Nilsson, and E. Sangfelt. Wireless underwater sensor networks. Technical Report FOI-R–2408–SE, FOI, Stockholm, December 2007.

R.D. Hippenstiel. *Detection theory: Applications and digital signal processing*. CRC Press, Boca Raton, FL, USA, 2002.

O. Hössjer, R. Lennartsson, D. Lindgren, T. Nyberg, S. Petrovic, and T. Öberg. Sequential distributed detection by local agents (preliminary title). to appear.

R.J.A. Little and D.B. Rubin. *Statistical analysis with missing data*. Wiley, Hoboken, NJ, USA, 2002.

L. Liu, S. Zhou, and J.H. Cui. Prospects and problems of wireless communication for underwater sensor networks. *Wireless Communications and Mobile Computing*, 8(8):977–994, 2008.

R. Niu, P.K. Varshney, and Q. Cheng. Distributed detection in a large wireless sensor network. *Information Fusion*, 7(4):380–394, 2006.

C. Rago, P. Willett, and Y. Bar-Shalom. Censoring sensors: a low-communication-rate scheme for distributed detection. *IEEE Transactions on Aerospace and Electronic Systems*, 32(2):554–568, 1996.

L.D. Stone, T.L. Corwin, and C.A. Barlow. *Bayesian multiple target tracking*. Artech House, Norwood, MA, USA, 1999.

J. Tsitsiklis and M. Athans. Convergence and asymptotic agreement in distributed decision problems. *IEEE Transactions on Automatic Control*, 29 (1):42–50, 1984.

J.N. Tsitsiklis. Decentralized detection by a large number of sensors. *Mathematics of Control, Signals, and Systems*, 1(2):167–182, 1988.

P. Varaiya. Consensus in distributed estimation. In A. Bagchi and H. Th. Jongen, editors, *Lecture Notes in Control and Information Sciences: Systems and Optimization, Proceedings of the Twente Workshop, Enschede, the Netherlands, April 16-18, 1984*, volume 66, pages 182–191, Berlin/Heidelberg, 1985. Springer.

V.V. Veeravalli. Decentralized quickest change detection. *IEEE Transactions on Information Theory*, 47(4):1657–1665, 2001.

R.B. Washburn and D. Teneketzis. Asymptotic agreement among communicating decisionmakers. *Stochastics*, 13:103–129, 1984.