Bayesian Inference in Dynamic Domains using Logical OR Gates

Rik Claessens[1,2], Alta de Waal[3], Pieter de Villiers[3,4], Ate Penders[2,5], Gregor Pavlin[2,6] and Karl Tuyls[1,5]

[1] *University of Liverpool, Liverpool, U.K.*
[2] *Thales Research & Technology, Delft, the Netherlands*
[3] *University of Pretoria, Pretoria, South Africa*
[4] *Council for Scientific and Industrial Research, Pretoria, South Africa*
[5] *Delft University of Technology, Delft, The Netherlands*
[6] *University of Amsterdam, Amsterdam, The Netherlands*

Keywords: Artificial Intelligence and Decision Support Systems, Multi-agent Systems, Strategic Decision Support Systems.

Abstract: The range of applications that require processing of temporally and spatially distributed sensory data is expanding. Common challenges in domains with these characteristics are sound reasoning about uncertain phenomena and coping with the dynamic nature of processes that influence these phenomena. To address these challenges we propose the use of causal Bayesian Networks for probabilistic reasoning and introduce the Logical OR gate in order to combine them with dynamic processes estimated by arbitrary Markov processes. To illustrate the genericness of the proposed approach, we apply it in a wildlife protection use case. Furthermore we show that the resulting model supports modularization of computations, which allows for efficient decentralized processing.

1 INTRODUCTION

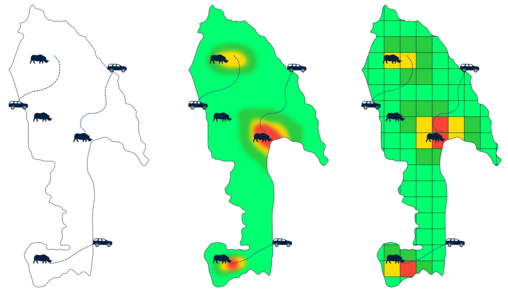
Recent advances in sensory, computing and communication technology have facilitated a new class of decision support applications that exploit rich and heterogeneous data to estimate the phenomena relevant for decision making and control. Such applications are gaining importance in various domains, such as security, smart homes, Internet of Things (IoT), etc. For example, imminent threats in security applications must be identified based on various sensory clues and intelligence. Similarly, in the domain of elderly care, heterogeneous data obtained via IoT devices can provide clues about anomalies corresponding to potentially dangerous states of an elderly person.

While the potential of this range of applications is huge, there are multiple challenges associated with correct and tractable processing of the correlated data, stemming from disparate sources (e.g. sensors, human observers, databases, social media, etc.) and collected at different locations and points in time. Such processing depends on domain models that describe correlations between the different data types and the inference algorithms that use such models to draw conclusions about the phenomena of interest. In the targeted domains, the modelling and inference are not trivial, however. These domains are characterized through many types of correlated phenomena and dynamic processes. The resulting domain models may contain many variables and the dynamics must be appropriately considered.

Without the loss of generality, we will use a running example from wildlife protection focusing on the prevention of rhino poaching to illustrate the challenges and solutions (Figure 1). As the resources for observing, patrolling or intervening in such environments are limited and the areas that need to be protected vast, a decision support system using different types of data is needed to help with the assessment of potentially critical locations where the poaching could take place. The threat depends on many different factors, such as environmental conditions as well as the presence of the rhinos, rangers and the poachers. Moreover, the observations of potential poachers are often sparse and uncertain. The same is true for the location of rhinos. Consequently, situation assessment requires collection of many different types

Claessens, R., Waal, A., Villiers, P., Penders, A., Pavlin, G. and Tuyls, K.
Bayesian Inference in Dynamic Domains using Logical OR Gates.
In *Proceedings of the 18th International Conference on Enterprise Information Systems (ICEIS 2016) - Volume 2*, pages 134-142
ISBN: 978-989-758-187-8

of data at different locations and points in time and reasoning about the observed patterns. As data collection and reasoning in such settings typically exceed the capabilities of human operators (Dörner and Schaub, 1994), an automated information fusion system is used.



(a) (b) (c)

Figure 1: An overview of a threat assessment system in the Kruger and Limpopo National Parks in South Africa. Figure 1(a) shows poacher and rhino locations (icons) and their tracks (dashed lines). A continuous heat-map visualizing high-risk zones and a discretized grid-based heat-map are shown in Figure 1(b) and 1(c) respectively.

This use case illustrates a class of applications that require inference combining (i) dynamic models describing the evolution of processes over space and time and (ii) location bound models that describe relations between different factors at a specific location.

This paper addresses multiple challenges associated with such analysis. Firstly, the dynamic and location bound models have to be combined in a theoretically sound and efficient manner. For this we propose the use of causal Bayesian Networks (BN) and so-called Logical OR gates, that allow creation of complex causal probabilistic domain models. Secondly, we show that the presented Bayesian approach facilitates modularization of models and inference processes, that can be computed in a distributed fashion.

The paper is structured as follows: Section 2 discusses related work on Bayesian Networks. In Section 3, we introduce the relevant probabilistic models and propose the use of Logical OR gates for combining dynamic elements with a static situation assessment BN. Section 4 discusses how the proposed methods can be applied in a rhino poaching domain. Afterwards we explain how the use of Logical OR gates results in a modular probabilistic network. Finally, we draw some conclusions and steps for future research in the last section.

2 RELATED WORK

Related work mainly comes from two research areas, i.e., Bayesian networks (specifically for environmental modelling) and distributed inference systems.

Bayesian networks (BNs) are well established method supporting systematic exploitation of correlated data and are often used as a modelling tool for environmental modelling with a wide range of case studies to be found in literature (Pearl, 1988; Johnson et al., 2010; Pullar and Phan, 2007; Borsuk et al., 2004; Borsuk et al., 2006). Furthermore, BNs are ideal to combine knowledge from diverse disciplines and sources (Düspohl et al., 2012) and hence Bayesian Network (BN) models can be learned from data and/or constructed with involvement of domain experts. This is often referred to as ‘participatory modelling’ (Bromley et al., 2005).

An important challenge of an inference system is to cope with large quantities of heterogeneous information that becomes available dynamically, at runtime. The inference systems must be adapted at runtime, which requires modular approaches, where loosely coupled inference modules collaboratively solve an assessment problem through message passing. There exist multiple approaches to achieve sound inference in modular systems, which perform exact inference with the help of secondary inference structures, such as junction trees, linked junction forests and spanning multiple processing modules (Xiang, 2002; Paskin and Guestrin, 2004). Compilation of such structures, however, requires expensive processing and massive messaging, which in turn can be impractical if constellations of information sources change rapidly. In this paper we use an alternative modularization approach to exact inference (Pavlin et al., 2010; de Oude and Pavlin, 2009), where compilation of secondary structures is avoided.

3 PROBABILISTIC MODELS

Models describing the correlations between the observations and hidden phenomena of interest are indispensable for sound inference. The various observations collected by the fusion system can be viewed as outcomes of a system of interrelated causal processes. Therefore it is reasonable to use Bayesian Networks and Hidden Markov Model approaches, as they can efficiently and systematically describe the dependencies between the phenomena. In the targeted domains, however, complex models are required, consisting of many variables and relations. Monolithic approaches to modelling and inference cannot cope with such complexity. However, it turns out that the overall domain models can be viewed as a composition of two types of models:

- Location Bound Model (LBM), that describes the relations between the various factors influencing the phenomena of interest at a specific location. Such a model correlates the phenomena of interest with the presence of one or multiple dynamic objects at the respective location and various environmental phenomena.
- Models of dynamic processes that describe the relations between states corresponding to multiple locations and different points in time, such as tracked objects.

In the presented approach, we assume that the area of interest is represented by a grid, where each cell corresponds to a location labelled a_k (Figure 1(a)). Each location is associated with an LBM that is used for the estimation of the states of hidden phenomena of interest at a certain location a_k at time t.

The two types of reasoning mentioned above require adequate representations and inference algorithms. Furthermore, a correct method that allows combining the LBMs with the dynamic models is required. These aspects are discussed in the following subsections.

3.1 Location Bound Models

An LBM describes correlations between the observable phenomena/events at a_k and the hidden phenomena that influence the observable events.

In the presented approach, location bound models (LBMs) are represented through causal BNs (Pearl, 1988). A BN is defined as a tuple $\langle G, P \rangle$, where $G = \langle \mathcal{V}, \mathbb{E} \rangle$ is a Directed Acyclic Graph (DAG) defining a domain $\mathcal{V} = \{V_1, \ldots, V_n\}$ and a set of directed edges $\langle V_i, V_j \rangle \in \mathbb{E}$ over the domain. The Joint Probability Distribution (JPD) $P(\mathcal{V})$ over the domain $\mathcal{V}$ is defined as $P(\mathcal{V}) = \prod_{\mathcal{V}} P(V_i|\pi(V_i))$, where $P(V_i|\pi(V_i))$ is the conditional probability distribution for node V_i given its parents $\pi(V_i)$, which can be represented by a Conditional Probability Table (CPT). BNs allow efficient representation of the states of heterogeneous phenomena and describe causal relations between these phenomena. Moreover, they support mathematically sound and efficient inference algorithms.

3.2 Dynamic Models

Dynamic models capture correlations between spatially and temporally distributed phenomena and events associated with evolving processes, such as moving objects. These evolutionary processes can be represented with the help of Hidden Markov Models or their generalization, Dynamic Bayesian Networks (DBNs) (Thrun et al., 2005). In these approaches the inference is carried out on models that are expanded with identically structured slices over time (Figure 2). Such inference is called tracking if we estimate the states of a moving object. In this paper we assume a common technique for approximate inference in such models, namely Particle Filters (Gustafsson et al., 2002). The Particle Filter (PF)-algorithm makes use of a set of particles, representing possible locations of the entity that is being tracked. The distribution or spread of the set of particles gives a measure for the uncertainty about the target's true location. The continuous probability distribution that a PF-algorithm approximates is given by the posterior probability distribution over the state of interest, in this case the location x_t of target j: $P(x_t|Z^j_{1:t})$, where $Z^j_{1:t}$ denotes the sequence of all observations of the tracked object. .

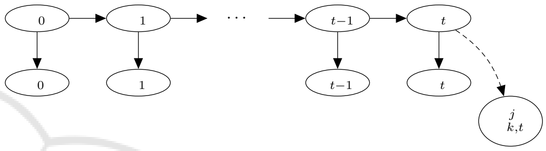


Figure 2: The structure of the DBN that is approximated by a PF-based tracking algorithm. The dashed edge represents the function $in(x^i_t, a_k)$ from Equation 2.

We introduce a binary variable, $T^j_{k,t}$, whose states represent the presence of an individual or a group of tracked object(s) (indexed by j) being present in area a_k at time t. The posterior probability of a $T^j_{k,t}$ is in principle an integral of the probability distribution, representing the spatial distribution of tracked objects, over area a_k. However, since the set of particles approximates a continuous probability distribution with a set of discrete particles, we approximate this integral with the number of particles that are inside a_k, divided by the total number of particles N:

$$
\begin{aligned}
P(T^j_{k,t} = true|Z^j_{1:t}) &= \int_{a_k} P(x_t|Z^j_{1:t})dx \\
&\approx \frac{1}{N} \sum_i in(x^i_t, a_k), \qquad (1)
\end{aligned}
$$

where the function $in(x^i_t, a_k)$ is defined as:

$$
in(x^i_t, a_k) = \begin{cases} 1, & \text{if } x^i_t \text{ is inside area } a_k \\ 0, & \text{otherwise} \end{cases}. \qquad (2)
$$

3.3 Combining Tracking and Location Bound Models

The question is, how the DBN shown in Figure 2 can be combined with an LBM in a mathematically sound way. An additional challenge is that multiple dynamic objects could be present at a specific location. Consequently, dependent on the situation, the LBM might have to be combined with multiple DBNs, each estimating the whereabouts of a different dynamic object. Thus, the estimation of the hidden phenomena of interest depends on evolving domain models. The solution is achieved in a number of steps.

First we consider the variable whose states are influenced by the outputs of tracking processes. As we assume that in an LBM there exists a "Track Present" (TP), a binary variable representing the presence of any object at the location associated with that LBM. In the discretized setting, the distribution over the binary variable is computed for a specific location and time, i.e. $TP_{k,t}$ represents the situation that any number of dynamic objects is present in area a_k at time t. Consequently, at any moment in time at any location, a varying number of tracks may influence the LBM that is performing the inference for that area.

Figure 3 shows the structure of the part of the BN described above. There is an edge from every $T^j_{k,t}$ variable (representing a single track) converging to $TP_{k,t}$. The dynamic nature of the tracks means that the structure of the BN changes as tracks enter and leave an area.

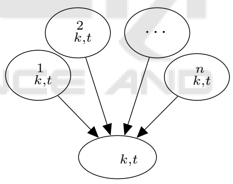


Figure 3: The BN structure of a connection between tracking processes $T^j_{k,t}$ and the BN variable $TP_{k,t}$ they influence.

The CPT of the variable $TP_{k,t}$ is determined by the set of incoming edges, i.e. the set of tracks that have an influence in an area a_k. In this paper, the CPT has a physical interpretation. For each area defined in the system, the tracked target is either present (with probability 1) or not present at all (thus, a probability of 0 for being present). This interpretation results in a straightforward definition of the entries in the CPT. Namely, the entries in the columns $P(TP_{k,t}|T^1_{k,t},\ldots,T^n_{k,t})$, for which $\exists\ T^j_{k,t} \in \{T^1_{k,t},\ldots,T^n_{k,t}\}$, s.t. $T^j_{k,t} = true$ are set as $[1\ 0]^T$. This includes all columns in the CPT, except one. This is the column, for which $\forall\ T^j_{k,t} \in \{T^1_{k,t},\ldots,T^n_{k,t}\}$, $T^j_{k,t} = false$. This assignment of random variables represents the situation where no tracked objects are present in a specific area. Therefore, we set this column as $[0\ 1]^T$.

Table 1: A schematic overview of the CPT of the variable $TP_{k,t}$, i.e. $P(TP_{k,t}|T_{1,k,t},\ldots,T_{j,k,t})$.

	$T^n_{k,t}$	t				f			
	$\vdots$	$\iddots$				$\ddots$			
	$T^2_{k,t}$	t		f		$\cdots$		f	
	$T^1_{k,t}$	t	f	t	f	$\cdots$	$\cdots$	t	f
$TP_{k,t}$	t	1	1	1	1	$\cdots$		1	0
	f	0	0	0	0			0	1

The resulting CPT is shown in Table 1 and is called a logical OR gate. Although the entries in the CPT consist only of 0s and 1s, uncertainty about a target's true location is introduced by the computation of the prior probability distribution of each $T^j_{k,t}$ node.

The logical OR gate can be used to integrate the LBM with dynamic models. This integration results in an overall more complex, but because of the presented approach easily tractable BN. The next section illustrates a BN in which tracks are incorporated.

3.4 Posteriors of Tracking Modules

The CPT from Table 1 is used to compute the belief over the tracked entity being present in area a_k. This marginalization is given by:

$$P(TP_{k,t} = true|Z_{1:t}) = \sum_{T^1_{k,t}} \cdots \sum_{T^j_{k,t}} \Big[P(TP_{k,t} = true|T^1_{k,t},\ldots,T^j_{k,t}) \cdot \prod_j P(T^j_{k,t}|Z^j_{1:t}) \Big]. \quad (3)$$

This marginalization is the product of probabilities of each permutation of tracked objects that are present in area a_k. By exploiting the structure of the CPT however, there is a much faster way to compute the posterior. Namely, it is possible to compute the belief that a tracked entity is present in area a_k as the product of the probabilities of each tracked object being absent in the area, $\prod_j \left(1 - P(T^j_{k,t} = true|Z^j_{1:t})\right)$. We show that this is equivalent to the marginalization for $TP_{k,t} = true$ over all $T^j_{k,t}$ variables.

We will exploit the characteristics of a binary variable:

$$P(TP_{k,t} = true_{k,t}|Z_{1:t}) = 1 - P(TP_{k,t} = false|Z_{1:t}) \quad (4)$$

From Table 1, it it is clear that $P(TP_{k,t} = false|T^1_{k,t},\ldots,T^j_{k,t})$ only resolves to 1, if and only if $\forall\, T^j_{k,t} \in \{T^1_{k,t}, \dots, T^n_{k,t}\}$, $T^j_{k,t} = false$. We combine Equations 3 and 4:

$$
\begin{aligned}
&P(TP_{k,t} = false|Z_{1:t}) \\
&= \sum_{T^1_{k,t}} \dots \sum_{T^j_{k,t}} \Big[P(TP_{k,t} = false|T^1_{k,t}, \dots, T^j_{k,t}) \\
&\quad \cdot \prod_j P(T^j_{k,t}|Z^j_{1:t}) \Big] \qquad (5)
\end{aligned}
$$

The marginalization in Equation 5 is a sum over zero-valued products, except the case where all T^j_k variables have value $T^j_k = false$. Because of this we are able to simplify Equation 5 to:

$$
P(TP_{k,t} = false|Z_{1:t}) = \prod_j P(T^j_{k,t} = false|Z^j_{1:t}) \quad (6)
$$

As we are interested in the posterior for $TP_{k,t} = true$, we subtract this value from 1. As a final step, by combining Equations 3, 4 and 5 and observing that, by definition $P(T^j_{k,t} = false|Z_{1:t}) = 1 - P(T^j_{k,t} = true|Z_{1:t})$, the computation of the posterior of $TP_{k,t} = true$ can be simplified as:

$$
\begin{aligned}
&P(TP_{k,t} = true|Z_{1:t})_k \\
&= 1 - \prod_j \big(1 - P(T^j_{k,t} = true|Z^j_{1:t})\big). \qquad (7)
\end{aligned}
$$

By using the models and equations described above it is possible to incorporate tracking information into an LBM, without the need of changing the model at runtime.

4 APPLICATION: RHINO POACHING

In this section we will apply the in the previous sections explained methods in a present-day security setting, rhino poaching in the Kruger Park, South Africa.

4.1 Bayesian Threat Assessment

An example of a BN that describes correlations between poaching events was introduced by (Koen et al., 2014). For the sake of simplicity, we use a derived BN, shown in Figure 4. To make the notation of variables more compact, we abbreviate all variable names to the bold and underlined parts in Figure 4 in the remaining part of this paper. Furthermore, we use superscripts R, P and Ra for variables relating to rhinos, poachers and rangers respectively.

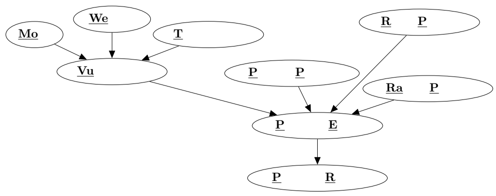


Figure 4: A BN describing phenomena that influence the likelihood of a poaching event.

The BN in Figure 4 corresponds to the following joint probability density (JPD) factorization:

$$
\begin{aligned}
JPD &= P(PE|Vu, PP, RP, RaP)P(PR|PE) \\
&\cdot P(Vu|Mo, We, T)P(Mo)P(We)P(T) \\
&\cdot P(PP)P(RP)P(RaP). \qquad (8)
\end{aligned}
$$

A separate instance of this model is used for each area, i.e. a cell, represented in the grid-based map shown in Figure 1(c). Each model correlates different types of observations obtained in the respective area a_k.

There are a number of entities of importance for which information should be gathered, such as the poachers and rhinos in Figure 1(a).

The states of nodes labelled *Poacher Present* and *Rhino Present* represent dynamic objects, as described by the variable "Track Present" (TP) in the previous section. However, the events corresponding to the states of these variables are not observed directly. The states of rhinos and poachers are typically estimated with the help of tracking processes based on filters that correlate spatio-temporal observations.

As described before, the dynamic nature of the tracks would require to modify the structure of the BN dynamically at runtime. However, considering the size of the area in which the system will be deployed, the number of areas and tracks might grow too large in order to perform computations of inference algorithms such as the sum-product algorithm efficiently. To avoid this computationally expensive task, we incorporate logical OR gates in the BN from Figure 4.

The JPD of the network shown in Figure 5 then resolves to:

$$
\begin{aligned}
JPD_{k,t} &= P(PE|Vu, PP, RP, RaP)P(PR|PE) \\
&\cdot P(Vu|Mo, We, T)P(Mo)P(We)P(T) \\
&\cdot P(PP|T^{1,P}_{k,t}, \dots, T^{j,P}_{k,t}) \prod_j P(T^{j,P}_{k,t}|Z^P_{1:t}) \\
&\cdot P(RP|T^{1,R}_{k,t}, \dots, T^{l,R}_{k,t}) \prod_l P(T^{l,R}_{k,t}|Z^R_{1:t}) \\
&\cdot P(RaP|T^{1,Ra}_{k,t}, \dots, T^{m,Ra}_{k,t}). \qquad (9)
\end{aligned}
$$

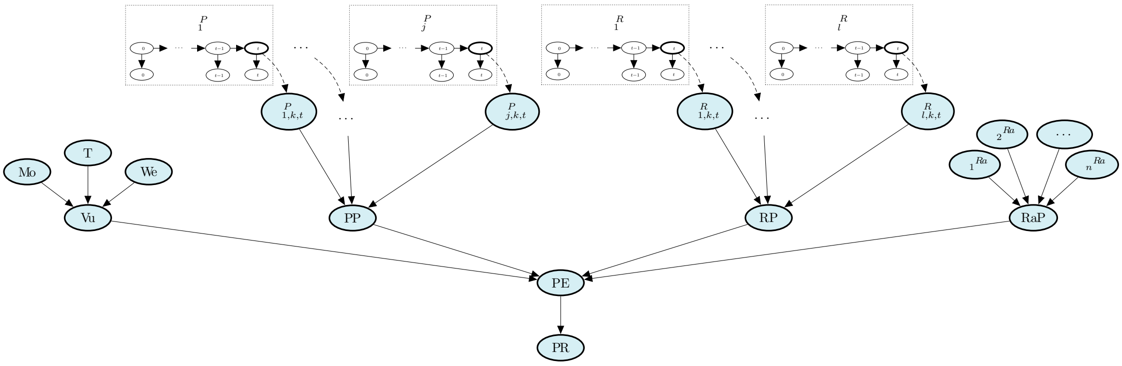


Figure 5: A BN using logical OR gates for inserting poacher, rhino and ranger tracks. Superscripts R, P and Ra denote variables for rhinos, poachers and rangers respectively.

By using the simplification from Equation 7, the posterior probability of the variable *Poaching Event* based on the BN from Figure 5 (Equation 9), is given by Equation 10 (in which η is the normalization factor).

$$
\begin{aligned}
&P(PE_{k,t} = true | Z^P_{1:t}, Z^R_{1:t}, E^P_{1:t}, E^R_{1:t}, E^{Vu}_{1:t}, X^{Ra}_{1:t})_k \\
&= \eta \sum_{TP} \sum_{RP} \sum_{Vu} \sum_{Ra} \Big[\\
&\quad P(PE_{k,t} = true | Vu_{k,t}, TP_{k,t}, RP_{k,t}, RaP_{k,t}) \\
&\quad \cdot P(PR_{k,t} | PE_{k,t} = true) \\
&\quad \cdot \sum_{We} \Big[P(Vu_{k,t} | Mo_{k,t}, We_{k,t}, T_{k,t}) \\
&\quad \cdot P(Mo_{k,t}) P(We_{k,t}) P(T_{k,t}) \Big] \\
&\quad \cdot \Big[1 - \prod_j \big(1 - P(T^{j,P}_{k,t} = true | Z^{j,P}_{1:t})\big) \Big] \\
&\quad \cdot \Big[1 - \prod_l \big(1 - P(T^{l,R}_{k,t} = true | Z^{l,R}_{1:t})\big) \Big] \\
&\quad \cdot \Big[1 - \prod_m \big(1 - P(T^{m,Ra}_{k,t} = true)\big) \Big] \Big] \qquad (10)
\end{aligned}
$$

5 DISTRIBUTED INFERENCE

In this section we discuss how the previously introduced domain models can be split up into modules that allow distributed computation. The notion of Markov boundaries is used to show that this type of distributed computation results in sound inference. Furthermore, the different types of agents that constitute this Multi-Agent System (MAS) are discussed as well as the complexity of the overall system.

By using logical OR gates for combining tracking processes and the LBMs, we obtain a system that is equivalent to a BN that describes the overall situation (Figure 5). This model captures correlations between disparate observations that are relevant for the estimation of the likelihood of a poaching event, such as the observations of poachers or rhinos as well as the phenomena that define the context of the event and further a priori knowledge.

This model is a basis for the computation of the probability of poaching which corresponds to the evaluation of Equation 10.

5.1 Model Decomposition

The fact that we can write the correlations in the form of a BN reveals important properties of the overall joint distribution over all variables in the BN. Namely, not all variables are directly dependent, which allows efficient modelling and inference.

We exploit the concept of d-separation to partition the BN shown in Figure 5 into smaller BN fragments according to the design rules presented in (Pavlin et al., 2010). We obtain fragment Ψ^{LBM}_k (denoted by the coloured nodes in figure 5) and tracker fragments $\Psi^{j,P}$ and $\Psi^{l,R}$ that correspond to the network shown in Figure 2. Each fragment Ψ^i is a BN defined over a set of variables $\mathcal{V}_i \in \mathcal{V}$, where $\mathcal{V}$ denotes the variables from the original BN. In each fragment Ψ^i we can identify a Markov Boundary (MB^i), a set of variables $\mathcal{X}_i \in \mathcal{V}_i$. If all variables in MB^i are instantiated with hard evidence, then the inference over other variables $\mathcal{V}_i$ in Ψ^i is rendered independent of other variables in the original BN (Pearl, 1988). In the used fragments we can identify the following Markov Boundaries:

- Each Ψ^{LBM}_k is associated with $MB^{LBM}_{k,t} = \{[T^{j,P}_{k,t}], [T^{l,R}_{k,t}], [T^{m,Ra}_{k,t}]\}$
- Each $\Psi^{j,P}$ is associated with $MB^P_{j,k,t} = \{T^{j,P}_{k,t}\}$
- Each $\Psi^{l,R}$ is associated with $MB^R_{l,k,t} = \{T^{l,R}_{k,t}\}$

Note, the rectangular brackets [ ] denote multiple variables associated with different tracker fragments contributing beliefs to the LBM at a_k. It turns out that in such domains (i) the intersection of MB of Ψ_k^{LBM} and the MB of any track fragment contains at most one uninstantiated variable ($T_{k,t}^{j,P}$ or $T_{k,t}^{l,R}$) and (ii) there are no intersections between the MBs of tracking fragments with uninstantiated variables. As it was shown in (Pavlin et al., 2010), such a system of Bayesian fragments allows exact Bayesian inference over $P(PE|E_{k,t})$, where $E_{k,t}$ denotes the set of all relevant observations that were collected at location k and throughout the system of tracking modules Ψ_j^P and Ψ_L^R contributing soft evidence in form of estimates of the likelihood of being at location k; i.e. the system of Bayesian modules supports inference that correctly takes into account correlations between disparate track observations, context data and hidden events of interest by simply passing of messages carrying outputs of trackers $P(T_{k,t}^{j,P}|Z_{1:t}^P)$, $P(T_{k,t}^{l,R}|Z_{1:t}^R)$ and $P(T_{k,t}^{m,Ra})$, respectively.

These messages correspond to the factors in Equation 10 that can be computed independently. Consequently, the computation of Equation 10 can be distributed over a system of processing modules.

This has important implications regarding the efficiency and flexibility of the envisioned assessment solutions. Namely, the computation can be distributed over an arbitrary system of networked machines and the equation can be adapted dynamically as new sources and tracks enter a specific area. As a new track enters a_k, its current estimate is simply plugged into the equation, which then automatically correlates all the data that the track was producing with the rest of the observations and hidden phenomena.

5.2 Distributed Inference

We can cast the overall computation as a service composition problem, where each service has specific domain knowledge and inference capabilities. Clearly, such dynamic computations of beliefs over states requires non-trivial computational systems and adequate information management and distribution between the many services. The modules must not only be able to discover other modules that can provide relevant data, i.e. beliefs over the relevant states, but also maintain and terminate information flows between these modules. Therefore we use the MAS paradigm to systematically implement such adaptive inference systems. The resulting MAS system supports distributed processing equivalent to Equation 10. The system of modules implements the sum-product algorithm in which disparate data collected by different modules in the system is correctly correlated.

The framework is used to systematically organize different types of computation. The Distributed Information Fusion System (DIFS) proposed consists of the following distinctive types of modules:

Location Bound Modules. Each LBM dedicated to an area a_k is represented and used by a specific module computing belief about a poaching event at location a_k at different time intervals. Each LBM module gathers information from relevant phenomena of interest in a_k and from all tracker modules whose estimates indicate that the chance of their track being in a_k exceeds some threshold.

Tracking Modules. Each track is estimated by a dedicated process that collects all relevant data and computes the posterior $P(T_{k,t}^{j,P} = true|Z_{1:t}^P)$. The tracking modules keep track of the constantly changing location of the target position estimates and the associated area of interest (AOI), defined as a set of all points in which the estimated probability of the target presence exceeds a certain threshold. A tracking module subscribes to all relevant types of data sources within an AOI, such as sensors, humans capable of producing structured reports, etc. The subscriptions dynamically change with the estimated AOI over time.

Sensor Modules. Each sensor is represented by a distinct module. Other modules that require sensor data subscribe to a sensor's output based on the sensor type and location.

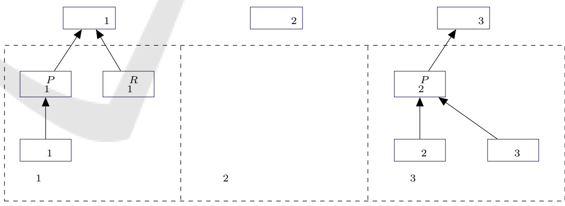


Figure 6: A schematic overview of the Distributed Information Fusion System architecture for 3 areas (a_k), including several tracking ($T^{j,P}$, $T^{l,R}$), sensor (S_i) and LBM (LBM_n) modules and their (possible) connections.

Each module provides a context for a specific process or sensor, i.e. the location/area it is associated with, time of availability and other parameters, such as cost, latency, etc., if required. Moreover, the modules dynamically create information flows between the right processes through service discovery and negotiation. The service discovery is based on the needs for certain types of data produced in the right context (e.g. a presence sensor data in cell a_k). A schematic overview of the proposed system is shown in Figure 6.

The inference equivalent to equation 10 is achieved through dynamic configuration of information flows, triggered by the need of various distributed inference processes and the availability of the relevant information. For example, at the initialization of the system, an LBM module at cell a_k subscribes to outputs of any tracker module whose AOI intersects a_k. The information flow from tracker modules is established and terminated as the AOI enters and leaves area a_k.

When relevant modules send new information to an LBM module, it processes the information immediately. This happens in the following situations: (i) a new track enters a_k, (ii) an existing track leaves a_k, (iii) the posterior probability of a $T_{k,t}^{j,P}$ or $T_{k,t}^{l,R}$ node changes sufficiently, (iv) one of the variables $\{Mo, T, We\}$ change their state.

5.3 Physical Distribution of Modules

A large scale system as the one introduced in Section 4, requires not only modularization to make the problem tractable, but also distribution over physical machines to make the system responsive. Running such a large scale scenario takes too much time to compute on a single machine to still have a useful and relevant output for the human expert. The distribution, however, brings additional complexity to the system with respect to discovery of and routing between different modules.

The presented MAS approach allows distribution of processing services. The actual distribution of the inference modules will depend on the application and various operational boundary conditions:

- The model complexity in a module dictates the computational effort.
- Modularization reduces the computational complexity, hence might speed-up the process, but require more messaging. Messaging is slow compared to intra-module communication, meaning the speed-up gained by reducing computational complexity is lost by introduction of too many small modules.
- The overall domain complexity implies different partitioning of the system; if the variables the system is reasoning about are densely connected, it might be difficult to create small modules, if at all possible.
- Distribution over multiple machines reduces the computation load, but introduces large variations in communication latency. If the communication between the machines does not support a sufficient bandwidth, the distribution might be impractical.
- The spatial and organizational proximity of different sources.
- Frequency of updates. High data acquisition frequency introduces more belief updates resulting in more messaging and processing costs.

6 DISCUSSION & CONCLUSION

In this paper we address several challenges in a class of domains that require reasoning about uncertain and dynamic phenomena. To cope with the dynamics of the domain, we propose the use of Logical OR gates to introduce the outcomes of inference dynamic models into probabilistic location bound models. These LBMs describe the relation of a set of phenomena of interest in a discretized area. The Logical OR gate is an efficient structure to combine the information gathered by the dynamic processes into an area's probabilistic model without changing the model itself. This is achieved by exploiting the structure of the conditional probability table of the OR gates. We show that the belief of a tracked object being present in an area can be computed by using the probabilities of all tracked objects being absent in that area. This method avoids unnecessary computations and the need for changing the LBM when tracks possibly enter or leave an area. Because of this, it is possible to separate the components over different modules. Very important is the fact that the outcome of these modules are identical to the result of the monolithic model. By using the proposed methods, a computationally intensive decision support application can be run by efficient distributed processing.

We illustrated an application of rhino poaching in South Africa by applying the Logical OR gates in an existing BN and showing how the resulting modules can be used to compute the results of the overall system in a distributed way. However, we believe the presented approach to be viable in a range of domains with the described characteristics.

REFERENCES

Borsuk, M. E., Reichert, P., Peter, A., Schager, E., and Burkhardt-Holm, P. (2006). Assessing the decline of brown trout *salmotrutta* in Swiss rivers using a Bayesian probability network. *Ecological Modelling*, 192(1):224–244.

Borsuk, M. E., Stow, C. A., and Reckhow, K. H. (2004). A Bayesian network of eutrophication models for synthesis, prediction, and uncertainty analysis. *Ecological Modelling*, 173(2):219–239.

Bromley, J., Jackson, N., Clymer, O., Giacomello, A., and Jensen, F. (2005). The use of Hugin®to develop Bayesian networks as aid to integrated water resource planning. *Environ. Model. Software*, pages 231 – 242.

de Oude, P. and Pavlin, G. (2009). Efficient Distributed Bayesian Reasoning via Targeted Instantiation of Variables. pages 323–330.

Dörner, D. and Schaub, H. (1994). Errors in Planning and Decision-making and the Nature of Human Information Processing. *Applied Psychology*, 43(4):433–453.

Düspohl, M., Frank, S., and Doell, P. (2012). A Review of Bayesian Networks as a Participatory Modeling Approach in Support of Sustainable Environmental Management. *Journal of Sustainable Development*, 5(12).

Gustafsson, F., Gunnarsson, F., Bergman, N., Forssell, U., Jansson, J., Karlsson, R., and Nordlund, P.-J. (2002). Particle Filters for Positioning, Navigation, and Tracking. *IEEE Transactions on Signal Processing*, 50(2):425–437.

Johnson, S., Mengersen, K., de Waal, A., Marnewick, K., Cilliers, D., Houser, A. M., and Boast, L. (2010). Modelling cheetah relocation success in southern Africa using an Iterative Bayesian Network Development Cycle. *Ecological Modelling*, 221(4):641 – 651.

Koen, H., de Villiers, P., Pavlin, G., de Waal, A., de Oude, P., and Mignet, F. (2014). A Framework for Inferring Predictive Distributions of Rhino Poaching Events through Causal Modelling. *17^{th} International Conference on Information Fusion, 2014*.

Paskin, M. and Guestrin, C. (2004). Robust Probabilistic Inference in Distributed Systems. In *Proceedings of the 20th Annual Conference on Uncertainty in Artificial Intelligence (UAI-04)*, pages 436–445, Banff, Canada. AUAI Press.

Pavlin, G., de Oude, P., Maris, M., Nunnink, J., and Hood, T. (2010). A Multi-Agent Systems Approach to Distributed Bayesian Information Fusion. *Information Fusion*, 11(3):267 – 282.

Pearl, J. (1988). *Probabilistic Reasoning in Intelligent Systems: Networks of Plausible Inference*. Morgan Kaufmann Publishers Inc., San Francisco, CA, USA.

Pullar, D. and Phan, T. (2007). Using a Bayesian Network in a GIS to model relationships and threats to koala populations close to urban environments. In Oxley, L. and Kulasiri, D., editors, *MODSIM 2007: Land, water and environmental management: Integrated systems for sustainability.*, pages 1370–1375.

Thrun, S., Burgard, W., and Fox, D. (2005). *Probabilistic Robotics (Intelligent Robotics and Autonomous Agents)*. The MIT Press.

Xiang, Y. (2002). *Probabilistic Reasoning in Multiagent Systems: A Graphical Models Approach*. Cambridge University Press.