June 2008 Volume 3 Number 1 ISSN 1557-6418

JAIF 08

Journal of Advances in Information Fusion

A semi-annual archival publication of the International Society of Information Fusion

Regular Papers | Page

From the Editor-In-Chief

Value of Peer Review...

www.isif.org

ISIF

INTERNATIONAL SOCIETY OF INFORMATION FUSION

The International Society of Information Fusion (ISIF) seeks to be the premier professional society and global information resource for multidisciplinary approaches for theoretical and applied INFORMATION FUSION technologies. Technical areas of interest include target tracking, detection theory, applications for information fusion methods, image fusion, fusion systems architectures and management issues, classification, learning, data mining, Bayesian and reasoning methods.

Manuscripts are submitted at http://jaif.msubmit.net. If in doubt about the proper editorial area of a contribution, submit it under the unknown area.

Journal of Information Advances in Information Fusion (ISSN 1557-6418) is published semi-annually by the International Society of Information Fusion. The responsibility for the contents rests upon the authors and not upon ISIF, the Society, or its members. ISIF is California Nonprofit Public Benefit Corporation at P.O. Box 4631, Mountain View, California 94040.

From the Editor-In-Chief:

June 2008

Value of Peer-Review

I have served as an editor for peer-reviewed journals since 1996 and during those thirteen years, I have been exposed to numerous complaints about the peer review and publication processes. The most common complaint relates to the delay in publication that results from the peer review process. Another common complaint relates to popular opinion that novel results from newcomers are rejected to protect the interests of those in the research community. In other words, the researchers are members of a "good old boys club" that rejects the results of those outside the "club." Before I address these criticisms, let us consider the value of the peer review and publication processes.

In order to illustrate the value of peer review and publication process for journals, I conducted a search of IEEE Xplore for papers with keywords of interest to those involved in information fusion. I restricted the searches to IEEE journal articles in 2008. I then restricted the search to IEEE conference articles in 2008. I also restricted the searches to IET journal articles or IET conference articles. The results are shown in the table. For "target tracking" as a keyword, 83 articles were found in IEEE journals and more than 500 articles were found in IEEE conference proceedings. These are the numbers of manuscripts published in only one year. Similar ratios of IEEE journal articles to IEEE conference articles were found for "data fusion," "information fusion," and "nonlinear filtering" as keywords. Since it is unrealistic for a researcher to read all of these manuscripts in their active areas, researchers must focus their energy on selected articles. By focusing on journal articles, researchers can greatly improve their efficiency.

Why should researchers focus their energy on the journal articles? First, each journal article has been reviewed by at least two peers who have recommended it for publication as an original contribution with accurate results. The standards for peer review of conference papers vary from a single review of an abstract to a review of the complete manuscript. Second, the referees of journal articles provide input to the authors for improving their manuscript and two or three review cycles are conducted to ensure that the comments of the referees are properly addressed. The peer review of conference papers is often time driven by the conference schedule and typically includes no checks of manuscript revisions. Third, the editorial staff of journals read the manuscript for inconsistencies, clarity, and typographical errors, while the final version articles of conference proceedings are seldom read by the technical program committee. Therefore, the peer review and publication processes of journals greatly improve the efficiency of researchers by helping them to focus their efforts on articles that have meet standards on quality and originality. With the high rate of publications in the areas of information fusion, it is unrealistic for researchers to spend time on all articles in their area and spending time on articles that are not original or incorrect will further degrade their efficiency.

Information Fusion Related Articles Found
in IEEE Xplore from 2008

Keyword	Number of Articles from 2008 in IEEE Xplore			
	IEEE Journals	IET Journals	IEEE Conferences	IET Conferences
Target Tracking	83	16	>500	19
Nonlinear Filtering	43	2	256	2
Data Fusion	42	2	383	5
Information Fusion	17	0	238	0

During my first few years as an editor, it was common for the first review cycle for a manuscript to take a year and completion of the peer review process would take more than two years. Today, with the use of web-based systems for managing the peer review process, the typical review periods have been reduced to four months for the first cycle of review and less than a year to complete the peer review process. Thus, significant progress has been made in reducing the delay in the publication of manuscripts due to the peer review. Considering the value of a rigorous peer review process, the current delays are very reasonable. This is particularly true for JAIF that is operated by editors and referees who are volunteers. Achieving further reductions in the time required by the peer review process presents a significant challenge because the more talented reviewers tend to be very busy with other projects for their employer and JAIF is competing for their time to serve our profession as a volunteer.

From history, we know that innovative solutions to challenging problems have been rejected by the peer review process. With today's standards of three or four reviewers per manuscript, the rejection of papers with innovative results is less likely. If one of the reviewers provides evidence of a thorough review and recommends the manuscript for publication, it is difficult for an Associate Editor to reject the paper unless a second reviewer finds a technical flaw in the results. Since all manuscript reviews are archived in the web-based review system, the peer review process can be audited at any time by the EIC. Furthermore, the EIC regularly monitors the performance of the Associate Editors in order to ensure the integrity of the peer review process. Also, the authors can appeal the publication decision of the Associate Editor to the Area Editor and EIC. Since the review process is thoroughly documented and archived, these appeals can be addressed fairly in a timely manner. Thus, the likelihood of a paper being rejected because of the personal biases of the reviewers or editors is greatly diminished with the peer review processes in use by JAIF.

Sincerely,
William Dale Blair
Editor-In-Chief

Optimal Policies for a Class of Restless Multiarmed Bandit Scheduling Problems with Applications to Sensor Management

R. B. WASHBURN
M. K. SCHNEIDER

We present verifiable sufficient conditions for determining optimal policies for finite horizon, discrete time Markov decision problems (MDPs) with terminal reward. In particular, a control policy is optimal for the MDP if (i) it is optimal at the terminal time, (ii) immediate decisions can be deferred to future times, and (iii) the probability transition functions are commutative with respect to different decisions. The result applies to a class of finite horizon restless multiarmed bandit problems that arise in sensor management applications, which we illustrate with a pair of examples.

Manuscript received January 26, 2006; revised February 25, 2008; released for publication May 2, 2008.

Refereeing of this contribution was handled by Chee Chong.

This material is based upon work supported by the United States Air Force under Contract No. F33615-02-C-1197.

Authors' addresses: R. B. Washburn, Parietal Systems Inc., 510 Turnpike Street, Suite 201, North Andover, MA 01845, robert.washburn@parietal-systems.com; M. K. Schneider, BAE Systems, 6 New England Executive Park, Burlington, MA 01803, michael.k.schneider@baesystems.com.

1557-6418/08/$17.00 © 2008 JAIF

1. INTRODUCTION

Consider the Markov decision problems (MDPs) arising in the areas of intelligence, surveillance, and reconnaissance in which one selects among different targets for observation so as to track their position and classify them from noisy data [9], [10]; medicine in which one selects among different regimens to treat a patient [1]; and computer network security in which one selects different computer processes for observation so as to find ones exhibiting malicious behavior [6]. These MDPs all have a special structure. Specifically, they are discrete-time MDPs in which one controls the evolution of a set of Markov processes. There are two possible transition probability functions for the processes. The control at a given time selects a subset of processes, which then transition independently according to the controlled transition probability; the remaining processes transition independently according to the uncontrolled transition probability. Rewards are additive across processes and accumulated over time. The control problem is one of determining a policy to select controls so as to maximize expected rewards. MDPs with this structure have been termed restless bandit problems [15]. Our particular interest in such problems is in developing methods for deriving optimal solutions to them. Such solutions may be important of themselves as a control solution or may be useful for analyzing a problem in the process of developing a good suboptimal controller.

Restless bandits problems are a variation of a classical stochastic scheduling problem called a multiarmed bandit problem. It differs from the restless bandits problems considered here in two key respects. The first is that the states of the unselected process in the multiarmed bandit problem do not change. Second, the rewards in a multiarmed bandit problem are accumulated over an infinite horizon, discounting future rewards. Note that this is a significant difference because the time remaining in the horizon is essentially a component of the state and does not change for a multiarmed bandit problem but does change for the finite horizon restless bandit problems considered here.

A number of techniques have been previously developed for computing solutions to restless bandits problems. For example, index rules have been shown to optimally solve classical multiarmed bandit scheduling problems [2], [4]. Generalizations of this result have been conjectured, and some of them have been proven to apply to other classes of restless bandit problems [14], [15]. Proofs establishing the optimality of controls for finite-horizon restless bandit problems with particular reward structures have also been presented [1], [3], [5]. Each of these results describes a set of conditions for a control to be optimal for a restless bandit problem.

This paper introduces a set of novel conditions that are sufficient for a control policy to be optimal for a finite-horizon MDP. The conditions are readily verified for a specified control policy and are convenient for verifying the optimality of controls such as priority index rules [3], [4], [15]. We have been able to apply the conditions to verify the optimality of controls for a number of different restless bandit problems [12]. In particular, our conditions can be used to verify special cases of previous results on the optimality of controls for MDPs in [1] and [3]. We have also been able to apply our results to verify the optimality of controls for MDPs arising in sensor management [9], [10] applications for which no existing proofs of optimality existed. General conditions have not been previously developed that can verify optimality of strategies for such a range of examples. These sufficient conditions may prove useful in helping to identify and verify optimal policies for similar multiarmed bandit problems and for developing good suboptimal solutions for more complex problems. The latter is illustrated by the work in [3] and [13]. In both cases, a priority index rule is proven to be optimal for special cases of a more general restless bandit problem. Although the policies are not optimal for the general problems, empirical results reported in the papers demonstrate that the policies perform well even for the more general case.

An example of the type of sensor management problem we are interested in is that of managing an airborne sensor to collect data on ground targets. The goals could be either to collect kinematic data so as to track the kinematic state of the targets or to collect discriminative data so as to classify them. The control problem is one of selecting a subset of the targets for observation, subject to sensor field of view constraints, given current estimates of target and class. The objective is to optimize the quality of the data collected within a finite time horizon. Such sensor management problems are naturally modeled as restless bandit problems [8], [7], [11]. The quality of the data for each target can be modeled as a Markov process, which transitions differently depending on whether the target is selected for observation or not.

The details of our results and applications of them to sensor management problems are provided in the rest of the paper. Section 2 presents our results on sufficient conditions for a control to be optimal for a Markov decision problem. Section 3 applies the result to a general sensor management problem. Finally, examples of managing a sensor to perform binary classification and target tracking are presented in Section 4-A. For the reader's convenience, we have relegated the proofs for the main theorem and all propositions to the Appendix.

2. SUFFICIENT CONDITION

We will denote a MDP with terminal reward by the tuple $(\mathbb{X},\mathbb{U},p_u,R,T)$ where $\mathbb{X}$ denotes the discrete (finite or countable) state space of the Markov chain, $\mathbb{U}$ denotes the finite set of possible decisions, $\{p_u : u \in \mathbb{U}\}$ is the collection of transition probabilities parameterized by the decision u, R is the terminal reward function $R : \mathbb{X} \to \mathbb{R}$, and the integer T is the terminal time.

If $X(t)$, $0 \le t \le T$ is the Markov process with decisions $U(t)$, $0 \le t \le T-1$, and terminal reward $R(X(T))$, the MDP problem is to select U to maximize the expected value $E\{R(X(T))\}$ of the terminal reward. We assume that the decision $U(t)$ depends only on $X(0),\ldots,X(t)$ and that

$$\Pr\{X(t+1)=\xi \mid X(t)=x, U(t)=u\} = p_u(\xi \mid x). \tag{1}$$

The dynamic programming equations for the optimal reward-to-go function $V(x,t)$ for the MDP $(\mathbb{X},\mathbb{U},p_u,R,T)$ are given as follows. The terminal condition is

$$V(x,T) = R(x). \tag{2}$$

The recursion is

$$V(x,t) = \max_u \{V_u(x,t)\} \tag{3}$$

for times $0 \le t \le T-1$, where we define

$$V_u(x,t) := \sum_\xi V(\xi, t+1) p_u(\xi \mid x). \tag{4}$$

Also, any u that achieves the maximum in (3) is defined to be an optimal decision at time t when in state x.

DEFINITION 1 Suppose that the MDP $(\mathbb{X},\mathbb{U},p_u,R,T)$ has the probability transition functions $p_u(\xi \mid x)$ for $x,\xi \in \mathbb{X}$, $u \in \mathbb{U}$, and terminal reward $R(x)$ for $x \in \mathbb{X}$. If $\Phi(x,t) \subset \mathbb{U}$ for each $x \in \mathbb{X}$ and $0 \le t \le T-1$, we say that Φ is a **strategy set** for the MDP.

DEFINITION 2 If Φ is a strategy set for the MDP $(\mathbb{X},\mathbb{U},p_u,R,T)$ and if for each $x \in \mathbb{X}$, the expected value for selecting each $u \in \Phi(x,T-1)$ achieves the maximum value, *i.e.*,

$$\sum_y R(y) p_u(y \mid x) = \max_v \sum_y R(y) p_v(y \mid x), \tag{5}$$

we say that the strategy set Φ is **terminally optimal** for the MDP.

DEFINITION 3 More generally, for, $0 \le t \le T-1$ and $x \in \mathbb{X}$, define $\Phi^*(x,t)$ to be the **set of optimal strategies**

$$\Phi^*(x,t) = \left\{u : V_u(x,t) = \max_w V_w(x,t)\right\}. \tag{6}$$

What follows is a pair of definitions for properties of the strategy set and MDP as well as a theorem concerning the optimality of the strategy set when these conditions hold. Note that the properties in the definitions are abstract at this point and are illustrated later in this section with an example.

DEFINITION 4 If Φ is a strategy set for the MDP $(\mathbb{X},\mathbb{U},p_u,R,T)$, and if for each t such that $0 \le t \le T-2$, each $x \in \mathbb{X}$,

$$\begin{aligned} &u \in \Phi(x,t), \qquad V_v(x,t) > V_u(x,t), \qquad \text{and} \\ &p_v(y \mid x) > 0 \quad \text{imply} \quad u \in \Phi(y,t+1), \end{aligned} \tag{7}$$

then we say that **decisions are deferrable** in the strategy set Φ.

REMARK 1 Definition 4 gives conditions under which if u is in the decision set at the current time but a different decision v is made, then u is still in the decision set at the next time. This condition allows using an interchange argument to prove the optimality of the decision set (Theorem 1). Unfortunately, Definition 4 is too hard to check in practice. However, it is implied by various stronger conditions that are easier to check. For example, if for each t such that $0 \le t \le T-2$, each $x \in \mathbb{X}$, and for all u, v, y,

$$\begin{aligned} &u \in \Phi(x,t),\ v \neq u, \qquad \text{and} \\ &p_v(y \mid x) > 0 \quad \text{imply} \quad u \in \Phi(y, t+1), \end{aligned} \tag{8}$$

then decisions are deferrable in the strategy set Φ. This condition is stronger than the definition, since $V_v(x,t) > V_u(x,t)$ obviously implies that $v \neq u$. At the end of this section we prove another stronger condition for problems with symmetry.

DEFINITION 5 We say that the probability transition functions $p_u(\xi \mid x)$ are **commutative** if for all $u, v \in \mathbb{U}$,

$$\sum_{\eta} p_u(\xi \mid \eta) p_v(\eta \mid x) = \sum_{\eta} p_v(\xi \mid \eta) p_u(\eta \mid x) \tag{9}$$

for all $x, \xi \in \mathbb{X}$.

THEOREM 1 *Suppose that Φ is a strategy set for an MDP $(\mathbb{X}, \mathbb{U}, p_u, R, T)$ with commutative transition probability functions p_u, such that Φ is terminally optimal and decisions in Φ are deferrable. Then the strategy set Φ is optimal in the sense that any decision $U(t) \in \Phi(X(t), t)$ for $0 \le t \le T-1$, is an optimal decision for $(\mathbb{X}, \mathbb{U}, p_u, R, T)$.*

REMARK 2 If $\Phi^*(x,t)$ is the optimal strategy set for $(\mathbb{X}, \mathbb{U}, p_u, R, T)$ as defined in (6), then Φ^* is necessarily terminally optimal. It also necessarily satisfies the condition for deferrable decisions, simply because the hypothesis of the condition,

$$u \in \Phi^*(x,t),\ V_v(x,t) > V_u(x,t), \tag{10}$$

is always false. As we indicated in Remark 1, this condition is difficult to check in practice, but we can replace it with stronger conditions which do not refer to the optimal reward function. With these stronger conditions, it is important to have the third condition, commutativity of the transition probabilities, to prove the optimality of a proposed strategy set.

To conclude this section we will prove another stronger condition for deferrable decisions in Φ based on symmetric MDP problems. Note that for these problems, the state space of the Markov chain $\mathbb{X}$ is a product space $\mathbf{X}^n$, and the ith component of an element $x \in \mathbb{X}$ is denoted by x_i.

DEFINITION 6 The MDP $(\mathbb{X}, \mathbb{U}, p_u, R, T)$ is symmetric if for some $n > 1$

$$\mathbb{X} = \mathbf{X}^n, \tag{11}$$

$$\mathbb{U} = \{1, \ldots, n\}, \tag{12}$$

$$p_{\pi(i)}(\pi y \mid \pi x) = p_i(y \mid x) \tag{13}$$

and

$$R(x) = R(\pi x) \tag{14}$$

where π permutes the components of x, y, namely

$$\pi x = (x_{\pi(1)}, \ldots, x_{\pi(n)}), \tag{15}$$

for any permutation π of $\{1, \ldots, n\}$ and all $x \in \mathbf{X}^n$.

REMARK 3 Note that the symmetry conditions in Definition 6 all hold for multiarmed bandit scheduling problems. However, the symmetric scheduling problem considered here still differs from the multiarmed bandit problem in two key respects. First, the states for unobserved processes may change, whereas, for multiarmed bandit problems, the states of unobserved processes remain the same. Second, the horizon here is finite whereas the horizon for multiarmed bandit problems is infinite.

PROPOSITION 1 *Suppose that the MDP $(\mathbb{X}, \mathbb{U}, p_u, R, T)$ is symmetric. Then if for $0 \le t \le T-2$ and all $x \in \mathbb{X}$*

$$\begin{aligned} &u \in \Phi(x,t),\ x_v \neq x_u \qquad \textit{and} \\ &p_v(y \mid x) > 0 \quad \textit{imply} \quad u \in \Phi(y, t+1), \end{aligned} \tag{16}$$

decisions are deferrable in Φ.

An example of a strategy for a symmetric MDP that is terminally optimal, deferrable, and commutative is as follows. Suppose $\mathbf{X} = \mathbb{N}$ and

$$R(x) = \sum_{i=1}^{n} \delta(x_i) \tag{17}$$

where

$$\delta(x_i) = \begin{cases} 1 & \text{if} \quad x_i = 0 \\ 0 & \text{otherwise.} \end{cases} \tag{18}$$

Moreover, define the transition probabilities as follows. If $x_i > 0$

$$p_i(y \mid x) = \begin{cases} 1/2 & \text{if} \quad y_i = x_i \pm 1 \\ 0 & \text{otherwise} \end{cases} \tag{19}$$

and if $x_i = 0$,

$$p_i(y \mid x) = \begin{cases} 1 & \text{if} \quad y_i = 0 \\ 0 & \text{otherwise.} \end{cases} \tag{20}$$

This is a MDP in which there are n independent Markov processes x_i evolving on the non-negative integers. A process x_i transitions only if it is selected by the control and is equally likely to increase or decrease. The value 0 is a trapping state. The objective is to drive as many processes as possible to the trapping state. Define the strategy set to be the non-zero processes of minimal value

$$\Phi(x,t) = \arg\min_i \{x_i > 0\}. \tag{21}$$

This strategy set is terminally optimal because selecting a process with value 1 is optimal at the last stage. The strategy set is deferrable because the condition of Proposition 1 holds. Specifically, if $u \in \Phi(x,t)$ and $v \in \mathbb{U}$ is such that $x_v \neq x_u$, then $x_v > x_u$ and $p_v(y \mid x) > 0$ implies $y_v \geq x_u$ so that

$$u \in \arg\min_i \{y_i > 0\} = \Phi(y, t+1). \tag{22}$$

Finally, the probability transitions are commutative because

$$\sum_\eta p_u(\xi \mid \eta) p_v(\eta \mid x)$$

$$= \begin{cases} 1/4 & \text{if} \quad \xi_u = x_u \pm 1, \xi_v = x_v \pm 1 \\ 0 & \text{otherwise} \end{cases} \tag{23}$$

$$= \sum_\eta p_v(\xi \mid \eta) p_u(\eta \mid x). \tag{24}$$

Thus, the strategy is optimal for this problem by Proposition 1.

Applications of the results in this section to sensor management problems follow.

3. APPLICATIONS TO SENSOR MANAGEMENT PROBLEMS

The results are stated for a very general situation in Section 2, where few assumptions are made concerning the statistics of the Markov chain. However, the optimality conditions are expected to be useful for analyzing special cases of more general problems, in part to develop good heuristics for the general case. The purpose of this section is to specialize the optimality conditions to problems where the Markov chain is a product of independent, identically distributed chains, which is a common situation arising in some important special cases of sensor management problems.

Specifically, consider the sensor management problem where there are n targets and we can only observe one target at a time. In the simplest case, the decision $U(t)$ to make at each time t is only which target $i = 1,\ldots,n$ to observe. There is a Markov chain $X_i(t)$ corresponding to each target i, where $X_i(t)$ represents the information state of target i at time t. Typically, we assume that the chains $X_i(t)$ are independent and identically distributed, and that the selected (i.e., observed) chain transitions according to $p(\xi \mid x)$ and the $n-1$ unobserved chains transition according to $q(\xi \mid x)$. Moreover, the reward is typically additive over the n targets, namely

$$R(X_1(T),\ldots,X_n(T)) = \sum_{i=1}^n r(X_i(T)). \tag{25}$$

The resulting MDP $(\mathbb{X}, \mathbb{U}, p_u, R, T)$ has special structure where

$$\mathbb{X} = \mathbf{X}^n \text{ and } \mathbf{X} \text{ is the state space of one Markov chain } X_i \tag{26}$$

$$\mathbb{U} = \{1,\ldots,n\} \tag{27}$$

$$p_i(\xi \mid x) = p(\xi_i \mid x_i) \prod_{j \neq i} q(\xi_j \mid x_j) \qquad \text{for} \quad i \in \mathbb{U},\ x,\xi \in \mathbf{X}^n \tag{28}$$

$$R(x) = \sum_{i=1}^n r(x_i) \qquad \text{for} \quad x \in \mathbf{X}^n. \tag{29}$$

REMARK 4 If $s = |\mathbf{X}|$ is the number of states for each single Markov chain, then the computational complexity of the dynamic programming solution is $O(ns^{2n}T)$. Thus, for fixed s and T, the complexity is exponential in n. Furthermore, the memory requirements are exponential, namely $O(s^n T)$. In some cases we can find an optimal strategy of the form $U(t) \in \Phi((X_1(t),\ldots,X_n(t)),t)$ where

$$\Phi(x,t) = \{i : M_i(x_i,t) = \max_j M_j(x_j,t)\}. \tag{30}$$

This is what we call a priority index rule strategy. The $M_i(x_i,t)$ are indices that can be computed for each target with complexity $O(s^2T)$ (i.e., equivalent to solving the dynamic program for one target). Thus, the complexity of the n target strategy is $O(ns^2T)$ rather than $O(ns^{2n}T)$, linear in n rather than exponential in n.

For the class of transition probabilities $p_i(\xi \mid x)$ with structure (28), commutativity is equivalent to the commutativity of the transition functions p and q, as the following simple result shows.

PROPOSITION 2 *If the transition probability functions $p_i(\xi \mid x)$ defined for $\xi, x \in \mathbb{X}^n$ and $i \in \{1,\ldots,n\}$ satisfy*

$$p_i(\xi \mid x) = p(\xi_i \mid x_i) \prod_{j \neq i} q(\xi_j \mid x_j), \tag{31}$$

and if for all $\xi_1, x_1 \in X$,

$$\sum_{\eta_1} q(\xi_1 \mid \eta_1) p(\eta_1 \mid x_1) = \sum_{\eta_1} p(\xi_1 \mid \eta_1) q(\eta_1 \mid x_1), \tag{32}$$

then $p_i(\xi \mid x)$ are commutative transition probability functions for $\xi, x \in \mathbb{X}^n$.

REMARK 5 Note that commutativity always holds if p or q is the identity transition $\delta(\xi_i \mid x_i) = 1$ for $\xi_i = x_i$ and 0 otherwise. Note that $q = \delta$ is assumed true in (non-restless) multiarmed bandit problems. Also, classification sensor management problems often satisfy $q = \delta$ (i.e., the classification information state remains unchanged while the target is unobserved).

REMARK 6 Transition probabilities of the form

$$p_i(\xi \mid x) = p(\xi_i \mid x_i) \prod_{j \neq i} \delta(\xi_j \mid x_j) \tag{33}$$

and reward functions

$$R(x) = \sum_{i=1}^{n} r(x_i) \tag{34}$$

are obviously symmetric.

For this class of MDPs corresponding to sensor management problems, the general result (Theorem 1) becomes the following.

COROLLARY 1 *Suppose that the MDP* $(\mathbb{X},\mathbb{U},p_u,R,T)$ *has special symmetric structure where*

$$\mathbb{X} = \mathbf{X}^n \text{ and } \mathbf{X} \textit{ is the state space of one one Markov chain } X_i \tag{35}$$

$$\mathbb{U} = \{1,\ldots,n\} \tag{36}$$

$$p_i(\xi \mid x) = p(\xi_i \mid x_i)\prod_{j\neq i} q(\xi_j \mid x_j) \qquad \textit{for} \quad i\in\mathbb{U},\ x,\xi\in\mathbf{X}^n \tag{37}$$

$$R(x) = \sum_{i=1}^{n} r(x_i) \text{ for } x\in\mathbf{X}^n. \tag{38}$$

Then the strategy set Φ *is optimal if the following three conditions are met. The first condition is that p and q are commutative so that*

$$\sum_{\eta} p(\xi \mid \eta)q(\eta \mid x) = \sum_{\eta} q(\xi \mid \eta)p(\eta \mid x). \tag{39}$$

Suppose that $\Phi(x,t)$ *is a strategy set for* $x = (x_1,\ldots,x_n)$*. Then, the second condition is that* $i \in \Phi(x,T-1)$ *implies*

$$\sum_{y_i} r(y_i)p(y_i \mid x_i) - r(x_i) \geq \sum_{y_j} r(y_j)p(y_j \mid x_j) - r(x_j) \tag{40}$$

for all $j \neq i$*, and the third condition is that*

$$i\in\Phi(x_1,\ldots,x_i,\ldots,x_j,\ldots,x_n,t),\ x_i\neq x_j, \qquad \textit{and} \quad p(y_j \mid x_j) > 0 \tag{41}$$

implies that

$$i\in\Phi(x_1,\ldots,x_i,\ldots,y_j,\ldots,x_n,t+1). \tag{42}$$

PROOF The condition on $p_i(\xi \mid x)$ implies that it is commutative. The second condition implies that Φ is terminally optimal for the terminal reward $R(x)$, and the third condition implies that decisions in Φ are deferrable (Proposition 1). The result follows from Theorem 1.

4. SENSOR MANAGEMENT EXAMPLES

What follows are two examples that illustrate the application of the conditions presented in the paper. The first is a binary classification example, which is a type of finite horizon sensor management problem for which the states of unobserved processes (other than the time remaining until the end of the horizon) do not change. The second is a tracking problem, for which the states of unobserved processes do change. The examples illustrate how the novel conditions presented in this paper apply to a large class of problems that includes these two. The utility of such an analysis is that it sheds insight into sensor management problems and suggests heuristics that could be used for more general sensor management problems.

A. Binary Classification Problem

This problem is to classify as many of n objects as possible over a finite time horizon T given binary measurements of the objects. This problem is similar to the classical treasure hunting problem [2]. In that problem, one selects among a finite number of areas to search for treasure, but the treasure may be missed with a fixed probability. This is a special type of a multiarmed bandit problem in which a so-called deteriorating condition holds so that the optimal policy is a greedy policy. The binary classification problem considered here differs from the treasure hunting problem in two respects. The first is that the horizon here is finite whereas it is infinite for the treasure hunting problem. The second is that processes in the two problems represent different quantities and so have different transition probabilities. In the problem here, each process represents the probability that a target is of a particular type. In the treasure hunting problem, each process represents the probability that the treasure is present at a particular location. The details of the binary classification problem follow.

First, note that this problem is a partially observed Markov decision process (POMDP) that can be interpreted as an MDP with a countable state space. Suppose there are n random variables Z_i with values 0, 1 and that $\Pr\{Z_i = 1\} = p$ for all $i = 1,\ldots,n$. Suppose that the $Y_i(t)$ are $0,1$ observations of Z_i, and $Y_i(t)$ are independent and identically distributed conditioned on Z_i with

$$\Pr\{Y_i(t) = y \mid Z_i = z\} = (1-\varepsilon)\cdot\delta_{y,z} + \varepsilon\cdot(1-\delta_{y,z}), \tag{43}$$

where we use the notation $\delta_{y,z} = 1$ if $y = z$ and 0 otherwise. We assume that $\varepsilon < \frac{1}{2}$. Note that ε is the probability of classification error for one measurement.

Define the information state $X_i(t)$ as the conditional probability

$$X_i(t) = \Pr\{Z_i = 1 \mid Y_i(1),\ldots,Y_i(t)\}. \tag{44}$$

The objective of the problem is to maximize the expected reward

$$E\left\{\sum_{i=1}^{n} r(X_i(T))\right\} \tag{45}$$

at the terminal time T, where $r(x_i)$ is the individual reward

$$r(x_i) = \max_{d_i=0,1}\{r(d_i,1)x_i + r(d_i,0)(1-x_i)\} \tag{46}$$

and $r(d,z)$ are the rewards for the different types of outcomes (i.e., deciding d_i when the true state of i is z_i).

The processes $X_i(t)$ satisfy

$$X_i(0) = p \tag{47}$$

and for $t \geq 0$,

$$X_i(t+1) = \begin{cases} \dfrac{(1-\varepsilon)X_i(t)}{(1-2\varepsilon)X_i(t)+\varepsilon} & \text{with probability } (1-2\varepsilon)X_i(t)+\varepsilon \\ \dfrac{\varepsilon X_i(t)}{(2\varepsilon-1)X_i(t)+1-\varepsilon} & \text{with probability } (2\varepsilon-1)X_i(t)+1-\varepsilon. \end{cases} \tag{48}$$

Note that although $X_i(t)$ take values in $\mathbb{R}$, there are only a countable number of possible values they can take. Thus, $X_i(t) \in \mathbf{X} \subset \mathbb{R}$ where $\mathbf{X}$ is a countable set. Thus, we have an MDP $(\mathbb{X}, \mathbb{U}, p_u, R, T)$ where

$$\mathbb{X} = \mathbf{X}^n \tag{49}$$

$$\mathbb{U} = \{1,2,\ldots,n\} \tag{50}$$

$$p_i(\xi \mid x) = p(\xi_i \mid x_i) \prod_{j \neq i} \delta(\xi_j \mid x_j) \qquad \text{for} \quad i \in \mathbb{U},\ x, \xi \in \mathbf{X}^n \tag{51}$$

$$R(x) = \sum_{i=1}^{n} r(x_i) \qquad \text{for} \quad x \in \mathbf{X}^n \tag{52}$$

where $p(\xi_i \mid x_i)$ is defined by

$$p\left(\frac{(1-\varepsilon)x_i}{(1-2\varepsilon)x_i+\varepsilon} \mid x_i\right) = (1-2\varepsilon)x_i + \varepsilon \tag{53}$$

$$p\left(\frac{\varepsilon x_i}{(2\varepsilon-1)x_i+1-\varepsilon} \mid x_i\right) = (2\varepsilon-1)x_i + 1 - \varepsilon \tag{54}$$

and $r(x_i)$ is defined by (46). We will consider the special case for which $r(1,1) = r(0,0) = 1$ and $r(0,1) = r(1,0) = 0$ so that

$$r(x_i) = \tfrac{1}{2} + |x_i - \tfrac{1}{2}|, \tag{55}$$

and we will assume that the prior probability $p = \frac{1}{2}$. Note that if $p = \frac{1}{2}$, then

$$\mathbf{X} = \left\{ \frac{1}{1 + \left(\dfrac{\varepsilon}{1-\varepsilon}\right)^m} : \quad m = 0, \pm 1, \pm 2, \ldots \right\}. \tag{56}$$

PROPOSITION 3 *The strategy set Φ defined by*

$$\Phi(x) = \{i : |x_i - \tfrac{1}{2}| = \min_j |x_j - \tfrac{1}{2}|\} \tag{57}$$

is optimal for the binary classification problem with $r(1,1) = r(0,0) = 1$, $r(0,1) = r(1,0) = 0$, and prior probability $p = \frac{1}{2}$ for each object i.

B. Tracking Problem

The following is an example in which one is managing a sensor to track targets. Specifically, one is tracking targets over a finite horizon with a noisy sensor. At the end of the time horizon, the tracks are to be handed over to another sensor. The handover is successful if the track mean square error is smaller than the required level. The objective is to maximize the number of tracks that are successfully handed over.

Note that this example differs from the binary classification one in that unobserved chains have nontrivial dynamics. Specifically, the dynamics are those of the track error covariances. The conditions are used to verify the optimality of a strategy for an approximate model of the track error covariance where the increase in error when a track is unobserved is given by that of a Kalman filter, but the error reduction is approximated as being constant, independent of the initial error. The details of this example are as follows.

Consider the one-dimensional tracking problem in which there are n targets each of which is moving as a one-dimensional Brownian motion with process noise variance Λ_p. Location measurements have additive noise with variance Λ_m. The state of each track i at time t, for the purposes of sensor management, is the error variance $X_i(t)$. All tracks are initialized with the same error variance Λ_0, and all have the same desired error variance Λ_h at the end of the horizon T. The objective of the problem is to maximize the expected reward

$$E\left\{\sum_{i=1}^{n} r(X_i(T))\right\} \tag{58}$$

at the terminal time T, where $r(x_i)$ is the individual reward

$$r(x_i) = \begin{cases} 1 & \text{if} \quad x_i \leq \Lambda_h \\ 0 & \text{otherwise.} \end{cases} \tag{59}$$

Now suppose that the track error is approximated so that the track error reduction for observed tracks is constant, given by the error reduction from the desired value Λ_h. That is, if the error variance is initially Λ_h, then after one measurement update and one prediction, it is reduced by the amount $\Lambda_p - \Lambda_h^2/(\Lambda_h + \Lambda_m)$. Then, the dynamics of the processes $X_i(t)$ satisfy

$$X_i(0) = \Lambda_0 \tag{60}$$

and for unobserved processes for $t \geq 0$,

$$X_i(t+1) = X_i(t) + \Lambda_p. \tag{61}$$

For the process observed at $t \geq 0$,

$$X_i(t+1) = X_i(t) + \Lambda_p - \frac{\Lambda_h^2}{\Lambda_h + \Lambda_m}. \tag{62}$$

We will assume that $\Lambda_p < \Lambda_h^2/(\Lambda_h + \Lambda_m)$ so that the error for observed processes is always decreasing. This is an MDP $(\mathbb{X}, \mathbb{U}, p_u, R, T)$ with

$$\mathbb{X} = \mathbb{R}^n \tag{63}$$

$$\mathbb{U} = \{1, 2, \ldots, n\} \tag{64}$$

$$p_i(\xi \mid x) = p(\xi_i \mid x_i) \prod_{j \neq i} q(\xi_j \mid x_j) \qquad \text{for} \quad i \in \mathbb{U},\ x, \xi \in \mathbb{X} \tag{65}$$

$$R(x) = \sum_{i=1}^{n} r(x_i) \qquad \text{for} \quad x \in \mathbb{X} \tag{66}$$

where $p(\xi_i \mid x_i)$ is defined by

$$p\left(x_i + \Lambda_p - \frac{\Lambda_h^2}{\Lambda_h + \Lambda_m} \mid x_i\right) = 1 \tag{67}$$

$q(\xi_i \mid x_i)$ is defined by

$$q(x_i + \Lambda_p \mid x_i) = 1, \tag{68}$$

and $r(x_i)$ is defined by (46).

PROPOSITION 4 *The strategy set Φ defined by*

$$\Phi(x) = \left\{ i : x_i = \min_j \{x_j : x_j > \Lambda_h - \Lambda_p\} \right\} \tag{69}$$

is optimal for this tracking problem.

REMARK 7 Note that under this strategy, the approximate error variance $X_i(t) \geq 0$ because $X_i(t)$ will decrease only if the process is chosen for observation, which will only occur if $X_i(t) > \Lambda_h - \Lambda_p$ and

$$X_i(t) + \Lambda_p - \frac{\Lambda_h^2}{\Lambda_h + \Lambda_m} \geq \Lambda_h - \frac{\Lambda_h^2}{\Lambda_h + \Lambda_m} > 0. \tag{70}$$

5. CONCLUSION

Thus, the sufficient conditions stated in Section 2 are useful for establishing optimality of sensor management strategies. Note that the optimal strategy for the binary classification and tracking examples presented in Section 4-A are priority index rule strategies, as defined in Section 3. Priority index rules are optimal strategies for other sensor management problems including those in [1], [5], [3]. However, the conditions in this paper do not imply optimality of these strategies except for some special cases of the sensor management problem being solved. Whether there exists a generalization of the results in this paper that implies optimality of priority index rules for general sensor management problems and other restless bandit problems is an open question.

APPENDIX. PROOFS OF RESULTS

A. Proof of Theorem 1

For, $0 \leq t \leq T-1$ and $x \in \mathbb{X}$, let $\Phi^*(x,t)$ be the set of optimal strategies, as defined in Definition 3. We want to prove that

$$\Phi(x,t) \subset \Phi^*(x,t). \tag{71}$$

The terminal optimality condition is equivalent to

$$\Phi(x, T-1) \subset \Phi^*(x, T-1). \tag{72}$$

Thus, assume that $\Phi(x,t+1) \subset \Phi^*(x,t+1)$ is true for $t < T-1$ and prove (71) from it. Suppose that $u \in \Phi(x,t)$ and $u \notin \Phi^*(x,t)$. Clearly $\Phi^*(x,t) \neq \emptyset$ and there is $v \in \Phi^*(x,t)$ such that $V_v(x,t) > V_u(x,t)$. The condition that decisions in Φ are deferrable implies that $u \in \Phi(X(t+1), t+1)$ where $X(t+1)$ results from using $U(t) = v$. The induction hypothesis implies that

$$\Phi(X(t+1), t+1) \subset \Phi^*(X(t+1), t+1), \tag{73}$$

so that $U(t+1) = u$ is an optimal decision.

We now can use the commutativity of the transitions p_w to show that the sequence of decisions $U(t) = u$, $U(t+1) = v$ has the same expected value-to-go as the sequence of decisions $U(t) = v$, $U(t+1) = u$ and must be optimal too. Specifically, note that starting from $X(t)$, if $X(t+2)$ is the state resulting from $U(t) = v$, $U(t+1) = u$ and $\tilde{X}(t+2)$ is the state resulting from $U(t) = u$, $U(t+1) = v$, then commutativity implies that $X(t+2)$ and $\tilde{X}(t+2)$ have the same distribution. By assumption (induction) the decisions $U(t) = v$, $U(t+1) = u$ are optimal and have the value-to-go

$$V(X(t),t) = E\{V(X(t+2), t+2) \mid X(t)\}. \tag{74}$$

Commutativity implies that

$$E\{V(X(t+2), t+2) \mid X(t)\} = E\{V(\tilde{X}(t+2), t+2) \mid X(t)\}, \tag{75}$$

which implies that $U(t) = u$, $U(t+1) = v$ must also be optimal decisions. Thus, u is optimal, contrary to assumption and we must have $u \in \Phi^*(x,t)$.

B. Proof of Proposition 1

First, we show by induction that the symmetry assumption implies that the optimal reward-to-go satisfies

$$V_v(x,t) = V_{\pi(v)}(\pi x, t) \tag{76}$$

for all permutations π. Let x denote a vector in $\mathbf{X}^n = \mathbb{X}$. By definition of symmetry

$$V_{\pi(u)}(\pi x, T-1) = \sum_y R(\pi y) p_{\pi(u)}(\pi y \mid \pi x) \tag{77}$$

$$= \sum_y R(y) p_u(y \mid x) \tag{78}$$

$$= V_u(x, T-1). \tag{79}$$

Now, assume that

$$V_v(x,t+1) = V_{\pi(v)}(\pi x, t+1) \tag{80}$$

for all x, π and prove it for t. Note that the induction hypothesis implies that

$$V(x,t+1) = \max_v V_v(x,t+1) \tag{81}$$

$$= \max_v V_{\pi(v)}(\pi x, t+1) \tag{82}$$

$$= V(\pi x, t+1). \tag{83}$$

By symmetry assumptions,

$$V_u(x,t) = \sum_y V(y,t+1) p_u(y \mid x) \tag{84}$$

$$= \sum_y V(\pi y, t+1) p_{\pi(u)}(\pi y \mid \pi x) \tag{85}$$

$$= \sum_y V(y,t+1) p_{\pi(u)}(y \mid \pi x) \tag{86}$$

$$= V_{\pi(u)}(\pi x, t) \tag{87}$$

which completes the induction.

Now, to prove the statement of Proposition 1, suppose that $u \in \Phi(x,t)$, $V_v(x,t) > V_u(x,t)$ and $p_v(y \mid x) > 0$. We have just shown that

$$V_v(x,t) = V_{\pi(v)}(\pi x, t) \tag{88}$$

for all permutations π. Let π be the permutation that interchanges v and u. Then if $x_v = x_u$, $V_v(x,t) = V_u(x,t)$. Thus, $V_v(x,t) > V_u(x,t)$ implies that $x_v \neq x_u$. By the proposition's assumption, it follows that $u \in \Phi(y,t+1)$, which proves the result.

C. Proof of Proposition 2

Note that for $i \neq j$,

$$\sum_\eta p_i(\xi \mid \eta) p_j(\eta \mid x) \tag{89}$$

$$= \sum_\eta p(\xi_i \mid \eta_i) \prod_{k \neq i} q(\xi_k \mid \eta_k) p(\eta_j \mid x_j) \prod_{k \neq j} q(\eta_k \mid x_k) \tag{90}$$

$$= \sum_{\eta_j} q(\xi_j \mid \eta_j) p(\eta_j \mid x_j) \sum_{\eta_i} p(\xi_i \mid \eta_i) q(\eta_i \mid x_i)$$

$$\times \prod_{k \neq i,j} \sum_{\eta_k} q(\xi_k \mid \eta_k) q(\eta_k \mid x_k). \tag{91}$$

By assumption

$$\sum_{\eta_j} q(\xi_j \mid \eta_j) p(\eta_j \mid x_j) = \sum_{\eta_j} p(\xi_j \mid \eta_j) q(\eta_j \mid x_j) \tag{92}$$

and

$$\sum_{\eta_i} q(\xi_i \mid \eta_i) p(\eta_i \mid x_i) = \sum_{\eta_i} p(\xi_i \mid \eta_i) q(\eta_i \mid x_i). \tag{93}$$

Thus,

$$\sum_{\eta_j} q(\xi_j \mid \eta_j) p(\eta_j \mid x_j) \sum_{\eta_i} p(\xi_i \mid \eta_i) q(\eta_i \mid x_i)$$

$$\times \prod_{k \neq i,j} \sum_{\eta_k} q(\xi_k \mid \eta_k) q(\eta_k \mid x_k) \tag{94}$$

$$= \sum_{\eta_j} p(\xi_j \mid \eta_j) q(\eta_j \mid x_j) \sum_{\eta_i} q(\xi_i \mid \eta_i) p(\eta_i \mid x_i)$$

$$\times \prod_{k \neq i,j} \sum_{\eta_k} q(\xi_k \mid \eta_k) q(\eta_k \mid x_k) \tag{95}$$

$$= \sum_\eta p_j(\xi \mid \eta) p_i(\eta \mid x), \tag{96}$$

proving that

$$\sum_\eta p_i(\xi \mid \eta) p_j(\eta \mid x) = \sum_\eta p_j(\xi \mid \eta) p_i(\eta \mid x). \tag{97}$$

D. Proof of Proposition 3

The proof verifies that the three conditions of Corollary 1 hold.

First, the transition probabilities $p_i(\xi \mid x)$ are obviously commutable and symmetric, and the reward function $R(x)$ is obviously symmetric.

Now, note that

$$\sum_{y_i} [R(y_i) - R(x_i)] p(y_i \mid x_i) \tag{98}$$

$$= -\tfrac{1}{2} - |x_i - \tfrac{1}{2}| \tag{99}$$

$$+ \left(\frac{1}{2} + \left| \frac{(1-\varepsilon)x_i}{(1-2\varepsilon)x_i + \varepsilon} - \frac{1}{2} \right| \right) ((1-2\varepsilon)x_i + \varepsilon) \tag{100}$$

$$+ \left(\frac{1}{2} + \left| \frac{\varepsilon x_i}{(2\varepsilon - 1)x_i + 1 - \varepsilon} - \frac{1}{2} \right| \right)$$

$$\times ((2\varepsilon - 1)x_i + 1 - \varepsilon). \tag{101}$$

This simplifies to

$$\sum_{y_i} [R(y_i) - R(x_i)] p(y_i \mid x_i)$$

$$= -|x_i - \tfrac{1}{2}| + \tfrac{1}{2}|x_i - \varepsilon| + \tfrac{1}{2}|(1 - x_i) - \varepsilon|, \tag{102}$$

which is equivalent to

$$\sum_y [R(y) - R(x_i)] p(y \mid x_i)$$

$$= \begin{cases} 0 & \text{for} \quad 0 \le x_i \le \varepsilon \\ x_i - \varepsilon & \text{for} \quad \varepsilon \le x_i \le \frac{1}{2} \\ 1 - x_i - \varepsilon & \text{for} \quad \frac{1}{2} \le x_i \le 1 - \varepsilon \\ 0 & \text{for} \quad 1 - \varepsilon \le x_i \le 1 \end{cases}. \tag{103}$$

Note that because

$$x_i = \frac{1}{1 + \left(\frac{\varepsilon}{1-\varepsilon}\right)^m}, \tag{104}$$

if $m < 0$, then

$$x_i \leq \frac{1}{1 + \left(\frac{\varepsilon}{1-\varepsilon}\right)^{-1}} = \varepsilon \tag{105}$$

and if $m > 0$, then

$$x_i \geq \frac{1}{1 + \left(\frac{\varepsilon}{1-\varepsilon}\right)} = 1 - \varepsilon. \tag{106}$$

Thus,

$$\sum_{y_i}[R(y_i) - R(x_i)]p(y_i \mid x_i) = \begin{cases} 0 & \text{for} \quad x_i \neq \frac{1}{2} \\ \frac{1}{2} - \varepsilon & \text{for} \quad x_i = \frac{1}{2} \end{cases}. \tag{107}$$

In particular,

$$\max_j \left\{ \sum_{y_j}[R(y_j) - R(x_j)]p(y_j \mid x_j) \right\} = \begin{cases} 0 & \text{if all} \quad x_i \neq \frac{1}{2} \\ \frac{1}{2} - \varepsilon & \text{if some} \quad x_i = \frac{1}{2} \end{cases} \tag{108}$$

and if

$$|x_i - \tfrac{1}{2}| = \min_j |x_j - \tfrac{1}{2}|, \tag{109}$$

then

$$\sum_{y_i}[R(y_i) - R(x_i)]p(y_i \mid x_i) = \max_j \left\{ \sum_{y_j}[R(y_j) - R(x_j)]p(y_j \mid x_j) \right\}. \tag{110}$$

This shows that the second condition of Corollary 1 holds.

To show that the third condition of Corollary 1 holds, suppose that $i \in \Phi(x)$ so that

$$|x_i - \tfrac{1}{2}| = \min_k |x_k - \tfrac{1}{2}| \tag{111}$$

and suppose $x_j \neq x_i$, $p(y_j \mid x_j) > 0$. Thus,

$$y_j = \frac{(1-\varepsilon)x_j}{(1-2\varepsilon)x_j + \varepsilon} \tag{112}$$

or

$$y_j = \frac{\varepsilon x_j}{(2\varepsilon - 1)x_j + 1 - \varepsilon}. \tag{113}$$

If $|x_j - \frac{1}{2}| > |x_i - \frac{1}{2}|$, then it is easy to see that $|y_j - \frac{1}{2}| \geq |x_i - \frac{1}{2}|$ and therefore $i \in \Phi(x_1, \ldots, y_j, \ldots, x_n)$. However, it is possible that $|x_j - \frac{1}{2}| = |x_i - \frac{1}{2}|$ and $x_i \neq x_j$. If $x_i \neq \frac{1}{2}$, then the conclusion is not true, because one of the two values of y_j is closer to $\frac{1}{2}$ than x_i.

However, we can easily extend the proposition to cover this case. Note that if $|x_j - \frac{1}{2}| = |x_i - \frac{1}{2}|$, then $x_j = 1 - x_i$. The classification problem is invariant under the transformation $x_i \to 1 - x_i$, and in particular,

$$V(\ldots, x_i, \ldots, \tau) = V(\ldots, 1 - x_i, \ldots, \tau). \tag{114}$$

Furthermore, if $p(y_i \mid x_i) > 0$, then $p(1 - y_i \mid 1 - x_i) > 0$. It follows that

$$V_i(\ldots, x_i, \ldots, \tau) = V_i(\ldots, 1 - x_i, \ldots, \tau). \tag{115}$$

As a consequence of this and the symmetry of V, we find that $|x_j - \frac{1}{2}| = |x_i - \frac{1}{2}|$ implies that

$$V_i(x, \tau) = V_j(x, \tau). \tag{116}$$

Thus, $i, j \in \Phi(x)$ implies that $V_i(x,\tau) = V_j(x,\tau)$. This is sufficient to extend the proposition because if $V_j(x,t) > V_i(x,t)$, then both $x_i \neq x_j$ and $|x_j - \frac{1}{2}| \neq |x_i - \frac{1}{2}|$. Thus, we can apply the earlier argument to show that $i \in \Phi(x_1, \ldots, y_j, \ldots, x_n)$. Consequently, the third and final condition of Corollary 1 holds so that Φ is optimal.

E. Proof of Proposition 4

The optimality of Proposition 4 will be established by verifying that the three conditions of Corollary 1 hold.

First, note that the distributions p and q are commutative since

$$\sum_\eta p(\xi \mid \eta) q(\eta \mid x) = \sum_\eta q(\xi \mid \eta) p(\eta \mid x) = \begin{cases} 1 & \text{if} \quad \xi = x + 2\Lambda_p - \dfrac{\Lambda_h^2}{\Lambda_h + \Lambda_p} \\ 0 & \text{otherwise.} \end{cases} \tag{117}$$

Moreover,

$$\sum_{y_i} r(y_i) p(y_i \mid x_i) - r(x_i) = \begin{cases} 0 & \text{if} \quad x_i \leq \Lambda_h - \Lambda_p \\ 1 & \text{if} \quad \Lambda_h - \Lambda_p < x_i \leq \Lambda_h - \Lambda_p + \dfrac{\Lambda_h^2}{\Lambda_h + \Lambda_p} \\ 0 & \text{if} \quad \Lambda_h - \Lambda_p + \dfrac{\Lambda_h^2}{\Lambda_h + \Lambda_p} < x_i. \end{cases} \tag{118}$$

Thus, if $i \in \Phi$, then for all $j \neq i$, either $x_i \leq x_j$ or $x_j \leq \Lambda_h - \Lambda_p$ so that

$$\sum_{y_i} r(y_i)p(y_i \mid x_i) - r(x_i) \geq \sum_{y_j} r(y_j)p(y_j \mid x_i) - r(x_i), \tag{119}$$

and the second condition of Corollary 1 holds. Finally, if

$$i \in \Phi(x_1,\ldots,x_i,\ldots,x_j,\ldots,x_n,t) \tag{120}$$

$x_j \neq x_i$ then $p(y_j \mid x_j) > 0$ implies one of the following. Either $x_i < x_j$ or $x_j \leq \Lambda_h - \Lambda_p$. In the first case, there exists $m \in \{0,1,2,\ldots\}$ so that $y_j - x_i = m\Lambda_h^2/(\Lambda_h + \Lambda_p) \geq 0$. In the second case, $y_j \leq x_j \leq \Lambda_h - \Lambda_p$. In either case,

$$i \in \Phi(x_1,\ldots,x_i,\ldots,y_j,\ldots,x_n,t+1). \tag{121}$$

and the third and final condition of Corollary 1 holds, and the strategy set is optimal.

REFERENCES

[1] D. A. Berry and B. Fristedt
Bandit Problems: Sequential Allocation of Experiments.
Chapman and Hall, 1985.

[2] D. P. Bertsekas
Dynamic Programming and Optimal Control.
Athena Scientific, Belmont, MA, 2001.

[3] D. A. Castanon
Optimal search strategies in dynamic hypothesis testing.
IEEE Transactions on Systems, Man, and Cybernetics, **25**, 7 (1995), 1130–1138.

[4] J. C. Gittins
Bandit processes and dynamic allocation indices.
Journal of the Royal Statistical Society: Series B (Methodological), **41**, 2 (1979), 148–177.

[5] S. Howard, S. Suvorova and B. Moran
Optimal policy for scheduling of Gauss-Markov systems.
In *Proceedings of the International Conference on Information Fusion*, 2004.

[6] O. P. Kreidl and T. M. Frazier
Feedback control applied to survivability: A host-based autonomic defense system.
IEEE Transactions on Reliability, **53**, 1 (2004), 148–166.

[7] V. Krishnamurthy
Algorithms for optimal scheduling and management of hidden Markov model sensors.
IEEE Transactions on Signal Processing, **50**, 6 (2002), 1382–1397.

[8] V. Krishnamurthy and R. J. Evans
Hidden Markov model multiarm bandits: A methodology for beam scheduling in multitarget tracking.
IEEE Transactions on Signal Processing, **49**, 12 (2001), 2893–2908.

[9] S. Musick and R. Malhotra
Chasing the elusive sensor manager.
In *Proceedings of the IEEE National Aerospace and Electronics Conference*, vol. 1, 1994, 606–613.

[10] R. Popoli
The sensor management imperative.
In Yaakov Bar-Shalom (Ed.), *Multitarget-Multisensor Tracking: Applications and Advances*, vol. 2, Artech House, 1992, 325–392.

[11] M. K. Schneider, G. L. Mealy and F. M. Pait
Closing the loop in sensor fusion systems: Stochastic dynamic programming approaches.
In *Proceedings of the American Control Conference*, 2004.

[12] M. K. Schneider, A. Nedich, D. A. Castanon and R. B. Washburn
Approximation methods for Markov decision problems in sensor management.
BAE Systems, Burlington, MA, Technical Report TR-1620, 2005.

[13] R. B. Washburn, M. K. Schneider and J. J. Fox
Stochastic dynamic programming based approaches to sensor resource management.
In *Proceedings of 5th Iternational Conference on Information Fusion*, 2002, 608–615.

[14] R. R. Weber and G. Weiss
On an index policy for restless bandits.
Journal of Applied Probability, **27** (1990), 637–648.

[15] P. Whittle
Restless bandits: Activity allocation in a changing world.
Journal of Applied Probability, **25A** (1988), 287–298.

Robert B. Washburn received his B.S. in 1973 from Yale University in mathematics and his Ph.D. in 1979 from the Massachusetts Institute of Technology in applied mathematics.

He has been a principal scientist at Parietal-Systems, Inc. (PSI) since January 2006. At PSI he is developing algorithms for constraint-based data association, sensor resource management, and pattern recognition for information fusion applications. Before joining PSI, Dr. Washburn worked at ALPHATECH for 23 years, where he was a Chief Scientist and during which time he led research and development in several areas of information fusion and sensor management. His work included the development of multiple hypothesis tracking (MHT) and correlation algorithms for several applications, and development and application of multiresolution statistical image fusion algorithms and performance estimation theory for model-based target recognition algorithms. He led the efforts on sensor resource management, using approximate dynamic programming approaches to develop efficient control algorithms for pointing sensors and selecting sensor modes for sensing multiple targets. Dr. Washburn is a member of the IEEE, American Mathematical Society, and Society for Industrial and Applied Mathematics.

Michael K. Schneider received his B.S.E. degree in electrical engineering and Certificate in applied and computational mathematics (Summa Cum Laude) from Princeton University in 1994, and his M.S. and Ph.D. degrees in electrical engineering and computer science from the Massachusetts Institute of Technology in 1996 and 2001, respectively.

He is a lead research engineer at BAE Systems Advanced Information Technologies (BAE AIT) where, since 2001, he has been working on problems in information fusion. At BAE AIT, he has been developing combinatorial optimization algorithms for sensor resource management as well as algorithms for signature-aided tracking and analysis of patterns in historical track data. He is a member of IEEE, SIAM, and INFORMS.

Fusion of Tracks with Road Constraints

CHUN YANG
ERIK BLASCH

This paper is concerned with tracking of ground targets on roads and investigates possible ways to improve target state estimation via fusing a target's track with information about the road along which the target is traveling. A target track is estimated using a surveillance radar whereas a digital map provides the road network of the region under surveillance. When the information about roads is as accurate as (or even better than) radar measurements, it is desired naturally to incorporate such information (fusion) into target state estimation. In this paper, roads are modeled with analytic functions and their fusion with a target track is cast as linear or nonlinear state constraints in an optimization procedure. The constrained optimization is then solved with the Lagrangian multiplier, leading to a closed-form solution for linear constraints and an iterative solution for second-order nonlinear constraints. Geometric interpretations of the solutions are provided for special cases. Compared to other methods, the track-to-road fusion using the constrained optimization technique can be easily implemented as an add-on module without changes to an existing tracker. For curved roads with coarse waypoints, the nonlinear constrained solution outperforms the piecewise linearized constrained approach. Computer simulation results are presented to illustrate the algorithms.

Manuscript received October 4, 2006; revised August 3, 2007 and June 12, 2008; released for publication on June 25, 2008.

Refereeing of this contribution was handled by Benjamin J. Slocumb.

Authors' addresses: C. Yang, Sigtem Technology, Inc., 1343 Parrott Drive, San Mateo, CA 94402, E-mail: (chunyang@sigtem.com); E. Blasch, Air Force Research Lab/RYAA, 2241 Avionics Circle, WPAFB, OH 45433, E-mail: (erik.blasch@wpafb.af.mil).

1. INTRODUCTION

With the rapid building up of geographic information system (GIS) including digital road maps (DRM) and digital terrain elevation data (DTED), information about roads becomes more accurate, up to date, and accessible. Looking for a map in the Internet is at fingertips with a least cost (i.e., distance or time) route plotted to a destination. Road and terrain information has been used in the past for navigation via terrain contour matching. Other examples include the increasingly popular use of digital maps for automobile navigation with a Global Positioning System (GPS) receiver and terrain-aided navigation for aircraft.

This paper is concerned with tracking of ground targets on roads and investigates possible ways to improve target state estimation via fusing a target's track with information about the road along which the target is traveling. A target track is estimated using a surveillance radar whereas a digital map provides the road network of the region under surveillance. Target tracking is not unfamiliar with road maps. For example, target tracks are represented by colorful dots and lines blinking along road networks on a big screen, often on top of a topographic or satellite image, in a situation room, in an air traffic control tower, and on a radar operator screen. In these applications, however, target tracks and road networks are merely displayed together with little or no interaction in the data processing level.

When the information about roads is as accurate as (or even better than) radar measurements, it is naturally desired to incorporate such information into target state estimation. When a vehicle travels off-road or on an unknown road, the state estimation problem is unconstrained. However, when the vehicle is traveling on a known road, be it straight or curved, the state estimation problem can be cast as constrained with the road network information available from digital road/terrain maps. In the past, such constraints are often ignored (or left for the users to perceive it as in the display example mentioned above). The resulting estimates, even obtained with the Kalman filter, cannot be optimal because they do not make full use of this additional information about state constraints.

To use such state constraints, previous attempts can be put into several groups. The first group is to incorporate road information into the state estimation process. One technique is to reduce the system model parameterization. Another technique is to translate the state constraints onto the state process and/or observation noise covariance matrix for the estimation filter [10]. The use of variable structure IMM (VS-IMM) methods also belongs to this group [7, 18, 19, 22]. Yet another technique is to project a dynamic system onto linear state constraints and then apply the Kalman filter to the projected systems [11]. Similarly, for nonlinear state constraints, there is the one-dimensional (1D) representation of a target motion along a curvilinear road [27].

The technique to model bounded random variables with truncated densities also belongs to this group, which is easily workable with such nonlinear filters as a particle filter [2, 4, 25, 26]. Road map information can also be integrated within Multi-Hypothesis Tracking (MHT) [12, 13].

The second group is to treat state constraints as pseudo measurements [8]. For a road segment, its analytic model not only constrains the target position but also the direction of the target's velocity. Indeed, the target velocity is closely aligned with the road orientation for a linear segment and with the tangent vector at the target position for a nonlinear segment. Furthermore, an estimate of centripetal acceleration can be obtained given the curvature and the target speed [27].

In the third group, an unconstrained Kalman filter solution is first obtained and then the unconstrained state estimate is projected onto the constrained surface [24]. This technique can also be viewed as post-processing (estimation or updating) correction [28] or track-to-road fusion as referred to in this paper. In conventional track fusion, two or more tracks are available, each consisting of an estimate of the underlying target trajectory with its estimation error covariance. The fused track is typically found that minimizes the sum of covariance-weighted state errors squared [5, 6]. In contrast to this conventional track fusion that operates on individual states, fusion of tracks with roads involves a state value (a point) and a subset of state values (an arc or interval). In this paper, roads are modeled with analytic functions and their fusion with a target track is therefore formulated as linear or nonlinear state constraints in an optimization procedure.

Although this paper presents a new technique for the third group, it is interesting to think of it relative to the first group in much the same way track fusion is compared with measurement fusion. In measurement fusion, measurements from all sensors are made available to a centralized tracker, which has the potential to fuse out the best estimate. However, measurement fusion may not be practical for distributed sensors wherein gathering all raw measurements is often limited by network transmission bandwidth and latency. Track fusion is frequently used as acceptable compromise between performance and cost.

Similarly, fusion of tracks with road constraints (in the third group) may not perform as well as an algorithm that incorporates road maps directly into the filtering process (in the first group). However, it has many merits of its own. First, it is simple and can be retrofitted into existing trackers as an add-on module without changes to the trackers. Since the tracks are obtained without constraints, it can easily switch between off-road and on-road operations when road information is available and the unconstrained tracks are deemed close to roads. Second, an up-to-date accurate road map may not be available to individual sensors but only at a fusion center. In this case, the algorithm of track-to-road fusion as presented in this paper can be applied directly whereas those in the first group cannot. Third, as noted in [18], the IMM methods based on road maps do not always perform better than those without road maps particularly when the updating interval is long. In contrast, constraining an on-road target's track onto a road (fusing) has no such a problem. Fourth, the track-to-road fusion algorithm goes beyond target tracking to navigation for instance where it can be used to loosely integrate GPS fixes and digital maps [16].

In this paper, we therefore focus on the third group and in particular present an optimization procedure for nonlinear state constraints which is shown to be superior to the linear approximation of nonlinear state constraints as suggested in [24].

There are a host of constrained nonlinear optimization techniques [15]. Primal methods search through the feasible region determined by the constraints. Penalty and barrier methods approximate constrained optimization problems by unconstrained problems through modifying the objective function (e.g., add a term for higher price if a constraint is violated). Instead of the original constrained problem, dual methods attempt to solve an alternate problem (the dual problem) whose unknowns are the Lagrangian multipliers of the first problem. Cutting plane algorithms work on a series of ever-improving approximating linear programs whose solutions converge to that of the original problem. Lagrangian relaxation methods are widely used in discrete constrained optimization problems.

In addition, moving horizon estimation reformulates the estimation problem as quadratic programming over a moving, fixed-size estimation window and has become an important approach to constrained nonlinear estimation [20]. Another approach to constrained linear estimation is to exploit the Lagrangian duality. Indeed, a constrained linear estimation problem is shown to be a particular nonlinear optimal control problem in [9]. Constrained state estimation has also been studied from a game-theoretical point of view (also called the minimax or H_∞ estimation) in [23].

In this paper, the constrained optimization is solved with the Lagrangian multiplier, leading to a closed-form solution for linear constraints and an iterative solution for nonlinear constraints. In the latter case, we present a method that allows for the use of second-order nonlinear state constraints exactly. The method can provide better approximation to higher order nonlinearities. The new method is based on a computational algorithm that iteratively finds the Lagrangian multiplier. The use of a second-order constraint versus linearization is a tradeoff between reducing approximation errors to higher-order nonlinearities and keeping the problem computationally tractable.

A nonlinear constraint can be approximated with linear constraints in a piecewise fashion. By judicious selection of the number of linear segments and their placement (i.e., the point around which to linearize), a reasonably good performance can be expected. In the limit, a nonlinear function is represented by a piecewise function composed of an infinite number of linear segments. This naturally leads to the use of nonlinear constraints. As such, the proposed nonlinear constrained solution for curved roads is not only more accurate but also less complicated in implementation than a piecewise linearized constrained approach, to be shown later in simulation examples.

Although the main results are restricted to state equality constraints, it can be extended to inequality constraints. According to [24], the inequality constraints can be checked at each time step of filtering. If the inequality constraints are satisfied at a given time step, no action is taken since the inequality constrained problem is solved. If the inequality constraints are not satisfied at a given time step, then the constrained solution is applied to enforce the constraints.

The paper is organized as follows. Section 2 presents linearly constrained state estimation for fusion of tracks with linear road segments. Section 3 presents an iterative solution for fusion of tracks with nonlinear road segments. In both cases, geometric interpretations of the solutions are provided for special cases. In Section 4, computer simulation results are presented to illustrate the algorithms. Finally, Section 5 provides concluding remarks and suggestions for future work.

2. FUSION OF TRACKS WITH LINEAR ROAD SEGMENTS

When a road segment is straight, it can be modeled as a linear state constraint. In this section, we first summarize the results for linearly constrained state estimation [24] as an approach to fusion of tracks with linear road segments. We then show that this linearly constrained state estimation is equivalent to use of constraints as measurements in state update. Finally, we provide a simple geometric interpretation of the linearly constrained state estimation for track-to-road fusion.

2.1. Linearly Constrained State Estimation for Track-to-Road Fusion

Consider a linear time-invariant discrete-time dynamic system together with its measurement as

$$\mathbf{x}_{k+1} = \mathbf{A}\mathbf{x}_k + \mathbf{B}\mathbf{u}_k + \mathbf{w}_k \tag{1a}$$

$$\mathbf{y}_k = \mathbf{C}\mathbf{x}_k + \mathbf{v}_k \tag{1b}$$

where the subscript k is the time index, $\mathbf{x}$ is the state vector, $\mathbf{u}$ is a known input, $\mathbf{y}$ is the measurement, and $\mathbf{w}$ and $\mathbf{v}$ are state and measurement noise processes, respectively. It is implied that all vectors and matrices have compatible dimensions, which are omitted for simplicity.

The goal is to find an estimate denoted by $\hat{\mathbf{x}}_k$ of $\mathbf{x}_k$ given the measurements up to time k denoted by $Y_k = \{\mathbf{y}_0,\ldots,\mathbf{y}_k\}$. Under the assumptions that the state and measurement noises are uncorrelated zero-mean white Gaussian with $\mathbf{w} \sim N\{0,\mathbf{Q}\}$ and $\mathbf{v} \sim N\{0,\mathbf{R}\}$ where $\mathbf{Q}$ and $\mathbf{R}$ are positive semi-definite covariance matrices, the Kalman filter provides an optimal estimator in the form of $\hat{\mathbf{x}}_k = E\{\mathbf{x}_k \mid Y_k\}$ [3]. Starting from an initial estimate $\hat{\mathbf{x}}_0 = E\{\mathbf{x}_0\}$ and its estimation error covariance matrix $\mathbf{P}_0 = E\{(\mathbf{x}_0 - \hat{\mathbf{x}}_0)(\mathbf{x}_0 - \hat{\mathbf{x}}_0)^T\}$ where the superscript ^{T} stands for matrix transpose, the Kalman filter equations specify the propagation of $\hat{\mathbf{x}}_k$ and $\mathbf{P}_k$ over time and the update of $\hat{\mathbf{x}}_k$ and $\mathbf{P}_k$ by measurement $\mathbf{y}_k$ as

$$\bar{\mathbf{x}}_{k+1} = \mathbf{A}\hat{\mathbf{x}}_k + \mathbf{B}\mathbf{u}_k \tag{2a}$$

$$\bar{\mathbf{P}}_{k+1} = \mathbf{A}\mathbf{P}_k\mathbf{A}^T + \mathbf{Q} \tag{2b}$$

$$\hat{\mathbf{x}}_{k+1} = \bar{\mathbf{x}}_{k+1} + \mathbf{K}_{k+1}(\mathbf{y}_{k+1} - \mathbf{C}\bar{\mathbf{x}}_{k+1}) \tag{2c}$$

$$\mathbf{P}_{k+1} = (\mathbf{I} - \mathbf{K}_{k+1}\mathbf{C})\bar{\mathbf{P}}_{k+1} \tag{2d}$$

$$\mathbf{K}_{k+1} = \bar{\mathbf{P}}_{k+1}\mathbf{C}^T(\mathbf{C}\mathbf{P}_k\mathbf{C}^T + \mathbf{R})^{-1} \tag{2e}$$

where $\bar{\mathbf{x}}_{k+1}$ and $\bar{\mathbf{P}}_{k+1}$ are the predicted state and prediction error covariance, respectively.

Now in addition to the dynamic system of (1), we are given the linear state constraint equation

$$\mathbf{D}\mathbf{x}_k = \mathbf{d} \tag{3a}$$

where $\mathbf{D}$ is a known constant matrix of full rank, $\mathbf{d}$ is a known vector, and the number of rows in $\mathbf{D}$ is the number of constraints, which is assumed to be less than the dimension of states. If $\mathbf{D}$ is a square matrix, the state is fully constrained and can thus be solved by inverting (3a). Although no time index is given to $\mathbf{D}$ and $\mathbf{d}$ in (3a), it is implied that they can be time-dependent, leading to piecewise linear constraints.

The information about a target traveling along a linear road segment is well modeled by (3a) and illustrated in Fig. 1. As shown, the road is specified by the orientation θ defined as the angle of its normal vector $\mathbf{n}$ relative to the x-axis and the distance to the origin r. The unit vectors pointing along the road and perpendicular to the road are given by $\boldsymbol{\mu} = [-\sin\theta,\cos\theta]^T$ and $\mathbf{n} = [\cos\theta,\sin\theta]^T$, respectively. Clearly, a target at position $\mathbf{p} = [x,y]^T$ with velocity $\mathbf{v} = [\dot{x},\dot{y}]^T$ satisfies the linear constraints $\mathbf{p}^T\mathbf{n} = r$ and $\mathbf{v}^T\mathbf{n} = 0$. These two equations can be easily put together into the format of (3a) with the corresponding $\mathbf{D}$ and $\mathbf{d}$ given below.

$$\mathbf{D} = \begin{bmatrix} \cos\theta & 0 & \sin\theta & 0 \\ 0 & \cos\theta & 0 & \sin\theta \end{bmatrix}, \qquad \mathbf{d} = \begin{bmatrix} r \\ 0 \end{bmatrix} \tag{3b}$$

The constrained Kalman filter according to [24] is constructed by directly projecting the unconstrained state estimate $\hat{\mathbf{x}}_k$ onto the constrained surface $\mathcal{S} = \{\mathbf{x} : \mathbf{D}\mathbf{x} = \mathbf{d}\}$. It is formulated as the solution to the problem

$$\breve{\mathbf{x}} = \arg\min_{\mathbf{x}\in\mathcal{S}} (\mathbf{x} - \hat{\mathbf{x}})^T\mathbf{W}(\mathbf{x} - \hat{\mathbf{x}}) \tag{4}$$

where $\mathbf{W}$ is a symmetric positive definite weighting matrix. The time index subscript k is dropped from

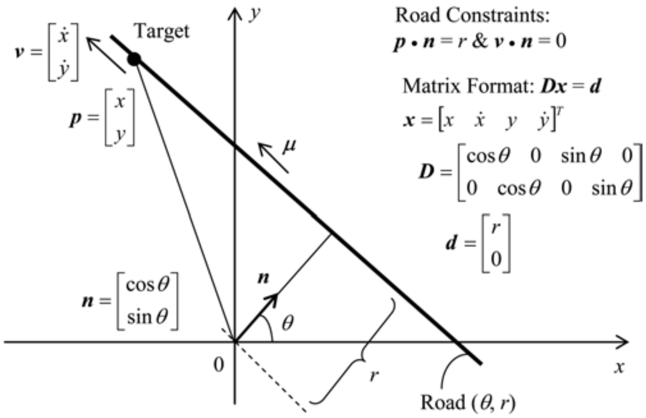

Fig. 1. Road models as linear constraints.

variables in (4) for simplicity. When $\mathbf{W} = \mathbf{I}$, the cost function of (4) is the standard least squares formulation. If $\mathbf{W}$ is chosen based on the estimation error covariance matrix $\mathbf{P}$, it becomes the weighted least squares solution.

Derived using the Lagrangian multiplier technique in Appendix A, the solution to the constrained optimization in (4) is given by [24]

$$\breve{\mathbf{x}} = \hat{\mathbf{x}} - \mathbf{W}^{-1}\mathbf{D}^T(\mathbf{D}\mathbf{W}^{-1}\mathbf{D}^T)^{-1}(\mathbf{D}\hat{\mathbf{x}} - \mathbf{d}). \tag{5}$$

Several interesting statistical properties of the constrained Kalman filter are presented in [24]. This includes the fact that the constrained state estimate as given by (5) is an unbiased state estimate for the system in (1) subject to the constraint in (3a) for a known symmetric positive definite weighting matrix $\mathbf{W}$. Furthermore when $\mathbf{W} = \mathbf{P}^{-1}$, the constrained state estimate has a smaller error covariance than that of the unconstrained state estimate, and it is actually the smallest for all constrained Kalman filters of this type.

2.2. Track-to-Road Fusion Architectures[1]

According to (4), the fusion of a target track (an unconstrained state estimate) $\hat{\mathbf{x}}_k$ with a road segment represented by a surface in the state space $\mathcal{S}$ is cast as a constrained least squares optimization problem, yielding the constrained solution $\breve{\mathbf{x}}$ and its estimation error covariance $P_{\breve{\mathbf{x}}}$. This leads to two possible implementation schemes. One is the open-loop architecture without feedback as shown in Fig. 2(a) and the other is the closed-loop architecture with feedback as shown in Fig. 2(b).

In the open-loop architecture of Fig. 2(a), the unconstrained solution can be used to help select the proper road constraints prior to fusion and the fused solution may be further used to refine road constraints for future target movements. However, the fused state is not fed back to the unconstrained tracker.

In contrast, the closed-loop architecture of Fig. 2(b) feeds back the fused state to the unconstrained Kalman tracker (i.e., to replace the state with the fused state). This has the advantage of keeping the one-step-ahead prediction closely aligned to the road estimates.

There are several issues to trade off when making the choice of one architecture versus the other. The closed-loop scheme needs to alter the unconstrained filter and its implementation therefore requires internal access. Further, a two-way data link may be necessary if the tracker and the fusion center are not co-located.

The open-loop architecture is simple and can be retrofitted into existing trackers as an add-on module without changes to the trackers. Since the tracks are obtained without constraints, it can easily switch between off-road and on-road operations. It is particularly useful for cases where no up-to-date accurate road map (e.g., the latest satellite imagery) is available to individual sensors but at a fusion center. In this paper, the open-loop scheme is implemented in the simulation examples presented in Section 4.

As shown in Fig. 2, an important step leading to the track-to-road fusion is the road constraint generation. It consists of two major parts, namely, creating an analytic

[1]This subsection is added based on the editor's comments.

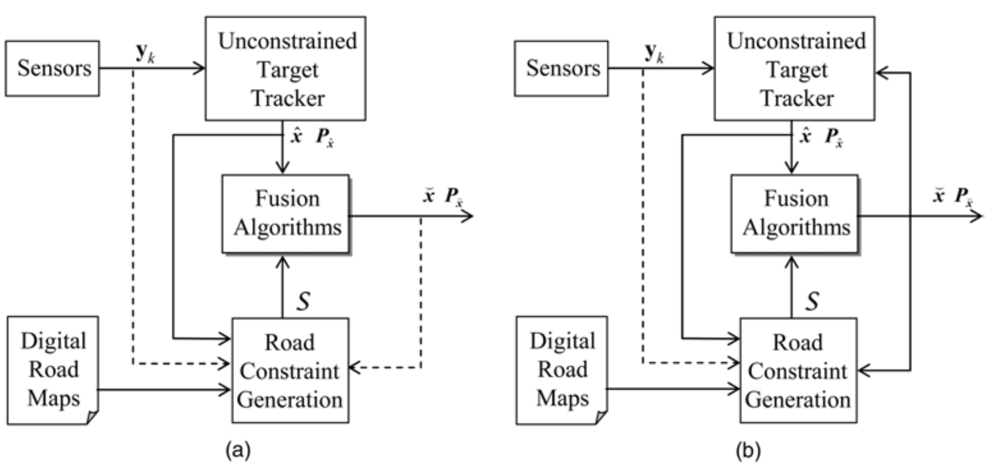

Fig. 2. Track-to-road fusion architectures.

representation for a given road segment and selecting the correct road segment(s) for fusion.

In a digital geographic database, road network is stored as a series of waypoints, a layer of the vector map (VMAP). The waypoints are typically extracted from survey data and aerial imagery among others and the density or spacing of waypoints is determined by the map resolution, which may be nonuniform. Although local survey data may contain the radius and turn center of a curved road segment, the waypoints themselves do not define the functions representing the road. To apply the road-constrained optimal fusion method, it is necessary to generate the constraint function based on the waypoints in the database. The most typical approach would use linear segments to connect the waypoints as evident from Google Map, MapQuest, or MSN Maps when zooming in. The waypoint connections defines a line representing road, which can be a simple line connecting two waypoints or a tangent passing through a waypoint. Alternately, a spline (a piecewise polynomial function) can be used to define the road in between the points, leading to a nonlinear function defining the road.

Ideally, the number of waypoints used to define the road is generated such that the maximum error between the actual road and the mathematical model for the road (linear segments, spline, etc.) is less than some allowable value. However, waypoints in most digital maps are pre-determined and fixed. Depending on the map resolution and sensor accuracy, when the error associated with the constraints becomes larger than the error in the sensor measurement, the benefits of using such constraints diminish. It is therefore desired to have a road modeling system that generates the waypoints to support "adaptive sampling" so that the error between the road and the road model is always less than some limit. The use of an analytic nonlinear representation, rather than fixed waypoints with linear segments, is a possible way toward adaptive sampling and resampling.

The second aspect of the road constraint generation shown in Fig. 2 is constraint selection, which identifies which road the target is on and the closest waypoints on the road and then produces the constraint function for those points of the road. Similar to the problem of target tracking with measurement origin uncertainties where data association is applied prior to measurement updating, the track-to-road fusion necessitates road constrained data association (RCDA) especially with closely spaced roads and around intersections. This association can be either measurement-based or predicted state-based and a data history may be needed to ascertain the winning hypothesis.

For an identified road, it then comes to select a piecewise constraint model. Without pre-determined analytic models available for the road segment, it is possible to perform on-line synthesis. For example, from two closest waypoints to a measurement, a line representation can be computed for those points. Or a spline representation of the road can be computed for the nearest three points of the digital map. A nonlinear representation (from the spline for instance) further allows for piecewise linearization with the point for the linearization chosen near the measurement or near the estimated track state. Iterative linearization can be used to refine the linearized constraints if necessary.

For lines between fixed waypoints or piecewise linearized segments from a nonlinear model, the linear constrained optimization method of this section can be applied. For curved roads, the nonlinear optimization method presented in Section 3 can be used advantageously when a nonlinear representation of a road is available.

The aspects of constraint modeling and selection are not further discussed in this paper. Another important issue that is not addressed either in this paper is possible errors in digital maps such as bias and misorientation for linear road segments and erroneous radius and turn center for curved segments. We leave it for future treatment but focus on fusion methodology in this paper.

2.3. Linear Road Constraint as Pseudo Measurement

As described above, the linear constrained estimator (5) can be obtained by different methods. It is shown in this section that it is also equivalent to the solution where the linear state constraints are considered as pseudo measurements.

For the linear time-invariant discrete-time dynamic system (1a) and its measurement (1b), consider the linear state constraint (3) as an additional measurement to the system, which can be used to perform the filter measurement update (2c) and (2d) right after (1b) without the filter time propagation (2a) and (2b) (i.e., stay the same). To apply (2), we identify the following equivalence:

$$\mathbf{C} = \mathbf{D}, \qquad \mathbf{R} = \mathbf{0}, \qquad \mathbf{y}_k = \mathbf{d}. \tag{6}$$

Consider $(\hat{\mathbf{x}}_k^{(i)}, \mathbf{P}_k^{(i)})$ as the constrained state and covariance after the ith iteration update with the constraints at time k. With this notation, $(\hat{\mathbf{x}}_k^{(i)}, \mathbf{P}_k^{(i)}) = (\hat{\mathbf{x}}_k, \mathbf{P}_k)$ estimate for $i = 0$ is the unconstrained state estimate and covariance at time k. The Kalman filter gain is given by

$$\mathbf{K}_k^{(i+1)} = \mathbf{P}_k^{(i)}\mathbf{D}^T(\mathbf{D}\mathbf{P}_k^{(i)}\mathbf{D}^T)^{-1}. \tag{7}$$

The updated state and error covariance becomes:

$$\hat{\mathbf{x}}_k^{(i+1)} = \hat{\mathbf{x}}_k^{(i)} + \mathbf{P}_k^{(i)}\mathbf{D}^T(\mathbf{D}\mathbf{P}_k^{(i)}\mathbf{D}^T)^{-1}(\mathbf{d} - \mathbf{D}\hat{\mathbf{x}}_k^{(i)}) \tag{8}$$

$$\mathbf{P}_k^{(i+1)} = \mathbf{P}_k^{(i)} - \mathbf{P}_k^{(i)}\mathbf{D}^T(\mathbf{D}\mathbf{P}_k^{(i)}\mathbf{D}^T)^{-1}\mathbf{D}\mathbf{P}_k^{(i)}. \tag{9}$$

If we choose $\mathbf{W} = (\mathbf{P}_k^{(i)})^{-1}$, (8) becomes

$$\hat{\mathbf{x}}_k^{(i+1)} = \hat{\mathbf{x}}_k^{(i)} + \mathbf{W}^{-1}\mathbf{D}^T(\mathbf{D}\mathbf{W}^{-1}\mathbf{D}^T)^{-1}(\mathbf{d} - \mathbf{D}\hat{\mathbf{x}}_k^{(i)}) \tag{10a}$$

$$= \hat{\mathbf{x}}_k^{(i)} - \mathbf{W}^{-1}\mathbf{D}^T(\mathbf{D}\mathbf{W}^{-1}\mathbf{D}^T)^{-1}(\mathbf{D}\hat{\mathbf{x}}_k^{(i)} - \mathbf{d}) \tag{10b}$$

$$= \hat{\mathbf{x}}_k - \mathbf{W}^{-1}\mathbf{D}^T(\mathbf{D}\mathbf{W}^{-1}\mathbf{D}^T)^{-1}(\mathbf{D}\hat{\mathbf{x}}_k - \mathbf{d}) \tag{10c}$$

which is exactly the same as the solution given by (5).

This equivalence affords a possible way to incorporate uncertainty in road modeling such as bias, width, and mis-orientation through pseudo measurement error covariance matrix $\mathbf{R}$. In the ideal case where roads are assumed to be known perfectly, this $\mathbf{R}$ is set to zero. Inequality constraint is another way to handle uncertainty if errors are within certain known bounds. Furthermore, when the track-to-road fusion is based on optimization with a least-squares criterion, it is possible to introduce weightings to account for directional errors given by covariance matrices of the track and/or the road.

2.4. Geometric Interpretation

Assume that the state dimension is n and the number of linear constraints is $m < n$. For $\mathbf{x} \in \mathbb{R}^n$, the constraint $\mathcal{S} = \{\mathbf{x} : \mathbf{D}\mathbf{x} = \mathbf{d}\}$ constitutes a surface in $\mathbb{R}^n$. It is shown in Appendix B that for the case where $\mathbf{W} = \mathbf{I}$, the linear constrained estimation (5) is the orthogonal projection of the unconstrained estimate onto the constraining surface. This offers a geometric interpretation and provides a theoretical justification of the intuitive practice of finding a point along the road that is of the shortest distance.

The theory still holds for $\mathbf{W} \neq \mathbf{I}$. The proof is given in Appendix C. The results presented in this and previous sections complement the work of [24], providing an interesting geometric interpretation to the linear constrained estimation by estimate projection.

3. FUSION OF TRACKS WITH NONLINEAR ROAD SEGMENTS

When a road segment is curved, it can be modeled as a nonlinear state constraint. In this section, we first analyze the linearizing approach and the associated constraint approximation error. We then present an iterative solution to a second order state constraint. Finally, we offer a geometric interpretation of the solution under a circular constraint and outline a simple approach to a more general second order state constraint problem of practical significance.

3.1. Approximation Errors in Constraint Linearization

To deal with nonlinearity, a simple approach is to project the unconstrained state estimate onto linearized state constraints. Once the constraints are linearized, the results presented in the previous section for linear cases can be applied. However, linearization introduces constraint approximation error, which is a function of the nonlinearity and, more importantly, of the point around which the linearization takes place. This may lead to an undesired divergence problem as analyzed below.

Consider the nonlinear state constraint of the form

$$\mathbf{g}(\mathbf{x}) = \mathbf{d}. \tag{11}$$

We can expand the nonlinear state constraints about a constrained state estimate $\check{\mathbf{x}}$ and for the ith row of (11), we have

$$g_i(\mathbf{x}) - d_i = g_i(\check{\mathbf{x}}) + \mathbf{g}_i'(\check{\mathbf{x}})^T(\mathbf{x} - \check{\mathbf{x}}) + \frac{1}{2!}(\mathbf{x} - \check{\mathbf{x}})^T\mathbf{g}_i''(\check{\mathbf{x}}) + (\mathbf{x} - \check{\mathbf{x}}) + \cdots - d_i = 0 \tag{12}$$

where the superscripts $'$ and $''$ denote the first and second partial derivatives.

Keeping only the first-order terms as suggested in [24], some rearrangement leads to

$$\mathbf{g}'(\check{\mathbf{x}})^T\mathbf{x} \approx \mathbf{d} - \mathbf{g}(\check{\mathbf{x}}) + \mathbf{g}'(\check{\mathbf{x}})^T\check{\mathbf{x}} \tag{13}$$

where $\mathbf{g}(\mathbf{x}) = [\ldots g_i(\mathbf{x})\ldots]^T$, $\mathbf{d} = [\ldots d_i \ldots]^T$, and $\mathbf{g}'(\mathbf{x}) = [\ldots \mathbf{g}_i'(\mathbf{x})\ldots]$. An approximate linear constraint is therefore formed by replacing $\mathbf{D}$ and $\mathbf{d}$ in (3) with $\mathbf{g}'(\mathbf{x})^T$ and $\mathbf{d} - \mathbf{g}(\check{\mathbf{x}}) + \mathbf{g}'(\check{\mathbf{x}})^T\check{\mathbf{x}}$, respectively.

Fig. 3 illustrates this linearization process and identifies possible errors associated with linear approximation of a nonlinear state constraint. As shown, a previous constrained state estimate $\check{\mathbf{x}}^-$ lies somewhere on the constrained surface but is away from the true state $\mathbf{x}$. The projection of the unconstrained state estimate $\hat{\mathbf{x}}$ onto the approximate linear state constraint produces the current constrained state estimate $\check{\mathbf{x}}^+$, which is however subject to the constraint approximation error. Clearly, the further away is $\check{\mathbf{x}}^-$ from $\mathbf{x}$, the larger is the approximation-introduced error. More critically, such an approximately linear constrained estimate may not satisfy the original nonlinear constraint specified in (11). It is therefore desired to reduce this approximation-introduced error by including higher-order terms while keeping the problem computationally tractable. One possible approach is presented in the next section.

As discussed in Section 2.2, when waypoints of a digital map are used to construct linear constraints directly, their modeling error is related to a large extent to the coarseness of waypoints, which is determined in turn by the map resolution.

Working with an analytic road model, simply curve-fitted from fixed waypoints for instance, provides the opportunity for possible "iterated linearization" or "adaptive sampling" so as to maintain small uniform linearization errors. As shown in Fig. 3, when the linearization point is far away from the true state, the lineariza-

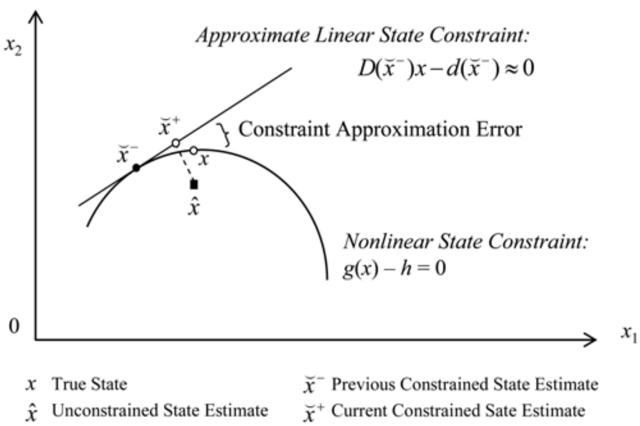

Fig. 3. Errors in linear approximation of nonlinear state constraints.

tion is poor. In this particular case, the linearization point is the predicted state $\breve{\mathbf{x}}^-$, which happens to be on road. However, due to target motion, this predicted state is offset from the true state $\mathbf{x}$. The linear constrained state estimate $\breve{\mathbf{x}}^+$ is now "closer" to the true state than the predicted one and can be used to re-linearize the function as done in an iterated extended Kalman filter. This iterated linearization may reduce linearization error in a sense but cannot guarantee a smaller state estimation error because the linear constrained state estimate and its iterations may not always fall onto the road.

At a first glance, a curved road can be well approximated with a sufficient number of waypoints where the linearization points are critically placed (i.e., the waypoint sample rate is sufficient to keep the error between the road and the road model small). In the limit, a piecewise linear approximation converges to a continuous function; and the direct use of a nonlinear constraint itself, rather than its approximation, becomes natural.

In practical cases, however, only a limited number of waypoints are available. For sharp turns, linear approximation errors dominate. As shown later in Section 4.1, the selection of a linear segment and in particular the transition from one segment to another is a rather involved process. On the other hand, the track-to-road fusion is considerably simplified with nonlinear constrained optimization as described below.

3.2. Iterative Solution to Second-Order Constraints

Naturally formed roads tend to have more bends and turns of irregular shapes (high nonlinearity). Even highways have to follow terrain contours when crossing mountains. Locally, however, it suffices to represent a curved road segment by a second-order state constraint function as

$$f(\mathbf{x}) = [\mathbf{x}^T \;\; 1]\begin{bmatrix} \mathbf{M} & \mathbf{m} \\ \mathbf{m}^T & m_0 \end{bmatrix}\begin{bmatrix} \mathbf{x} \\ 1 \end{bmatrix}$$

$$= \mathbf{x}^T\mathbf{M}\mathbf{x} + \mathbf{m}^T\mathbf{x} + \mathbf{x}^T\mathbf{m} + m_0 = 0 \qquad (14)$$

which can be viewed as a second-order approximation to an arbitrary nonlinearity in a digital terrain map.

Similar to (4), we can formulate the projection of an unconstrained state estimation onto a nonlinear constraint surface as the constrained least-squares optimization problem

$$\hat{\mathbf{x}} = \arg\min_{\mathbf{x}} (\mathbf{z} - \mathbf{H}\mathbf{x})^T(\mathbf{z} - \mathbf{H}\mathbf{x}) \qquad (15a)$$

$$\text{subject to} \quad f(\mathbf{x}) = 0. \qquad (15b)$$

If we let $\mathbf{W} = \mathbf{H}^T\mathbf{H}$ and $\mathbf{z} = \mathbf{H}\hat{\mathbf{x}}$, the formulation in (15) becomes the same as in (4). In a sense, (15) is a more general formulation because it can also be interpreted as a nonlinear constrained measurement update or a projection in the predicted measurement domain.

The solution to the constrained optimization (15) can be obtained again using the Lagrangian multiplier technique, which is detailed in Appendix D, as

$$\hat{\mathbf{x}} = \mathbf{G}^{-1}\mathbf{V}(\mathbf{I} + \lambda\mathbf{\Sigma}^T\mathbf{\Sigma})^{-1}\mathbf{e}(\lambda) \qquad (16a)$$

$$q(\lambda) = \sum_i \frac{e_i^2(\lambda)\sigma_i^2}{(1 + \lambda\sigma_i^2)^2} + 2\sum_i \frac{e_i(\lambda)t_j}{1 + \lambda\sigma_i^2} + m_0 = 0 \qquad (16b)$$

where $\mathbf{G}$ is an upper right diagonal matrix resulting from the Cholesky factorization of $\mathbf{W} = \mathbf{H}^T\mathbf{H}$ as

$$\mathbf{W} = \mathbf{G}^T\mathbf{G} \qquad (16c)$$

$\mathbf{V}$, an orthonormal matrix, and $\mathbf{\Sigma}$, a diagonal matrix with its diagonal elements denoted by σ_i, are obtained from the singular value decomposition (SVD) of the matrix $\mathbf{L}G^{-1}$ as

$$\mathbf{L}G^{-1} = \mathbf{U}\mathbf{\Sigma}\mathbf{V}^T \qquad (16d)$$

where $\mathbf{U}$ is the other orthonormal matrix of the SVD and $\mathbf{L}$ results from the factorization $\mathbf{M} = \mathbf{L}^T\mathbf{L}$, and

$$\mathbf{e}(\lambda) = [\ldots e_i(\lambda), \ldots]^T = \mathbf{V}^T(\mathbf{G}^T)^{-1}(\mathbf{H}^T\mathbf{z} - \lambda\mathbf{m}) \qquad (16e)$$

$$\mathbf{t} = [\ldots t_i \ldots]^T = \mathbf{V}^T(\mathbf{G}^T)^{-1}\mathbf{m}. \qquad (16f)$$

As a nonlinear equation in λ, it is difficult to find a closed-form solution in general for the nonlinear equation $q(\lambda) = 0$ in (16b). Numerical root-finding algorithms may be used instead. For example, the Newton's method is used below. Denote the derivative of $q(\lambda)$ with respect to λ as

$$\dot{q}(\lambda) = 2\sum_i \frac{e_i(\lambda)\dot{e}_i(1 + \lambda\sigma_i^2)\sigma_i^2 - e_i^2(\lambda)\sigma_i^4}{(1 + \lambda\sigma_i^2)^3} + 2\sum_i \frac{\dot{e}_i t_i(1 + \lambda\sigma_i^2) - e_i(\lambda)t_i\sigma_i^2}{(1 + \lambda\sigma_i^2)^2} \qquad (17a)$$

where

$$\dot{\mathbf{e}} = [\ldots \dot{e}_i \ldots]^T = -\mathbf{V}^T(\mathbf{G}^T)^{-1}\mathbf{m}. \qquad (17b)$$

Then the iterative solution for λ is given by

$$\lambda_{k+1} = \lambda_k - \frac{q(\lambda_k)}{\dot{q}(\lambda_k)} \qquad \text{starting with} \quad \lambda_0 = 0. \qquad (18)$$

The iteration stops when $|\lambda_{k+1} - \lambda_k| < \tau$, a given small value or the number of iterations reaches a prespecified number. Then bringing the converged Lagrangian multiplier λ back to (16a) provides the constrained optimal solution.

Now consider the special case where $\mathbf{W} = \mathbf{H}^T\mathbf{H}$, $\mathbf{z} = \mathbf{H}\hat{\mathbf{x}}$, and $\mathbf{m} = \mathbf{0}$, that is, a quadratic constraint on the state. Under these conditions, $\mathbf{t} = 0$ and $\mathbf{e}$ is no longer a function of λ so its derivative relative to λ vanishes, $\dot{\mathbf{e}} = 0$. The quadratic constrained solution is then given by

$$\breve{\mathbf{x}} = (\mathbf{W} + \lambda\mathbf{M})^{-1}\mathbf{W}\hat{\mathbf{x}} \tag{19a}$$

where the Lagrangian multiplier λ is obtained iteratively as in (18) with the corresponding $q(\lambda)$ and $\dot{q}(\lambda)$ given by

$$q(\lambda) = \sum_i \frac{e_i^2\sigma_i^2}{(1+\lambda\sigma_i^2)^2} + m_0 = 0 \tag{19b}$$

$$\dot{q}(\lambda) = -2\sum_i \frac{e_i^2\sigma_i^4}{(1+\lambda\sigma_i^2)^3}. \tag{19c}$$

The solution of (19) is also called the constrained least squares [17: pp 765–766], which was previously applied for the joint estimation of angles of arrival and calibration of channel biases of a linear array [29]. Similar techniques have been used for the design of filters for radar applications [1] and in robust minimum variance beamforming [14]. When $\mathbf{M} = 0$, the constraint in (14) degenerates to a linear one. The constrained solution is still valid. However, the iterative solution for finding λ is no longer applicable but a closed-form solution is available instead as given in (5).

3.3. Geometric Interpolation for Simple Cases

Consider a simple example where a target travels along a circle. For this case, in fact, a closed-form solution can be derived. Assume that $\mathbf{W} = \mathbf{I}$, $\mathbf{M} = \mathbf{I}$, $\mathbf{m} = \mathbf{0}$, and $m_0 = -r^2$. Let $\mathbf{p}$ be the position components of the state $\mathbf{x}$, to which the constraint is applied. The nonlinear constraint can be equivalently written as:

$$\mathbf{p}^T\mathbf{p} = r^2. \tag{20}$$

The quadratic constrained estimate given in (19a) becomes:

$$\breve{\mathbf{p}} = (\mathbf{W} + \lambda\mathbf{M})^{-1}\mathbf{W}\hat{\mathbf{p}} = (1+\lambda)^{-1}\hat{\mathbf{p}} \tag{21}$$

where λ is the Lagrangian multiplier.

Bringing (21) back to (20) gives:

$$\breve{\mathbf{p}}^T\breve{\mathbf{p}} = \left(\frac{\hat{\mathbf{p}}}{1+\lambda}\right)^T \frac{\hat{\mathbf{p}}}{1+\lambda} = r^2. \tag{22}$$

One solution for λ is:

$$\lambda = \frac{\sqrt{\hat{\mathbf{p}}^T\hat{\mathbf{p}}}}{r} - 1 = \frac{\|\hat{\mathbf{p}}\|_2}{r} - 1 \tag{23}$$

where $\|\cdot\|_2$ stands for the L_2-norm or length for the vector.

Bringing the solution for λ in (23) back to (21) gives:

$$\breve{\mathbf{p}} = r\frac{\hat{\mathbf{p}}}{\|\hat{\mathbf{p}}\|_2}. \tag{24}$$

This indicates that for this particular case with a circular constraint, the constraining results in normalization.

This further suggests a simple solution for some practical applications. When a target is traveling along a circular path (or approximately so), one can first find the equivalent center of the circle around which to establish a new coordinate system. Then express the unconstrained solution in the new coordinate and normalize it as the constrained solution. Finally convert it back to the original coordinates. For non-circular but second-order paths, eigenvalue-based scaling may be effected following coordinate translation and rotation in order to apply this circular normalization. Reverse operations are in order to transform back to the original coordinates. For applications of high dimensionality, the scalar iterative solution of (17) may be more efficient.

4. SIMULATION RESULTS

In this section, two simulation examples are presented in the context of on-road ground vehicle tracking. The first example compares linearized and nonlinear constraining schemes for a simple tracker and the second example compares unconstrained and constrained IMM trackers.

4.1. Linearized versus Nonlinear Constraints for a Simple Tracker

In this simulation example, a ground vehicle is assumed to travel along a circular road segment as shown in Fig. 3. The turn center is chosen as the origin of the x-y coordinates and the turn radius is $r = 100$ m. The target maintains a constant turn rate of 5.7296 deg/s with an equivalent linear speed of 10 m/s. The initial state is

$$\mathbf{x}_{k=0} = [x\ \ \dot{x}\ \ y\ \ \dot{y}]^T = [100\text{ m}\ \ 0\text{ m/s}\ \ 0\text{ m}\ \ 10\text{ m/s}]^T. \tag{25}$$

The vehicle is tracked by a radar sensor with a sampling interval of $T = 1$ s. The sensor provides position measurements of the vehicle as

$$\mathbf{y}_k = \begin{bmatrix} 1 & 0 & 0 & 0 \\ 0 & 0 & 1 & 0 \end{bmatrix}\mathbf{x}_k + \mathbf{v}_k \tag{26}$$

where the measurement error $\mathbf{v} \sim N(0,\mathbf{R})$ is a zero-mean Gaussian noise, independent in the x- and y-axis. The covariance matrix $\mathbf{R} = \mathrm{diag}([\sigma_x^2\ \ \sigma_y^2])$ uses the particular values of $\sigma_x = \sigma_y = 7$ m in the simulation. To use the position measurement model (26), it is assumed that the radar-produced measurements in a polar frame are converted to the Cartesian frame and the errors associated with the conversion are ignored.

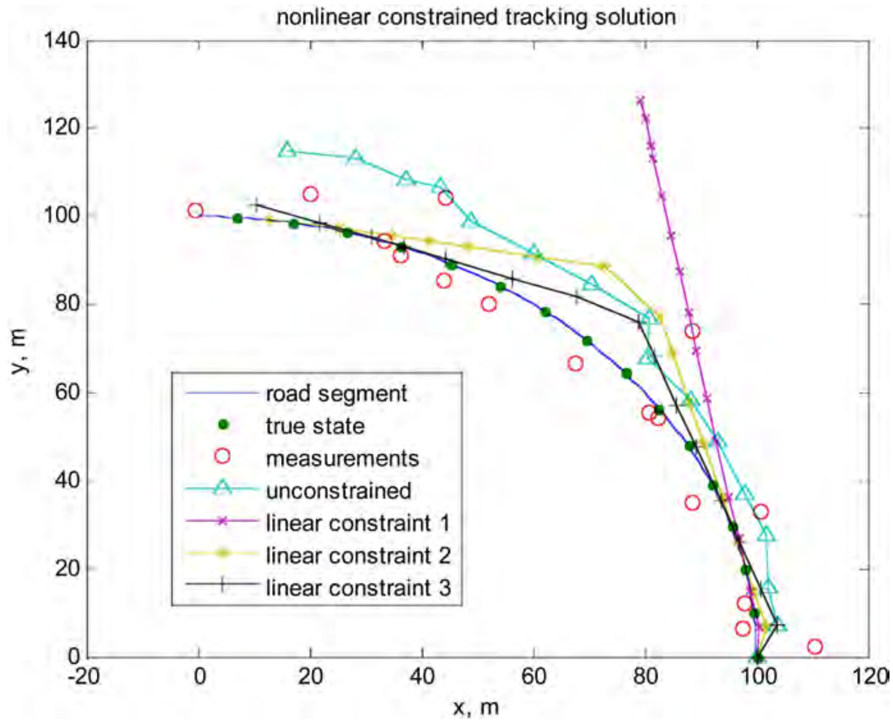

Fig. 4. Sample trajectories for linear constrained Kalman filter.

The radar implements a simple tracker based on the following discrete-time second-order kinematic model (nearly constant velocity)

$$\mathbf{x}_{k+1} = \begin{bmatrix} 1 & T & 0 & 0 \\ 0 & 1 & 0 & 0 \\ 0 & 0 & 1 & T \\ 0 & 0 & 0 & 1 \end{bmatrix} \mathbf{x}_k + \begin{bmatrix} \frac{1}{2}T^2 & 0 \\ T & 0 \\ 0 & \frac{1}{2}T^2 \\ 0 & T \end{bmatrix} \mathbf{w}_k \tag{27}$$

where the process noise $\mathbf{w} \sim N(0,\mathbf{Q})$ is also a zero-mean Gaussian noise, independent of the measurement noise $\mathbf{v}$. The covariance matrix $\mathbf{Q} = \mathrm{diag}([\sigma_{\ddot{x}}^2 \ \ \sigma_{\ddot{y}}^2])$ uses the particular values of $\sigma_{\ddot{x}} = \sigma_{\ddot{y}} = 0.32$ m/s^2 in the simulation.

When represented in a Cartesian coordinate system, a target traveling along a curved road is certainly subject to acceleration in both the x- and y-axis. However, no effect is made in this simulation to optimize the tracker for maneuver but merely to select $\mathbf{Q}$ and the initial conditions so as to focus on constraining the estimates. The use of an IMM filter [5] with "coordinated turn" models will be presented next in Section 4.2. The initial state is selected to be the same as the true state, i.e., $\hat{\mathbf{x}}_0 = \mathbf{x}_0$ for this example, again to focus on the aspect of track-to-road fusion, not on that of tracker design. The initial estimation error covariance is selected to be

$$\mathbf{P}_0 = \mathrm{diag}([5^2 \ \mathrm{m}^2 \ \ 1^2 \ (\mathrm{m/s})^2 \ \ 5^2 \ \mathrm{m}^2 \ \ 1^2 \ (\mathrm{m/s})^2]). \tag{28}$$

Fig. 4 shows sample trajectories of the linear constrained Kalman filter. There are 5 curves and 2 series of data points in the figure. The true state is represented by a series of dots ($\cdot$) at consecutive sampling instants, which is plotted on the solid line being the road segment. The corresponding measurements are a series of circles (o).

The estimates of the unconstrained Kalman filter are shown as the connected triangles ($\triangle$) whereas those of linearly constrained Kalman filters are shown as the connected crosses ($\times$), stars (*), and pluses (+) for three linear approximations of the nonlinear constraint of the curved road, respectively.

In the first approximation (the line with cross $\times$ labeled "linear constraint 1"), a single linearizing point at $\theta_1 = 10°$ is chosen to cover the entire curved road, where θ is the angle made relative to the x-axis, positive in the counter-clock direction. The linearized state constraint at θ_1 can be written as

$$\begin{bmatrix} \cos\theta_1 & 0 & \sin\theta_1 & 0 \\ 0 & \cos\theta_1 & 0 & \sin\theta_1 \end{bmatrix} \mathbf{x} = \begin{bmatrix} r \\ 0 \end{bmatrix}. \tag{29}$$

Although all estimates are faithfully projected by the constrained filter onto this linear constraint, tangential to the curve at the linearizing point, it runs away from the true trajectory and the resulting errors continue to grow. The apparent divergence is caused by the choice of linearization.

In the second approximation (the line with star * labeled "linear constraint 2"), two linearizing points at $\theta_1 = 15°$ and $\theta_2 = 80°$ are chosen to cover the curved road with two linear segments. The switching point from one linear segment to the other in this case is at $\theta = 45°$. As shown, the estimates are projected onto one of the two linear segments. Except near the corner

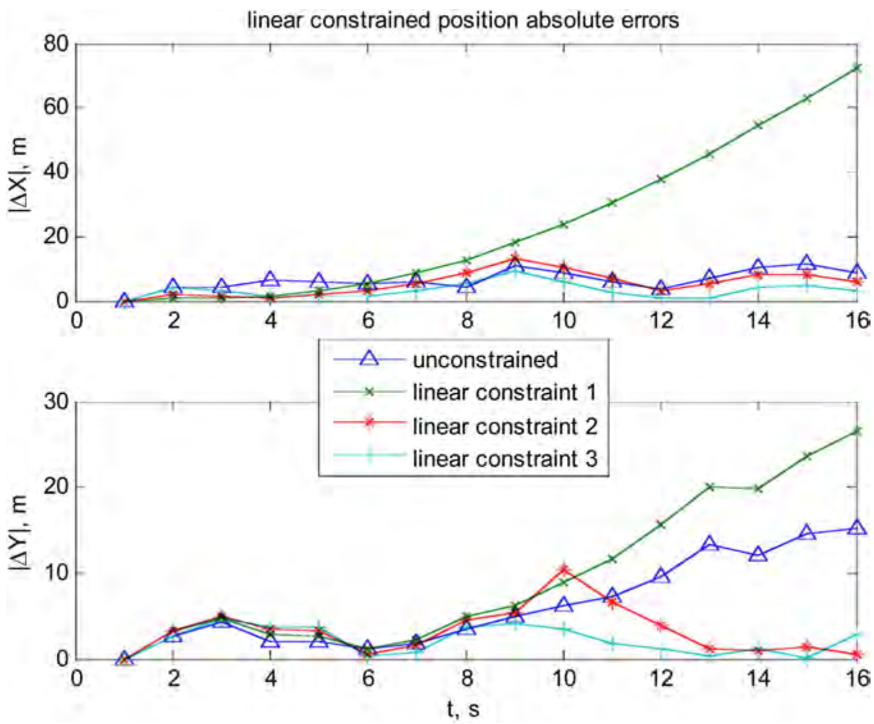

Fig. 5. Linear constrained position errors versus time.

where the two linear approximations intersect (which is far away from both linearizing points), the linear constrained estimates typically outperform the unconstrained estimates (closer to the true state). This is better illustrated in Fig. 5 where the upper plot is for the absolute position error in x while the lower plot is for the absolute position error in y, both plotted as a function of time.

Still with two linearizing points and the same switching point at $\theta = 45°$, the third approximation (the line with plus+labeled "linear constraint 3") adjusts linearizing points to $\theta_1 = 20°$ and $\theta_2 = 70°$. A better overall performance is achieved as shown in Fig. 5.

It is clear from Fig. 4 that a nonlinear constraint can be approximated with linear constraints in a piecewise fashion. By judicious selection of the number of linear segments and their placement (i.e., the point around which to linearize), a reasonably good performance can be expected. In the limit, a nonlinear function is represented by a piecewise function composed of an infinite number of linear segments. This naturally leads to the use of nonlinear constraints.

Fig. 6 shows sample trajectories of the nonlinear constrained Kalman filter. There are 2 curves and 4 series of data points in the figure. The true state is still represented by a series of dots ($\cdot$) at the sampling instants, which is plotted on the solid line of road segment. The corresponding measurements are again a series of circles (o). The unconstrained Kalman filter is shown as the connected crosses ($\times$) whereas the estimates of nonlinearly constrained Kalman filters are shown as a series of pluses (+) and stars (*) for two implementations, respectively.

The first implementation (the series of pluses +) only applies the nonlinear constraint to the position estimate whereas the second implementation (the series of stars *) applies constraints to both the position and velocity estimates. In fact, we encounter a hybrid (mixed) linear and nonlinear state constraint situation. The constrained position estimate is given by (19) for the quadratic case (equivalent to (24) for a circular road). Since the velocity direction is along the tangent of the road curve, the constrained velocity estimate is obtained by the following projection

$$\hat{\mathbf{v}}_{\text{constrained}} = (\hat{\mathbf{v}}^T_{\text{unconstrained}}\boldsymbol{\mu})\boldsymbol{\mu} \tag{30}$$

where $\hat{\mathbf{v}} = [\hat{\dot{x}}\ \ \hat{\dot{y}}]^T$ is the estimated velocity vector and $\boldsymbol{\mu} = [-\sin\theta\ \ \cos\theta]^T$ is the constrained unit direction vector associated with the constrained position at $\theta = \tan^{-1}(\hat{y}/\hat{x})$.

In the present simulation, the open-loop architecture without feedback is used. In this implementation, the unconstrained estimation error covariance is not modified after the constrained estimate is obtained using the projection algorithms (19). The implementation is therefore pessimistic (suboptimal) in the sense that it does not take into account the reduction in the estimation error covariance brought in by constraining. One consequence of this simplification is more volatile state estimates. To quantify this effect, one approach is to project the unconstrained probability density function (i.e., a normal distribution with support on the whole state space) onto the nonlinear constraint. Statistics can then be calculated from the constrained probability density function with the constraint as its support. Again,

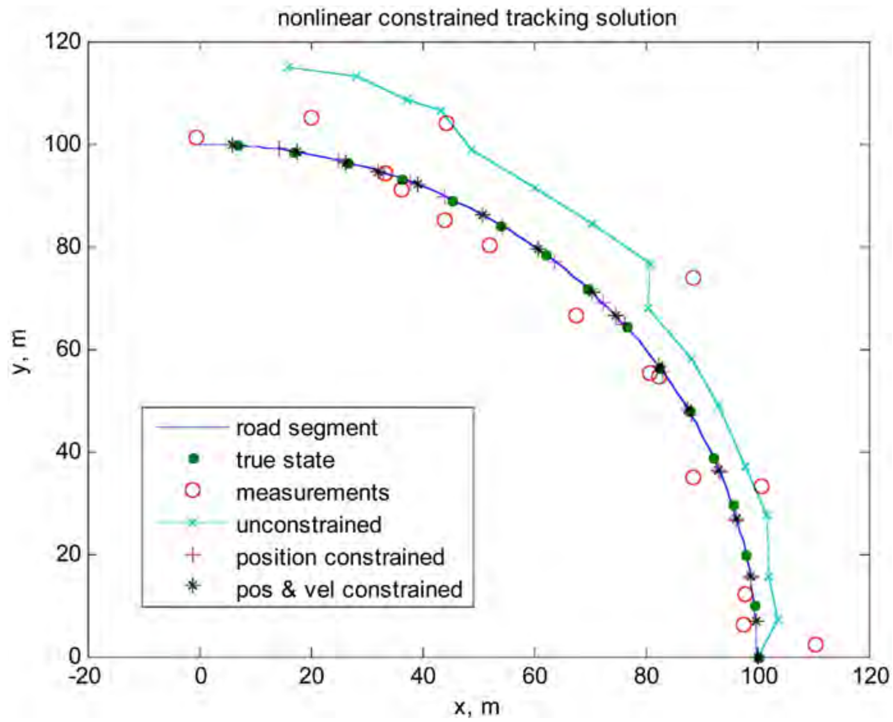

Fig. 6. Sample trajectories for nonlinear constrained Kalman filter.

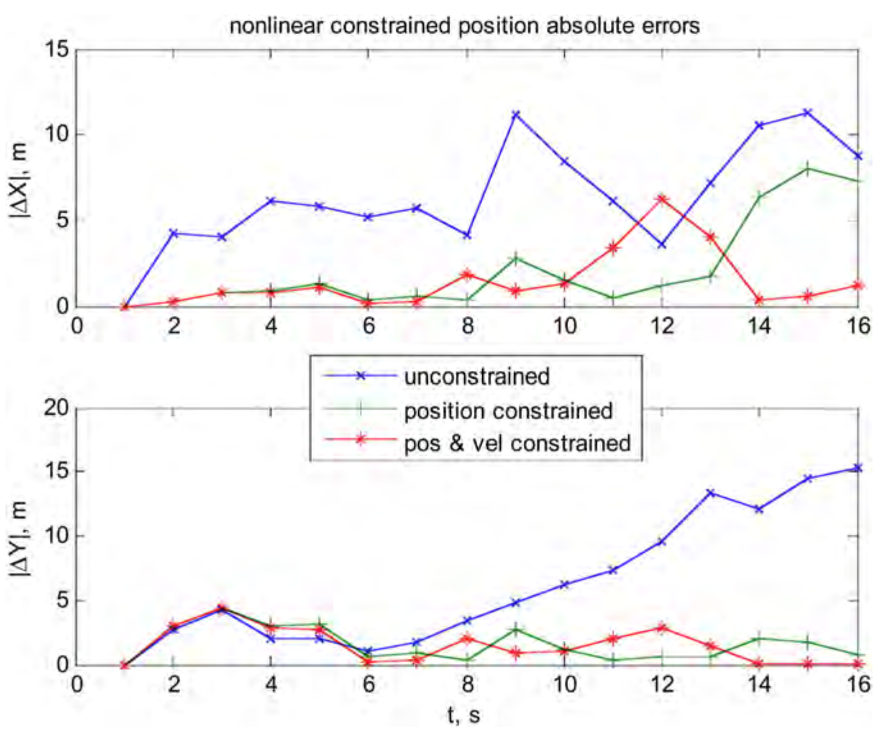

Fig. 7. Nonlinear constrained position errors versus time.

the resulting error ellipse represented by the covariance matrix is only an approximation to the second order. As explained in Section 2.2, the open-loop architecture without feedback has many merits of its own and it here provides a reference point for fusion architecture study.

As shown in Fig. 6, both the nonlinear constrained estimates fall onto the road as expected. Overall the position and velocity constrained estimates are better (closer to the true state) than the position-only constrained estimates. This is illustrated in Fig. 7 where the upper plot is for the absolute position error in x while the lower plot is for the absolute position error in y.

A Monte Carlo simulation is used to generate the root mean square (RMS) errors of state estimation. The results are based on a total of 100 runs across 16

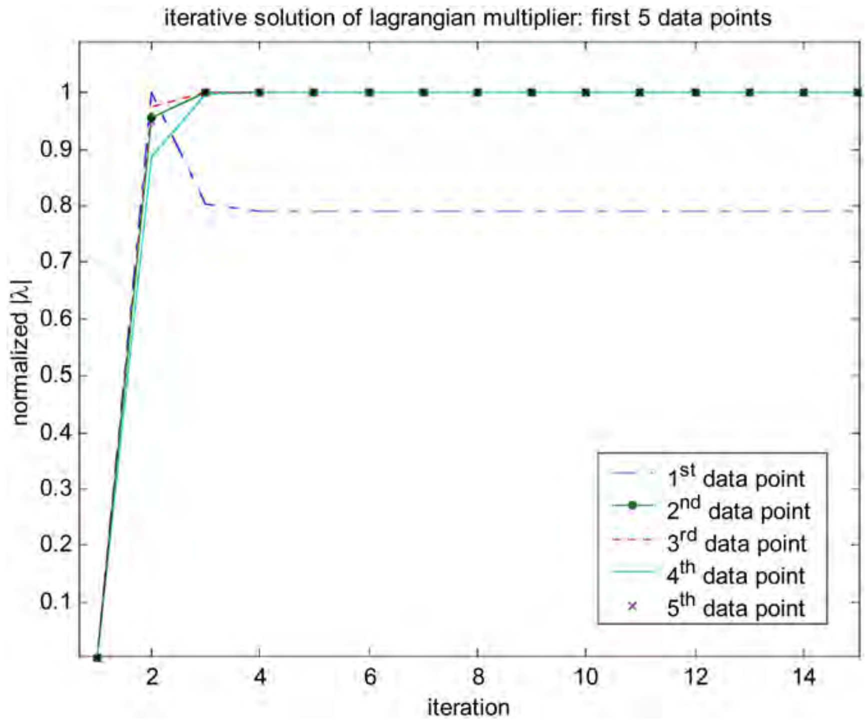

Fig. 8. Convergence in iterative Lagrangian multiplier.

TABLE I
RMS Estimation Errors

Estimators	RMS Estimation Error	
	Position (m)	Velocity (m/s)
Unconstrained	8.4	4.3
Best Linear Constrained	5.5	2.5
Nonlinear Constrained	1.8	0.4

updates and summarized in Table I. The performance improvement of the nonlinear constrained filter over the linearized constrained filter is demonstrated.

Finally for this simulation example, we use Fig. 8 to show an example of the Lagrangian multiplier as it is calculated iteratively using (19). The runs for five unconstrained state estimates are plotted in the same figure and to make it fit, the normalized absolute values of λ are taken. As shown, starting from zero, it typically takes 4 iterations for the algorithm to converge in the example presented.

4.2. Unconstrained Versus Constrained IMM Trackers

In the previous example, a nearly constant velocity model (27) was used in the filter. Obviously, a tracker that uses a maneuvering model can do better in tracking a turning target. However, it may still not be able to produce a track that falls on road all the time. The track-to-road fusion algorithm described in this paper can be applied in conjunction with a maneuvering target tracker to further improve target state estimation as illustrated in the following simulation example.

An IMM filter is constructed based on the "coordinated turn" models. For a ground vehicle, its wide turning maneuver is reasonably well modeled by a coordinated turn, i.e., at a constant turn rate with a constant speed. For the state vector $\mathbf{x}_k$ defined in (25), the coordinated turn model is given by

$$\mathbf{x}_{k+1} = \begin{bmatrix} 1 & \dfrac{\sin\omega T}{\omega} & 0 & -\dfrac{1-\cos\omega T}{\omega} \\ 0 & \cos\omega T & 0 & -\sin\omega T \\ 0 & \dfrac{1-\cos\omega T}{\omega} & 1 & \dfrac{\sin\omega T}{\omega} \\ 0 & \sin\omega T & 0 & \cos\omega T \end{bmatrix} \mathbf{x}_k + \begin{bmatrix} \frac{1}{2}T^2 & 0 \\ T & 0 \\ 0 & \frac{1}{2}T^2 \\ 0 & T \end{bmatrix} \mathbf{w}_k \tag{31}$$

where ω is the turn rate considered to be a known modeling parameter and $\mathbf{w}_k$ is defined as for (27).

For the IMM filter, three models are specified by choosing different values for ω. In the first model, setting $\omega = 0$ in (31) leads to the nearly constant velocity or non-maneuver model (27). In the second model, $\omega = 5.7$ deg/s represents a left turn maneuver while in the third model, $\omega = -5.7$ deg/s represents a right turn maneuver. The three models have an equal initial model probability of 1/3 and the model transition probability matrix is taken as

$$\boldsymbol{\Pi} = \begin{bmatrix} 0.8 & 0.1 & 0.1 \\ 0.1 & 0.8 & 0.1 \\ 0.1 & 0.1 & 0.8 \end{bmatrix}. \tag{32}$$

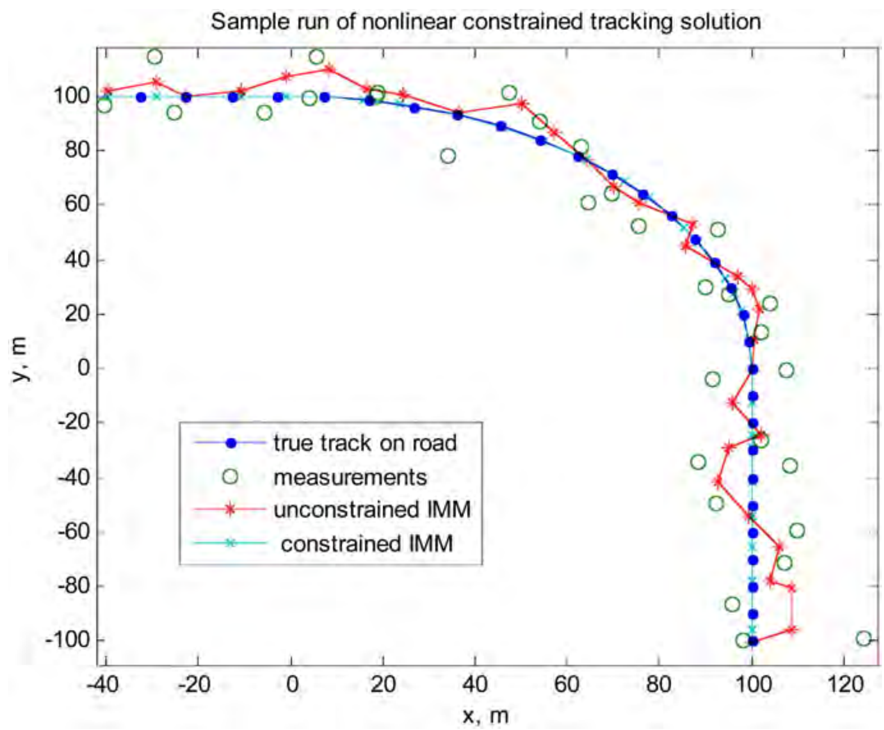

Fig. 9. Sample trajectories for unconstrained versus constrained IMM.

The three interacting filters inside the IMM tracker are identically initialized as $\hat{\mathbf{x}}_i \sim N(\mathbf{x}_0, \mathbf{P}_0)$ for $i = 1,2,3$. The same sensor model as (26) is used for generating radar measurements in this simulation. Hybrid constraints are applied with the nonlinear constraint (14) for position estimates and the linear constraint (30) for velocity estimates. In the Monte Carlo simulation, the truth track of the target remains the same but the initial estimate is drawn from the distribution for each run as described above. So is the measurement noise at each sampling instant for each run.

Fig. 9 shows sample trajectories wherein the target starts from an initial position at $(x_0, y_0) = (100$ m, -100 m) heading due north along a straight road with $\dot{y} = 10$ m/s. At $k = 10$ s, it follows the curve and makes a left turn at a rate of 5.7 deg/s for 16 s. At $k = 17$ s, it comes to another straight road heading due west with $\dot{x} = -10$ m/s for 5 s.

The true state is represented by a series of dots $(\cdot)$ plotted on the solid line of road segment. The corresponding measurements are a series of circles (o). The unconstrained IMM filter is shown as a series of connected stars (*) whereas the constrained IMM filter is shown as a series of connected crosses $(\times)$. When the target is on linear road segments, the linear constrained solution (5) is applied to the combined state of the IMM filter while on the curved road segment, a hybrid constrained solution is used (nonlinear for position and linear for velocity). From Fig. 9, the typical behavior of an unconstrained IMM filter can be seen. It converges rather quickly from the initialization of large errors, develops an overshoot right after the maneuver but corrects itself towards the true trajectory, and converges again after the maneuver terminated. However, these unconstrained IMM estimates (*) are off road while the target is on road.

In contrast, the constrained IMM estimates $(\times)$ are always on road even though they do not fall exactly on top of the true positions $(\cdot)$. As a result, the constrained position errors are smaller than the unconstrained ones as shown in Fig. 10, which are obtained by a Monte Carlo simulation with 100 runs. In particular, the velocity errors of the unconstrained IMM solution grow during the maneuver period whereas those of the constrained solution appear to be uniform.

The RMS errors averaged over the entire trajectory are summarized in Table II. The values in Table II are bigger than those in Table I because of larger initialization errors and longer simulation run. It shows an improvement of approximately 3 folds in position and in velocity.

5. CONCLUSIONS

In this paper, we presented an approach to incorporating road information into target tracking via track-to-road fusion. In this approach, road segments were modeled with analytic functions and their fusion with a target track was cast as a linearly or nonlinearly state constrained optimization procedure. With the Lagrangian multiplier, a closed-form solution was found for linear constraints and an iterative solution for nonlinear constraints. Geometric interpretations of the solutions were provided for simple cases. Computer simulation results demonstrate the performance of the algorithms.

Future work includes both algorithms development and practical applications. It is of interest to extend

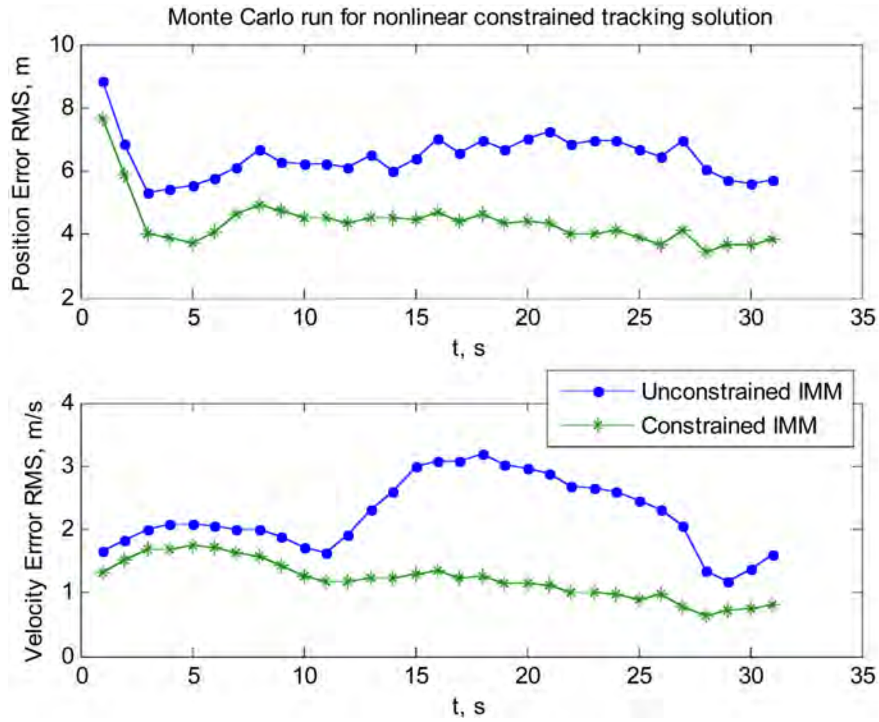

Fig. 10. Position and velocity error RMS versus time (100 Monte Carlo runs).

the iterative method presented in the paper for second-order nonlinear state constraints to other types of non-linear constraints of practical significance and to search for more efficient root-finding algorithms to solve for the Lagrangian multiplier. Similarly, the simple fusion of a single track to a single road as presented in this paper is being extended to multiple targets moving along closely-spaced road networks with intersections and by-passes. In this case, the fusion (or constraining) can take place in the measurement level as well as in the track level, involving road constrained data association (RCDA). Results will be reported in future papers.

ACKNOWLEDGMENTS

Research supported in part under Contract No. FA8650-05-C-1808, which is gratefully acknowledged. Thanks also go to the reviewers and the editor for their careful reading, insightful comments, and constructive suggestions, which were incorporated into the text.

APPENDIX A

To solve the constrained optimization problem in (4), we form the cost function including the Lagrangian multiplier

$$J(\mathbf{x},\lambda) = (\mathbf{x}-\hat{\mathbf{x}})^T\mathbf{W}(\mathbf{x}-\hat{\mathbf{x}}) + 2\lambda^T(\mathbf{D}\mathbf{x}-\mathbf{d}). \quad \text{(A1)}$$

The first order conditions necessary for a minimum are given by

$$\frac{\partial J}{\partial \mathbf{x}} = 0 \Rightarrow \mathbf{W}(\mathbf{x}-\hat{\mathbf{x}}) + \mathbf{D}^T\lambda = 0 \quad \text{(A2a)}$$

$$\frac{\partial J}{\partial \lambda} = 0 \Rightarrow \mathbf{D}\mathbf{x}-\mathbf{d} = 0. \quad \text{(A2b)}$$

The solution for the optimal Lagrangian multiplier λ can be found first as

$$\lambda = (\mathbf{D}\mathbf{W}^{-1}\mathbf{D}^T)^{-1}(\mathbf{D}\hat{\mathbf{x}}-\mathbf{d}). \quad \text{(A3)}$$

Bringing this solution back to (A1) leads to the constrained solution of the state in (5).

Note that the above derivation does not depend on the conditional Gaussian nature of the unconstrained estimate $\hat{\mathbf{x}}$. It was shown in [24] that when $\mathbf{W} = \mathbf{I}$, the solution in (5) is the same as what is obtained by the mean square method, which attempts to minimize the conditional mean square error subject to the state constraints, that is,

$$\min_{\mathbf{x}} E\{\|\mathbf{x}-\hat{\mathbf{x}}\|_2^2 \mid Y\} \qquad \text{such that} \quad \mathbf{D}\mathbf{x} = \mathbf{d}. \quad \text{(A4)}$$

Furthermore, when $\mathbf{W} = \mathbf{P}^{-1}$, i.e., the inverse of the unconstrained state estimation error covariance, the solution in (5) reduces to the result given by the maximum conditional probability method

$$\max_{\mathbf{x}} \ln \mathrm{Prob}\{\mathbf{x} \mid Y\} \qquad \text{such that} \quad \mathbf{D}\mathbf{x} = \mathbf{d}. \quad \text{(A5)}$$

TABLE II
RMS Estimation Errors

Estimators	RMS Estimation Error: Position (m)	Velocity (m/s)
Unconstrained IMM	9.0	6.2
Nonlinear Constrained IMM	3.2	1.7

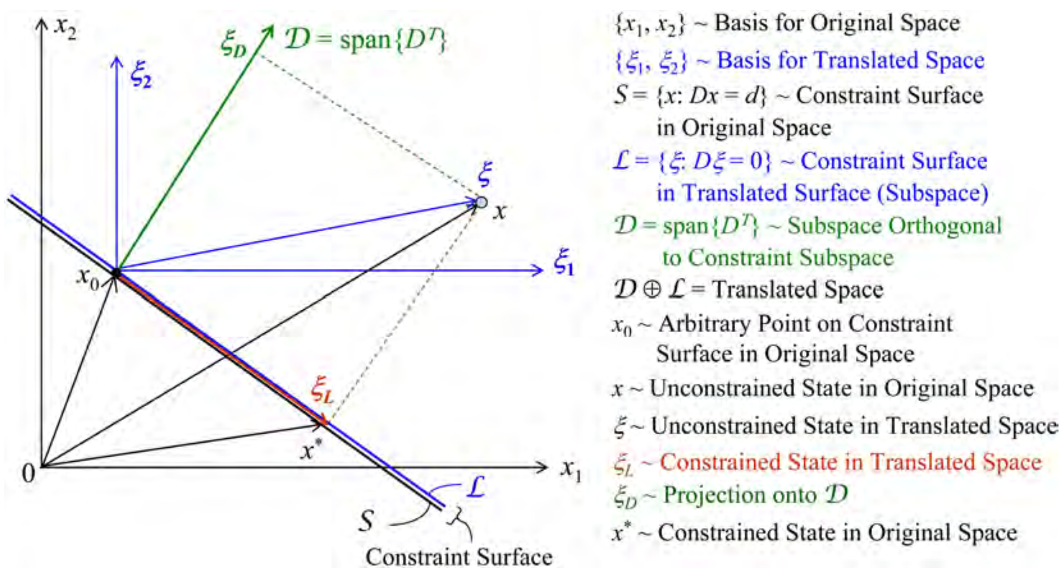

Fig. B1. Geometrical interpretation of linear constrained solution.

More results and proofs can be found in [24].

APPENDIX B

For $\mathbf{x} \in \mathbb{R}^n$, the constraint surface $\mathcal{S} = \{\mathbf{x} : \mathbf{D}\mathbf{x} = \mathbf{d}\}$ with the number of linear constraints $m < n$ is not a subspace simply because for $\mathbf{d} \neq \mathbf{0}$, the null vector is not inside $\mathcal{S}$. To construct a subspace, first find an arbitrary point $\mathbf{x}_0 \in \mathcal{S}$ and then define $\boldsymbol{\xi} = \mathbf{x} - \mathbf{x}_0$. This is equivalent to shifting the origin of the coordinates to $\mathbf{x}_0$, thus performing an affine transformation, denoted by $\mathbf{T}$. For all $\mathbf{x} \in \mathcal{S}$, the corresponding shifted vector $\boldsymbol{\xi}$ has the following property:

$$\mathbf{D}\boldsymbol{\xi} = \mathbf{D}(\mathbf{x} - \mathbf{x}_0) = \mathbf{D}\mathbf{x} - \mathbf{D}\mathbf{x}_0 = \mathbf{d} - \mathbf{d} = 0. \quad \text{(B1)}$$

In other words, the constraint surface after the affine transformation $\mathbf{T}$ becomes a subspace, denoted by $\mathcal{L} = \mathbf{T}\mathcal{S} = \{\boldsymbol{\xi} : \mathbf{D}\boldsymbol{\xi} = \mathbf{0}\}$, which has a dimension $n - m$. The affine transformation is illustrated in Fig. B1.

We are now to express $\mathcal{L}$. But first, the row vectors of $\mathbf{D}$ can be expressed as:

$$\mathbf{D}^T = [\mathbf{d}_1 \ \ \mathbf{d}_2 \cdots \mathbf{d}_m]. \quad \text{(B2)}$$

Since $\mathbf{D}$ is of full rank by assumption, the row vectors of $\mathbf{D}$ can be used as the non-orthogonal basis for a subspace denoted by $\mathcal{D} = \text{span}\{\mathbf{d}_1, \mathbf{d}_2, \ldots, \mathbf{d}_m\}$. In light of (B1) and by definition of $\mathcal{L}$, it is easy to see that $\mathcal{D}$ is an orthogonal complement of $\mathcal{L}$, that is, $\mathcal{D} \perp \mathcal{L}$ and $\mathcal{D} \oplus \mathcal{L} = \mathbb{R}^n$ where $\oplus$ stands for direct sum between two orthogonal subspaces.

For $\boldsymbol{\delta} \in \mathcal{D}$, it can be written as:

$$\boldsymbol{\delta} = \sum_{i=1}^{m} c_i \mathbf{d}_i = [\mathbf{d}_1 \ \ \mathbf{d}_2 \cdots \mathbf{d}_m] \begin{bmatrix} c_1 \\ c_2 \\ \vdots \\ c_m \end{bmatrix} = \mathbf{D}^T \mathbf{c}. \quad \text{(B3)}$$

Then for $\boldsymbol{\xi} \in \mathcal{L}$, we have

$$\langle \boldsymbol{\delta}, \boldsymbol{\xi} \rangle = \langle \mathbf{D}^T \mathbf{c}, \boldsymbol{\xi} \rangle = \boldsymbol{\delta}^T \boldsymbol{\xi} = \mathbf{c}^T \mathbf{D} \boldsymbol{\xi} = 0 \quad \text{(B4)}$$

where $\langle \mathbf{a}, \mathbf{b} \rangle = \mathbf{a}^T \mathbf{b}$ is the inner product defined on $\mathbb{R}^n$.

By the principle of orthogonality, an arbitrary vector $\boldsymbol{\xi}$ can be decomposed into its projections onto the orthogonal complements $\mathcal{D}$ and $\mathcal{L}$, denoted by $\boldsymbol{\xi}_D$ and $\boldsymbol{\xi}_L$, respectively, as

$$\boldsymbol{\xi} = \boldsymbol{\xi}_D + \boldsymbol{\xi}_L. \quad \text{(B5)}$$

Adding $\mathbf{x}_0$ to both sides of (B5), we can express the vectors in the original coordinates as:

$$\mathbf{x} = \boldsymbol{\xi} + \mathbf{x}_0 = \boldsymbol{\xi}_D + \boldsymbol{\xi}_L + \mathbf{x}_0 = \boldsymbol{\xi}_D + \mathbf{x}^*. \quad \text{(B6)}$$

The projection of the arbitrary vector on the constraint subspace $\mathcal{L}$ and the constraint surface $\mathcal{S}$ can be obtained, respectively, as:

$$\boldsymbol{\xi}_L = \boldsymbol{\xi} - \boldsymbol{\xi}_D \quad \text{(B7a)}$$

$$\mathbf{x}^* = \mathbf{x} - \boldsymbol{\xi}_D. \quad \text{(B7b)}$$

To obtain $\boldsymbol{\xi}_D$, express it as a linear combination of the non-orthogonal bases of $\mathbf{D}^T$ with the coefficient vector $\mathbf{c}$ as:

$$\boldsymbol{\xi}_D = \sum_{i=1}^{m} c_i \mathbf{d}_i = [\mathbf{d}_1 \ \ \mathbf{d}_2 \cdots \mathbf{d}_m] \begin{bmatrix} c_1 \\ c_2 \\ \vdots \\ c_m \end{bmatrix} = \mathbf{D}^T \mathbf{c}. \quad \text{(B8)}$$

Again, by the principle of orthogonality, the projection error vector $\boldsymbol{\xi} - \boldsymbol{\xi}_D$ is orthogonal to $\mathcal{D}$, i.e., each and every basis of it:

$$\langle \boldsymbol{\xi} - \boldsymbol{\xi}_D, \mathbf{d}_i \rangle = \langle \boldsymbol{\xi} - \mathbf{D}^T \mathbf{c}, \mathbf{d}_i \rangle = \mathbf{d}_i^T (\boldsymbol{\xi} - \mathbf{D}^T \mathbf{c}) = 0, \quad i = 1, \ldots, m. \quad \text{(B9)}$$

Stacking these orthogonality conditions, we obtain

$$\begin{bmatrix} \mathbf{d}_1^T \\ \mathbf{d}_2^T \\ \vdots \\ \mathbf{d}_m^T \end{bmatrix} (\boldsymbol{\xi} - \mathbf{D}^T\mathbf{c}) = 0 \qquad \text{or} \qquad \mathbf{D}(\boldsymbol{\xi} - \mathbf{D}^T\mathbf{c}) = 0. \tag{B10}$$

Since $\mathbf{D}\mathbf{D}^T$ is an $m \times m$ matrix and invertible, the coefficient vector can be obtained as:

$$\mathbf{c} = (\mathbf{D}\mathbf{D}^T)^{-1}\mathbf{D}\boldsymbol{\xi}. \tag{B11}$$

Bringing (B11) back to (B8) gives the projection vector as:

$$\boldsymbol{\xi}_D = \mathbf{D}^T(\mathbf{D}\mathbf{D}^T)^{-1}\mathbf{D}\boldsymbol{\xi} = \mathbf{P}\boldsymbol{\xi} \tag{B12}$$

where $\mathbf{P} = \mathbf{D}^T(\mathbf{D}\mathbf{D}^T)^{-1}\mathbf{D}$ is usually referred to as the projection matrix onto $\mathcal{D}$ and $(\mathbf{I} - \mathbf{P})$ is the projection matrix onto $\mathcal{L}$.

Bringing (B12) back to (B7) gives

$$\boldsymbol{\xi}_L = \boldsymbol{\xi} - \mathbf{P}\boldsymbol{\xi} = (\mathbf{I} - \mathbf{P})\boldsymbol{\xi} \tag{B13a}$$

$$\mathbf{x}^* = \mathbf{x} - \mathbf{P}\boldsymbol{\xi} = \mathbf{x} - \mathbf{P}(\mathbf{x} - \mathbf{x}_0). \tag{B13b}$$

Bringing the expression for $\mathbf{P}$ into (B13b) gives

$$\begin{aligned} \mathbf{x}^* &= \mathbf{x} - \mathbf{D}^T(\mathbf{D}\mathbf{D}^T)^{-1}\mathbf{D}(\mathbf{x} - \mathbf{x}_0) \\ &= \mathbf{x} - \mathbf{D}^T(\mathbf{D}\mathbf{D}^T)^{-1}(\mathbf{D}\mathbf{x} - \mathbf{D}\mathbf{x}_0) \\ &= \mathbf{x} - \mathbf{D}^T(\mathbf{D}\mathbf{D}^T)^{-1}(\mathbf{D}\mathbf{x} - \mathbf{d}) \end{aligned} \tag{B14}$$

where $\mathbf{D}\mathbf{x}_0 = \mathbf{d}$ is used to arrive at the last equation because of $\mathbf{x}_0 \in \mathcal{S}$.

Clearly, (B14) is exactly the same as (5) when $\mathbf{W} = \mathbf{I}$. This offers a geometric interpretation that the linear constrained estimation is the orthogonal projection of the unconstrained estimate onto the constrained surface. It provides a theoretical justification of the intuitive practice of finding a point along the road that is of the shortest distance.

APPENDIX C

When $\mathbf{W} \neq \mathbf{I}$, we can rewrite the weighted square error formulation as

$$\begin{aligned} \breve{\mathbf{x}} &= \arg\min_{\mathbf{x}\in S}(\mathbf{x} - \hat{\mathbf{x}})^T\mathbf{W}(\mathbf{x} - \hat{\mathbf{x}}) \\ &= \arg\min_{\mathbf{x}\in S}[\mathbf{W}^{1/2}(\mathbf{x} - \hat{\mathbf{x}})]^T\mathbf{W}^{1/2}(\mathbf{x} - \hat{\mathbf{x}}) = \arg\min_{\mathbf{z}\in\bar{S}}\mathbf{z}^T\mathbf{z} \end{aligned} \tag{C1}$$

where $\mathbf{W} = \mathbf{W}^{1/2}\mathbf{W}^{1/2}$ is a symmetric positive definite weighting matrix. This can be understood as if we perform an equivalent un-weighted optimization on the transformed state:

$$\mathbf{z} = \mathbf{W}^{1/2}\mathbf{x}. \tag{C2}$$

The constraint can be written as:

$$\mathbf{D}\mathbf{x} = \mathbf{D}\mathbf{W}^{-1/2}\mathbf{W}^{1/2}\mathbf{x} = \mathbf{M}\mathbf{z} = \mathbf{d} \tag{C3}$$

where $\mathbf{M} = \mathbf{D}\mathbf{W}^{-1/2}$ by definition. The constrained surface $\bar{S} = \{\mathbf{z} : \mathbf{M}\mathbf{z} = \mathbf{d}\}$ is used in the last equality of (C1).

Since the constrained solution in (B14) holds for $\mathbf{z}$ with $\mathbf{M}$ and $\mathbf{d}$, we have

$$\mathbf{z}^* = \mathbf{z} - \mathbf{M}^T(\mathbf{M}\mathbf{M}^T)^{-1}(\mathbf{M}\mathbf{z} - \mathbf{d}). \tag{C4}$$

Putting (C2) and (C3) into (C4) gives

$$\begin{aligned} \mathbf{W}^{1/2}\mathbf{x}^* &= \mathbf{W}^{1/2}\mathbf{x} - \mathbf{W}^{-1/2}\mathbf{D}^T(\mathbf{D}\mathbf{W}^{-1/2}\mathbf{W}^{-1/2}\mathbf{D}^T)^{-1} \\ &\quad \times (\mathbf{D}\mathbf{W}^{-1/2}\mathbf{W}^{1/2}\mathbf{x} - \mathbf{d}). \end{aligned} \tag{C5}$$

Multiplying both sides by $\mathbf{W}^{-1/2}$ gives the weighted constrained solution as:

$$\mathbf{x}^* = \mathbf{x} - \mathbf{W}^{-1}\mathbf{D}^T(\mathbf{D}\mathbf{W}^{-1}\mathbf{D}^T)^{-1}(\mathbf{D}\mathbf{x} - \mathbf{d}) \tag{C6}$$

which is exactly the same as (5).

It is interesting to note that the use of $\mathbf{W} = \mathbf{P}^{-1}$ has the effect of pre-whitening in the sense that

$$\mathrm{E}\{\mathbf{z}\mathbf{z}^T\} = \mathbf{P}^{-1/2}\mathrm{E}\{\mathbf{x}\mathbf{x}^T\}\mathbf{P}^{-1/2} = \mathbf{P}^{-1/2}\mathbf{P}\mathbf{P}^{-1/2} = \mathbf{I}. \tag{C7}$$

APPENDIX D

Construct the Lagrangian with the Lagrangian multiplier λ as

$$J(\mathbf{x},\lambda) = (\mathbf{z} - \mathbf{H}\mathbf{x})^T(\mathbf{z} - \mathbf{H}\mathbf{x}) + \lambda f(\mathbf{x}). \tag{D1}$$

Taking the partial derivatives of $J(\mathbf{x},\lambda)$ with respect to $\mathbf{x}$ and λ, respectively, setting them to zero leads to the necessary conditions

$$-\mathbf{H}^T\mathbf{z} + \lambda\mathbf{m} + (\mathbf{H}^T\mathbf{H} + \lambda\mathbf{M})\mathbf{x} = \mathbf{0} \tag{D2a}$$

$$\mathbf{x}^T\mathbf{M}\mathbf{x} + \mathbf{m}^T\mathbf{x} + \mathbf{x}^T\mathbf{m} + m_0 = 0. \tag{D2b}$$

Assume that the inverse matrix of $\mathbf{H}^T\mathbf{H} + \lambda\mathbf{M}$ exists. Then, $\mathbf{x}$ can be solved from (D2a), giving the constrained solution in terms of the unknown λ as

$$\mathbf{x} = (\mathbf{H}^T\mathbf{H} + \lambda\mathbf{M})^{-1}(\mathbf{H}^T\mathbf{z} - \lambda\mathbf{m}) \tag{D3}$$

which reduces to the unconstrained least-squares solution when $\lambda = 0$.

Assume that the matrix $\mathbf{M}$ admits the factorization $\mathbf{M} = \mathbf{L}^T\mathbf{L}$ and apply the Cholesky factorization to $\mathbf{W} = \mathbf{H}^T\mathbf{H}$ as

$$\mathbf{W} = \mathbf{G}^T\mathbf{G} \tag{D4}$$

where $\mathbf{G}$ is an upper right diagonal matrix. We then perform the SVD [17] of the matrix $\mathbf{L}\mathbf{G}^{-1}$ as

$$\mathbf{L}\mathbf{G}^{-1} = \mathbf{U}\boldsymbol{\Sigma}\mathbf{V}^T \tag{D5}$$

where $\mathbf{U}$ and $\mathbf{V}$ are orthonormal matrices and $\boldsymbol{\Sigma}$ is a diagonal matrix with its diagonal elements denoted by σ_i. For a square matrix, this becomes the eigenvalue decomposition.

Introduce two new vectors

$$\mathbf{e}(\lambda) = [\ldots e_i(\lambda),\ldots]^T = \mathbf{V}^T(\mathbf{G}^T)^{-1}(\mathbf{H}^T\mathbf{z} - \lambda\mathbf{m}) \tag{D6a}$$

$$\mathbf{t} = [\ldots t_i \ldots]^T = \mathbf{V}^T(\mathbf{G}^T)^{-1}\mathbf{m}. \tag{D6b}$$

With these factorizations and new matrix and vector notations, the constrained solution in (D3) can be simplified into (16a), which is repeated below for easy reference as

$$\mathbf{x} = \mathbf{G}^{-1}\mathbf{V}(\mathbf{I} + \lambda\boldsymbol{\Sigma}^T\boldsymbol{\Sigma})^{-1}\mathbf{e}(\lambda). \tag{D7}$$

The first and second order terms of $\mathbf{x}$ in (D2b) can be expressed in λ as:

$$\begin{aligned}\mathbf{x}^T\mathbf{M}\mathbf{x} &= \mathbf{e}(\lambda)^T(\mathbf{I} + \lambda\boldsymbol{\Sigma}^T\boldsymbol{\Sigma})^{-T}\boldsymbol{\Sigma}^T\boldsymbol{\Sigma}(\mathbf{I} + \lambda\boldsymbol{\Sigma}^T\boldsymbol{\Sigma})^{-1}\mathbf{e}(\lambda)\\ &= \sum_i \frac{e_i^2(\lambda)\sigma_i^2}{(1+\lambda\sigma_i^2)^2}\end{aligned} \tag{D8a}$$

$$\mathbf{m}^T\mathbf{x} = \mathbf{t}^T(\mathbf{I} + \lambda\boldsymbol{\Sigma}^T\boldsymbol{\Sigma})^{-1}\mathbf{e}(\lambda) = \sum_i \frac{e_i(\lambda)t_i}{1+\lambda\sigma_i^2} \tag{D8b}$$

$$\mathbf{x}^T\mathbf{m} = \mathbf{e}(\lambda)^T(\mathbf{I} + \lambda\boldsymbol{\Sigma}^T\boldsymbol{\Sigma})^{-1}\mathbf{t} = \sum_i \frac{e_i(\lambda)t_i}{1+\lambda\sigma_i^2}. \tag{D8c}$$

Bringing these terms into the constrained equation in (D2b) gives rise to the constraint equation, now expressed in terms of the unknown Lagrangian multiplier λ, as

$$\begin{aligned}q(\lambda) &= (\mathbf{z}^T\mathbf{H} - \lambda\mathbf{m}^T)(\mathbf{H}^T\mathbf{H} + \lambda\mathbf{M})^{-2}(\mathbf{H}^T\mathbf{z} - \lambda\mathbf{m})\\ &\quad + \mathbf{m}^T(\mathbf{H}^T\mathbf{H} + \lambda\mathbf{M})^{-1}(\mathbf{H}^T\mathbf{z} - \lambda\mathbf{m})\\ &\quad + (\mathbf{z}^T\mathbf{H} - \lambda\mathbf{m}^T)(\mathbf{H}^T\mathbf{H} + \lambda\mathbf{M})^{-1}\mathbf{m} + m_0\\ &= \mathbf{e}(\lambda)^T(\mathbf{I} + \lambda\boldsymbol{\Sigma}^T\boldsymbol{\Sigma})^{-1}\boldsymbol{\Sigma}^T\boldsymbol{\Sigma}(\mathbf{I} + \lambda\boldsymbol{\Sigma}^T\boldsymbol{\Sigma})^{-1}\mathbf{e}(\lambda)\\ &\quad + \mathbf{t}^T(\mathbf{I} + \lambda\boldsymbol{\Sigma}^T\boldsymbol{\Sigma})^{-1}\mathbf{e}(\lambda) + \mathbf{e}(\lambda)^T(\mathbf{I} + \lambda\boldsymbol{\Sigma}^T\boldsymbol{\Sigma})^{-1}\mathbf{t} + m_0\\ &= \sum_i \frac{e_i^2(\lambda)\sigma_i^2}{(1+\lambda\sigma_i^2)^2} + 2\sum_i \frac{e_i(\lambda)t_j}{1+\lambda\sigma_i^2} + m_0 = 0.\end{aligned} \tag{D9}$$

which is (16b) given in Section 3.

REFERENCES

[1] Y. I. Abromovich and M. B. Sverdlik
Synthesis of a filter response which maximizes the signal to noise ratio under additional quadratic constraints.
Radio Eng. and Electron. Phys., **5** (Nov. 1970), 1977–1984.

[2] C. Agate and K. J. Sullivan
Road-constraint target tracking and identification using a particle filter.
In O. E. Drummond (Ed.), *SPIE Proceedings of Signal and Data Processing of Small Targets*, vol. 5204, 2003, 532–543.

[3] B. Anderson and J. Moore
Optimal Filtering.
Englewood Cliffs, NJ: Prentice-Hall, 1979.

[4] S. Arulampalam, N. Gordon, M. Orton and B. Ristic
A variable structure multiple model particle filter for GMTI tracking.
In *Proceedings of the 5th International Conference on Information Fusion*, Annapolis, MD, July 2002, 927–934.

[5] Y. Bar-Shalom and X. R. Li
Multitarget-Multisensor Tracking: Principles and Techniques.
Storrs, CT: YBS Publishing, 1995.

[6] S. Blackman and R. Popoli
Design and Analysis of Modern Tracking Systems.
Boston, MA: Artrch House, 1999.

[7] T. Connare and E. Blasch
Group IMM tracking utilizing track and identification fusion.
In *Proceedings of the Workshop on Estimation, Tracking, and Fusion; A Tribute to Yaakov Bar Shalom*, Monterey, CA, May 2001, 205–220.

[8] A. Farina, L. Ferranti and G. Golino
Constrained tracking filters for A-SMGCS.
In *Proceedings of the 6th International Conference on Information Fusion*, Cairns, Queensland, Australia, July 2003, 414–421.

[9] G. C. Goodwin, J. A. De Dona, M. M. Seron and X. W. Zhuo
Lagrangian Duality between constrained estimation and control.
Automatica, **41** (June 2005), 935–944.

[10] T. Kirubarajan, Y. Bar-Shalom, K. R. Pattipati and I. Kadar
Ground target tracking with variable structure IMM estimator.
IEEE Transactions on Aerospace and Electronic Systems, **36** (Jan. 2002), 26–46.

[11] S. Ko and R. Bitmead
State estimation for linear systems with state equality constraints.
Automatica, **43**, 8 (Aug. 2007), 1363–1368.

[12] K. Koch
Ground target tracking with STAP-Radar: Selected tracking aspects.
In R. Klemm (Ed.), *Applications of Space-Time Adaptive Processing*, London: Institute of Electrical Engineers, 2004.

[13] M. Ulmke and W. Koch
Multi hypothesis track extraction and maintenance of GMTI sensor data.
In *Proceedings of the 8th International Conference on Information Fusion*, Philadelphia, PA, July 2005.

[14] R. G Lorenz and S. P. Boyd
Robust minimum variance beamforming.
In J. Li and P. Stoica (Eds.), *Robust Adaptive Beamforming*, Hoboken, NJ: Wiley-Interscience, 2006.

[15] D. G. Luenberger
Linear and Nonlinear Programming (2nd ed.).
Reading, MA: Addison-Wesley, 1989.

[16] M. Miller, E. Blasch, T. Nguyen and C. Yang
Constrained estimation for GPS/digital map integration.
In *Proceedings of the National Technical Meeting of the Institute of Navigation*, San Diego, CA, Jan. 2007, 1119–1127.

[17] T. K. Moon and W. C. Stirling
Mathematical Methods and Algorithms for Signal Processing.
Upper Saddle River, NJ: Prentice-Hall, 2000.

[18] B. J. Noe and N. Collins
Variable structure interacting multiple model filter (VS-IMM) for tracking targets with transportation network constraints.
In O. E. Drummond (Ed.), *SPIE Proceedings of Signal and Data Processing of Small Targets*, vol. 4048, 2000, 247–257.

[19] B. Pannetier, K. Benameur, V. Nimier and M. Rombau
VS-IMM using road map information for a ground target tracking.
In *Proceedings of the 8th International Conference on Information Fusion*, Philadelphia, PA, July 2005, A1-2 (1–8).

[20] C. V. Rao, J. B. Rawlings and D. Q. Mayne
Constrained state estimation for nonlinear discrete-time systems: Stability and moving horizon approximations.
IEEE Transactions on Automatic Control, **48** (Feb. 2003), 246–258.

[21] B. Ristic, S. Arullampalam and N. Gordon
Beyond the Kalman Filter—Particle Filters for Tracking Applications.
Boston, MA: Artech House, 2004.

[22] P. J. Shea, T. Zadra, D. Klamer, E. Frangione and R. Brouillard
Improved state estimation through use of roads in ground tracking.
In O. E. Drummond (Ed.), *SPIE Proceedings of Signal and Data Processing of Small Targets*, vol. 4048, 2000, 321–332.

[23] D. Simon
A game theory approach to constrained minimax state estimation.
IEEE Transactions on Signal Processing, **54** (Feb. 2006), 405–412.

[24] D. Simon and T. L. Chia
Kalman filtering with state equality constraints.
IEEE Transactions on Aerospace and Electronic Systems, **38** (Jan. 2002), 128–136.

[25] M. Ulmke
Improved GMTI-tracking using road maps and topographic information.
In O. E. Drummond (Ed.), *SPIE Proceedings of Signal and Data Processing of Small Targets*, vol. 5204, 2003.

[26] M. Ulmke and W. Koch
Road-map assisted ground moving target tracking.
IEEE Transactions on Aerospace and Electronic Systems, **42**, 4 (Oct. 2006), 1264–1274.

[27] C. Yang, M. Bakich and E. Blasch
Nonlinear constrained tracking of targets on roads.
In *Proceedings of the 8th International Conf. on Information Fusion*, Philadelphia, PA, July 2005, A8-3 (1–8).

[28] C. Yang and E. Blasch
Kalman filtering with nonlinear state constraints.
In *Proceedings of the 9th International Conference on Information Fusion*, Florence, Italy, July 2006, 8–15.

[29] C. Yang and D. M. Lin
Constrained optimization for joint estimation of channel biases and angles of arrival for small GPS antenna arrays.
In *Proceedings of the 60th Annual Meeting of the Institute of Navigation*, Dayton, OH, June 2004, 548–559.

Chun Yang received his Bachelor of Engineering from Northeastern University, Shenyang, China, in 1984, majored in industrial automation. In 1986, he received his Diplôme d'Études Approfondies from Institut National Polytechnique de Grenoble, Grenoble, France, majored in automatique et traitement du signal (LAG/ENSIEG). In 1989, he received his title of Docteur en Science from Université de Paris, Orsay, France, in sciences physiques (LSS/CNRS-ESE).

After two years of postdoctoral research at University of Connecticut, Storrs, CT, he moved on with his industrial R&D career. Since 1994, he has been with Sigtem Technology, Inc. and has been working on adaptive array and baseband signal processing for GNSS receivers and radar systems as well as on nonlinear state estimation with applications in target tracking, integrated inertial navigation, and information fusion. He served as tutorial chair for the ISIF/IEEE Fusion Conferences in Annapolis, MD (2000) and in Philadelphia, PA (2005), respectively. Dr. Yang is also an Adjunct Professor of Electrical and Computer Engineering at Miami University. He is the co-inventor of seven issued and pending U.S. patents, the co-author of an ION-GNSS best presentation paper (2007), the coauthor of an IEEE PLANS/ION AM best paper (2006), and the co-recipient of the ION Samuel M. Burka Award (2007).

Erik Blasch received his B.S. in mechanical engineering from the Massachusetts Institute of Technology and Masters Degrees in mechanical, health science, and industrial engineering from Georgia Tech. He attended the University of Wisconsin for an M.D./Ph.D. in mechanical engineering/neurosciences until being called to active duty in the United States Air Force. He completed an M.B.A., M.S.E.E., M.S. econ, M.S./Ph.D. psychology (ABD), and a Ph.D. in electrical engineering from Wright State University.

He is an information fusion evaluation tech lead for the Air Force Research Laboratory—COMprehensive Performance Assessment of Sensor Exploitation (COMPASE) Center (AFRL/RYAA), adjunct EE and BME professor at Wright State University (WSU) and Air Force Institute of Technology (AFIT), and a reserve Major with the Air Force Office of Scientific Research (AFRL/AFOSR). He was a founding member of the International Society of Information Fusion (ISIF) in 1998 and the 2007 ISIF president.

Dr. Blasch has many military and civilian career awards; but engineering highlights include team member of the winning 1991 American Tour del Sol solar car competition, 1994 AIAA mobile robotics contest, and the 1993 Aerial Unmanned Vehicle competition where they were first in the world to automatically control a helicopter. Since that time, Dr. Blasch has focused on automatic target recognition, targeting tracking, and information fusion research compiling 200+ scientific papers and book chapters. He is active in IEEE (AES and SMC) and SPIE including regional activities, conference boards, journal reviews, and scholarship committees; and participates in the Data Fusion Information Group (DFIG), the program committee for the DOD Automatic Target Recognition Working Group (ATRWG), and the SENSIAC National Symposium on Sensor and Data Fusion (NSSDF). He is a Fellow of SPIE.

Dynamic Scheduling of Multiple Hidden Markov Model-Based Sensors

WOOSUN AN

SATNAM SINGH

KRISHNA R. PATTIPATI

DAVID L. KLEINMAN

SWAPNA S. GOKHALE

In this paper, a hidden Markov model (HMM)-based dynamic sensor scheduling problem is formulated, and solved using information gain and rollout concepts to overcome the computational intractability of the dynamic programming recursion. The problem involves dynamically sequencing a set of sensors to monitor multiples tasks, which are modeled as multiple HMMs with multiple emission matrices corresponding to each of the sensors. The dynamic sequencing problem is to minimize the sum of sensor usage costs and the task state estimation error costs. The rollout information gain algorithm proposed herein employs the information gain heuristic as the base algorithm to solve the dynamic sensor sequencing problem. The information gain heuristic selects the best sensor assignment at each time epoch that maximizes the sum of information gains per unit sensor usage cost, subject to the assignment constraints that at most one sensor can be assigned to a HMM and that at most one HMM can be assigned to a sensor. The rollout strategy involves combining the information gain heuristic with the Jonker-Volgenant-Castañòn (JVC) assignment algorithm and a modified Murty's algorithm to compute the κ-best assignments at each decision epoch of rollout. The capabilities of the rollout information gain algorithm are illustrated using a hypothetical scenario to monitor intelligence, surveillance, and reconnaissance (ISR) activities in multiple fishing villages and refugee camps for the presence of weapons and terrorists or refugees.

Manuscript received September 25, 2007; revised June 26, 2008; released for publication July 16, 2008.

Refereeing of this contribution was handled by Dr. Shozo Mori.

Authors' address: Electrical and Computer Engineering Department, University of Connecticut, Storrs, CT 06269-2157, E-mail: (woa05001@engr.uconn.edu, satnam@engr.uconn.edu,krishna@engr.uconn.edu, dlkleinm@nps.edu, ssa@engr.uconn.edu).

A version of this paper was published in *Proceedings of the IEEE International Conference on Systems, Man, and Cybernetics*, Montreal, Quebec, Canada, 2007. The study reported in this paper was supported, in part, by the Office of Naval Research under contract no.N00014-06-1-0080. Any opinions expressed in this paper are solely those of the authors and do not represent those of the sponsor.

1. INTRODUCTION

Complex applications involving threat detection, such as multi-target tracking and unmanned aerial vehicles for surveillance in remote or hostile environments, include heterogeneous sensors, which trade off performance (e.g., detection, identification, and tracking accuracies) versus the sensor usage cost (e.g., power and bandwidth consumption, distance traveled, risk of exposure, deployment requirements). The objective of dynamic sensor scheduling is to judiciously allocate sensing resources to exploit the individual sensors' capabilities, while minimizing their usage cost. As an example, consider a target identification scenario where an incoming aircraft needs to be identified as an enemy or a friendly target using active or passive sensors available at a surveillance station [16]. This scenario requires sensor scheduling because active sensors (e.g., radar) tend to reveal clues about the location of the surveillance station to a potential enemy aircraft, whereas the more stealthy passive sensors tend to be inaccurate [16]. Thus, in this case, the sensor scheduling algorithm needs to trade-off accuracy versus risk of exposure. As another example, unmanned aerial vehicles (UAVs) are preferred assets for monitoring nearly all the intelligence, surveillance, and reconnaissance (ISR) activities; however, they cannot be deployed in large numbers due to their limited availability. Thus, astute allocation of scarce resources is a major issue in sensor scheduling.

In this paper, we consider the sensor scheduling problem faced by an ISR officer of an expeditionary strike group (ESG) in coordinating the use of surveillance assets (sensors) to improve situational awareness [14]. An ESG provides a flexible Navy-Marine force, capable of tailoring itself to a wide variety of missions. An important ESG mission involves dealing with asymmetric threats, such as terrorist groups who carry out attacks while trying to avoid direct confrontation. Terrorist groups are elusive, secretive, amorphously structured and decentralized entities that often appear unconnected. This stealthy behavior makes it very difficult to predict when and where they will strike. Moreover, the increased geographical range and unpredictable nature of this behavior require effective allocation and appropriate scheduling of sensors to achieve mission objectives. Effectively performing the ISR activities is a key step to gain situational awareness, which, in turn, enables the allocation of resources for the interdiction of potential threats.

We model the asymmetric threats using hidden Markov models, because these activities are concealed and their true states can only be inferred through the observations obtained using various ISR sensors. A pattern of these observations and its dynamic evolution over time provides the information base for inferring a potential realization of a threat [25]. Performing the ISR activities requires multiple sensors to provide observations needed for accurately estimating the status of suspicious activities. The available sensors are limited in number, and possess different attributes requiring judicious sensor allocation over time. Therefore, an effective scheduling of ISR sensors over time is essential for accurate situation assessment and to the success of the overall mission.

1.1. Previous Work

The dynamic sensor scheduling problem, which has been widely studied in the area of target tracking (e.g., [8], [28]), is to solve a sequential stochastic optimization problem that seeks to minimize the expected scheduling cost under a given set of resource constraints over time [8]. For linear Gaussian state space systems, one can obtain an analytic solution for the posterior distribution of system state given sensor measurements, and a scheduling sequence via a Kalman filter [18]. Shakeri *et al.* [24] formulated the sensor scheduling problem subject to a fixed total budget and the cost of individual sensor varying inversely with its measurement variance. They obtained the optimal measurement variance distribution that minimizes the trace of a weighted sum of the estimation error covariance matrices of a discrete-time vector stochastic process, when the auto-correlation matrix of the process is given. The study showed that the problem can be transformed into an optimization problem with linear equality and inequality constraints. In the special case of a linear finite-dimensional stochastic system, they showed that the problem can be formulated as an optimal control problem, where the gradient and Hessian of the objective function with respect to the sensor accuracy parameters can be derived via a two-point boundary value problem. The resulting optimization problem was solved via a projected Newton Method [4], [24].

In [26], Singh *et al*. provided a summary of previous research on sensor scheduling for tracking targets, whose dynamics are modeled by linear Gauss-Markov processes. They formulated the sensor scheduling problem as one of minimizing the variance of the estimation error of hidden states of a continuous-time HMM with respect to a given action sequence [26]. The authors proposed a stochastic gradient algorithm to determine the optimal schedule for the HMM. Another effort, related to our work, using a discrete HMM framework was considered by Krishnamurthy in [16]. Here, the author proposed a stochastic dynamic programming (DP) framework to solve the sensor scheduling problem for a single HMM, which is intractable for all but simple HMMs with a few states (e.g., at most 15 states).

Sub-optimal approaches, based on information-theoretic criteria, have been developed to overcome the computational intractability of determining the optimal sensor schedule. For a linear Gauss-Markov system, Logothetis *et al.* [17] formulated the sensor scheduling problem as one of determining a sequence of active sensors to maximize the mutual information between the states of the unobserved dynamic process and the observation process generated by the sensors. In the context of sensor networks, Zhao *et al.* [29] and Chu *et al.* [9] formulated the target tracking problem as a sequential Bayesian estimation problem, where the participants for sensor collaboration are determined by minimizing an objective function comprised of information utility, measured in terms of entropy, Mahalanobis distance and the sensor usage cost.

Rollout algorithms were first proposed for the approximate solution of dynamic programming recursions by Bertsekas *et al.* in [5], [6]. They are a class of suboptimal solution methods inspired by the policy iteration of dynamic programming and the approximate policy iteration of neuro-dynamic programming. The rollout algorithm, combined with the information gain heuristic (IG), was first proposed in our previous research on sequential fault diagnosis [27], where the system state is fixed (i.e., static), but unknown. In [27], we showed that rollout strategy, which can be combined with the one-step or multi-step look-ahead heuristic algorithms as base algorithms, can solve test sequencing problems in real-world systems with a higher computational efficiency than the optimal strategies, while being superior to those using the base algorithms only. In order to coordinate multiple sensor resources to track and discriminate targets modeled as continuous-state HMMs, Schneider *et al.* [23] presented a rollout approach to approximate the dynamic programming recursion using a cost-to-go function based on feasible candidate scenarios. In contrast, our approach employs discrete-state HMMs to model tasks and an information gain heuristic to estimate the cost-to-go function.

In this paper, two-dimensional assignment algorithms, exemplified by the Jonker-Volgenant-Castañon (JVC) [15] and the auction [2], [3], are used to obtain an assignment for maximizing the sum of information gains per unit sensor usage cost accrued by assigning multiple sensors to multiple HMMs. The JVC and the auction are the most efficient algorithms for solving the two-dimensional (2-D) assignment problems. The JVC algorithm is a primal-dual optimization method that includes an effective initialization of dual variables, and an augmentation phase based on the Dijkstra's shortest path algorithm [11]. The auction algorithm, proposed by Bertsekas *et al.* [2], [3], consists of a bidding phase and an assignment phase, where an optimal assignment is found by employing a coordinate descent method on the dual function. However, scaling of the information gain matrix is critical to the success of the auction algorithm. The κ-best assignment algorithm, first proposed

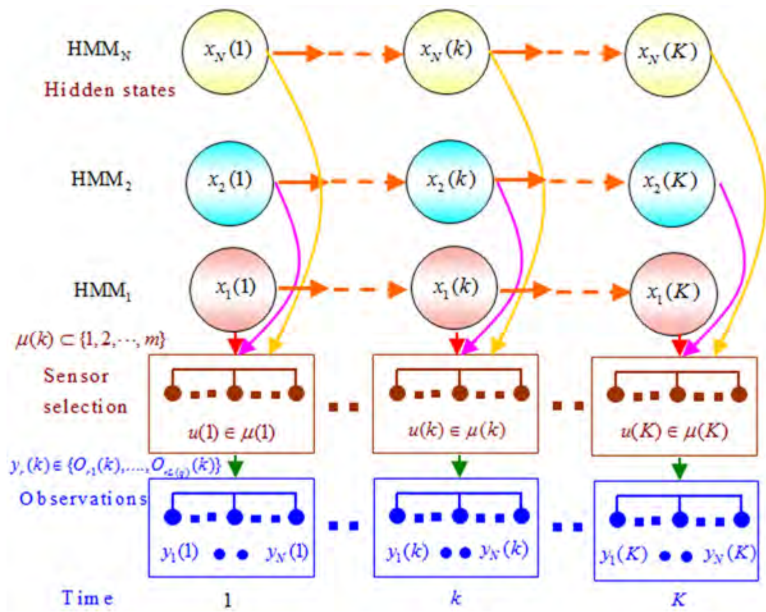

Fig. 1. Sensor scheduling problem for multiple HMMs.

by Murty [20], is independent of the algorithm chosen for solving the assignment problem. This algorithm ranks all the assignments in the order of decreasing objective function value by a clever partitioning of the search space of feasible assignments. The computational efficiency of Murty's algorithm has been enhanced by Cox *et al.* [10], Miller *et al.* [19], and Popp *et al.* [21], where the κ-best assignment algorithm is used to rank order assignment solutions for data association.

1.2. Scope and Organization of the Paper

This paper makes three novel contributions. First, motivated by the intractability of DP recursion even for a single HMM-based sensor scheduling problem [16] and its success in sequential probing for fault diagnosis [27], we propose a greedy heuristic algorithm based on information gain per unit sensor usage cost. We derive the information gain of a sensor for HMM models in the predictor-corrector form of state estimation equations, which are ideally suited for on-line implementation. Second, we improve the information gain heuristic algorithm by embedding it in a rollout algorithm to improve its scheduling performance. This is accomplished via the solution of a κ-best assignment algorithm. The multiple HMM scheduling problems using the combined rollout and assignment approach proposed herein have not been considered in the literature. Finally, the algorithms are applied to realistic ISR mission scenarios arising in ESG missions.

The paper is organized as follows. In Section 2, the multiple sensor scheduling problem is formulated. In Section 3, the DP recursion is developed. In Section 4, we present the rollout information gain heuristic algorithm based on JVC and κ-best assignment algorithm. We apply our solution approach to the ISR mission scenario, and present its results in Section 5. Finally, Section 6 concludes with a summary.

2. MULTIPLE HMM SENSOR SCHEDULING PROBLEM

2.1. The Factorial Hidden Markov Model (FHMM) for Dynamic Sensor Scheduling

Consider a scenario with N marginally independent discrete HMMs evolving independently and coupled via the observation process, as shown in Fig. 1. This model is also known as FHMM in the machine learning literature [13]. However, our framework is valid for coupled HMMs [7] and hierarchical HMMs [12] as well. Suppose there are m sensors, and $\mu(k) \subseteq \{1,2,\ldots,m\}$ are the set of available sensors at decision epoch $k \in \{1,2,\ldots,K\}$. We assume that at most a single sensor out of available sensors, $\mu(k)$, is assigned for observing the hidden state of a HMM at time epoch k. The FHMM is parameterized by the set of transition probability matrices $\mathbf{A}(k)$, the set of emission matrices $\mathbf{B}(k)$, and the set of initial probability vectors φ. We assume that the FHMM parameter sets $\Lambda(k) = (\mathbf{A}(k), \mathbf{B}(k), \varphi)$ ($k = 1,2,\ldots,K$), are known *a priori*; however, they could also be estimated based on historical data using the Baum-Welch algorithm [1].

The set of transition probability matrices of the underlying Markov chains associated with the N HMMs is given by $\mathbf{A}(k) = \{A_1(k),\ldots,A_r(k),\ldots,A_N(k)\}$ at time epoch k, where $A_r(k)$ denotes the transition probability matrix of the rth HMM:

$$A_r(k) = [a_{rij}(k)] = [P(x_r(k) = s_{rj} \mid x_r(k-1) = s_{ri})], \quad i,j = 1,2,\ldots,n_r \qquad (1)$$

where $x_r(k)$ is the hidden state of rth HMM at time epoch k. The hidden state $x_r(k) \in \{s_{ri} : i = 1,2,\ldots,n_r\}$, where n_r is the number of states used for modeling the rth HMM. We denote the subset of emission matrices, corresponding to each of the m sensors associated with the rth HMM, as $B_r(k) = \{B_{r1}(k),\ldots,B_{rq}(k),\ldots,B_{rm}(k)\}$ at time epoch k. The set of emission matrices for the N HMMs is denoted by $\mathbf{B}(k) = \{B_1(k),\ldots,B_r(k),\ldots, B_N(k)\}$. The observation, measured by sensor $u_r(k) = q$ assigned to the rth HMM at time epoch k, is denoted by $y_r(k) \in \{O_{r1}(k),\ldots,O_{rL(q)}(k)\}$, i.e., it belongs to one of $L(u_r(k) = q)$ symbols. Evidently, the number of observation symbols $L(q)$ can be a function of the sensors. This models a realistic scenario in which different sensors have different capabilities in generating different observation symbols. If none of the sensors is assigned to a HMM at a given epoch, we assume that the observed symbol is null (ϕ). The probability of observing the symbol $O_{rl}(k)$ $(l = 1,2,\ldots,L(q))$ with the sensor $u_r(k) = q$ assigned to the rth HMM, given the state $x_r(k) = s_{ri}$, denoted by $b_{rliq}(k)$, is an element of the emission matrix, $B_{rq}(k)$. That is,

$$B_{rq}(k) = [b_{rliq}(k)] = [P(y_r(k) = O_{rl}(k) \mid x_r(k) = s_{ri}, u_r(k) = q],$$
$$i = 1,2,\ldots,n_r; \quad l = 1,2,\ldots,L(q);$$
$$q = 1,2,\ldots,m; \quad r = 1,2,\ldots,N. \qquad (2)$$

The key point here is that the observation $y_r(k)$ depends upon the current state $x_r(k)$ and the selected sensor $u_r(k)$ from among the available sensors at time k. At time epoch k, we have, for each HMM ($r = 1,2,\ldots,N$), the information sets $\{Y_r^{k-1}, U_r^{k-1}\}$, where $Y_r^{k-1} = \{y_r(1),\ldots,y_r(k-1)\}$ and $U_r^{k-1} = \{u_r(1),\ldots, u_r(k-1)\}$, the previously observed symbols and the sensor sequence used from time epoch $t = 1$ to time epoch $t = k-1$. Evidently, $Y_r^0 = U_r^0 = \phi$. The initial probability of the underlying Markov states of the rth HMM at time $t = 0$ is denoted by

$$\underline{\varphi}_r = [\varphi_{ri} = p(x_r(0) = s_{ri})],$$
$$i = 1,2,\ldots,n_r; \quad r = 1,2,\ldots,N. \qquad (3)$$

We denote the set of initial probability vectors of the N HMMs as $\boldsymbol{\varphi} = \{\underline{\varphi}_1,\ldots,\underline{\varphi}_r,\ldots,\underline{\varphi}_N\}$.

2.2. Dynamic Sensor Scheduling Cost

The sensor scheduling problem is the following: How to find the policy to optimally allocate the m sensors to the N HMMs from time epoch 1 to time epoch K, based on $\{\mathbf{Y}^{k-1},\mathbf{U}^{k-1}\}_{k=1}^{K}$, where $(\mathbf{Y}^{k-1},\mathbf{U}^{k-1}) = \{Y_r^{k-1}, U_r^{k-1}\}_{r=1}^{N}$, the information available to optimize the sensor schedule at time epoch k. The sensor scheduling cost function is a sum of sensor usage costs and the state estimation errors over the planning horizon. The information states $\mathbf{\Pi}(k \mid k-1) = \{\underline{\pi}_1(k \mid k-1),\ldots, \underline{\pi}_r(k \mid k-1),\ldots,\underline{\pi}_N(k \mid k-1)\}^T$ are sufficient statistics to describe the current state of the N HMMs, where $\underline{\pi}_r(k \mid k-1) = \{\pi_{r1}(k \mid k-1),\ldots,\pi_{rn_r}(k \mid k-1)\}^T$. Indeed, the information state is the *predicted* probability of the hidden state $\mathbf{X}(k) = \{\underline{x}_1(k),\ldots,\underline{x}_r(k),\ldots,\underline{x}_N(k)\}^T$ given the available information, $\{\mathbf{Y}^{k-1},\mathbf{U}^{k-1}\}$, where $\underline{x}_r(k) = \{s_{r1}(k),\ldots,s_{rn_r}(k)\}^T$, i.e.,

$$\mathbf{\Pi}(k \mid k-1) = \mathbf{P}(\mathbf{X}(k) \mid \mathbf{Y}^{k-1},\mathbf{U}^{k-1}) \qquad (4)$$

where $\mathbf{P}(\mathbf{X}(k) \mid \mathbf{Y}^{k-1},\mathbf{U}^{k-1}) = \{P(\underline{x}_1(k) \mid Y_1^{k-1},U_1^{k-1}),\ldots, P(\underline{x}_r(k) \mid Y_r^{k-1},U_r^{k-1}),\ldots,P(\underline{x}_N(k) \mid Y_N^{k-1},U_N^{k-1})\}^T$. Let us denote the sensor scheduling policy from time epoch 1 to time epoch K by $\xi = \{\xi(k)\}_{k=1}^{K}$. For a given policy, the cumulative expected schedule cost from time epoch 1 to time epoch K, denoted by J_ξ, is assumed to be of the form:

$$J_\xi = E\left[\sum_{r=1}^{N}\left[\beta f_{rK}(\underline{\pi}_r(K \mid K)) + \sum_{k=0}^{K-1}\beta f_{rk}(\underline{\pi}_r(k \mid k)) + \sum_{k=1}^{K} g_{rk}(u_r(k),\underline{\pi}_r(k \mid k-1))\right]\right] \qquad (5)$$

where $\underline{\pi}_r(k \mid k)$ is the updated (corrected) information state, $f_{rk}(\underline{\pi}_r(k \mid k))$ is the state estimation error, $g_{rk}(u_r(k),\underline{\pi}_r(k \mid k-1))$ is the sensor cost of the rth HMM, and β is a positive scalar weight. Here, the expectation is over the stochastic realizations of measurement sequences. Typical cost functions for the state estimation error are as follows:

$$f_{rk}(\underline{\pi}_r(k \mid k)) = 1 - \underline{\pi}_r^T(k \mid k)\underline{\pi}_r(k \mid k), \qquad (6)$$

$$f_{rk}(\underline{\pi}_r(k \mid k)) = 1 - \max_{i \in \{1,\ldots,n_r\}} \pi_{ri}(k \mid k), \qquad (7)$$

$$f_{rk}(\underline{\pi}_r(k \mid k)) = \min_{1 \le i \le n_r} \sum_{j=1}^{n_r} \pi_{rj}(k \mid k)\lambda_{ij}. \qquad (8)$$

The first criterion as given in (6) can be interpreted as the L_2-norm of the updated state estimation error; the second criterion in (7) as the error probability of a maximum *a posteriori* probability (*MAP*)-based decision rule; while the third criterion in (8) as the expected cost of errors in estimating the information state. In (8), λ_{ij} represents the cost of erroneously estimating the hidden state as s_{ri} when the true state is s_{rj}. The sensor cost $g_{rk}(u_r(k),\underline{\pi}_r(k \mid k-1))$ is the sum of sensor usage cost $h_{rk}(u_r(k),\underline{\pi}_r(k \mid k-1))$ and sensor travel (movement) cost $c_m(u_r(k))$, i.e.,

$$g_{rk}(u_r(k),\underline{\pi}_r(k \mid k-1)) = h_{rk}(u_r(k),\underline{\pi}_r(k \mid k-1)) + c_m(u_r(k)) \qquad (9)$$

where the sensor usage cost is given by

$$h_{rk}(u_r(k),\underline{\pi}_r(k \mid k-1)) = \sum_{i=1}^{n_r} c_{rk}(s_{ri},u_r(k))\pi_{ri}(k \mid k-1) = \underline{c}_k^T(u_r(k))\underline{\pi}_r(k \mid k-1) \qquad (10)$$

and $\underline{c}_k^T(u_r(k)) = \{c_{rk}(s_{r1}, u_r(k)), c_{rk}(s_{r2}, u_r(k)), \ldots, c_{rk}(s_{rn_r}, u_r(k))\}$ is the usage cost of sensor $u_r(k)$ corresponding to each of the states $\{s_{ri}\}_{i=1}^{n_r}$. We also considered the cost of moving a sensor, denoted by $c_m(u_r(k))$, from its current location to the location of the task. This cost is computed via a simplified travel cost model as

$$c_m(u_r(k)) = \frac{\|(a_r, b_r) - (a_{u_r(k)}, b_{u_r(k)})\|_2}{v(u_r(k))} w(u_r(k)) \tag{11}$$

where (a_r, b_r) and $(a_{u_r(k)}, b_{u_r(k)})$ denote the cartesian coordinates of the task location (indexed by the HMM) and the selected sensor $u_r(k)$ for monitoring the rth HMM, respectively. Here, $w(u_r(k))$ is a priority parameter that accounts for the scarcity of the sensor, $v(u_r(k))$ denotes the velocity of the sensor (or the mobile platform on which it is resident), and $\|\cdot\|_2$ denotes the Euclidean (2-) norm.

3. DYNAMIC PROGRAMMING ALGORITHMS FOR OPTIMAL SOLUTION

The optimal solution to the sensor scheduling problem is to find the sensor assignment policy ψ^* which minimizes the sensor scheduling cost defined in (5). Let us define a optimal cost-to-go function $J^*(\Pi(k \mid k))$as follows:

$$\begin{aligned} J^*(\mathbf{\Pi}(k \mid k)) &= E\left[\sum_{r=1}^{N}\left[\beta f_{rK}(\underline{\pi}_r(K \mid K)) + \sum_{l=k}^{K-1} \beta f_{rl}(\underline{\pi}_r(l \mid l)) \right.\right. \\ &\quad \left.\left. + \sum_{l=k}^{K} g_{rl}(\psi_r^*(\mathbf{\Pi}(l \mid l-1)), \underline{\pi}_r(l \mid l-1))\right]\right] \end{aligned} \tag{12}$$

where $\psi_r^*(\Pi(l \mid l-1)) = u_r^*(l)$ is the optimal sensor allocated to rth HMM in the optimal policy. The optimal cost-to-go function $J^*(\mathbf{\Pi}(k \mid k))$ satisfies the dynamic programming (DP) recursion:

$$\begin{aligned} J^*(\mathbf{\Pi}(k \mid k)) &= E\left[\sum_{r=1}^{N}[\beta f_{rk}(\underline{\pi}_r(k \mid k)) + g_{rk}(\psi_r^*(\mathbf{\Pi}(k \mid k-1)), \underline{\pi}_r(k \mid k-1))]\right. \\ &\quad \left. + J^*(\mathbf{\Pi}(k+1 \mid k+1))\right] \end{aligned} \tag{13}$$

with the terminal condition $J^*(\mathbf{\Pi}(K \mid K)) = \sum_{r=1}^{N} \beta f_{rK} \cdot (\underline{\pi}_r(K \mid K)) + g_{rK}(\psi_r^*(\mathbf{\Pi}(K \mid K-1)), \quad \underline{\pi}_r(K \mid K-1))$. Hence, the optimal solution of (5) can be obtained using the dynamic programming (DP) technique; however, the computational complexity is $\mathbf{O}(\prod_{r=1}^{N} D^{(n_r-1)} nLmK)$. Here, D is the number of quantization levels used to discretize the continuous-valued information probability state, m is the number of sensors, K is the number of time epochs, N is the number of HMMs, n_r is the number of states of rth HMM, $n = \max_{r \in \{1,2,\ldots,N\}} n_r$, $L = \max_{r \in \{1,2,\ldots,N\}, q \in \{1,2,\ldots,m\}} (L(u_r(.) = q))$, and $L(u_r(.) = q)$ is the number of observation symbols when sensor q is allocated to the rth HMM at any epoch. The computational complexity is intractable in both n and N. This motivates us to investigate suboptimal algorithms to solve the dynamic sensor scheduling problem. We propose the rollout information gain (RIG) algorithm with computational complexity of $\mathbf{O}(NnLm^2K)$ per rollout, which is significantly lower than that for the DP technique.

4. ROLLOUT STRATEGIES TO SOLVE SENSOR SCHEDULING PROBLEM WITH MULTIPLE HMMS

4.1. Information Gain Heuristic as a Base Policy

Multiple HMM sensor scheduling involves two-dimensional (2-D) assignment or a weighted bipartite matching problem, where one set of nodes corresponds to sensors and the other set to HMMs. When allocating m sensors among N HMMs at each time epoch using the information gain heuristic algorithm, one needs to consider the $m \times N$ matrix of information gains for each sensor-HMM pair, where the elements of qth row correspond to information gains obtained by assigning sensor q to each of the N HMMs, as shown in Fig. 2. The information gain heuristic algorithm selects the best sensor assignment at each time epoch k, $\delta^*(k)$, that maximizes the sum of information gains per unit sensor usage cost, subject to the assignment constraints that at most one sensor can be assigned to a HMM and that at most one HMM can be assigned to a sensor. The assignment problem at time epoch k is (assuming without loss of generality that $m < N$)[1]

$$\begin{aligned} \delta^*(k) &= \arg\max_{\delta(k) \in \xi(k)} \sum_{q=1}^{m} \sum_{r=1}^{N} \frac{I_{qr}(\underline{\pi}_r(k \mid k-1), u_r(k) = q)}{g_{rk}(u_r(k), \underline{\pi}_r(k \mid k-1))} \delta_{qr}(k) \\ &\text{subject to} \quad \sum_{q=1}^{m} \delta_{qr}(k) \leq 1, \qquad r = 1, 2, \ldots, N; \\ &\qquad\qquad \sum_{r=1}^{N} \delta_{qr}(k) = 1, \qquad q = 1, 2, \ldots, m \end{aligned} \tag{14}$$

where $g_{rk}(u_r(k), \underline{\pi}_r(k \mid k-1))$ is the sensor usage cost when it is assigned to the rth HMM as defined in (9), and $I_{qr}(\underline{\pi}_r(k \mid k-1), u_r(k) = q)$ is the information gain

[1]This formulation can be extended to the case where multiple sensors may be needed to estimate a HMM state or a single sensor can estimate states of multiple HMMs. See [4].

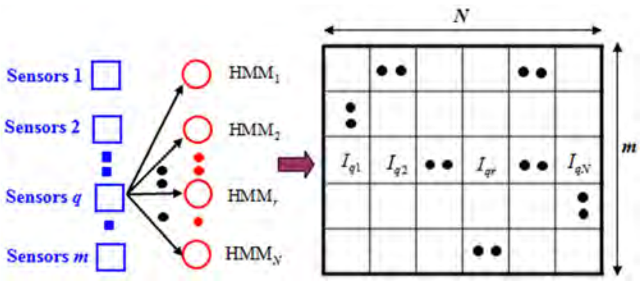

Fig. 2. Information gain matrix for multiple HMM-multiple sensor case.

given by:

$$
\begin{aligned}
&I_{qr}(\underline{\pi}_r(k \mid k-1), u_r(k) = q) \\
&\quad = \sum_{i=1}^{n_r} \pi_{ri}(k \mid k-1) \sum_{l=1}^{L(q)} b_{rliq}(k) \log_2 b_{rliq}(k) \\
&\qquad - \sum_{l=1}^{L(q)} \left(\sum_{i=1}^{n_r} b_{rliq}(k) \pi_{ri}(k \mid k-1) \right) \\
&\qquad \cdot \log_2 \left(\sum_{i=1}^{n_r} b_{rliq}(k) \pi_{ri}(k \mid k-1) \right). \qquad (15)
\end{aligned}
$$

The derivation of information gain equation is provided in the Appendix. The formulation in (13) is an asymmetric assignment problem, because none of the sensors may be assigned to some HMMs, leading to *null* observations at that time epoch for the corresponding unassigned HMMs.

The Jonker-Volgenant-Castañòn (JVC) and the auction are the most efficient algorithms for solving the (2-D) assignment problems. The JVC algorithm [15] is a primal-dual method that includes an effective initialization of dual variables, and an augmentation phase based on the Dijkstra's shortest path algorithm [11]. The auction algorithm, proposed by Bertsekas *et al.* [2] [3], consists of a bidding phase and an assignment phase, where an optimal assignment is found by employing a coordinate descent method on the dual function. However, scaling of the weight (in our case the information gain per unit cost) matrix is critical to the success of the auction algorithm.

The JVC algorithm is used here for finding the best assignment of sensors among multiple HMMs at each time epoch. Thus, in the multiple HMM case, the information gain heuristic algorithm can be implemented using the following five steps (see Fig. 3 for IG heuristic processing steps of a single (rth) HMM):[2]

Step 1 (State Prediction): Predict the information state vector set $\mathbf{\Pi}(k \mid k-1) = \{\underline{\pi}_1(k \mid k-1), \ldots, \underline{\pi}_N(k \mid k-1)\}^T$. Here, $\underline{\pi}_r(k \mid k-1)$ at time epoch k is predicted using the current updated information state vector at time epoch $(k-1)$, $\underline{\pi}_r(k-1 \mid k-1)$, and the transition matrix $A_r(k)$:

$$\underline{\pi}_r(k \mid k-1) = A_r^T(k)\underline{\pi}_r(k-1 \mid k-1), \quad 1 \le r \le N. \tag{16}$$

Here N is the number of HMMs being tracked and $\underline{\pi}_r(k \mid k-1) = \{\pi_{r1}(k \mid k-1), \ldots, \pi_{rn_r}(k \mid k-1)\}^T$. Evidently, the updated information state $\mathbf{\Pi}(k-1 \mid k-1)$ uses all the information available up to time epoch $k-1$, i.e., $\{\mathbf{Y}^{k-1}, \mathbf{U}^{k-1}\}$.

Step 2 (Generation of Information Gain Matrix): We construct the matrix of information gains per unit sensor cost for all sensor-HMM pairs,

$$\frac{I_{qr}(\underline{\pi}_r(k \mid k-1), u_r(k) = q)}{g_{rk}(u_r(k), \underline{\pi}_r(k \mid k-1))}, \qquad r = 1,2,\ldots,N; \quad q = 1,2,\ldots,m.$$

Step 3 (Sensor Assignment): Select the best sensor assignment $\delta^*(k)$ that maximizes the sum of information gains in (14) using the JVC assignment algorithm.

Step 4 (Observation): The set of observations $\{y_1(k), y_2(k), \ldots, y_N(k)\}$ at time epoch k are obtained using the sensor set $u_r(k)$ $(r = 1,2,\ldots,N)$ based on the emission probability matrices given in (2).

Step 5 (State Update): Obtain the updated information state, $\pi_{ri}(k \mid k)$, by using the forward algorithm [22] as follows:

$$\pi_{ri}(k \mid k) = \frac{b_{rliq*}(k)\pi_{ri}(k \mid k-1)}{\sum_{j=1}^{n_r} b_{rljq*}(k)\pi_{rj}(k \mid k-1)} \tag{17}$$

where $\pi_{ri}(k \mid k)$ is the ith element of $\underline{\pi}_r(k \mid k-1) = \{\pi_{r1}(k \mid k-1), \ldots, \pi_{ri}(k \mid k-1), \ldots, \pi_{rn_r}(k \mid k-1)\}^T$ and the b_{rliq*} is the (l,i) element of emission matrix $B_{rq*}(k)$. It is the probability of the symbol $O_{rl}(k)$ $(l = 1,2,\ldots, L(q))$ when the sensor $u_r(k) = q*$ is assigned to the rth HMM, given the state $x_r(k) = s_{ri}$.

[2]The information gain heuristic is derived in terms of predictor-corrector form of discrete HMM equations. These are similar to the dynamic Bayesian state estimation equations, when the HMM states and observations are continuous. In the latter case, $\pi_r(k \mid k-1) \triangleq p(\underline{x}_r(k) \mid Y_r^{k-1}, U_r^{k-1})$ and $\pi_r(k \mid k) \triangleq p(\underline{x}_r(k) \mid Y_r^k, U_r^k)$ should be interpreted as probability density functions of system state. The predicted information state for the next time epoch k can be obtained as:

$$\pi_r(k \mid k-1) = \int_{\underline{x}_r(k-1)} p(\underline{x}_r(k) \mid \underline{x}_r(k-1))\pi_r(k-1 \mid k-1)d\underline{x}_r(k-1); \qquad 1 \le r \le N$$

Once the observation for selected sensor $u_r(k)$ is obtained, the updated information state is:

$$\pi_r(k \mid k) = \frac{p(\underline{y}_r(k) \mid \underline{x}_r(k), u_r(k) = q)}{\int_{\underline{x}_r(k)} p(\underline{y}_r(k) \mid \underline{x}_r(k), u_r(k) = q)\pi_r(k \mid k-1)d\underline{x}_r(k)}\pi_r(k \mid k-1); \qquad 1 \le r \le N$$

Typically, these integrals are intractable. However, if we can obtain analytical approximations to the above equations (e.g., Gaussian sum approximation), information gain heuristic would still be a tractable approach.

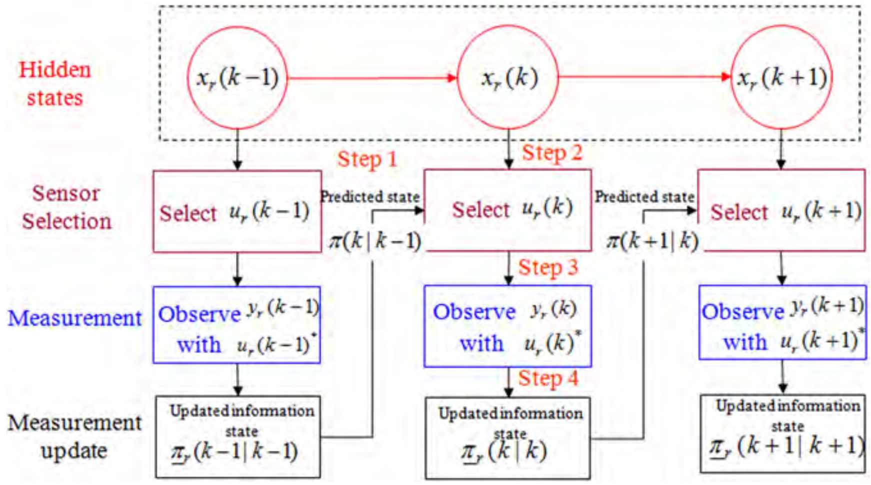

Fig. 3. Information gain heuristic (IG) processing steps for rth HMM.

4.2. Rollout Algorithms

Rollout algorithms are a class of suboptimal methods which are capable of improving the effectiveness of any given heuristic through sequential application. This is due to the policy improvement mechanism of the underlying policy iteration process [27]. They can be viewed as a single step of the classical policy iteration method, wherein we start from a given easily implementable and computationally tractable policy, and then try to improve on that policy using online learning and simulation. The attractive aspects of rollout algorithms are simplicity, broad applicability, and suitability for online implementation. The details of the rollout algorithms are provided in [5], [6], [27]. In our rollout framework, the information gain heuristic is used as a base policy where optimal cost-to-go function $J^*(\boldsymbol{\Pi}(k+1 \mid k+1))$ is approximated by the cost-to-go function $J(\boldsymbol{\Pi}(k+1 \mid k+1))$ of the information gain heuristic. The rollout policy for approximating (13) can be written in terms of Q-factor as follows:

$$
\begin{aligned}
\delta^*(k) &= \underset{\delta^i(k)\in\delta(k)}{\arg\min}\, Q(\boldsymbol{\Pi}(k \mid k-1), \delta^i(k)), \qquad i = 1,\ldots,\kappa \\
&= \underset{\delta^i(k)\in\delta(k)}{\arg\min}\, E\left[\sum_{r=1}^{N}[f_{rk}(\underline{\pi}_r(k \mid k)) \right. \\
&\qquad + g_{rk}(\psi_r^i(\boldsymbol{\Pi}(k \mid k-1)), \underline{\pi}_r(k \mid k-1))] \\
&\qquad \left. + J(\boldsymbol{\Pi}(k+1 \mid k+1))\right] \qquad (18)
\end{aligned}
$$

where $\delta(k) = \{\delta^1(k),\ldots,\delta^\kappa(k)\}$ are the κ-best assignments used to reduce the search space, and $\psi_r^i(\boldsymbol{\Pi}(k \mid k-1)) = u_r^i(k)$ is the sensor assigned to monitor the rth HMM in the ith-best assignment. The problem of computing the κ-best assignments is solved by combining the JVC algorithm with a modified Murty's algorithm [20], [10], [21]. However, the Q-factor driven by ith-best assignment $\delta^i(k)$ at time epoch k can not be computed in closed-form. A straightforward approach for computing the Q-factors is to use Monte Carlo simulations for $J(\boldsymbol{\Pi}(k+1 \mid k+1))$. Unfortunately, the method suffers from increase in computational complexity. In our paper, given information state vector $\boldsymbol{\Pi}(k \mid k)$ at time epoch k, we approximated $J(\boldsymbol{\Pi}(k+1 \mid k+1))$ by generating a single schedule trajectory computed from the information gain heuristic starting from $k+1$ to K. The rollout assignment is obtained by minimizing the approximated Q-factor in (18) from the κ-best assignments at time epoch k.

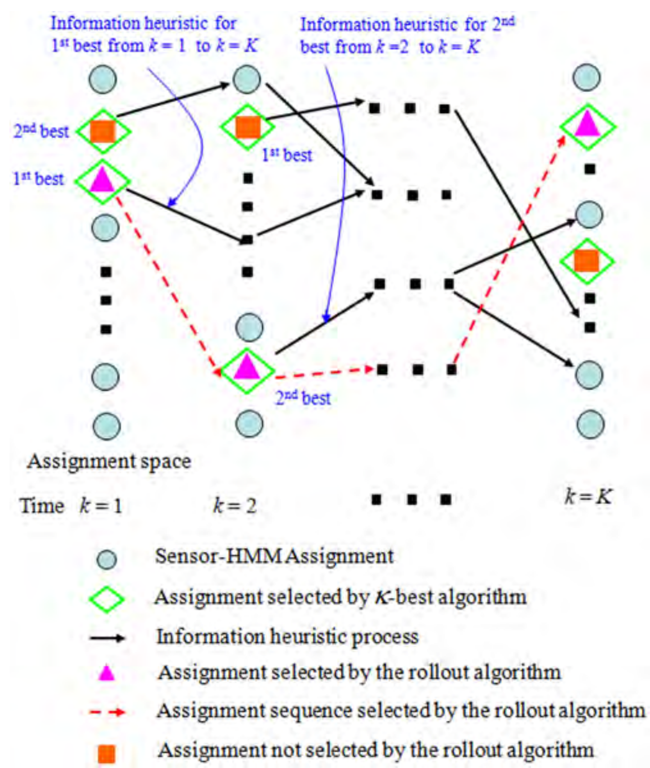

Fig. 4. Rollout information gain (RIG) algorithm coupled with κ-best assignment algorithm.

Fig. 4 graphically illustrates the RIG algorithm with two rollouts at each time epoch. The pseudo code of the

Rollout Information Gain Algorithm for Multiple HMMs:

Step 1: Predict the information state of each HMM using (16) as described in Section *3.1*.

Step 2: Generate the 2-D information gain matrix by computing the information gain per unit sensor cost for all sensor-HMM pairs at time k,

$$\frac{I_{qr}(\underline{\pi}_r(k|k-1),u_r(k)=q)}{g_{rk}(u_r(k),\underline{\pi}_r(k|k-1))}, \quad r=1,2,\ldots,N; q=1,2,\ldots,m.$$

Step 3: Find the κ-best sensor assignments, using the modified Murty's algorithm and the JVC algorithm. Usually, κ is in the range 2$\sim$5 to avoid excessive rollouts.

Step 4: Compute the approximated Q-factor in (18) for the κ-best sensor assignments. Select the sensor assignment which minimizes the approximated Q-factor in (18).

Step 5: If the cost has not improved over Z time epochs (Z in the range 3$\sim$5), stop. Else, update time epoch $k \rightarrow k+1$ and go back to *Step 1* until $k=K$.

Fig. 5. Pseudo code for the rollout information gain (RIG) algorithm.

Fig. 6. Notional area for scenario development.

RIG algorithm using the κ-best assignment algorithm is shown in Fig. 5.

5. COMPUTATIONAL RESULTS

5.1. A Hypothetical Mission Scenario

This scenario, motivated by ESG missions, involves simultaneous monitoring of multiple geographically dispersed threat activities. Here, an ISR officer needs to dynamically allocate sensors to monitor threats in a notional area (e.g., fishing villages, refugee camps) that involves primarily two fictitious countries, Asiland and Bartola [14]. Asiland is an unstable state, where maritime smugglers and anti-western terrorist groups have supported the insurgent factions hostile to the government of Bartola. Local terrorists and sea rovers use Asiland as a base. The scenario considers that nearly a month ago, the northern shore of Asiland was struck by a tsunami that destroyed several fishing villages and caused enormous casualties. Large numbers of Asiland citizens sought refuge in the south for help and assistance. However, this exodus quickly drained the resources of Asiland. Consequently, many Asiland refugees began to move to fishing villages and refugee camps in Bartola. Within a few days, insurgents and terrorist factions in and around Asiland began to exploit the situation, infiltrating their operations into Bartola by disguising as refugees and smuggling weapons onboard fishing boats and merchant ships. Bartola's military was overwhelmed by the massive influx of refugee boats, as well as tracking the terrorist/insurgent's activities using these boats and ships for illegal transfers. The government of Bartola sought help from the United States to provide Humanitarian Assistance/Disaster Relief (HA/DR) to Bartola and the organizations operating relief activities within it. The ESG sensor assets are deployed and begin to monitor strategically significant areas (e.g., major sea and air lanes as well as several major ports, villages, refugee camps, roads, and cities/sites) as shown in Fig. 6.

5.2. Single HMM Scenario: Monitoring a Fishing Village

We consider a scenario where an ISR officer needs to dynamically allocate sensors to monitor asymmet-

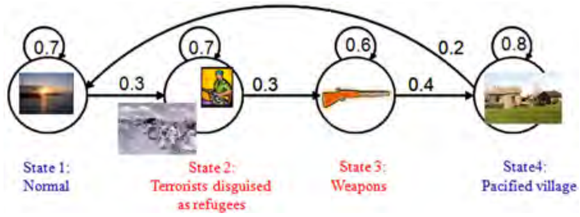

Fig. 7. A fishing village scenario for sensor scheduling problem.

ric threat activities in the notional area. The problem of monitoring the presence of terrorists and weapons in a fishing village is modeled using a four-state HMM. Activities such as the presence of terrorists and weapons, and ascertaining the crowd demeanor (normal or protesters or terrorists) are continually monitored using six sensors; labeled 1 through 6. As shown in Fig. 7, *State* 1 represents the normal state of the crowd in the fishing village. In *State* 2, refugees move into the village. Terrorists disguised as refugees arrive in the village and they also smuggle weapons into the village; this is modeled as *State* 3. In *State* 4, the weapons and terrorists are prosecuted/pacified and the village resumes normalcy, which is modeled by a transition to *State* 1. We specify the transition probability matrix, $A(k)$, based on state transitions and time spent in each state. In a discrete HMM model, the probability of staying in state j for a duration d is given by $p(d) = (a_{jj})^{d-1}(1 - a_{jj})$, where the expected duration is obtained from the following equations:

$$E[d] = \sum_{d=1}^{\infty} d(a_{jj})^{d-1}(1 - a_{jj}) = \frac{1}{(1 - a_{jj})}. \quad (19)$$

The self transition probability of state j is set by substituting $E[d]$ in (19) with the duration provided by the scenario. The state transition probabilities depend on how many links exist from state j to other states. Suppose that state j has n state transitions and state i is linked to state j, then the probability a_{ji} is assumed to be given by $a_{ji} = (1 - a_{jj})/n$. For simulations, we set the weighted priority vector $\underline{w}(u(k))$ in (11) to be a vector of constants and β is set to 10 in (5). The velocities of the six sensors are set as $\underline{v}^T(u(k)) = [300, 200, 300, 100, 450, 80]$. The sensor usage costs in (10) are selected as $\underline{c}_k(u(k)) = \{11, 4, 8, 5, 6, 2\}$, where for simplicity, each sensor usage cost vector is assumed to be independent of state. We used $\kappa = 6$ for the RIG algorithm. We assume that the initial probability distribution is known. The emission matrices are set by considering the sensing capability of each sensor, which are modeled by the probability of detection of the true states of each HMM. In this simulation, we assume that the observation capabilitities of a sensor decreases as the sensor label increases and each sensor provides one of four observation symbols at each time epoch k. The planning horizon $K = 15$ is set by considering the sum of expected durations as given in (19). Fig. 8 shows the total scheduling cost averaged over 1000 Monte Carlo runs. To assess the robustness of the algorithm, the cost was obtained by varying the observation probability using the variable p. Here, *Sensor* 1 or *Sensor* 5 curves indicate that a static schedule that employs *Sensor* 1 or *Sensor* 5 for all time epochs is substantially worse than a dynamic schedule. The rollout information gain algorithm (RIG)-based sensor schedule has approximately 5–18% lower cost as compared to one using only the information gain heuristic (IG); it also has $\approx 49\%$ lower cost as compared to a static schedule that employs *Sensor* 5 throughout. The dual advantages of using the RIG algorithm are that it significantly reduces the complexity of dynamic programming, while improving the accuracy over the base heuristic, viz., the information gain heuristic (IG) algorithm.

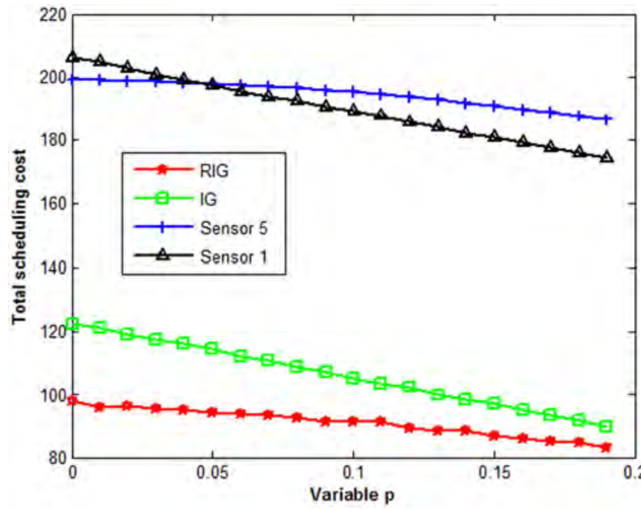

Fig. 8. Variation of the total scheduling cost with sensor accuracy variable p.

5.3 Multiple HMM Scenario: Monitoring Multiple Fishing Villages and Refugee Camps

In this scenario, we solve the problem of monitoring multiple fishing villages (FVs) and refugee camps (RCs) using multiple sensors. The problem of monitoring for the presence of refugees, weapons, and learning the crowd demeanor (normal or protesters or agitators or terrorists) in FVs and RCs is modeled using 17 HMMs, where their states are represented in a vector form (e.g., (refugees, weapons, crowd)), as shown in Fig. 9. For example, $(1, 1, 3)$ corresponds to the 16th state that indicates that refugees, weapons, and terrorists are present. Threat activities are continually monitored using a total of 17 sensors, which are comprised of 9 different types, as described in Table I. The states of 10 FVs and 7 RCs change dynamically by the departure and entry of refugees, and terrorists/insurgents (disguised as refugees). Fig. 10 shows the state transitions considered in this scenario.

The schedule cost considered for this scenario is given in (5), where sensor usage costs, $\underline{c}_k(u_r(k))$, are

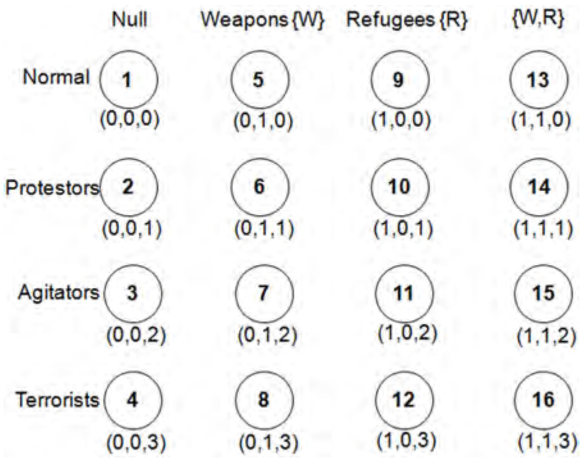

Fig. 9. States used for modeling multiple HMM scenario.

set by evaluating a weighted sum of unit price of the sensor and the crew required to operate the sensor. We employed the state estimation error criterion, given in (7). The cost of moving the sensors, $c_m(u_r(k))$, is set based on the actual mobility of the sensors and the distance from the task, as listed in Table I. The emission matrices are set by considering the sensing capability of each sensor as well as allocation preferences and geometrical constraints of sensing operations. If a sensor, $u_r(k) = q$, is irrelevant to monitoring a HMM (say, rth HMM), the entries of the emission matrix, given in (2), are set to uniform values, i.e., the emission matrix is a doubly stochastic matrix. We assume that each sensor provides one of the 16 observation symbols from the scenario. We specify the transition probability matrices, using the same process as that used in the single HMM case. However, since the scenario does not provide information on all the self-transition probabilities, the undefined self-transition probabilities are set to reasonable values and the remaining transition probabilities in the same row are uniformly distributed. For example, the transition matrix of Glorisabay, A_7, is set as shown in Table II, based on the state transition sequence of Glorisabay as shown in Fig. 10.

Fig. 11 shows the assignment distributions over 1500 mission scenarios, each averaged over 50 Monte Carlo runs. The value of β is set as 40 in (5) and the values of priority vector are set to a constant vector as in the single HMM case. The assignment distributions of RIG algorithm are obtained using $\kappa = 2$-best assignments. We assume that the initial probability distribution is known. The planning horizon $K = 30$ is set by considering the sum of expected durations as given in (19). Table III shows the state transitions and RIG-based sensor assignments in Glorisabay FV. Note assignments of *Sensor* 8 for RCs and *Sensor* 1 for FVs. The assignments are reasonable because *Sensor* 8 has difficulty in sensing the operations in RCs (e.g., patrol ships) and an ISR officer always assigns *Sensor* 1 to RCs as a first priority. To model the sensing constraints of *Sensor* 1 and *Sensor* 8 in the extreme case, the *Sensor* 1 emission matrix probabilities for FVs and the *Sensor* 8 emission matrix entries for RCs are distributed uniformly. The realization of assignment constraints by the setup of emission matrix is shown in Fig. 11, where columns correspond to sensors and rows correspond to HMMs. The brightness represents the number of assignments of HMM-sensor pair. Fig. 12 shows the total scheduling cost. The cost of assignment by distance was obtained using sensor assignments to minimize the sum

TABLE I
Setup of Usage Cost and Sensing Capability

		Sensor 1	Sensor 2	Sensor 3	Sensor 4	Sensor 5	Sensor 6	Sensor 7	Sensor 8	Sensor 9
	Crew	5	0	20	2	10	1	20	5	15
	Unit Price	51	13	0	12	36	22	40	26	40
Usage Cost	0.2 X Cr +0.16XUP	11	2	10	3	11	4	17	7	14
Mobility	Speed (kts)	333	49	30	180	330	547	30	35	30
Sensing Capability		Long range radar	Radar Camera	Human	Short Range Radar	Radar Sonar Buoy	Mid-range radar	Long Range Radar	Short range Radar	Mid-range radar
	Range	9	3	1	3	6	6	9	3	6
	Resolution	9	9	6	5	4	3	2	2	2
Sensing Capability	0.1 X Ra +0.9 X Re	9	8	6	5	4	3	3	2	2
Quantity		3	1	3	2	2	3	2	1	1

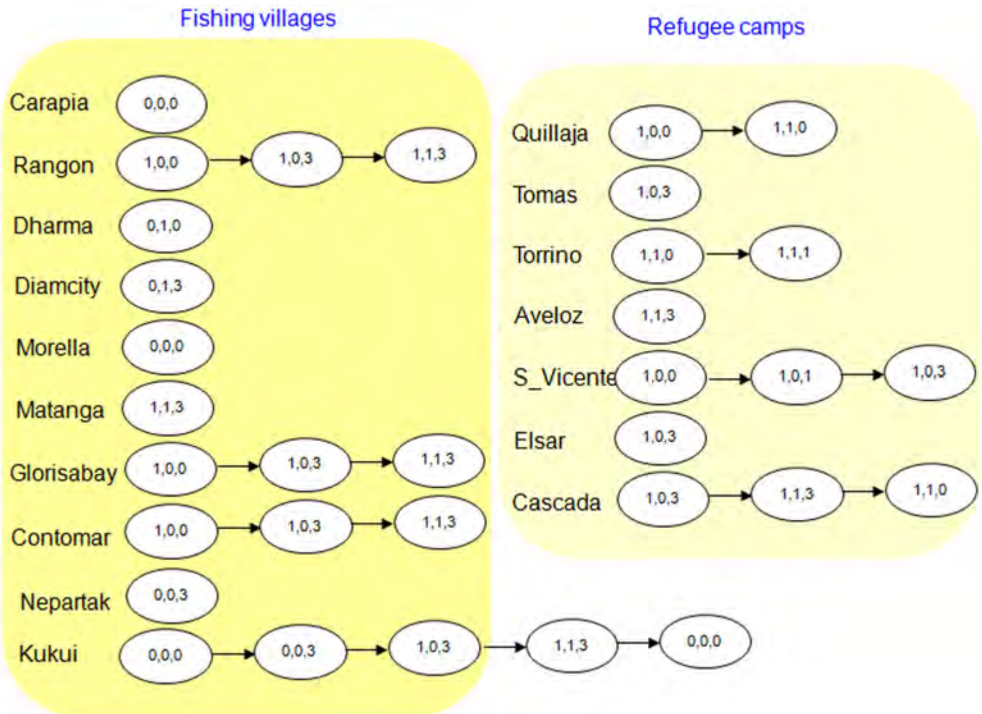

Fig. 10. States transitions of multiple fishing villages and refugee camps.

TABLE II
Transition Matrix A_7 Set Up of a Fishing Village Glorisabay

0.8	.013	.013	.013	.013	.013	.013	.013	.013	.013	.013	.013	.013	.013	.013	.013
.013	0.8	.013	.013	.013	.013	.013	.013	.013	.013	.013	.013	.013	.013	.013	.013
.013	.013	0.8	.013	.013	.013	.013	.013	.013	.013	.013	.013	.013	.013	.013	.013
.013	.013	.013	0.8	.013	.013	.013	.013	.013	.013	.013	.013	.013	.013	.013	.013
.013	.013	.013	.013	0.8	.013	.013	.013	.013	.013	.013	.013	.013	.013	.013	.013
.013	.013	.013	.013	.013	0.8	.013	.013	.013	.013	.013	.013	.013	.013	.013	.013
.013	.013	.013	.013	.013	.013	0.8	.013	.013	.013	.013	.013	.013	.013	.013	.013
.013	.013	.013	.013	.013	.013	.013	0.8	.013	.013	.013	.013	.013	.013	.013	.013
0	0	0	0	0	0	0	0	.89	0	0	.11	0	0	0	0
.013	.013	.013	.013	.013	.013	.013	.013	.013	0.8	.013	.013	.013	.013	.013	.013
.013	.013	.013	.013	.013	.013	.013	.013	.013	.013	0.8	.013	.013	.013	.013	.013
0	0	0	0	0	0	0	0	0	0	0	.89	0	0	0	.11
.013	.013	.013	.013	.013	.013	.013	.013	.013	.013	.013	.013	0.8	.013	.013	.013
.013	.013	.013	.013	.013	.013	.013	.013	.013	.013	.013	.013	.013	0.8	.013	.013
.013	.013	.013	.013	.013	.013	.013	.013	.013	.013	.013	.013	.013	.013	0.8	.013
.000 6	.000 6	.000 6	.000 6	.000 6	.000 6	.000 6	.000 6	.000 6	.000 6	.000 6	.000 6	.000 6	.000 6	.000 6	.99

of travelled distances. Sensor scheduling via the RIG algorithm ($\kappa = 7$) has approximately $\approx 2.1\%$ lower cost as compared to one using only the information gain heuristic (IG) and $\approx 4\%$ lower cost as compared to scheduling by distance. The results also suggest that, while the RIG algorithm in multiple HMM sensor scheduling problem improves the performance of information gain heuristic, it is less effectiveness when compared to a single

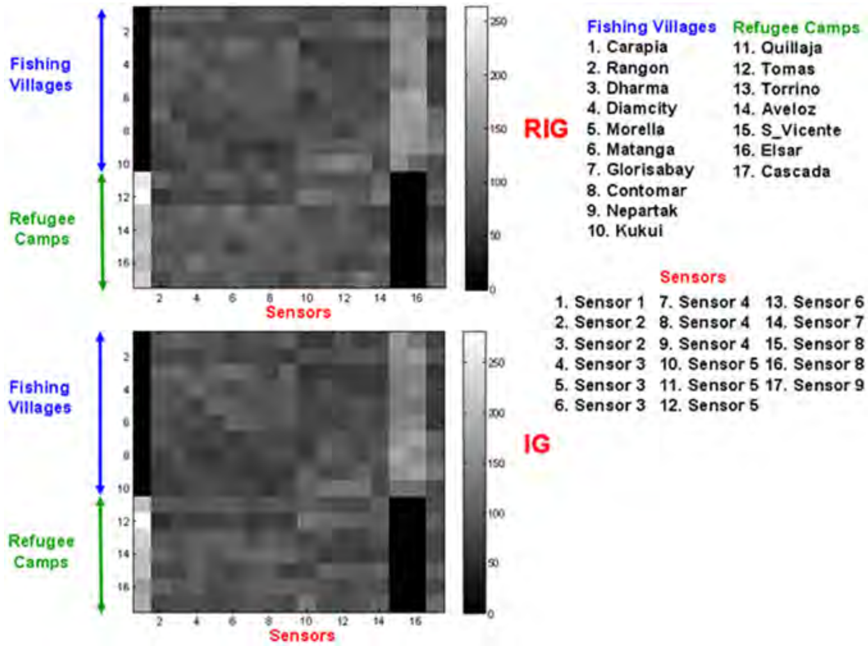

Fig. 11. Assignment distribution of IG ($\kappa = 1$) and RIG ($\kappa = 2$).

TABLE III
State Transition and RIG Sensor Assignment in Fishing Village Glorisabay

k	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15
$x_7(k)$	9	9	9	9	9	9	9	9	9	9	9	9	9	12	12
$u_7(k)$	9	17	13	7	14	6	2	17	12	4	4	11	14	8	2
k	16	17	18	19	20	21	22	23	24	25	26	27	28	29	30
$x_7(k)$	12	12	12	12	12	12	12	16	16	16	16	16	16	16	16
$u_7(k)$	8	11	5	10	16	9	1	5	10	12	7	16	3	12	17

HMM sensor scheduling problem. This is due to the assignment of all available sensors to multiple HMMs. In addition, the differences in information gains, obtained from the κ-best assignments, are much less than those obtained from the κ-best sensors in the single HMM case. However, the fact that RIG and IG have nearly identical performance gives us confidence that the IG-based sensor schedules are near-optimal.

6. CONCLUSION

This paper formulated the sensor scheduling problem using HMM formalisms. The optimal solution of the sensor scheduling problem via dynamic programming (DP) is intractable for both single and multiple HMM scheduling problems due to computational explosion caused by the curse of dimensionality. To overcome this, we proposed a RIG algorithm by combining rollout concepts with the JVC and the κ-best assignment algo-

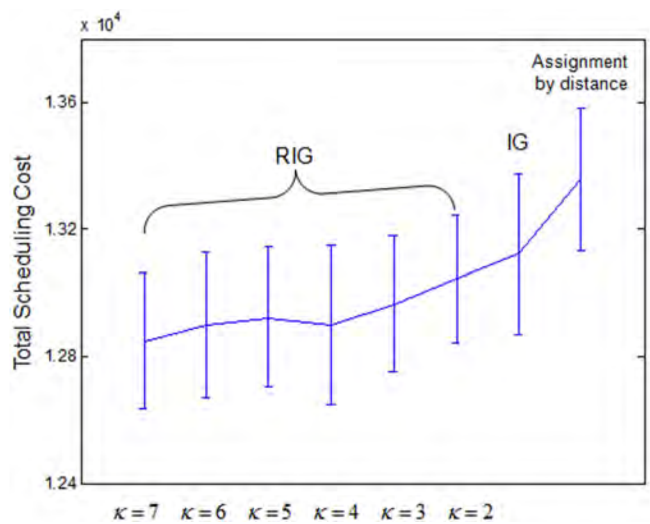

Fig. 12. Total scheduling cost of RIG, IG, and assignment by distance.

rithms. We illustrated its application on realistic mission scenarios involving the monitoring of threat activities in a fishing village modeled using a single HMM, and multiple fishing villages and refugee camps modeled using multiple HMMs.

Our work on HMM-based dynamic sensor scheduling model assumed that the tasks are independent. In practice, however, tasks may exhibit dependencies or may have a hierarchical structure. We plan to develop extensions to HMM-based sensor scheduling model to handle dependencies and hierarchical structure among tasks using coupled HMMs [7] and hierarchically structured HMMs [12]. In addition, we assumed that at most one sensor is assigned to a HMM at each time epoch. However, due to the various assignment constraints, imposed by organizational structure, this assumption may need to be relaxed. Finally, sensor scheduling is a cooperative process among multiple decision makers. Auction-based algorithms may provide a mechanism for implementing distributed and coordinated sensor scheduling algorithms. We plan to pursue these extensions in the future.

APPENDIX

Let $H(x) = -\sum_i p_i(x)\log_2 p_i(x)$ denote the entropy of the state with a probability mass function $\{p_i(x)\}$. Let us derive the information gain defined in (14) obtained by assigning a sensor $u_r(k) = q$ to rth HMM, where the information state is defined in (15). The entropy of the information state is:

$$H(\underline{\pi}_r(k \mid k-1)) = -\sum_{i=1}^{n_r} \pi_{ri}(k \mid k-1)\log_2 \pi_{ri}(k \mid k-1). \tag{20}$$

Recall that the joint entropy is given by:

$$H(x,u) = H(x) + H(u \mid x) = H(u) + H(x \mid u) \tag{21}$$

and mutual information (or information gain) is:

$$I(x,u) = H(x) - H(x \mid u). \tag{22}$$

$$= H(u) - H(u \mid x). \tag{23}$$

Using (21) and (22),

$$I_{qr}(\underline{\pi}_r(k \mid k-1), u_r(k) = q) = H(\underline{\pi}_r(k \mid k-1)) - H(\underline{\pi}_r(k \mid k-1) \mid u_r(k) = q), \tag{24}$$

$$I_{qr}(\underline{\pi}_r(k \mid k-1), u_r(k) = q) = H(u_r(k) = q) - H(u_r(k) = q \mid \underline{\pi}_r(k \mid k-1)) \tag{25}$$

where

$$\begin{aligned} H(u_r(k) = q) &= -\sum_{l=1}^{L(q)} P(y_r(k) = O_{rl}(k) \mid Y_r^{k-1}, U_r^{k-1}, u_r(k) = q) \\ &\quad \cdot \log_2 P(y_r(k) = O_{rl}(k) \mid Y_r^{k-1}, U_r^{k-1}, u_r(k) = q) \end{aligned} \tag{26}$$

and

$$\begin{aligned} P\left(y_r(k) = O_{rl}(k) \mid Y_r^{k-1}, U_r^{k-1}, u_r(k) = q\right) &= \sum_{i=1}^{n_r} P(y_r(k) = O_{rl}(k), x_r(k) = s_{ri} \mid Y_r^{k-1}, U_r^{k-1}, u_r(k) = q) \\ &= \sum_{i=1}^{n_r} P(y_r(k) = O_{rl}(k) \mid x_r(k) = s_{ri}, u_r(k) = q)\pi_{ri}(k \mid k-1) \\ &= \sum_{i=1}^{n_r} b_{rliq}(k)\pi_{ri}(k \mid k-1). \end{aligned} \tag{27}$$

The conditional entropy of a random variable X, conditioned on a random variable Y

$$H(X \mid Y) = \sum_y p_Y(y) H(X \mid Y = y). \tag{28}$$

Using (26):

$$H(u_r(k) = q \mid \underline{\pi}_r(k \mid k-1)) = -\sum_{i=1}^{n_r} \pi_{ri}(k \mid k-1)\left[\sum_{l=1}^{L(q)} b_{rliq}(k)\log_2 b_{rliq}(k)\right]. \tag{29}$$

Using (24) and (27), the information gain is:

$$\begin{aligned} I_{qr}(\underline{\pi}_r(k \mid k-1), u_r(k) = q) &= \sum_{i=1}^{n_r} \pi_{ri}(k \mid k-1)\sum_{l=1}^{L(q)} b_{rliq}(k)\log_2 b_{rliq}(k) \\ &\quad - \sum_{l=1}^{L(q)}\left(\sum_{i=1}^{n_r} b_{rliq}(k)\pi_{ri}(k \mid k-1)\right) \\ &\quad \cdot \log_2\left(\sum_{i=1}^{n_r} b_{rliq}(k)\pi_{ri}(k \mid k-1)\right). \end{aligned} \tag{30}$$

We can also derive the information gain using (23). In this case,

$$H(\underline{\pi}_r(k \mid k-1)) = -\sum_{i=1}^{n_r} \pi_{ri}(k \mid k-1)\log_2 \pi_{ri}(k \mid k-1), \tag{31}$$

$$H(\underline{\pi}_r(k \mid k-1) \mid u_r(k) = q)$$
$$= -\sum_{l=1}^{L(q)} P(y_r(k) = O_{rl}(k) \mid Y_r^{k-1}, u_r(k) = q)$$
$$\cdot \sum_{i=1}^{n} P(x_r(k) = s_{ri} \mid Y_r^{k-1}, u_r(k) = q, y_r(k) = O_{rl}(k))$$
$$\cdot \log_2 P(x_r(k) = s_{ri} \mid Y_r^{k-1}, u_r(k) = q, y_r(k) = O_{rl}(k)). \tag{32}$$

Using the forward algorithm [22],

$$P(x_r(k) = s_{ri} \mid Y_r^{k-1}, u_r(k) = q, y_r(k) = O_{rl}(k)) = \frac{b_{rliq}(k)\pi_{ri}(k \mid k-1)}{\sum_{j=1}^{n_r} b_{rljq}(k)\pi_{rj}(k \mid k-1)} \tag{33}$$

and

$$P(y_r(k) = O_{rl}(k) \mid Y_r^{k-1}, u_r(k) = q) = \sum_{j=1}^{n_r} b_{rljq}(k)\pi_{rj}(k \mid k-1). \tag{34}$$

Inserting (31) and (32) in (30), we get:

$$H(\underline{\pi}_r(k \mid k-1) \mid u_r(k) = q)$$
$$= -\sum_{l=1}^{L(q)} \sum_{i=1}^{n_r} b_{rliq}(k)\pi_{ri}(k \mid k-1) \cdot \log_2 \frac{b_{rliq}(k)\pi_{ri}(k \mid k-1)}{\sum_{j=1}^{n_r} b_{rljq}(k)\pi_{rj}(k \mid k-1)}$$
$$= -\sum_{i=1}^{n_r} \pi_{ri}(k \mid k-1) \sum_{l=1}^{L(q)} b_{rliq}(k) \log_2 b_{rliq}(k)$$
$$- \sum_{i=1}^{n_r} \pi_{ri}(k \mid k-1) \log_2 \pi_{ri}(k \mid k-1)$$
$$+ \sum_{l=1}^{L(q)} \left(\sum_{i=1}^{n_r} b_{rliq}(k)\pi_{ri}(k \mid k-1) \right) \cdot \log_2 \left(\sum_{i=1}^{n_r} b_{rliq}(k)\pi_{ri}(k \mid k-1) \right). \tag{35}$$

Since $\sum_{l=1}^{L(q)} b_{rliq}(k) = 1$ for $i = 1, 2, \ldots, n_r$, combining (35) and (31) indeed gives (30).

REFERENCES

[1] L. E. Baum, T. Petrie, G. Soules and N. Weiss
A maximization technique occurring in the statistical analysis of probabilistic functions of Markov chains.
Ann. of Math. Stat, **41** (1970), 164–171.

[2] D. P. Bertsekas
An auction algorithm for shortest paths.
SIAM Journal on Optimization, **1** (1991), 425–447.

[3] D. P. Bertsekas
Auction algorithms for network flow problems: A tutorial introduction.
Computational Optimization and Application, **1** (1992), 7–66.

[4] D. P. Bertsekas
Network optimization: Continuous and discrete models.
Athena Scientific, Belmont MA, 1998.

[5] D. P. Bertsekas and D. A. Castanon
Rollout algorithms for stochastic scheduling problems.
Journal of Heuristics, **5** (1999), 89–108.

[6] D. P. Bertsekas, J. N. Tsitsiklis and C. Wu
Rollout algorithms for combinatorial optimization.
Journal of Heuristics, **3** (1997), 245–262.

[7] M. Brand, N. Oliver and A. Pentland
Coupled hidden Markov models for complex action recognition.
In *Proceedings of IEEE Conference on Computer Vision and Pattern Recognition*, San Juan, Puerto Rico, 1997, 994–999.

[8] S. Chhetri
Sensor scheduling and efficient algorithm implementation for target tracking.
Ph. D. Dissertation, Arizona State University, 2006.

[9] M. Chu, H. Haussecker and F. Zhao
Scalable information-driven sensor querying and routing for Ad Hoc heterogeneous sensor networks.
International Journal of High Performance Computing Applications, **16** (2002), 293–313.

[10] I. Cox and M. Miller
On finding ranked assignments with application to multitarget tracking and motion correspondence.
IEEE Transactions on Aerospace and Electronic Systems, **31** (1995), 486–489.

[11] E. W. Dijkstra
A note on two problems in connexion with graphs.
Numerische Mathematik, **1** (1959), 296–271.

[12] S. Fine, Y. Singer and N. Tishby
The hierarchical hidden Markov model: Analysis and applications.
Machine Learning, **32** (1997), 41–62.

[13] Z. Ghahramani and M. I. Jordan
Factorial hidden Markov models.
Machine Learning, **29** (1997), 245–273.

[14] S. G. Hutchins, S. A. Weil, D. L. Kleinman, S. P. Hocevar, W. G. Kemple, K. Pfeiffer, D. Kennedy, H. Oonk, G. Averett and E. E. Entin
An investigation of ISR coordination and information presentation strategies to support expeditionary strike groups.
12th ICCRTS, Newport, RI, 2007.

[15] R. Jonker and A. Volgenant
A shortest augmenting path algorithm for dense and sparse linear assignment problems.
Computing, **38** (1987), 325–340.

[16] V. Krishnamurthy
Algorithms for optimal scheduling and management of hidden Markov model sensors.
IEEE Transactions on Signal Processing, **50** (2002), 1382–1397.

[17] A. Logothetis and A. Isaksson
On sensor scheduling via information theoretic criteria.
IEEE American Control Conference, San Diego, CA, vol. 4, 1999, 2402–2406.

[18] L. Meier, J. Perschon and R. M. Dressier
Optimal control of measurement subsystems.
IEEE Transactions on Automatic Control, **12** (1967), 528–536.

[19] M. L. Miller, H. S. Stone and I. J. Cox
Optimizing Murty's ranked assignment method.
IEEE Transactions on Aerospace and Electronic Systems, **33** (1997), 851–862.

[20] K. G. Murty
Algorithm for ranking all the assignments in order of increasing cost.
Operations Research, **16** (1968), 682–687.

[21] R. L. Popp, K. R. Pattipati and Y. Bar-Shalom
Dynamically adaptable M-Best 2D assignment and multi-level parallelization.
IEEE Transactions on Aerospace and Electronic Systems, **35** (1999), 1145–1160.

[22] L. R. Rabiner
A tutorial on hidden Markov models and selected applications in speech recognition.
Proceedings of the IEEE, **77** (1989), 257–286.

[23] M. Schneider and C. Chong
A rollout algorithm to coordinate multiple sensor resources to track and discriminate targets.
In *Proceeding of SPIE*, vol. 6235, Orlando, FL, 2006, 62350E.

[24] M. Shakeri, K. R. Pattipati and D. L. Kleinman
Optimal measurement scheduling for state estimation.
IEEE Transactions on Aerospace and Electronic System, **31** (1995), 716–729.

[25] S. Singh, W. Donat, H. Tu, K. Pattipati and P. Willett
Anomaly detection via feature-aided tracking and hidden Markov models.
IEEE Aerospace Conference, Big Sky, MT, 2007, 1–18.

[26] S. S. Singh, N. Kantas, B. Vo, A. Doucet and R. J. Evans
Simulation based optimal sensor scheduling with application to observer trajectory planning.
Technical Report, Cambridge University, 2005.

[27] F. Tu and K. R. Pattipati
Rollout strategies for sequential fault diagnosis.
IEEE Transactions of System, Man and Cybernetics, Part A, **33** (2003), 86–99.

[28] He. Ying and K. P. Chong
Sensor scheduling for target tracking in sensor networks.
IEEE Conference on Decision and Control, **1** (2004), 743–748.

[29] F. Zhao, J. Shin and J. Reich
Information-driven dynamic sensor collaboration.
IEEE Signal Processing Magazine, **19** (2002), 61–72.

Woosun An is currently a Ph.D. student in electrical and computer engineering at the University of Connecticut (UConn). He received his B.S. degree in electrical engineering from the University of Chung Ang, South Korea in 2001. He received his M.S. degree in electrical engineering from the University of Florida in 2005.

He received UConn engineering summer fellowship award in 2007 and 2008. His research interests include dynamic sensor scheduling, graphical models, and optimization theory.

Satnam Singh is currently a Ph.D. student in electrical and computer engineering at the University of Connecticut (UConn). He received his B.S. degree in chemical engineering from the Indian Institute of Technology (IIT) Rookree, India in 1996. He received his M.S. degree in electrical engineering from the University of Wyoming in 2003.

Prior to his graduate studies, he worked as a design engineer in Engineers India Limited, New Delhi, India for 4 years. He is a reviewer for IEEE Potentials magazine for the past three years and he has also served as the President of IEEE Student Branch at UConn during 2003–04. He received UConn engineering summer fellowship award in 2006. His research interests include data mining, graphical models, pattern recognition, fault detection and diagnosis, and optimization theory.

Krishna R. Pattipati (S'77—M'80—SM'91—F'95) received the B.Tech. degree in electrical engineering with highest honors from the Indian Institute of Technology, Kharagpur, in 1975, and the M.S. and Ph.D. degrees in systems engineering from the University of Connecticut in 1977 and 1980, respectively.

From 1980–1986 he was employed by ALPHATECH, Inc., Burlington, MA. Since 1986, he has been with the University of Connecticut, where he is a Professor of Electrical and Computer Engineering. His current research interests are in the areas of adaptive organizations for dynamic and uncertain environments, multi-user detection in wireless communications, signal processing and diagnosis techniques for complex system monitoring, and multi-object tracking. Dr. Pattipati has published over 370 articles, primarily in the application of systems theory and optimization (continuous and discrete) techniques to large-scale systems. He has served as a consultant to Alphatech, Inc., Aptima, Inc., and IBM Research and Development. He is a cofounder of Qualtech Systems, Inc., a small business specializing in intelligent diagnostic software tools.

Dr. Pattipati was selected by the IEEE Systems, Man, and Cybernetics Society as the Outstanding Young Engineer of 1984, and received the Centennial Key to the Future award. He was elected a Fellow of the IEEE in 1995 for his contributions to discrete-optimization algorithms for large-scale systems and team decision making. Dr. Pattipati has served as the Editor-in-Chief of the IEEE Transactions on SMC: Part B—Cybernetics during 1998–2001, Vice-President for Technical Activities of the IEEE SMC Society (1998–1999), and as Vice-President for Conferences and Meetings of the IEEE SMC Society (2000–2001). He was corecipient of the Andrew P. Sage award for the Best SMC Transactions Paper for 1999, Barry Carlton award for the Best AES Transactions Paper for 2000, the 2002 and 2008 NASA Space Act Awards for "A Comprehensive Toolset for Model-based Health Monitoring and Diagnosis," the 2003 AAUP Research Excellence Award and the 2005 School of Engineering Teaching Excellence Award at the University of Connecticut. He also won the best technical paper awards at the 1985, 1990, 1994, 2002, 2004 and 2005 IEEE AUTOTEST Conferences, and at the 1997 and 2004 Command and Control Conferences.

David L. Kleinman (S'62—M'68—SM'93—F'94) received his B.E.E. degree from The Cooper Union, New York, in 1962 followed by M.S. and Ph.D. degrees in electrical engineering (control systems) from MIT in 1963 and 1967, respectively.

He works part-time as a research professor in the Information Sciences Department at the Naval Postgraduate School in Monterey, CA, and is also professor emeritus in the Electrical and Computer Engineering Department at the University of Connecticut. His current research is in the areas of team decision-making/coordination in dynamic task environments, and structural adaptation processes in human organizations. From 1967 to 1971, he worked at Bolt Beranek and Newman, Cambridge, MA where he pioneered in the application of modern control and estimation theory to develop and validate an analytical model for describing human control and information processing performance in manned vehicle systems. From 1971–73 he directed the East Coast office of Systems Control, Inc., where he led applied research projects in both manual and automatic control. From 1973–1994, while a Professor at UConn, he directed the CYBERLAB, a laboratory for empirical research in cybernetic systems. In 1993 Dr. Kleinman was named an IEEE Fellow for his work in human modeling. He has authored or coauthored over 200 journal and conference papers, many of which are in the area of military Command and Control.

Dr. Kleinman has been a Vice-President of Finance and Long-range Planning, for the IEEE-SMC Society. He was the Program Chairman for the 1989 IEEE International Conference on Systems, Man, and Cybernetics, and for the 1990 JDL Symposium on Command and Control Research.

Dr. Kleinman was a cofounder of Alphatech, Inc., Qualtech Systems, Inc., and Aptima, Inc. He continues to serve on the Board of Directors of the latter two companies.

Swapna S. Gokhale received her B.E. (Hons.) in electrical and electronics engineering, and computer science from the Birla Institute of Technology and Science, Pilani, India in 1994, and her M.S. and Ph.D. in electrical and computer engineering from Duke University in 1996 and 1998 respectively.

She is currently an assistant professor in the Department of Computer Science and Engineering at the University of Connecticut. Prior to joining UConn, she was a research scientist at Telcordia Technologies (formerly Bell Communications Research). Her dissertation research was in the area of software reliability and performance. Prior to joining Telcordia, she was a Post Graduate Researcher at the University of California, Riverside. Her research interests include software testing and reliability, Quality of Service issues in wireless and wireline networks and Voice-over-IP. She received the Best Paper awards at Intl. Symposium on Software Reliability Engineering (ISSRE) in 1998 and at Intl. Conference on Reliability and Safety Engineering (INCRESE) in 2005. She received a CAREER award from the National Science Foundation (NSF) to conduct research in the area of architecture-based software reliability assessment in January 2007.

Correction of Selection Bias in Traffic Data by Bayesian Network Data Fusion

MAREK JUNGHANS
HANS-JOACHIM JENTSCHEL

In this paper a method is introduced based on the concept of Bayesian Networks (BNs), which is applied to model sensor fusion. Sensors can be characterised as real time variant systems with specific physical functional principles, allowing to determine the value of a physical state of interest within certain ranges of tolerance. The measurements of the sensors are affected by external, e.g. environmental conditions, and internal conditions, e.g. the physical life of the sensor and its components. These effects can cause selection bias, which yields corrupted data. For this reason, the underlying process, the measurements, the external and internal conditions are considered in the BN model for data fusion. The effectiveness of the approach is underlined on the basis of vehicle classification in traffic surveillance. The results of our simulations show, that the accuracy of the estimates of the vehicle classes is increased by more than 60%.

Manuscript received October 5, 2007; revised August 24, 2008; released for publication November 1, 2008.

Refereeing of this contribution was handled by Professor David Hall.

Authors' addresses: Institut für Verkehrstelematik, Fakultät für Verkehrswissenschaften "Friedrich List," Technische Universität Dresden, 01062 Dresden, Germany, Email: (marek.junghans@tu-dresden.de, hans-joachim.jentschel@tu-dresden.de).

1. INTRODUCTION

Bayesian Data Fusion (BDF) is a well-established method in decision-level fusion to increase the quality of measured data of several equal or different sensors, e.g. [7], [13]. Although the method is powerful, the results of the fusion process are only (1) as good as the sensors are; (2) as good as the a priori knowledge about the sensors is and (3) as good as the a priori knowledge about the underlying process is. For instance, in case of vehicle classification for traffic surveillance by several more or less accurate sensors (item 1), accurate relative frequencies of correct and wrong classifications (phantom detections, incorrect classified vehicles) are required to achieve beneficial fusion results (item 2). This statement is supplemented by an adequate characterisation and quantification of the underlying unknown traffic process (item 3).

For an adequate traffic management, there is a particular need for highly accurate traffic data, measured by accurate and reliable sensors, yielding a high degree of acceptance and credibility concerning the significance of the measured traffic parameters. There are a lot of different sensor technologies with different physical functional principles, different performance, problems and thus, differing operational areas [18], [19]. Two currently important coexisting sensor technologies are for instance the inductive loop detectors and video sensors. Loop detectors measure the traffic process temporally, while video sensors enable temporal and wide area measurements, yielding more comprehensive data about the underlying traffic process than loop detectors. Both sensors provide a data quality in accordance with their physical functional principle and in accordance with the influences of the affecting surrounding environment. For instance, an inductive loop detector works properly under fluid traffic conditions, whereas the measurements are not accurate, if there is stop-and-go traffic. Furthermore, vehicle detection and classification may be problematic in case of overtaking procedures, when the loops are overrun only partly, [11], [12]. That means an inductive loop detector is a sensor, which is influenced by the traffic process itself. In contrast to loop detectors, it is a well-known fact, that the most currently employed video sensors usually work poorly under bad weather conditions (e.g. heavy rain, fog, etc.), changing illuminations (e.g. reflections on the road surface) and traffic process dependent problems (e.g. occlusions among the vehicles on the road). Although new methods have recently been developed to overcome the addressed problems [11], the detection errors of currently used video sensors increase to more than 1000%, if the weather and illumination conditions are bad [6]. In contrast, they perform much better (they can reach even the same accuracy as an inductive loop detector), if the conditions for an optimal operation are maintained.

Not only weather and illumination conditions affect the accuracy of the measurements, but also other environmental conditions (e.g. temperature, luminosity, hygrometry, etc.) and systematic causes (e.g. the installation of the sensor for overhead or sidefire detection) may distort the detection. Without consideration of particular sensor properties and dependencies and the influences of the environmental conditions on the sensor, the data are manipulated by selection bias. Consequently, physically and environmentally affected sensors must be considered in the (probabilistic) fusion model to correct selection bias and to decrease the frequency of faulty sensor data.

In this paper, the concept of BNs is applied to merge biased traffic sensor data, which are affected by the surrounding environment. The conditions affecting the sensors are modelled in the BN data fusion model. It will be shown that the correction of selection bias can improve the accuracy of data fusion by more than 60%.

The paper is structured as follows: In section 2 some background on BNs is given. Subsequently, in section 3, the naive (classical) concept of BDF is introduced and then, extended to Bayesian Network Data Fusion (BNDF) considering additional nodes containing additional information, which are important for the fusion process. In section 4, a BNDF model is developed for the qualification of traffic data. Thereby, some environmental conditions (e.g. weather conditions, reflections on the road surface) and some traffic process related conditions (e.g. occlusions among the vehicles, the dependency on the traffic state) are modelled as additional nodes in the considered network. Then, in section 5, simulation results are presented. Finally, in section 6, conclusions and and future prospects are given.

2. BAYESIAN NETWORKS (BNS)

A Bayesian Network (BN) is a graphical formalism of handling and processing uncertain and incomplete knowledge in causal reasoning. BNs consist of a set of discrete random variables or nodes and a set of directed links or arrows. Each node is described by a set of mutually exclusive states. Some of the nodes are connected with other nodes by arrows. The arrows characterise the conditional dependencies among the nodes. So for instance, in the BN shown in Fig. 1, there is an arrow from node X to node Z_1, this indicates X causes Z_1. In this case, X is called a parent node, because it is the cause and Z_1 is the child node, because it describes the effect. The cause-and-effect relationships are modelled by the quantification of conditional probability tables (CPTs) to each single node. The nodes together with the arrows form the directed acyclic graph (DAG) [5].

Neapolitan [22] gives an adequate mathematical definition of BNs: (1) Let $\mathcal{P} = P(\mathbf{x})$ be the joint probability distribution (JPD) of the space of all possible state values $\mathbf{x}$ of the random variables in some set $\mathbf{X} = \{X_1, \ldots, X_n\}$, which are connected by a set of arrows

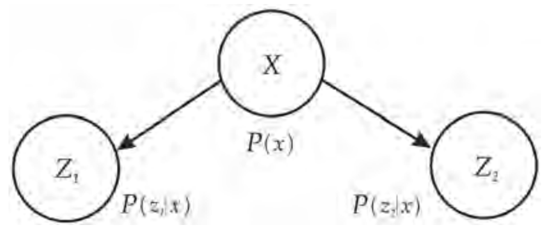

Fig. 1. A simple BN, which consists of three nodes. The variables Z_1 and Z_2 are effects of the common cause X. The (conditional) probabilities are given.

$\mathbf{A} = \{(X_i, X_j) \mid X_i, X_j \subset \mathbf{X}, i \neq j\}$ and the arrows pointing from X_i to X_j. (2) Let $\mathcal{G} = (\mathbf{X}, \mathbf{A})$ be a DAG. Then, (3) $(\mathcal{G}, \mathcal{P})$ is a BN, if $(\mathcal{G}, \mathcal{P})$ satisfies the Markov condition, i.e. a variable is conditionally independent of its nondescendents given its parents. Thus, the JPD $P(\mathbf{x})$ is characterised by

$$P(\mathbf{x}) = \prod_{x_i \in \mathbf{x}} P(x_i \mid \mathbf{pa}(X_i)) \tag{1}$$

with $\mathbf{pa}(X_i)$ denoting the set of the parents states of node X_i. If node X_i has no parent nodes, then $\mathbf{pa}(X_i) = \emptyset$. If X_i is a node with $m_i = |X_i|$ states, i.e. $X_i = \{x_{i,1}, \ldots, x_{i,m_i}\}$, $P(X_i = x_{i,k})$ denotes the probability of the certain state $x_{i,k}$. The conditional probability $P(x_i \mid x_j)$ denotes the conditional probability table of all conditional probabilities $P(x_{i,k} \mid x_{j,l})$, with $k = 1, \ldots, m_k$ and $l = 1, \ldots, m_l$.

The simple BN shown in Fig. 1 consists of three nodes, the parent node X and its child nodes Z_1 and Z_2. The states of each node are characterised by small letters x, z_1 and z_2 respectively. The causal relationships are given by directed links and the JPD of this BN is computed by equation (1), which can be rewritten as:

$$P(x, z_1, z_2) = P(x)P(z_1 \mid x)P(z_2 \mid x). \tag{2}$$

The BN in Fig. 1, which is characterised by the JPD in equation (2), satisfies the Markov condition.

For further reading in general theory on BN the reader is referred to [4], [22], [23].

3. BAYESIAN NETWORK DATA FUSION (BNDF)

In the following section 3.1, the naive or classical Bayesian approach for data fusion is introduced and then, in 3.2, extended to the more generalised Bayesian Network Data Fusion (BNDF).

3.1. Naive Bayesian Data Fusion (BDF)

Bayesian Data Fusion (BDF) makes use of Bayes' rule and combines objective and/or subjective knowledge of the underlying and possibly unknown process—its a priori probabilities and likelihoods—in a probabilistic model. The method can principally be characterised as:

$$P(x \mid z) = \alpha \cdot P(x)P(z \mid x) \tag{3}$$

to infer $x \in X$, which is the unknown state among $|X|$ possible states by the observation $z \in Z$ among $|Z|$ possible observations, which are also called evidences. All the Ps are discrete probability distributions, but can be considered in the continuous world as well. $P(z \mid x)$ is the conditional probability distribution (likelihood function) of a sensor measurement z given the true state x. It reflects the correct and false measurements, which can be characterised as the quantification of the accuracy of the sensor. $P(x)$ is the prior distribution of x describing our expectation of the unknown variable X. $P(x \mid z)$ is the inference distribution (a posteriori distribution) of the unknown state x given a specific measurement z. It can be characterised as the trust in a specific measurement $P(z \mid x)$ expecting the prior $P(x)$. α is a normalising constant, which ensures, that $\sum_i P(x_i \mid z) = 1$.

When the a priori probabilities and likelihoods are determined, the given measurements z allow to infer the unknown state x according to equation (3). That means knowledge, which is based on evidences from the observable variable Z is propagated towards the unknown variable X. For more information on BDF see [7], [13], [15], [25].

The BN in Fig. 1 is the simplest BN, which models naive BDF according to equation (3). The shown BN consists of the variable X, which represents the unknown process of interest and the two sensor variables (evidence nodes) Z_1 and Z_2. However, the advantage of BNDF is to extend the naive Bayesian model by a more detailed or more characteristic modelling of the sensor nodes and/or the underlying process. This problem is addressed in the following section.

3.2. From BDF to BNDF

Remind the BN in Fig. 1 for naive BDF. Imagine the two sensor variables Z_1 and Z_2 model an inductive loop detector and a video based sensor respectively, measuring the vehicle classes on a road of interest. Then, the variable X represents the underlying and unknown traffic process. Now, imagine, that some properties of the detectors are influenced by their surrounding environment, which might yield selection bias in the measurements, resulting in faulty or corrupted data usually being undesirable for an adequate traffic management. But, if we know more about the underlying process, the applied sensors and their properties and their surrounding environment, we can include this knowledge in a BN, which contains additional nodes, modelling these influences. Consequently, the advantage of BNDF is to extend the naive BDF model by a more detailed, more characteristic and more realistic modelling of the sensor nodes and/or the underlying process. This results in a data fusion model, which is capable of correcting selection bias. As a consequence, the resulting merged data are more accurate.

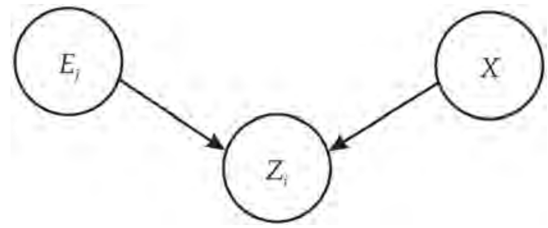

Fig. 2. An extended BN for data fusion according to Fig. 1 with an arbitrary sensor node Z_i and an environmental influence node E_j. The causal dependence between the environment E and the measurement Z is shown by the directed link connecting them. Without loss of generality, the sensor and environment nodes can be connected several times with the traffic node X, depending on how many sensors are in use and how many environmental dependencies affect the performance of the sensor.

The application of these particular BNs for traffic surveillance is a novel solution for the correction of selection bias in manipulated traffic data. Comparable, but different investigations were done for landmine detection, e.g. [5], and in case of the detection of acoustic signals, e.g. [16].

In Fig. 1, the environmental dependencies affecting the performance of the sensors are not yet considered. A more realistic and thus, more complex BN for data fusion considering environmental influences is shown in Fig. 2. According to Fig. 2 and the text above, equation (3) has to be modified to equation (4), which enables an improved data fusion:

$$P(x \mid \mathbf{z}, \mathbf{e}) = \alpha \cdot P(x) P(\mathbf{z} \mid x, \mathbf{e}) \tag{4}$$

with x representing the unknown vehicle class, $\mathbf{e}$ describing the set of the environmental influences on the performance of the sensor and $\mathbf{z}$ the affected measurements of a set of sensors. The calculation of equation (4) yields a correction of selection bias by the influence of the sensors' surrounding environment and hence, the improvement of the estimates of the unknown state variable X under these conditions.

In the following section, the influences of the affecting environment on a video based traffic detector are more specified. Later, a comparison between the performance of a weather independent inductive loop detector and a weather dependent video sensor is made. The correction of selection bias and thus, the improvement of the fusion process are shown on the basis of synthetic traffic data.

4. BNDF TO CORRECT SELECTION BIAS IN TRAFFIC DATA

In this section, the BN according to Fig. 2 and its inherent fusion equation (4) are applied to merge traffic data and to improve the accuracy of the fusion process. In the following paragraphs the modelling of the prior probability distribution, the likelihood probability of the environmental affected sensor, the resulting BNDF model and the inference of the unknown state values for the traffic process are discussed.

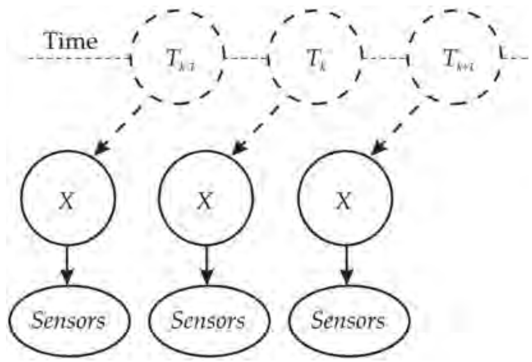

Fig. 3. A BN modelling the time dependence of the traffic process (node X). For each point in time T_k another BN characterises the traffic process with different probability distributions $P(x \mid T_k)$.

4.1. Modelling the Prior Probability

The prior $P(x)$ represents the knowledge about the underlying traffic process. If we do not know anything about it, it is legitimate to model the prior as a uniform probability distribution, weighting each state value equally. But if we know more, the process can be modelled considering different objective (physical) or subjective (Bayesian) assumptions. So, for instance, the prior can be modelled as a probability distribution containing all the relative frequencies of the most expected vehicle on a road

- depending on the ratio of actual vehicles counted in the referred area, city, country, etc.,
- depending on the time of the day,
- depending on the type of the observed road (e.g. it can be distinguished between play streets and transit roads),
- depending on incidents, events and structural measures, stoppages, etc.,
- or even mixtures of some of the mentioned dependencies.

An example for modelling the traffic process depending on the time of day is given in Fig. 3. The time dependence can be modelled as additional control nodes (which are not BN nodes) resulting in a different BN at each point in time. These kinds of BNs are called Dynamic Bayesian Networks (DBNs), but shall not be considered throughout this paper. See [21], [26] for deeper information.

The formalism of BNs, introduced in section 2, allows one to model the prior probability distribution of the traffic process and its dependencies with additional nodes and attached known or learned probability tables to manipulate the a posteriori probability by the given assumptions and information in a useful way.

The necessary data for the quantification of the probability tables can be learned, adapting the underlying traffic process. Adaptive learning methods, e.g. for learning time variant prior probabilities and CPTs are addressed in, e.g. [3], [10], [24].

Fig. 4. Reflections on the road surface usually make the determination of relevant traffic data difficult.

4.2. Modelling the Influenced Sensors

As already stated in section 1, the performance of a sensor is dependent on its functional principle, the surrounding environment and other phenomena. Here, a traffic state dependent sensor and an environmental affected sensor are modelled.

1) *Modelling the Environmental Influenced Sensor*: A video detector, as an example for an environmental influenced sensor, can be characterised by the following dependencies [2], [6], [14], [20]:

- Different or changing weather conditions (e.g. heavy rain, fog, snow, etc.) mainly cause false, multiple and phantom detections.
- Different or changing illuminations, e.g. darkness, at nightfall, glare of the sun, sun rise and sundown, shadows of moving or immobile objects, reflections on the road, e.g. as shown in Fig. 4, etc., usually cause false, multiple and phantom detections.
- Camera motion and camera vibrations can be caused by heavy winds. Particularly in wide area traffic surveillance erroneous detections occur.
- Particular traffic conditions can cause partial or even total occlusions among the vehicles, yielding an underdetection of some or even all vehicle classes.
- Video sensors are mounted for overhead, sidefire and wide area detection. Depending on the installation height partial and total occlusions occur, yielding an underdetection of some vehicle classes.
- The physical environment, e.g. temperature, luminosity, hygrometry, etc., can have a great influence on the measurements, because the sensor is built within a specific range of tolerance.
- The driver behaviour, e.g. overtaking and turning procedures, can lead to multiple or false detections, when the vehicles pass through several fixed detection areas partly. Usually specific vehicle classes are overcounted, some others are undercounted.

- The wear and tear of the sensor and the components during its operating life (Mean Time Between Failure—MTBF) can cause corrupted data, lack of data or even data terminations.
- A bad calibration of the camera, particularly of the detection areas, cause wrong vehicle classifications, because of multiple or underdetections and faulty velocity and length measurements.

Some of the effects are time-dependent (e.g. shadows of moving or standing objects, because of sun rise), some occur by accident (rain, reflections on the road surface) and some have systematic causes (calibration of a camera for overhead or side fire detection).

2) *Modelling the Traffic State Influenced Sensor*: An inductive loop detector, as an example for a traffic state dependent sensor, is an LC-oscillator, which is buried underneath the road surface. Its resonance frequency changes, if there is an metallic object in working area of the loop. These changes are evaluated and compared with known pattern, thus, a vehicle classification is possible. Besides environmental and other affecting dependencies, an inductive loop detector can be characterised by, e.g. [11]:

- Free flow conditions usually yield optimal detection results, while stop-and-go traffic distorts the measurements. Usually, there occur overdetections, misclassifications of the vehicles, enduring occupancies, yielding for instance false estimations of the traffic density.
- The driver behaviour in different traffic conditions, e.g. if loop detectors are overrun only partly, can lead to multiple or false detections and misclassifications of the detected vehicles.

3) *Comments on Modelling the Considered Sensors*: The most effects, which have an influence on the sensors' performance and the quality of the measured data, are not methodical, but stochastical uncertain and can be considered and quantified in the BNDF model. The quantification of the influences of the most physical conditions for an optimal sensor performance is possible by studying the data sheets, for instance, delivered by the original end manufacturer. In contrast, the quantification of the performance of the sensor under different environmental conditions is difficult, because extensive field tests with highly accurate sensors or manual references need to be realised. So for instance, by means of a rain or weather sensor, the current weather situation can be determined and relative frequencies of correct, false and phantom vehicle classifications could be made to decide whether the sensor is more or less influenced by weather conditions. Then, the values for the likelihood $P(z \mid x, \mathbf{e})$ of the considered sensor can be used for inference.

4.3. Inference and Sensor Data Fusion

The resulting BNDF model can be used to compute the state values of the unknown traffic process X within the surrounding known (or even unknown) environment (data $\mathbf{e}$). The sensors provide measuring data (evidences) $\mathbf{z}$, allowing the vehicle class x to be inferred from the BNDF model according to the Markov condition of the JPD in equation (1) and the extended Bayes' rule in equation (4).

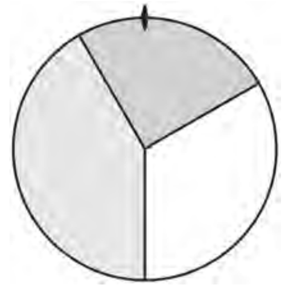

Fig. 5. The probability wheel (from Heckerman [8]).

In general, the computation and evaluation of a BN is NP-hard [22], [23]. Depending of the number of nodes, the number of states of each node and the algorithm used, the evaluation of a BN can be quick or time-consuming. The consideration of these facts and the requirements, defined by the user, e.g. concerning the real-time applicability, the accuracy of the fusion results, etc. determine the structure (e.g. the number of nodes and states) and the computation methods of the BN in question (e.g. exact or approximate inference algorithms).

Computing equation (4) by the use of exact or approximate inference algorithms, the probability distribution of the inferred state $x \in X$ is estimated. Typically, the unknown value x is determined by a maximum a posteriori estimation (MAP) of the a posteriori probability

$$\hat{x} = \arg\max_x P(x \mid \mathbf{z}, \mathbf{e}) \tag{5}$$

by maximising the confidence in the measurement. Another method to calculate $\hat{x}$, which keeps the principle of probability alive, is the so called probability wheel, introduced in [8]. In this method, the probabilities of the states of a variable are considered as regions of different percent areas on a symmetric wheel (see Fig. 5). The symmetry assumption implies, that any position where the wheel can stop is equally likely. Consequently, the probability of which state $\hat{x}$ will be chosen, depends on the percent area where the wheel will stop. Hence, in comparison to MAP estimation the probability wheel may stop at the percent area for even very unlikely states, which is particularly advantageous in case of flat probability density functions, that characterise a higher degree of uncertainty. The application of the probability wheel can be expressed by

$$\hat{x} = \arg \mathrm{PW}_x P(x \mid \mathbf{z}, \mathbf{e}) \tag{6}$$

where PW labels the probability wheel operator. There are different methods and algorithms for realising exact or approximate inference, which are not discussed in this paper. Good descriptions can be found in [21]–[23].

5. RESULTS

In this section simulation results for BNDF in comparison with the naive BDF in the case of vehicle classification are presented. Thereby, on the one hand, the influence of environmental conditions on the performance of a video sensor and on the other hand, the influence of the traffic conditions on the performance of an inductive loop detector are investigated. In the case of BNDF these influences, which affect the measurements of the sensors, are considered as additional nodes in the BN model. The inference of the resulting BNDF model yields a correction of selection bias in the merged traffic data. In contrast, naive BDF is not capable of correcting bias, yielding manipulated data.

The investigations were made on the basis of a data base (video based measurements) containing 65,000 measurements generated synthetically and additionally, real 24-hour traffic data [27] containing approximately 120,000 measurements. The real traffic data were recorded with an ASIM TT 298 combination detector[1] and an ordinary inductive loop detector at an intersection between a federal road and a less frequented city road (Radeburger Strasse/Meinholdstrasse) in Dresden, Germany, on 20 May 2005. This data base is characterised by peak traffic in the morning and in the evening and was used for learning the prior probability $P(x)$.

The BNDF model, developed in the last two sections and the given data base are used for learning and validation of cases with sensor data. The real data set is not used for validation. For simulations, the tools Mathematica 4.0 and Netica 3.19 were used. The results are compared with the naive approach of BDF without correction of selection bias.

5.1. State Declaration and Assumptions

In the following, the states of the traffic process node X and the sensors Z_1 (loop detector) and Z_2 (video sensor) are represented by different nine vehicle classes, which are given by the following set of symbols:

$$X = \{\mathrm{C}, \mathrm{C+}, \mathrm{V}, \mathrm{L}, \mathrm{L+}, \mathrm{D}, \mathrm{B}, \mathrm{M}, \mathrm{N}\}$$

representing C (car), C+ (car with a trailer), V (van), L (lorry), L+ (lorry with a trailer), D (double train), B (bus), M (motorcycle) and N (not classifiable). One more virtual class Ø, representing the case *nothing detected*, was used to decide, whether a sensor did not detect anything, although a vehicle was present. Thus,

$$Z_1 = Z_2 = \{X, \text{Ø}\}.$$

[1]A final report about testing the ASIM TT 298 detector in accordance with the German TLS standard [1], developed by Munich University of Technology is available on http://www.asim.ch/traffic/pdf/report_tt298_d.pdf [17].

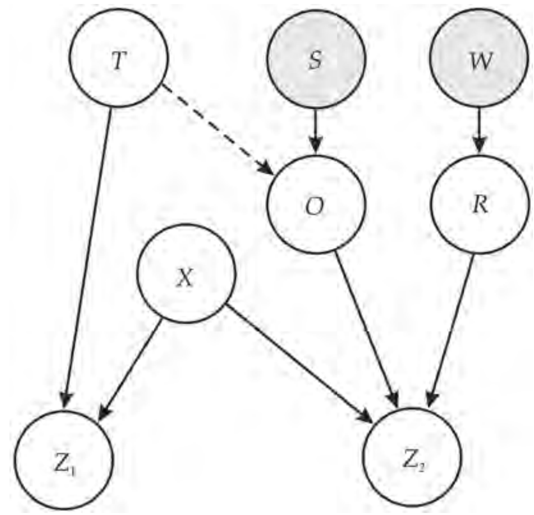

Fig. 6. Qualified data fusion with the two affected sensors Z_1 and Z_2. In contrast to the classical BDF model according to Fig. 1 the BN contains the environmental nodes W and R, modelling the Weather conditions and the Reflections on the road surface; the traffic process dependent nodes T and O, modelling the Traffic state and Occlusions among the vehicles on the road; as well as the node S, modelling the Sensor installation for sidefire or overhead detection. The grey coloured nodes and the dashed directed link are inconsequential, if the the nodes O, R and T are evidence nodes.

The quantification of the prior $P(x)$ was made by EM- (expectation maximisation) learning of the real 24-hour data [27]. The vehicle class C was expectedly strongly overrepresented by approximately 85%, while the other eight classes share the remaining 15%. The relative frequency of the most rare class N reached only 0.1% (see equation (8) in the appendix).

We chose the simple BN in Fig. 6, which is the result of the following assumptions and determinations made:

- The loop detector Z_1 is affected only by the traffic state T, which is characterised by the states t_1 (free flow) and t_2 (stop-and-go traffic), i.e. $T = \{t_1, t_2\}$. It works optimally in the case of $T = t_1$. There should be an explicit underdetection of the vehicle classes C+, L+ and D in the case of $T = t_2$ (see equation (10) in the appendix). All other classes should be slightly underdetected. The influence set for sensor Z_1 thus is $\mathbf{E}_1 = T$.
- In contrast, the video sensor Z_2 should be affected by reflections on the road surface R and current occlusions among the vehicles O. Reflections on the road surface should be caused by the current weather conditions W. Occlusions should be caused by the sensor installation S and the traffic state T. Since we consider the nodes O and R as evidence nodes, the influences of T, S and W on Z_2 are “explained away” [22], [23]. Thus, the influence set for the video detector is given by $\mathbf{E}_2 = \{R, O\}$.
- The resulting influence set is given by $\mathbf{E} = \{\mathbf{E}_1, \mathbf{E}_2\} = \{T, R, O\}$.
- The nodes O and R are binary nodes, which are characterised by the states o_1 (there are not any occlusions among the vehicles at all) and o_2 (heavy

occlusions) and r_1 (there are not any reflections on the road surface) and r_2 (heavy reflections).

- The video sensor has the same performance as the loop detector in the case of optimal conditions with no bias. Thus, the likelihood probability of the video sensor is given by $P(z_2 \mid x, o_1, r_1) = P(z_1 \mid x, t_1)$ yielding optimal fusion results (see equations (9) and (11) in the appendix). For this reason, the fusion process needs only to be simulated in the case of the worst conditions concerning the three nodes with the states o_2, r_2 and t_2, where the correction of selection bias is to be proved.
- In the case of reflections on the road surface, because of darkness and bad weather conditions and in the case of occlusions among vehicles, it often happens, that some vehicles are overcounted and some are undercounted [2], [6], [14], [20]. Here it is assumed, that larger vehicles, e.g. lorries and buses, are overcounted, while smaller vehicles, like cars, motorcycles and vans, are undercounted (see equations (13) and (14) in the appendix). This causes changes in the quantification of the likelihoods of the video sensor and the loop detector and yields different joint likelihoods $P(z_1, z_2 \mid x, t, o, r)$.
- Phantom detections should not be present.

With the assumptions and determinations made we simulated the naive BDF approach in comparison with the extended and qualified BNDF model, which considers the traffic state node, the occlusions node and the reflections node.

In the following paragraph the results for two sensor data fusion are presented.

5.2. BNDF vs. BDF

According to the learned a priori distribution $P(x)$ of the underlying traffic process and the modelled likelihoods of the the loop detector $P(z_1 \mid x, t)$ and the video sensor $P(z_2 \mid x, o, r)$, which consider the made assumptions of the preceeding paragraph 5.1, we simulated the fusion process with two sensors for the following cases (see equations (9) to (14) for the applied sensor likelihoods):

0. Both sensors work optimally, i.e. there are not any internal and external influences, which affect the measurements of the sensor. This case is only used for reference.
1. The inductive loop detector works optimally, but the video detector is affected
   a) by occlusions, i.e. $\mathbf{e} = \{o_2\}$.
   b) by reflections on the road surface, i.e. $\mathbf{e} = \{r_2\}$.
   c) by occlusions and reflections on the road surface, i.e. $\mathbf{e} = \{o_2, r_2\}$.
2. The video sensor works optimally, but the inductive loop detector is affected by the traffic state, i.e. $\mathbf{e} = \{t_2\}$.
3. Both sensors are affected, i.e the loop detector is influenced as in 2.) and the video sensor is influenced as in 1.) by:
   a) by occlusions, i.e. $\mathbf{e} = \{t_2, o_2\}$.
   b) by reflections on the road surface, i.e. $\mathbf{e} = \{t_2, r_2\}$.
   c) by occlusions and reflections on the road surface, i.e. $\mathbf{e} = \{t_2, o_2, r_2\}$.

In case 3.c) the conditions for the detection and classification of vehicles are the worst, because both sensors are affected by reflections on the road, heavy traffic conditions and occlusions among the vehicles.

The simulations were done with the same number of 65,000 measurements under the prevailing circumstances. Since the sensors have the same performance in the case of optimal conditions (see paragraph 5.1), yielding the best fusion results, it is necessary to investigate the fusion process for the cases 1.a) till 3.c). Case 0.) is used only for reference. We used the probability wheel, according to equation (6), for the estimation of the optimal state x, since highly influenced traffic sensor data are merged reflecting the expected higher degree of uncertainty of the data.

The tables I to VII show the achieved estimation errors of the vehicle classes with naive BDF (row BDF) according to equation (3) in comparison to the extended BNDF (row BNDF), which considers the the nodes T, O and R, according to equation (4) and Fig. 6.

We considered two kinds of estimation errors for the evaluation of the comparison between BNDF and BDF. The relative Class Related Error CRE(x) of vehicle class x is given by:

$$\mathrm{CRE}(x) = \frac{\mathrm{FDV}(x)}{\mathrm{CDV}(x) + \mathrm{FDV}(x)} \qquad \forall \quad x \in X,$$

with CDV(x) denoting the number of correctly detected vehicles of class x and FDV(x) denoting the number of false detected vehicles of class x. The calculation of the CREs of the vehicle class of interest informs us about the accuracy of the fusion process in a class related context. Since some vehicle classes can be detected more or less better than other, the CREs differ. The accumulation of FDV(x), $\forall x \in X$, in relation to the sum of all detected vehicles, yields the Total Classification Error (TCE):

$$\mathrm{TCE} = \frac{\sum_{x \in X} \mathrm{FDV}(x)}{\sum_{x \in X} (\mathrm{CDV}(x) + \mathrm{FDV}(x))} \tag{7}$$

which allows us to state something about how accurate and successful the fusion process is overall. Although the TCE can be decreased by far, it might happen, that there are vehicle classes, whose CREs increase by far or which cannot be determined anymore, i.e. their CREs reach 100%. Depending on the fusion task to be solved and the underlying operational areas of the applied sensors, the fusion process can be successful if the TCE decreases and some CREs increase. For instance, this might be the case for an average estimation of the travel times for traffic management. On the other hand, the fusion process can be unsuccessful, if there is at least one CRE increasing or reaching even 100%, e.g. in the case of enforcement at tollgates on motorways. Because of these facts, the two terms *CRE-successful fusion* and *TCE-successful fusion* are introduced to distinguish between the two error metrics above.

The relative discrepancies between the TCEs and the CREs in the case of BNDF and in the case of the naive BDF are given by ΔTCE and ΔCRE respectively.

In the following, the simulation results of the considered cases are given and interpreted. It is shown, that modelling the affecting conditions as additional nodes in a BN usually, but not generally, yield improvements of the vehicle class estimates. Although the TCEs decrease in any cases in BNDF, the CREs of some vehicle classes may increase.

1) *None Affected Sensors—Case 0.)*: As mentioned above, if both sensors work optimally, i.e. there is no affecting influence set **E**, optimal results for vehicle classification are obtained. In this case the results for BDF and BNDF are identical. 807 of 65,000 vehicle class estimates are erroneous (TCE = 1.24%). The CREs for each vehicle classes are the following: CRE(C) = 0.5%, CRE(C+) = 19.0%, CRE(V) = 2.6%, CRE(L) = 3.9%, CRE(L+) = 18.0%, CRE(D) = 5.3%, CRE(B) = 20.3%, CRE(M) = 18.3% and CRE(N) = 29.2%. These results are used for reference.

2) *Affected Video Detector—Cases 1.a) to 1.c)*: As expected, if there is only one sensor affected by some influence set **E**, the vehicle classification with BNDF yields much better results, than by BDF, which had already been stated in the underlying paper [9]. Here, the TCEs decrease from 8.7% (5,644 erroneous vehicles), 10.0% (6,517) and 10.9% (7,097) to 5.0% (3,241), 3.7% (2,430) and 4.2% (2,751) for the cases 1.a), 1.b) and 1.c) respectively. See the tables I, II and III for more results. In almost any case the CREs decrease by far, up to ΔCRE(M) = −88% in case 1.b), whereas there is exactly one significant increase of the CRE of vehicle class L in the case 1.c). Since the class C ist strongly overrepresented by the prior probability $P(x = C) = 85\%$, it is not surprising, that there can be achieved an enormous error reduction for this vehicle class up to ΔCRE(C) = −79% in case 1.c) as well. There are also some other cases, where the vehicle classes C+ and N cannot be detected at all, i.e. their CREs reached 100%. Furthermore, if the classification results achieved here are compared with the unaffected reference case 0., it can be ascertained, that there is an reduction of the CREs of the class M from 29.2% to 11.8%, i.e. we could even improve the unaffected fusion results by far. Considering the three cases, we can speak of a TCE-successful fusion and an almost totally CRE-succesful fusion for vehicle classification, if the sensor properties, affected by the modelled conditions are considered in the BNDF model. As a consequence, we are able to improve the measurements of environmental influenced sensors, whose properties and dependencies are modelled in a particular BN, by environmental independent sensors.

3) *Affected Loop Detector—Case 2.)*: If the video detector works optimally and the loop detector is affected by the traffic state, i.e. the influence set is $\mathbf{e} = \{t_2\}$, the simulation results of the cases 1.a) to 1.c) are mostly verified. Altogether, there is a reduction of the TCE from 3.5% (2,298 erroneous vehicles) to 3.1% (1,989). See table IV for results. The CRE for class L+ is decreased by far: ΔCRE(L+) = −62%. But there are also increases of the CREs for the vehicle classes V, L and N. As a result, vehicle class N cannot be correctly classified anymore. The fusion results can be said to be TCE-succesful and almost CRE-successful. Completely CRE-unsuccessful are the CREs for the classes L and N.

4) *Both Sensors Affected—Cases 3.a) to 3.c)*: If both sensors are affected by some influence set **E**, we have the worst conditions for detecting and classifying vehicles on the road. Here, the TCEs decrease from 17.4%, 27.1% and 26.3% to 10.2%, 8.8% and 9.5% in the cases 3.a), 3.b) and 3.c) respectively. That means the TCEs reduce by 41%, 67% and 64%, resprectively. See table V, VI and VII for results. The overall results show incredible improvements, due to the consideration of the sensors' dependencies in the BNDF model, but there are also weightily drawbacks in accordance with the CREs of some vehicle classes. Since class C is strongly overrepresented by the prior probabiliy $P(x)$, the superposition of the influences make the correct classification of C much easier, thus the CREs of class C in BNDF are very low. In contrast, the CREs of other vehicle classes increase and some reach 100%. Noticable is for instance the increase of the CRE(L) by 184% and 44% in the cases 3.b) and 3.c) respectively.

If the fusion in these cases is evaluated, we can state, that we have very TCE-successful fusion, but CRE-unsuccessful fusion in almost any case and for almost each vehicle class, which is usually unacceptable. Due to the fact, that BDF also behaves poorly, we cannot even speak of good fusion results in general. Under such difficult circumstances and again, depending on the fusion task to solve in accordance with the underlying traffic related problem, one should think about the reduction of the nine vehicle classes to maybe two, for instance combining car similar vehicles to the first class and lorry similar vehicle to the second class.

5) *Summary of the Results in the Tables I to VII*: Summarising the seven tables, we can state, that usually BDF performs poorly, because of the inherent selection bias, yielding the addressed over- and undercounting of specified vehicle classes. In contrast, in the case of BNDF, the consideration of the sensors' surrounding environment and other phenomena like traffic process related dependencies, affecting the sensors' performance, in the fusion model, yielded an explicit

TABLE I

CREs for Vehicle Classification in the Case 1.a) [%]									
	C	C+	V	L	L+	D	B	M	N
BDF	3.8	94.0	26.1	25.4	86.3	39.7	88.1	99.5	98.9
BNDF	2.7	100	9.0	17.3	46.6	26.1	42.7	17.9	100
ΔCRE	−27	+6	−66	−32	−46	−34	−52	−82	+1
ΔTCE	By consideration of $\mathbf{e} = \{o_2\}$ in BNDF: −42.6%								

TABLE II

CREs for Vehicle Classification in the Case 1.b) [%]									
	C	C+	V	L	L+	D	B	M	N
BDF	8.6	98.3	7.9	8.3	63.4	14.1	66.9	95.8	92.4
BNDF	2.5	19.2	7.3	8.6	20.1	15.9	53.5	11.8	38.0
ΔCRE	−71	−81	−8	+4	−68	+12	−20	−88	−59
ΔTCE	By Consideration of $\mathbf{e} = \{r_2\}$ in BNDF: −62.7%								

TABLE III

CREs for Vehicle Classification in the Case 1.c) [%]									
	C	C+	V	L	L+	D	B	M	N
BDF	7.9	99.6	16.1	21.9	72.6	26.8	80.8	96.2	94.2
BNDF	1.7	37.1	8.5	35.2	34.5	32.2	32.0	35.9	100
ΔCRE	−79	−63	−48	+61	−52	−14	−60	−634	+6
ΔTCE	By Consideration of $\mathbf{e} = \{o_2, r_2\}$ in BNDF: −61.2%								

TABLE IV

CREs for Vehicle Classification in the Case 2.) [%]									
	C	C+	V	L	L+	D	B	M	N
BDF	1.6	71.3	5.9	8.6	79.2	28.0	36.2	30.7	88.6
BNDF	1.2	71.3	6.6	11.2	29.4	22.2	36.2	30.7	100
ΔCRE	−25	±0	+13	+31	−62	−21	±0	±0	+13
ΔTCE	By Consideration of $\mathbf{e} = \{t_2\}$ in BNDF: −13.4%								

TABLE V

CREs for Vehicle Classification in the Case 3.a) [%]									
	C	C+	V	L	L+	D	B	M	N
BDF	9.9	97.6	38.6	37.6	95.9	74.8	90.1	100	100
BNDF	2.8	100	36.4	44.3	100	100	100	100	100
ΔCRE	−76	+2	−6	+18	+4	+34	+11	±0	±0
ΔTCE	By Consideration of $\mathbf{e} = \{t_2, o_2\}$ in BNDF: −41.1%								

TABLE VI

CREs for Vehicle Classification in the Case 3.b) [%]									
	C	C+	V	L	L+	D	B	M	N
BDF	26.0	99.6	22.5	18.7	90.6	57.0	65.8	96.8	100
BNDF	3.9	73.7	21.8	53.1	100	46.2	65.8	59.3	100
ΔCRE	−85	−26	−3	+184	+10	−19	±0	−39	±0
ΔTCE	By Consideration of $\mathbf{e} = \{t_2, r_2\}$ in BNDF: −67.4%								

TABLE VII

CREs for Vehicle Classification in the Case 3.c) [%]									
	C	C+	V	L	L+	D	B	M	N
BDF	23.5	99.6	29.0	31.9	92.8	64.3	84.8	98.5	98.0
BNDF	3.3	100	31.3	45.9	100	66.3	84.8	47.8	100
ΔCRE	−86	+0.4	+8	+44	+8	+3	±0	−52	+2
ΔTCE	By Consideration of $\mathbf{e} = \{t_2, o_2, r_2\}$ in BNDF: −63.8%								

TCE-successful fusion for any considered case. See Fig. 7 for the results. The best results were obtained, if the worst conditions were considered in the fusion model. In the case of two sensors, one traffic process dependent loop detector and one weather and traffic process affected video sensor, we achieved an improvement of the fusion process by more than 60%. Consequently, we are able to enhance environmental independent sensors by strongly environmental dependent sensors, whose properties and dependencies are modelled in a particular BN. That means since there is no single sensor, which is environmentally independent, the sensors should be affected in different ways and/or different domains.

On the other hand, the results show, that BNDF (and also BDF) is not CRE-successful in any case, particularly if the conditions for both sensors are bad. Depending on the fusion task to be solved and the underlying task in traffic management, it must be decided, whether the fusion results are beneficial or not. If necessary, the classification domain for vehicles must be reduced to two classes for instance.

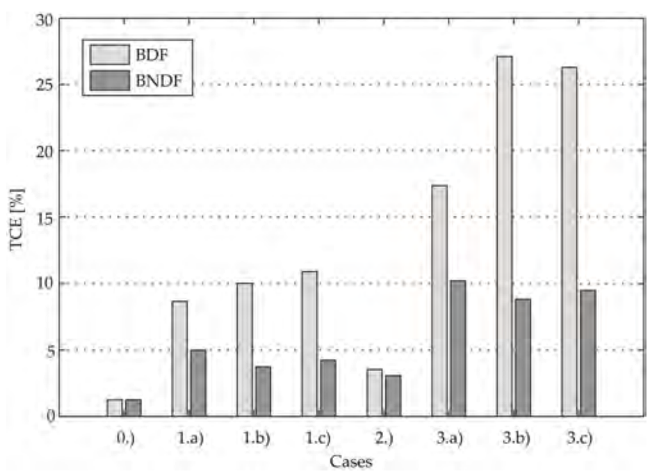

Fig. 7. The TCEs of the cases 0.) (for reference) to 3.c) are plotted for BDF and BNDF. The most improvements occur in 3.a) to 3.c).

The results show, what magnitude of improvements in data fusion can be achieved, if external and internal affects are considered in the fusion model. Nevertheless, the obtained error reductions of the vehicle class estimates must be interpreted in a correct manner, because the simulations were done on the basis of synthetic data (which was supplemented with real 24-hour traffic data). When using real traffic data, the fusion error reduction results are supposed to be slightly worse.

6) *Computational Performance*: The computation and evaluation of the simple BN in Fig. 6 can be done in real-time, since it is small and consists of mainly binary evidence nodes. However, if one wishes to consider more external and internal influences, which are characterised by more states, affecting the performance of the sensors, the computational complexity grows exponentially. Consequently, the use of larger, more realistic BN for data fusion in traffic surveillance, has to be investigated with regard to the accuracy and reliability of the traffic data, real-time applicability, etc.

6. CONCLUSIONS AND FUTURE PROSPECTS

In this paper a data fusion method was introduced, that is based on the concept of Bayesian Networks, called Bayesian Network Data Fusion (BNDF). Since sensors are time variant systems with particular functional principles, the measurements are as good as the a priori knowledge about the sensors and the underlying process are and as good as the measurements of the sensors are. The sensors are affected by the external, e.g. environmental conditions, and internal conditions, e.g. the physical life of the sensor, yielding selection bias in the resulting measuring data. Thus, the consideration of these dependencies in a BN model are indispensable to correct selection bias and thus, improve the fusion process and the resulting data.

The obtained results for vehicle classification show, that the BNDF model is able to infer vehicle classes by systematically taking into account sensor measurements (the vehicle evidences), environmental conditions (the environmental evidences) and traffic process related conditions (traffic state evidences and occlusions). By the combination of two heterogeneous sensors—here, a weather and traffic process dependent video sensor and a traffic process dependent inductive loop detector—the accuracy of the estimates of the vehicle classes is improved by up to more than 60%, i.e. the fusion process is TCE-successful in any case. The fusion results are also CRE-successful, if it can be ensured, that the sensors are affected by some differing influence sets. Under certain difficult circumstances, the fusion process is usually not CRE-successful, which means, the CREs increase by far or reach even 100%, i.e. the vehicles cannot be classified correctly at all. As a consequence, the applied sensors should differ in their internal and external influences. Furthermore, the sensors must be used carefully. Depending on the fusion task to be solved and the underlying traffic related application, one must decide, whether a not CRE-successful fusion is sufficently satisfied. The decisions of a traffic manager will differ in the case of simply measuring averaged travel times and in the case of the classifaction of vehicles for enforcment and monetary applications.

The obtained results must be interpreted carefully, because the simulations were done on the basis of synthetic traffic data, supplemented with real traffic data. Moreover, the conditions for an optimal performance of the video sensor were intentionally violated by the modelled bad conditions. Vice versa, if the conditions are optimal, the fusion results will be even better. The investigation of a two homogeneous sensor fusion was not of interest in this article. Since, homogeneous detectors are affected by the same internal and external conditions, selection bias cannot be corrected in general. Nevertheless, an improvement of the fusion process is obvious.

The results further show, what magnitude of improvements in data fusion can be achieved, if external and internal affects are considered in the fusion model. But, in case of real traffic data, the fusion results are supposably slightly worse.

Our current work is characterised by the application of the proposed method (using probability wheel and MAP estimation) to real traffic data. Thereby not only video sensors and their dependencies on weather conditions, but also other sensors, e.g. inductive loop detectors, infrared sensors, etc. and their influences by external and internal conditions need to be investigated and quantified in a BNDF model, because the concept of BNDF is not restricted to any particular sensor type, but generally valid. Thus, the creation and application of an adequate BN sensor model is supposed to improve the fusion results and to correct selection bias in general. In this regard it has to be stated, that the simulations done and the results achieved cannot be extrapolated, since they refer only on two specific sensors, with certain properties and applications.

Furthermore, the concept of adaptive probability learning is to be applied to the considered probabilistic sensor model to investigate the considered results for an instationary traffic process with time dependent prior probabilities [10]. This comes along with the coupling of the environmental nodes, e.g. the weather node, with meteorological models to achieve more sophisticated cases.

APPENDIX

In the following the prior probability and the CPTs of the simulated inductive loop detector Z_1 and the video sensor Z_2 are given.

Prior Probability

$$P(x) = (.847\ \ .004\ \ .094\ \ .026\ \ .005\ \ .012\ \ .005\ \ .006\ \ .001)^T \tag{8}$$

CPTs of the Inductive Loop Detector Z_1

$P(z_1 \mid x, t_1)$

$$= \begin{pmatrix} .91 & .01 & .01 & .01 & .01 & .01 & .01 & .01 & .01 & .01 \\ .01 & .91 & .01 & .01 & .01 & .01 & .01 & .01 & .01 & .01 \\ .01 & .01 & .91 & .01 & .01 & .01 & .01 & .01 & .01 & .01 \\ .01 & .01 & .01 & .91 & .01 & .01 & .01 & .01 & .01 & .01 \\ .01 & .01 & .01 & .01 & .91 & .01 & .01 & .01 & .01 & .01 \\ .01 & .01 & .01 & .01 & .01 & .91 & .01 & .01 & .01 & .01 \\ .01 & .01 & .01 & .01 & .01 & .01 & .91 & .01 & .01 & .01 \\ .01 & .01 & .01 & .01 & .01 & .01 & .01 & .91 & .01 & .01 \\ .01 & .01 & .01 & .01 & .01 & .01 & .01 & .01 & .91 & .01 \end{pmatrix} \tag{9}$$

$P(z_1 \mid x, t_2)$

$$= \begin{pmatrix} .73 & .03 & .03 & .03 & .03 & .03 & .03 & .03 & .03 & .03 \\ .18 & .36 & .18 & .04 & .04 & .04 & .04 & .04 & .04 & .04 \\ .03 & .03 & .73 & .03 & .03 & .03 & .03 & .03 & .03 & .03 \\ .03 & .03 & .03 & .73 & .03 & .03 & .03 & .03 & .03 & .03 \\ .04 & .05 & .05 & .13 & .27 & .27 & .13 & .02 & .03 & .01 \\ .04 & .05 & .05 & .13 & .27 & .27 & .13 & .02 & .03 & .01 \\ .03 & .03 & .03 & .03 & .03 & .03 & .73 & .03 & .03 & .03 \\ .10 & .01 & .03 & .03 & .01 & .01 & .03 & .73 & .03 & .02 \\ .11 & .11 & .11 & .11 & .11 & .11 & .11 & .11 & .11 & .01 \end{pmatrix}. \tag{10}$$

CPTs of the Video Sensor Z_2

$P(z_2 \mid x, o_1, r_1)$

$$= \begin{pmatrix} .91 & .01 & .01 & .01 & .01 & .01 & .01 & .01 & .01 & .01 \\ .01 & .91 & .01 & .01 & .01 & .01 & .01 & .01 & .01 & .01 \\ .01 & .01 & .91 & .01 & .01 & .01 & .01 & .01 & .01 & .01 \\ .01 & .01 & .01 & .91 & .01 & .01 & .01 & .01 & .01 & .01 \\ .01 & .01 & .01 & .01 & .91 & .01 & .01 & .01 & .01 & .01 \\ .01 & .01 & .01 & .01 & .01 & .91 & .01 & .01 & .01 & .01 \\ .01 & .01 & .01 & .01 & .01 & .01 & .91 & .01 & .01 & .01 \\ .01 & .01 & .01 & .01 & .01 & .01 & .01 & .91 & .01 & .01 \\ .01 & .01 & .01 & .01 & .01 & .01 & .01 & .01 & .91 & .01 \end{pmatrix} \tag{11}$$

$P(z_2 \mid x, o_2, r_1)$

$$= \begin{pmatrix} .20 & .06 & .06 & .06 & .05 & .05 & .06 & .20 & .06 & .20 \\ .20 & .05 & .05 & .10 & .05 & .05 & .05 & .20 & .05 & .20 \\ .20 & .05 & .10 & .10 & .10 & .07 & .10 & .07 & .14 & .07 \\ .10 & .05 & .10 & .15 & .125 & .125 & .15 & .05 & .10 & .05 \\ .10 & .05 & .10 & .15 & .15 & .15 & .15 & .05 & .05 & .05 \\ .10 & .05 & .10 & .15 & .15 & .15 & .15 & .05 & .05 & .05 \\ .10 & .05 & .15 & .15 & .10 & .10 & .15 & .05 & .10 & .05 \\ .01 & .01 & .01 & .01 & .01 & .01 & .01 & .01 & .01 & .91 \\ .12 & .12 & .12 & .12 & .12 & .12 & .12 & .12 & .02 & .02 \end{pmatrix} \tag{12}$$

$P(z_2 \mid x, o_1, r_2)$

$$= \begin{pmatrix} .01 & .01 & .01 & .91 & .01 & .01 & .01 & .01 & .01 & .01 \\ .01 & .01 & .01 & .07 & .40 & .40 & .07 & .01 & .01 & .01 \\ .01 & .01 & .22 & .70 & .01 & .01 & .01 & .01 & .01 & .01 \\ .01 & .01 & .01 & .40 & .07 & .07 & .40 & .01 & .01 & .01 \\ .01 & .01 & .01 & .07 & .40 & .40 & .07 & .01 & .01 & .01 \\ .01 & .01 & .01 & .07 & .40 & .40 & .07 & .01 & .01 & .01 \\ .01 & .01 & .01 & .40 & .07 & .07 & .40 & .01 & .01 & .01 \\ .55 & .05 & .05 & .05 & .05 & .05 & .05 & .05 & .05 & .05 \\ .10 & .10 & .10 & .10 & .10 & .10 & .10 & .10 & .10 & .10 \end{pmatrix} \tag{13}$$

$P(z_2 \mid x, o_2, r_2)$

$$= \begin{pmatrix} .03 & .03 & .03 & .73 & .03 & .03 & .03 & .03 & .03 & .03 \\ .15 & .01 & .03 & .07 & .30 & .30 & .07 & .05 & .01 & .01 \\ .10 & .01 & .18 & .65 & .01 & .01 & .01 & .01 & .01 & .01 \\ .08 & .01 & .07 & .25 & .19 & .17 & .20 & .01 & .01 & .01 \\ .05 & .03 & .05 & .12 & .30 & .30 & .12 & .01 & .01 & .01 \\ .05 & .03 & .05 & .12 & .30 & .30 & .12 & .01 & .01 & .01 \\ .08 & .01 & .07 & .20 & .19 & .17 & .25 & .01 & .01 & .01 \\ .03 & .03 & .03 & .03 & .03 & .03 & .03 & .03 & .03 & .73 \\ .10 & .10 & .10 & .10 & .10 & .10 & .10 & .10 & .10 & .10 \end{pmatrix}. \tag{14}$$

REFERENCES

[1] Bundesanstalt für Straenwesen (BAST)
Technische Lieferbedingungen für Streckenstationen.
Abschnitt IV: Prüfungen Verkehrserfassungsgerẗte, 2002.

[2] S.-C. Cheung and C. Kamath
Robust techniques for background subtraction in urban traffic video.
In *Video Communications and Image Processing, SPIE Electronic Imaging*, 2004.

[3] I. Cohen, A. Bronstein and F. G. Cozman
Adaptive online learning of bayesian network parameters.
http://www.hpl.hp.com/techreports/2001/HPL-2001-156.html, 2001.

[4] R. G. Cowell, A. P. Dawid, S. L. Lauritzen and D. J. Spiegelhalter
Probabilistic Networks and Expert Systems.
New York: Springer, 1999.

[5] S. Ferrari and A. Vaghi
Demining sensor modeling and feature-level fusion by Bayesian networks.
IEEE Sensors Journal, **6**, 2 (Apr. 2006), 471–483.

[6] J. Grenard, D. Bullock and A. P. Tarko
Evaluation of selected video detection systems at signalized intersections.
Final Report, FHWA/IN/JTRP-2001/22 (West Lafayette. IN: Indiana Department of Transportation, Division of Research, and Purdue University, Nov. 2001).

[7] D. L. Hall
Mathematical Techniques in Multisensor Data Fusion.
Massachusetts: Artech House, 1992.

[8] D. Heckerman
A tutorial on learning with Bayesian networks.
Technical Report MSR-TR-95-06, Redmond, 1995.

[9] M. Junghans and H.-J. Jentschel
Qualification of traffic data by bayesian network data fusion.
In *Proceedings of the 10th International Conference on Information Fusion*, Québec, Canada, July 2007.

[10] M. Junghans and H.-J. Jentschel
Adaptive learning of Bayesian networks for the qualification of traffic data by contaminated Dirichlet density functions.
In *Proceedings of the IEEE 2008 1st Internatational Conference on Information Technology* (IT 2008), Gdansk, Poland, May 2008, 369–372.

[11] A. Leich
Ein Beitrag zur Realisierung der videobasierten weitrüumigen verkehrsbeobachtung.
Ph.D. thesis, Dresden University of Technology, 2006.

[12] A. Leich, T. Fliess and H.-J. Jentschel
Bildverarbeitung im strassenverkehr—Überblick über den stand der technik.
Zwischenbericht, Dresden University of Technology, 2001.

[13] L. A. Klein
Sensor and Data Fusion: A Tool for Information Assessment and Decision Making.
Bellingham, WA: SPIE Press, 2004.

[14] L. A. Klein
Sensor Technologies and Data Requirements for ITS.
London: Artech House, 2001.

[15] D. Koks and S. Challa
An introduction to Bayesian and Dempster-Shafer data fusion.
Defence, Science & Technology (2002), DSTO-TR-1436.

[16] M. J. Larkin
Sensor fusion and classification of acoustic signals using Bayesian networks.
In *IEEE Proceedings Signals, Systems & Computers*, vol. 2, 1998, 1359–1362.

[17] A. Leonhardt
Detektortest—Kombinationsdetektor ASIM modell 298 TT.
Schlussbericht, Munich University of Technology, 2004.

[18] P. Martin, Y. Feng and X. Wang
Detector technology evaluation.
Research Paper, University of Utah, 2003.

[19] L. E. Y. Mimbela, L. A. Klein, P. Kent, J. L. Hamrick, K. M. Luces and S. Herrera
Summary of Vehicle Detection and Surveillance Technologies used in Intelligent Transportation Systems.
The Vehicle Detector Clearinghouse Federal Highway Administration (FHWA) Intelligent Transportation Systems Program Office, 2003.

[20] P. Molnar and T. R. Collins
Traffic data collection using image processing technology.
In *Traffic Data and Applications*, Berlin-Heidelberg: Springer, 1999, 357–364.

[21] K. P. Murphy
Dynamic Bayesian networks: Representation, inference and learning.
Ph.D. thesis, University of California, Berkeley, 2002.

[22] R. Neapolitan
Learning Bayesian Networks.
Prentice-Hall: Pearson Education, 2004.

[23] J. Pearl
Probabilistic Reasoning in Intelligent Systems: Networks of Plausible Inference.
San Francisco, CA: Morgan Kaufmann Publishers, 1991.

[24] J. Roure and R. Sanguesa
Incremental methods for Bayesian network learning.
In *UAI'99*, http://citeseer.ist.psu.edu/roure99incremental.html, 1999.

[25] L. D. Stone, C. A. Barlow and T. L. Corwin
Bayesian Multiple Target Tracking.
London: Artech House, 1999.

[26] Y. Zhang and Q. Ji
Active and Dynamic information fusion for multisensor systems with dynamic Bayesian networks.
IEEE Transactions on Systems, Man, and Cybernetcis, B, **36**, 2 (2006), 467–472.

[27] Y. Zhang and Q. Ji
Traffic data of Radeburger Strasse/Meinholdstrasse region Dresden.
Civil Engineering Office Dresden, Germany, May 20, 2005.

Marek Junghans was born in Karl-Marx-Stadt, the former GDR, in 1977. He received the Dipl.-Ing. degree in intelligent transportation systems with high distinction from Dresden University of Technology, Dresden, Germany, in 2002 and received the "Friedrich List" Award for his diploma thesis in 2003.

Since 2003 he has been a scientific assistant at the institute for Traffic Communications Engineering in Dresden. His researching interests include methods and applications of stochastic signal processing, adaptive systems, digital image processing for wide area traffic surveillance and data fusion with focus on Bayesian Networks. He is currently a Ph.D. candidate under the supervision of Professor Hans-Joachim Jentschel.

Hans-Joachim Jentschel received all scientific degrees from Dresden University of Transport and Communications "Friedrich List": Dipl.-Ing. for electrical engineering in 1969, Dr.-Ing. for network theory in 1974 and habilitation as Dr.-Ing. habil. for information technology in 1986.

During 1975–79 he was a scientific staff member in the group of Basic Research for Electrical Engineering with ROBOTRON Dresden Company, Dresden, GDR. From 1979 to 1984 he served as a project manager in the area of interactive graphic man-machine interfaces in the Joint Institute for Nuclear Research (JINR) in Dubna, Russia. In 1984 he returned to Dresden University of Transport and Communications working as a lecturer. Since 1987 Dr. Jentschel had been the head of the professorship "Information Technologies." Since 1992 he has been working as a full professor at Dresden University of Technology, Institute of Traffic Information Engineering. He is a member of IEEE, VDE and ITG. Current scientific interests include stochastic signal processing and adaptive systems.

Information for Authors

The Journal of Advances in Information Fusion (JAIF) is a semi-annual, peer-reviewed archival journal that publishes papers concerned with the various aspects of information fusion. The boundaries of acceptable subject matter have been intentionally left flexible so that the journal can follow the research activities to better meet the needs of the research community and members of the International Society of Information Fusion (ISIF).

Restrictions on Publication: The International Society of Information Fusion (ISIF) publishes in JAIF only original material that has not been either published or submitted for publication in an archival journal. However, prior publication of an abbreviated and/or preliminary form of the material in conference proceedings or digests shall not preclude publication in JAIF when notice is made at the time of submission and the previous publications are properly referenced in the manuscript. Notice of prior publication of an abbreviated version of the manuscript must be given at the time of submission to the transactions. Submission of the manuscript elsewhere is precluded unless it is refused for publication or withdrawn by the author. Concurrent submission to other publications and this journal is viewed as a serious breach of ethics and, if detected, the manuscript under consideration will be rejection.

Sanctions for Plagiarism: If a manuscript is substantially similar to any previous journal submission by any of the authors, its history and some explanation must be provided. If a manuscript was previously reviewed and rejected by a different journal, the authors shall provide to the area editor copies of all correspondence involving the earlier submission. In addition, the authors must discuss the reasons for resubmission and be prepared to deliver further material if more is requested. Submission of a previously rejected manuscript is acceptable as long as the motive for submission to JAIF is explained and the authors have addressed the feedback of the earlier reviewers. The authors of a manuscript that is found to have been plagiarized from others or contain a crossover of more than 25% with another journal manuscript by any of the authors will incur sanctions by the ISIF. The following sanction will be applied for any of the above-noted infractions: (1) immediate rejection of the manuscript in question; (2) immediate withdrawal of all other submitted manuscripts by any of the authors to any of the ISIF publications (journals, conferences, workshops); (3) prohibition against all of the authors for any new submissions to ISIF publications, either individually, in combination with the authors of the plagiarizing manuscript as well as in combination with new coauthors. The prohibition shall continue for at least two years from notice of suspension.

Types of Contributions: Contributions may be in the form of regular papers or correspondence. The distinction between regular papers and correspondence is not one of quality, but of nature. Regular papers are to be a well-rounded treatment of a problem area, while a correspondence item makes one or two points concisely. In regular papers, the title, abstract, and introduction should be sufficiently informative to illuminate the essence of the manuscript to the broadest possible audience and to place the contributions in context with related work. The body of the manuscript should be understandable without undue effort by its intended audience. Correspondence should be less discursive but equally lucid. Authors should be aware that a well-written correspondence are usually published more rapidly than a regular paper.

Submission of Manuscript for Review: Manuscripts are submitted electronically via the internet at http://jaif.msubmit.net. For peer review, manuscripts should be formatted in a single column and double spaced.

General Information for JAIF Authors and Submission of Final Manuscript: General information for JAIF authors is available at http://www.isif.org/jaif.htm, and specific document preparation information for final manuscript for publication is given below and can be found at http://www.isif.org/JAIF Manuscript Preparation1.doc.

Copyright: It is the policy of the ISIF to own the copyright to the technical contributions it publishes; authors are required to sign an ISIF copyright transfer form before publication. This form appears in JAIF from time to time and is available on the web site listed above. If a copyright form is not submitted with the manuscript, it will be requested upon contribution acceptance. Publication will not take place without a completed copyright form.

Page Charges: Since ISIF has elected to support the publication of JAIF with the revenues of conferences and membership dues, page charges are not mandated for publication in JAIF. Page charges for journals of professional societies are traditional, and at some time in the future, page charges may be introduced to cover the expenses associated with the publication of JAIF. Payment of page charges will not be a prerequisite for publication.

Discloseability: The ISIF must, of necessity, assume that material submitted for publication is properly available for general dissemination to the audiences for which JAIF serves. It is the responsibility of the author, not ISIF, to determine whether disclosure of materials requires prior consent of other parties, and, if so, obtain it.

Preparation of Manuscript for Publication: After a manuscript is recommended for publication by the editorial staff, the corresponding author prepares the manuscript according to the guidelines below and submits the final version of the manuscript as a pdf file and the supporting manuscript files. Before preparing their manuscript for publication, authors should check for the more update version of the guidelines at http://jaif.msubmit.net.

All manuscripts must be submitted electronically in pdf format with the supporting files in one of the following formats: Plain TeX, AMS-TeX, LaTeX, or MS-Word with the MathType extension for equations and in-text mathematical material. The pdf file must be created with graphics resolution set to 300 dpi or greater. The author must approve the pdf file because it will be used as the basis for rectifying any inconsistencies in the supporting files. The supporting files must be prepared according to the following guidelines.

- Text files should include title, authors, affiliations, addresses, a footnote giving the dates for submission and revision and name of the editor that handed the review, and an optional footnote acknowledging a sponsor for the research. The information must be included in the text files in this order.
- Abstract must be included. The abstract must include the key words for the manuscript and include no more than 300 words for a regular paper and 150 words for a correspondence.
- Authors should number main section headings as 1, 2, 3, etc., and subsections as 1.1, 1.2, or 2.1. All headings should be typed with title format (i.e., first letter caps and lower case)—not in ALL CAPS. The typesetter will convert to all caps in final formatting as required.
- Authors should format references very carefully according to the examples given below. Style for authors' names are initials followed by last name and it must be followed precisely. References must be listed alphabetically by (i) last name of first author, (ii) initials of first author, (iii) last name of second author, (iv) initials of second author, etc. For manuscripts with common authors and order of authors, the date of publication should be used to select order with the earlier publications being list first. The names of publications must be spelled completely. In the references, the author should use only approved abbreviations that include: No., Vol., p., pp., AIAA, IEEE, IEE, and SPIE.
- Authors who use one of the TeX variants must provide a list of their macros and the files for the macros. Authors who use LaTeX must include the bbl and aux files.
- All figures must be submitted electronically as HIGH resolution (300 dpi or better) in Color graphics or GRAYSCALE (where shading is involved) or BLACK AND WHITE (if simple line art) graphics files (tif, eps, jpg, bmp). Each figure must be supplied as a separate graphics file. Graphics (or captions) should NOT be embedded in the text files. The figures must be included in the pdf file of the full article and the pdf file must be created with graphics resolution set to 300 dpi or greater.
- A separate file including all figure captions must be included.
- Each table must be submitted in a separate file. Tables (or captions) should NOT be included in text files and should be in form of DATA—rather than graphics—files.
- A separate file including all table captions must be included.
- A separate text of the biography of each author must be submitted. The text file should be less than 500 words.
- Separate graphics files of each author's photo should be provided as a grayscale graphics file or a color graphics file.

Examples of the references are alphabetized correctly and listed below.

BOOK:

[1] R. E. Blahut
Theory and Practice of Error Control Codes. Reading, MA: Addison-Wesley, 1983.

PROCEEDINGS ARTICLE:

[2] T. Fichna, M. Gartner, F. Gliem, and F. Rombeck
Fault-tolerance of spacebome semiconductor mass memories.
In Twenty-Eighth Annual International Symposium on Fault-Tolerant Computing, Digest of Papers, 1998, 408–413.

BOOK:

[3] P. K. Lala
Fault Tolerant and Fault Testable Hardware Design. Englewood Cliffs, NJ: Prentice-Hall, 1985.

WEB SITE:

[4] National Semiconductors Inc.
Homepage: http://www.national.com.

PROCEEDINGS ARTICLE:

[5] C. Paar and M. Rosner
Comparison of arithmetic architectures for reed-solomon decoders in reconfigurable hardware.
In *Proceedings of the Symposium on Field-Programmable Custom Computing Machines*, Apr. 1997, 219–225.

JOURNAL ARTICLE:

[6] N. R. Saxena and E. J. McCluskey
Parallel signature analysis design with bounds on aliasing.
IEEE Transactions on Computers, **46**, 4 (Apr. 1997), 425–438.

[7] C. I. Underwood and M. K. Oldfield
Observations on the reliability of cots-device-based solid state data recorders operating in low-Earth orbit.
IEEE Transactions on Nuclear Science, **47**, 4 (June 2000), 647–653.

WEB SITE:

[8] Xilinx Inc.
Homepage: http://www.xilinx.com.

INTERNATIONAL SOCIETY OF INFORMATION FUSION

ISIF Website: http://www.isif.org

SOCIETY VISION

The International Society of Information Fusion (ISIF) seeks to be the premier professional society and global information resource for multidisciplinary approaches for theoretical and applied information fusion technologies.

SOCIETY MISSION

Advocate

To advance the profession of fusion technologies, propose approaches for solving real-world problems, recognize emerging technologies, and foster the transfer of information.

Serve

To serve its members and engineering, business, and scientific communities by providing high-quality information, educational products, and services.

Communicate

To create international communication forums and hold international conferences in countries that provide for interaction of members of fusion communities with each other, with those in other disciplines, and with those in industry and academia.

Educate

To promote undergraduate and graduate education related to information fusion technologies at universities around the world. Sponsor educational courses and tutorials at conferences.

Integrate

Integrate ideas from various approaches for information fusion, and look for common threads and themes—look for overall principles, rather than a multitude of point solutions. Serve as the central focus for coordinating the activities of world-wide information fusion related societies or organizations. Serve as a professional liaison to industry, academia, and government.

Disseminate

To propagate the ideas for integrated approaches to information fusion so that others can build on them in both industry and academia.

Call for Papers

The Journal of Advances in Information Fusion (JAIF) seeks original contributions in the technical areas of research related to information fusion. Authors of papers in one of the technical areas listed on the inside cover of JAIF are encouraged to submit their papers for peer review at http://jaif.msubmit.net.

Call for Reviewers

The success of JAIF and its value to the research community is strongly dependant on the quality of its peer review process. Researchers in the technical areas related to information fusion are encouraged to register as a reviewer for JAIF at http://jaif.msubmit.net. Potential reviewers should notify via email the appropriate editors of their offer to serve as a reviewer.

Additional Copies

If you need additional copies, please contact Tammy Blair at 404-407-7726 or tammy.blair@gtri.gatech.edu for pricing information.

Journal of Advances in Information Fusion Volume 3 Number 1 June 2008 ISSN 1557-6418