INTERACTING MULTIPLE MODEL METHODS IN MULTITARGET - MULTISENSOR TRACKING: SURVEY

E. Mazor[§] A. Averbuch [†] Y. Bar-Shalom[‡] J. Dayan[§]

[§]Department of Mechanical Engineering
Technion, Israel Institute of Technology
Haifa 32000, Israel

[†]School of Mathematical Sciences
Tel Aviv University, Tel Aviv 69978, Israel

[‡]Department of Electrical and Systems Engineering
University of Connecticut
Storrs, Connecticut, USA

Abstract

The Interacting Multiple Model (IMM) estimator is a suboptimal hybrid filter that has been shown to be one of the most cost-effective hybrid state estimation schemes. The importance of this approach is that it is the best compromise available currently between complexity and performance: its computational requirements are linear in the size of the problem (number of models) while its performance is almost the same as that of an algorithm with quadratic complexity.

The objective of this paper is to survey and put in perspective the existing IMM methods for Multitarget-Multisensor Tracking problems. Special attention is given to the assumptions underlying each algorithm and its applicability to various situations.

1 Introduction

Considerable research has been undertaken in the field of optimal estimation theory in relation to the Multitarget-Multisensor Tracking (MTMST) problem. This is of interest in both military and civilian applications. Ballistic Missile Defense and Airborne Surveillance require identification and tracking of several hundred targets in real time. A typical scenario includes maneuvering and non-maneuvering targets as well as environmental clutter and false alarms produced by the sensor receiver noise and atmospheric disturbances. These problems have a wide variety of direct applications such as sophisticated weapon delivery systems, satellite surveillance systems, air defense, ocean/battlefield surveillance, air traffic control (ATC), and non-military vehicle tracking systems. In addition, the techniques have general application to statistical pattern recognition and robotics, etc. The field has been investigated by many researchers during the past two decades. The main purpose of a tracking system for traffic control or air defense is the estimation of target trajectories in the controlled area and their prediction into the near future. In a civilian ATC system, the estimated trajectories (tracks) are used to check the standard separation between pairs of targets for maintenance of safety conditions (collision avoidance) and regularity of traffic flow. In a maritime collision avoidance system, each ship monitors the traffic situation in order to detect the possibility of colliding targets. In an air defense system, the estimated trajectories are generally used to help perform some of the following functions [FARINA 80]: threat identification, threat evaluation, weapon assignment, calculation of the predicted position (for fire or launch of missile), kill evaluation.

Although the mathematical structure of the optimal solution of the MTMST problem is well understood [BARSHALOM 88, BARSHALOM 95, BLACKMAN 86], the excessive computational complexity of the optimal tracking algorithm — the Multiple Hypothesis Tracker (MHT) prohibits its practical realization using even the largest and fastest computers available in the foreseeable future. Suboptimal implementations of the MHT have many potential pitfalls. For this reason, many different tracking algorithms have been developed which sacrifice optimal performance for the sake of computational feasibility. Intuition suggests that some suboptimal approximations to the optimal algorithms result in little performance loss, but there exists no reliable quantitative analysis of suboptimal performance that demonstrates that this is, in fact, the case. The basic research problem is to find effective analytical techniques for the estimation problems. MTMST problems involve both the estimation of continuous-valued parameters such as target positions, velocities and accelerations, and the testing of discrete hypotheses, such as the association of measurement data with targets. Thus, it is natural to pose the problem of MTMST as a hybrid state estimation problem, that is, estimation for a partially observed stochastic process with discrete- and continuous-valued states. The hybrid state viewpoint [BARSHALOM 89b, CAMPO 90, DUFOUR 92, PATTIPATI 83, WASHBURN 85] provides both a natural formulation for many types of surveillance and tracking problems and a powerful framework for deriving theoretically optimal and practical suboptimal tracking algorithms. Hybrid systems are characterized by:

- State (consisting of kinematic and, possibly, feature components) that evolves according to a model, and

- Model that is governed by a discrete stochastic process: it is one of a finite number of possible models (each corresponds to a behavior mode), that undergo jumps (switches) from one model (behavior mode) to another according to a set of transition probabilities (the Multiple Model (MM) approach).

The value of hybrid models for tracking algorithms is that the occurrence of target maneuvers can be explicitly included in the kinematic equations through regime jumps [BLOM 92, MARITON 92]. The MM adaptive estimation approach is based on the fact that the behavior of a target cannot be characterized at all times by a single model, but a finite number of models can adequately describe its behavior in different regimes. Several approaches have been proposed to perform the state estimation of a system together with identification of each model (out of a finite set). Although static (i.e., nonswitching) MM estimation algorithms have been known since the mid 1960s, practical algorithms for the switching case have only recently become available. A unified treatment and a brief survey of the various suboptimal state estimation and system structure detection algorithms were presented in [TUGNAIT 82] for discrete systems with abruptly changing structure where the changes are modeled by a finite state Markov chain. The algorithms considered in this paper assume that the transition probability matrix governing the Markov chain is known. The proposed approach assumes that the actual (true) system is one of finite number of possible models and that the a priori probability of each model is known. Modeling with Markov jump parameters may be thought of as an extension of the MM approach (or partitioning approach). In the former, Tugnait allows switchings among the hypothesized models unlike the static MM approach where a candidate model is assumed to describe the true system behavior for all time. In several applications, switching parameter modeling is more realistic [PATTIPATI 83] because the underlying true system does have time-varying characteristics. Note that partitioning is a powerful approach resulting in parallel processing algorithms and robust solutions [LU 92, MAYBECK 90, SUNG 87]. The case when the transition probability matrix is unknown deserves attention and has been considered by several authors for some restricted versions of the general problem [MILLNERT 80, TUGNAIT 82]. Extension to the case where jumps in the system structure are modeled by a semi-Markov chain has been discussed in [MOOSE 75].

A unified tutorial presentation on the topic of MM estimation and other related techniques from a target tracking oriented perspective can be found in [BARSHALOM 88, BARSHALOM 95, BLACKMAN 86]. Filtering algorithms for hybrid-state estimation in general consist of a bank of Kalman filters (KF) and some algorithm to organize the co-operation between the individual KF. Compared to other MM approaches the main advantage of hybrid system-based solutions is that they are not subject to the so-called "obliviousness to detection" difficulty: the a priori jump probabilities used in the regime dynamics make hybrid algorithms alert to regime changes whereas classical solutions are biased toward the no-maneuver hypothesis after long quiescent periods [BOGLER 87]. One of the significant schemes is the so-called Generalized Pseudo Bayesian (GPB) method [CHANG 78, TUGNAIT 82] and the other is the Interacting Multiple Model (IMM) algorithm [BLOM 84, BLOM 88]. The general structure of these algorithms consists of a bank of filters for the state cooperating with a filter for the parameters. A GPB algorithm of order k (GPBk) needs N_f^k filters in its bank [TUGNAIT 82], where N_f is the number of models in the bank and k is the number of sampling periods over which mode jumps are considered. The IMM algorithm is conceptually similar to GPB2 and performs nearly as well as the GPB2 method, but requires only N_f filters to operate in parallel,i.e., it has notably less computations, namely, the cost of GPB1 [BLOM 88]. The IMM algorithm, originally proposed by [BLOM 84], is a suboptimal hybrid filter that was shown to achieve the best compromise between performance and complexity. Its complexity is linear in the size of the problem (number of models) while its performance is almost the same as that of the GPB2. Another advantage of the IMM is its modularity. It can be set up using KF or extended KF (EKF) as its building blocks to account for nonlinearities in the motion equation (e.g., for coordinated turns) and in the measurement equation for range-azimuthal-elevation observations, or probabilistic data association filters (PDAF) and joint PDAF (JPDAF), based on KF or EKF, if data association (due to false alarms or neighboring targets) is a significant problem.

In the framework of the Hadamard project of the French Civil Administration, advanced tracking methods were investigated [VACHER 92]. The project deals with the aircraft motion modeling and shows that stringent speed estimation can be obtained only with multiple model algorithms. Vacher's work [VACHER 92] is devoted to a multimodel representation of trajectories and the description of the IMM algorithm that was proven suitable for this problem.

The formulation was developed by [BLOM 84], who based it on the assumption that the governing system equations describe a Markov chain process. The state of this chain represents the mode of the target flight during a particular time interval. This mode is assumed to be among the possible modes of the set of all modal states at all times. The set of modes consists, for example, of a constant velocity model and a unique maneuver model. The process is characterized by its a priori transition probabilities, which constitute the transition matrix. These are design parameters for the algorithm.

2 Problem Formulation

A general description for a hybrid system with additive noise is [LI 92a]

$$x(k+1) = f[k, x(k), m(k+1)] + g[k, x(k), m(k+1)]v[k, m(k+1)] \tag{2.1}$$

with mode-dependent noisy measurements

$$z(k) = h[(k, x(k), m(k)] + w[k, m(k)] \tag{2.2}$$

and a (possibly base state dependent) Markovian transition of the system mode

$$P\{m_j(k+1)|m_i(k)\} = \phi[k, x(k), m_i, m_j] \quad \forall m_i, m_j \in M_s \tag{2.3}$$

where $x(\cdot)$ is the base state, $m(k)$ is the modal state (system mode index) at time k, which denotes the mode in effect during the sampling period ending at k, $P\{\cdot\}$ is probability, $m_j(k) \triangleq \{m(k) = j\}$ is the event that mode j is in effect at time k, M_s is the set of all modal states at all times, $v[\cdot]$ and $w[\cdot]$ are the mode-dependent process and measurement noise sequences with means $\bar{v}$ and $\bar{w}$ and covariances Q and R, respectively. It is assumed that v and w, uncorrelated with $x(0)$, are mutually uncorrelated and that f, g and h are known. Also $x(0)$ is assumed to be Gaussian with appropriate mean and covariance. All vectors and matrices are assumed to have appropriate dimensions. Note that (2.2) implies the availability of noisy, base state information only and that the mode information is embedded in the measurement sequence. This description covers the case of nonstationary noises by modeling the noise sequences as jumping from one set of parameters to another one.

In a fixed structure (or fixed mode-set) hybrid system, a set of modes must be determined in advance. Denoting by M_f this fixed set of N_f modes, the system (2.1)–(2.3) is then approximated as

$$x(k+1) = f_j[k, x(k)] + g_j[k, x(k)]v_j(k) \quad \forall j \in M_f \tag{2.4}$$

$$z(k) = h_j[k, x(k)] + w_j(k) \quad \forall j \in M_f \tag{2.5}$$

and jumps between the system modes are modeled by switching from one model to another, governed by the same law as given in (2.3). Here, N_f is the cardinality of the set M_f, i.e., the possible number of system modes, and subscript j denoted quantities pertaining to model (mode) m_j.

A jump linear fixed structure hybrid system with mode transition modeled by a semi-Markov process, such as the state-dependent Markov chain, can be described by the equations [BLOM 85, BLOM 90, BLOM 91, BLOM 92, CAMPO 88, CAMPO 90, CAMPO 91, MOOKERJEE 87]

$$x(k+1) = F_j(k)x(k) + v_j(k) \quad \forall j \in M_j \tag{2.6}$$

$$z(k) = H_j(k)x(k) + w_j(k) \quad \forall j \in M_j \tag{2.7}$$

The model switching process considered is of the semi-Markov type. The process is specified by a family of transition matrices $p_{ij}(\tau_i)$, where τ_i is the system's sojourn time in model i, i.e., it is a "sojourn-time-dependent Markov" (STDM) chain, which belongs to the semi-Markov class. The current probabilities of transition for the STDM process (chain) are defined as

$$p_{ij}(\tau) = P\{m_j(k+1)|m_i(k), \tau_i(k) = \tau\} \quad \forall i, j \in M_f \tag{2.8}$$

where $\tau_i(k)$ is the sojourn time in state i at time k. It is assumed that at $k = 0$ the sojourn time (in whatever state the system model is) is $\tau = 1$. Thus, the values τ are taken from 1 to the maximum, which at time k is then $k + 1$. The process and measurement noises are Gaussian mutually uncorrelated with zero mean and known covariances.

The simplest hybrid system is one that can be described by a set of linear models as

$$x(k+1) = F_j(k)x(k) + \Gamma_j(k)v_j(k) \quad \forall j \in M_f \tag{2.9}$$

$$z(k) = H_j(k)x(k) + w_j(k) \quad \forall j \in M_f \tag{2.10}$$

with the mode transition governed by a first-order homogeneous Markov chain

$$P_{ij}\{m_j(k+1)|m_i(k)\} = p_{ij} \quad \forall i, j \in M_f \tag{2.11}$$

where p_{ij} is the Markov transition probability from mode i to mode j. It is assumed, as before, that the process and measurement noises are Gaussian mutually uncorrelated with zero mean and known covariances.

When the Markovian switching coefficients can be isolated to a system bias, such linear hybrid system can be represented as [BLAIR 92]

$$x(k+1) = F(k)x(k) + \Gamma_j(k)b(k) + \Gamma^x(k)v^x(k) \tag{2.12}$$

$$b(k+1) = D_j(k)b(k) + \Gamma_j^b(k)v^b(k) \tag{2.13}$$

$$z(k) = H(k)x(k) + G_j^b(k)b(k) + w(k) \tag{2.14}$$

with bias transition governed by a finite state Markov chain

$$P_{ij}\{m_j^b(k+1)|m_i^b(k)\} = P_{ij} \qquad \forall i,j \in M_f \tag{2.15}$$

where $b(k)$ is the system bias, p_{ij} is the Markov transition probability from bias model m_i^b to bias model m_j^b; $v^x(k)$ and $v^b(k)$ are white Gaussian errors for the system state and bias process respectively; $w(k)$ is the white Gaussian errors in the measurements and independent of $v^x(k)$ and $v^b(x)$; it is assumed that covariances of $v^x(k)$, $v^b(k)$ and $w(k)$ are known $(Q^x(k), Q^b(k)$ and $R(k)$ respectively); the functions $F(k), \Gamma_j(k), \Gamma^x(k)$ are known also.

The problem of hybrid state estimation is to estimate the base state and the modal state based on the measurement sequence.

3 IMM Algorithm and Its Varieties

The IMM algorithm has been shown to be one of the most cost-effective and simple schemes for the estimation in hybrid systems [AVERBUCH 91a, BARSHALOM 90a, BARSHALOM 90b, BARSHALOM 92a, BARSHALOM 95, GUU91, HOULES 89] and therefore it is suitable to be used in MTMST. The IMM design parameters are:

1. The set of models for the various regimes and their structure

2. The process noise intensities for the various models, in particular the nonmaneuvering model with low-level process noise and the maneuvering model(s) with certain higher noise levels determined by the assumed maneuverability of the targets.

3. The jump structure (usually Markov) and probabilities between the models from the selected set. These probabilities are chosen according to the designer's beliefs about the frequency of the regime switches and can be subsequently adjusted based on Monte Carlo simulation results. When the transition probabilities of the Markov chain are zero the IMM algorithm reduces to the well-known static MM algorithm.

The IMM algorithm has three desirable properties: it is recursive, modular and has fixed computational requirements per cycle. In each cycle it consists of three major steps: interaction (mixing), filtering, and combination. At each time, the initial condition for the filter matched to a certain mode (a module) is obtained by mixing the state estimates of all filters at the previous time under the assumption that this particular mode is in effect at the current time. This is followed by a regular filtering (prediction and update) step, performed in parallel for each mode. Then, a combination (weighted sum) of the updated state estimates of all filters yields the state estimate. The probability of a mode, which is being in effect plays a key role as the weighting of the mixing and the combination of states and covariances.

3.1 Baseline IMM Algorithm

The baseline IMM algorithm [BARSHALOM 89a, BARSHALOM 93, BLOM 84, BLOM 88] assumes the simplest form of hybrid system (2.9)-(2.11); each mode-matched filter is a standard KF.

Notations: Z^k denotes the measurement sequence through time k; $\hat{x}(i|j)$ denotes the state estimate at time i conditioned on Z^j, and $P(i|j)$ is the associated covariance matrix; quantities pertinent to mode j are denoted with subscript j; $N(y;\bar{y},P)$ denotes the (multivariate) Gaussian density function of y with mean $\bar{y}$ and covariance P; $\sum_i$ stands for $\sum_{m_i \in M_f}$; $\hat{x}_j(k|k)$ and $P_j(k|k)$ are the state estimate and its covariance in mode-matched filter j at time k; $\hat{x}_{0j}(k|k)$, $P_{0j}(k|k)$ are the mixed initial condition for mode-matched filter j at time k; $\hat{x}(k|k)$, $P(k|k)$ are the combined state estimate and its covariance; $\mu_j(k)$ is the mode probability at time k; $\mu_{i|j}(k|k)$ is the mixing probability at time k; $\Lambda_j(k)$ is the likelihood function of mode-matched filter j.

Summary of the basic IMM algorithm (one cycle):

Interaction - $\forall i,j \in M_f$ $\mu_{i|j}(k-1|k-1) = \frac{1}{\bar{c}_j} p_{ij}\mu_i(k-1)$ (mixing probability), where $\bar{c}_j$ is a normalization factor.

$$\bar{c}_j = \sum_i p_{ij}\mu_i(k-1)$$

$$\hat{x}_{0j}(k-1|k-1) = \sum_i \hat{x}_i(k-1|k-1)\mu_{i|j}(k-1|k-1)$$

$$P_{0j}(k-1|k-1) = \sum_i \Big\{P_i(k-1|k-1) + [\hat{x}_i(k-1|k-1) - \hat{x}_{0j}(k-1|k-1)]$$
$$\cdot [\hat{x}_i(k-1|k-1) - \hat{x}_{0j}(k-1|k-1)]^T\Big\}\mu_{i|j}(k-1|k-1)$$

Filtering - $\forall i,j \in M_f$

$$\hat{x}_j(k|k-1) = F_j(k-1)\hat{x}_{0j}(k-1|k-1) + \Gamma_j(k-1)\bar{v}_j(k-1)$$

$$P_j(k|k-1) = F_j(k-1)P_{0j}(k-1|k-1)F_j(k-1)^T$$
$$+\Gamma_j(k-1)\left[W_j(k-1) + \bar{v}_j(k-1)\bar{v}_j(k-1)^T\right]\Gamma_j(k-1)^T$$

$$\hat{x}_j(k|k) = \hat{x}_j(k|k-1) + W_j(k)r_j(k)$$

$$P_j(k|k) = P_j(k|k-1) - W_j(k)S_j(k)W_j(k)^T$$

$$r_j(k) = z(k) - \hat{z}_j(k|k-1) \text{ (residual)}$$

$$\hat{z}_j(k|k-1) = H_j(k)\hat{x}_j(k|k-1) \text{ (measurement prediction)}$$

$$S_j(k) = H_j(k)P_j(k|k-1)H_j(k)^T + R_j(k) \text{ (residual covariance)}$$

$$W_j(k) = P_j(k|k-1)H_j(k)^T S_j(k)^{-1} \text{ (filter gain)}$$

$$\Lambda_j(k) = N(r_j(k);0,S_j(k)) \text{ (likelihood function)}$$

$$\mu_j(k) = \frac{1}{c}\Lambda_j(k)\sum_i p_{ij}\mu_i(k-1) = \frac{1}{c}\Lambda_j(k)\bar{c}_j \text{ (mode probability, } c \text{ is a normalizing factor)}$$

Combination - $\forall j \in M_f$

$$\hat{x}(k|k) = \sum_j \hat{x}_j(k|k)\mu_j(k)$$

$$P(k|k) = \sum_j \left\{P_j(k|k) + [\hat{x}_j(k|k) - \hat{x}(k|k)]\,[\hat{x}_j(k|k) - \hat{x}(k|k)]^T\right\}\mu_j(k)$$

Following [BLOM 92] the outline principles of the track maintenance system are: Once a track has been initiated, the track maintenance system has to continue following that track as long as it is observed by at least one sensor. Hence, a multisensor track-continuation problem is reduced to a "single-sensor" problem, where the updating is sequential across the sensors. Track continuation has to cope with several uncertainties, of which the following four cause the major complications:

- Nonlinear target dynamics during a turn,
- The association of measurements with existing tracks,
- Gaussian-mixture-type measurement noise,
- Sudden starts and stops of maneuvers (mode switching).

In developing a high-quality track continuator the Bayesian approach is employed. It consists of building appropriate stochastic models; developing the exact filter solution from the Bayesian theory estimation for Markov processes; and finally introducing efficient numerical approximations of the exact solution. By following these three steps, several approximate Bayesian methods were developed for each of the subproblems the above mentioned (major complications) of tracking. The various approximate Bayesian methods show a trade-off between computational load and performance, depending on the particular application.

To cope with the first subproblem, the well-known extended Kalman method of linearization along the predicted path is adopted as a good compromise. Of the various approximate methods to cope with the second subproblem, a multiple-model probabilistic data association method [BARSHALOM 88, BARSHALOM 93, BARSHALOM 95] is judged to be the best compromise to update an aircraft track from new observations. Moreover, the PDA approach allows us to include an efficient solution to the third subproblem. To cope with the fourth subproblem, the aircraft behavior is modeled as a hybrid state Markov process. An aircraft trajectory can be subdivided into distinct segments, corresponding to modes of flight. An appropriate model for the switching between modes is a finite-state (semi-) Markov process. An aircraft is then modeled as having a hybrid state; consisting of a Euclidean-values (diffusive) state component and a discrete-values state component. The discrete state describes the mode of flight. The diffuse state describes the horizontal position, ground speed, course, and, for some modes, bank angle, flight-path angle, and transversal acceleration. The evolution in time of the discrete state (i.e., switching between modes) and the diffusive state are determined by the physical relations between the state components, the jump-type changes of the control commands of the pilot, and disturbances caused by both the pilot and the wind. In this model, switching from one mode to another generally causes a simultaneous random jump in bank angle, flight-path angle, and acceleration. The combination of discrete and diffusive states in a maneuvering trajectory model causes the associated tracking problem to fall within the class of so-called hybrid-state estimation problems. Filtering algorithms for hybrid-state estimation, in general, consist of a bank of Kalman filters and some algorithm to organize the cooperation between the individual Kalman filters.

Efficient cooperation between the KFs is realized by an interaction between the estimates (representing the conditional mean and covariance given mode is in effect) for the different modes at the beginning of each filter cycle. The interaction is determined by the conditional probabilities of switching between modes. For problems like tracking, the IMM interaction is so effective, that IMM algorithm performs almost like the exact Bayesian filter [BLOM 88]. Moreover, the IMM requires a far lower computational power than other, recently developed, high-performance algorithms for tracking maneuvering aircraft [BLOM 88, BOGLER 87, BARSHALOM 89a].

3.2 STDM-Based IMM Estimator

In this approach [BLOM 85, BLOM 90, BLOM 91, BLOM 92, CAMPO 88, CAMPO 90, CAMPO 91, MOOKERJEE 87] the hybrid system is modeled by the equations (2.6)-(2.8); each mode-matched filter is a standard KF. Each model transition probability is a known function of the sojourn time given by (2.8) and has a sojourn time $\tau_i(k)$ in state i which is, however, not known. The conditional probability mass function (pmf) of the sojourn time in state $m(k) = i$ based on the available information Z^k at time k, is

$$g_i^k(\tau) \triangleq \left\{\tau_i(k) = \tau \mid m(k) = i, Z^k\right\} = P\left\{\tau_i(k) = \tau \mid m(k) = i, Z^{k-1}\right\}$$

$$= P\left\{m(k-1) = i, \ldots, m(k-\tau+1) = i, m(k-\tau) \neq i \mid m(k) = i, Z^{k-1}\right\} \tag{3.1}$$

where the perfect knowledge of the state $m(k)$ allows one to go down to one index less in the conditioning, i.e., Z^{k-1}. The conditional pmf of the sojourn time τ in state i at time k (3.1) is given by the expressions in [CAMPO 88, CAMPO 91]. Following (2.8) the conditional probability of transition from i to j at time $k-1$, given the observations Z^{k-1} is, in terms of (3.1),

$$\hat{p}_{ij}(k-1) \triangleq P\left\{m(k) = j \mid m(k-1) = i, Z^k\right\} = \sum_{\tau=1}^{k} p_{ij}(\tau) g_i^{k-1}(\tau) \tag{3.2}$$

Note that the argument of p_{ij}, defined in (2.8), is the sojourn time, while the argument of $\hat{p}_{ij}$, defined above is the current time. The filter has access only to the observations form which the conditional pmf of the sojourn time can be obtained: this, in turn, is to be used in calculation of the conditional transition probabilities (3.2).

There are numerous examples of systems which have discrete models that randomly vary with time and experience switchings, between different models, after a random sojourn time. In some situations the switching probabilities depend on the sojourn time. Such situations are commonly experienced in MTMST [BARSHALOM 88, BARSHALOM 95, BLACKMAN 86, BOGLER 90, WALTZ 90]. The model switching process is characterized by a family of transition probability matrices $p_{ij}(\tau_i)$, i.e., it is STDM chain, which belongs to the semi-Markov class. The example [CAMPO 88, CAMPO 91] shows that STDM-based IMM estimator gives a stable estimation filter for both two- and three-model cases. The three-model filter is shown to give a small probability of error in system structure detection and thus the true system model becomes rapidly known, which is critical in failure detection schemes.

Summary of the STDM-based IMM estimator (one cycle):

Interaction - $\forall i, j \in M_f$

$$\mu_{i|j}(k-1|k-1) = \frac{1}{\bar{c}}\hat{p}_{ij}(k-1)\mu_i(k-1) \text{ (mixing probability)}$$

The conditional probability of transition from i to j at time $k-1$, given the observations Z^{k-1} is

$$\hat{p}_{ij}(k-1) = \sum_{\tau=1}^{k} p_{ij}(\tau) g_i^{k-1}(\tau)$$

$$\bar{c}_j = \sum_i \hat{p}_{ij}(k-1)\mu_i(k-1)$$

$\hat{x}_{0j}(k-1|k-1)$ and $P_{0j}(k-1|k-1)$ are defined in section 3.1.

Filtering - from section 3.1

$$\mu_j(k) = \frac{1}{c}\Lambda_j(k)\sum_i \hat{p}_{ij}(k-1)\mu_i(k-1) = \frac{1}{c}\Lambda_j \bar{c}_j$$

Combination - (see section 3.1)

3.3 IMM Algorithms for Systems with Markovian Switching Coefficients Isolated to a System Bias

The Interacting Multiple Bias Model (IMBM) algorithm [BLAIR 92] integrates the baseline IMM algorithm and recent developments in two-stage state estimation. The IMBM algorithm consists of a filter for the bias-free portion of the state model, a filter for each bias model, a model probability evaluator for each bias model filter, an estimate mixer at the input of the bias filters, and an estimate combined at the output of the bias filters. With the assumption that the model switching is governed by an underlying Markov chain, the mixer uses the model probabilities and the model switching probabilities to compute a mixed bias estimate for each bias filter. The bias-free filter error covariance is also modified to reflect the spread of the means of the bias filters. At the beginning of filtering cycle, each bias filter uses a mixed bias estimate and the bias-free measurement residual to compute a new estimate of the bias and a likelihood for the bias model. The likelihoods, prior model probabilities, and the model switching probabilities are then used to compute new model probabilities. The overall bias estimate and compensated state estimate are then computed with the output of the bias-free filter, the outputs of the bias filters, and their model probabilities.

The IMBM algorithm was applied to tracking maneuvering targets, where the target acceleration was treated as a bias with Markovian switching coefficients. This algorithm was called the Interacting Multiple Acceleration Model (IMAM) algorithm. In the IMAM algorithm the bias-free filter corresponds to the constant velocity filter, while the bias filters correspond to a zero acceleration model and two constant acceleration models. The constant acceleration models will have different process noises covariances, while one constant acceleration filter will utilize the kinematic constraint for constant speed targets [ALOUANI 91a, ALOUANI 91b]. Thus, this IMM algorithm will include a constant velocity model and three bias models. The first bias model, $m_1(k)$, corresponds to a zero mean and zero error covariance bias, does not not require a filter. The second bias model, $m_2(k)$, corresponds to a constant state with the kinematic constraint, while the third bias model, $m_3(k)$, corresponds to a constant state. With these model configurations, the estimate of the velocity for time k based on bias model j $(V_j(k))$ can be shown to be equivalent for the three different models. Also, the use of the kinematic constraint does not affect the constant velocity filter. Thus, the pseudo measurement for the kinematic constraint is input directly into the third bias filter. Since the estimate of the velocity for bias-free filter at time k $(V(k))$ is common to all three bias models, significant implications can be made in the IMBM algorithm. This IMAM algorithm, which utilizes a zero bias model, $m_1(k)$, and two constant acceleration models, is presented in [BLAIR 92]. The tracking performance of the IMAM algorithm is compared with the tracking performance of a similar baseline IMM algorithm. Simulation results show that IMAM provides tracking that is very similar to that provided by the baseline IMM algorithm, while requiring about 43% of the computations of the IMM algorithm, when a constant velocity model and two constant acceleration models are used. The computational requirements of the IMAM algorithm may allow multiple model algorithms to be implemented for target tracking in existing systems, where the IMM algorithm cannot be justified from a computational standpoint. The results are preliminary. With further experience with model selection and tuning, the performance of the IMAM algorithm should be able to closely match that of the IMM algorithm.

3.4 IMM Estimators with Mode-Set Adaptation

Often, for practical problems an algorithm using a fixed set of a small number of models can not yield accurate results. Apart from the increase in computation, use of more models does not guarantee better performance—actually, it may yield even poorer results. The major reason for the poor performance of the fixed mode-set algorithms in some cases is that when a large number of modes are used at any time, many of them are so different from the system mode in effect at a particular time, that not only the computation for the filters matched to these modes provides no gain but also the unnecessary "competition" among modes degrades the performance. This degradation arises from the fact that the same (large) mode set is used for all time periods. To find a way out of this dilemma, the paper [LI 92a] introduces the concept of variable structure, i.e., adaptation of the model set. Li and Bar-Shalom [LI 92a] have proposed several variable structure algorithms, in contrast to the existing efforts of developing better implementable versions of the so-called optimal (fixed structure) estimator. One solution is to use a time-varying mode set. That is, M_f in (2.4)-(2.5) is replaced by a time-varying set $M(k)$, to be determined from all the information available—in particular, the sequence of the measurements.

The optimal estimators described above have been considered to yield the performance bound of all IMM estimators. Unfortunatly, these "optimal" estimators may give unsatisfactory results in some cases. Clearly, the existing efforts of developing good real-time implementable versions of these estimators are hampered in these cases. Fortunately, these so-called "optimal" estimators are the best only within the class of fixed structure algorithms. By using a variable structure, it is possible to develop algorithms that can outperform these estimators. The variable structure (or adaptive mode-set) estimator can be viewed as a two-level hierarchical algorithm: multiple model at the lower level and multiple mode-set at the higher level. As was shown in [LI 92a], the fixed structure estimator is only a special case of the variable structure estimator using a trivial partition consisting of the following single sequence of mode sets

$$M^k = \{M_f, M_f, \ldots, M_f\} \tag{3.3}$$

where M^k denotes a sequence of the mode sets through time k. That is, the variable structure estimator reduces to one of the fixed structure estimators if only sequences of fixed mode sets are allowed. Not that (3.3) is not the only possible partition. Suppose $M_s = \{1, 2, 3, 4\}$. The following two sequences of mode sets constitute one of the many possible partitions at time $k = 2$:

$$M_1^2 = \{\{1, 2, 3, 4\}, \{1, 2\}\}$$
$$M_2^2 = \{\{1, 2, 3, 4\}, \{3, 4\}\}$$

Thus, even the choice of the partition can also be made upon the receipt of the measurement sequence. The variable structure estimator is too complicated to be feasible. Nevertheless, it suggests the use of time-varying mode-set sequences since such sequences can lead to better results than a fixed sequence. The number of mode histories in the adaptive mode-set algorithm at k is $\prod_{t=1}^{k} N(t)$ $\quad$ ($N(t)$ is the number of modes at time t), which is much smaller than N_f^k since $N(t) < N_f$. Thus, the adaptive mode-set approach can provide a tremendous saving in computation over a fixed mode-set approach, apart form superior performance. Meanwhile, the same management techniques of mode histories as in the fixed mode-set algorithms can be retained.

Another approach, suggest by [MUNIR 94], is to use the so-called adaptive interacting multiple model (AIMM) algorithm. Each model is assigned a fixed deterministic acceleration to cope with the different target accelerations. A rough estimate of the target acceleration is taken from a biased filter which uses combined information from the models. This estimated acceleration is then taken into account in the models. Different maneuvering target scenarios were generated to compare the performance of the AIMM algorithm with the IMM algorithm. The performance of the AIMM algorithm in tracking is better than the IMM algorithm, which uses submodels with different noise levels, when accuracy and computation time are used in the assessment. A major advantage of the AIMM algorithm is that it does not need pre-defined fixed deterministic accelerations for the submodels. The IMM algorithm may provide better accuracy when one of the submodels is well matched with the motion of the target. However, the AIMM algorithm can provide better coverage of variable target accelerations using fewer submodels.

A method named Selected Filter IMM (SFIMM) has been proposed in [LIN 93]. The SFIMM algorithm is a modified IMM algorithm which uses a subset of filters with the specific subset chosen using decision rules. Particular attention is focused on the design parameters, namely, the subset type and the decision rule in this study, the coverage of thirteen filters was investigated. Two different subset types were examined; the subset type No. 1 was suggested in [LI 92a], where no model overlapping was allowed. It was found to have a higher possibility of shift and so when shifts occur a larger portion of models needed initialization, larger tracking errors induced. The subset No. 2, which used a symmetric structure, with some small overlapping, resulted in a decreased shift probability so that the tracking performance was improved. Two different decision rules were used to decide whether a shift was required or not, first the Maximum Likelihood (ML) criterion which checked the most likely among all the models in the same subset to make a decision. In the IMM algorithm the likelihood function for each model has some contributions from all the models in the same subset, so it is a mixed likelihood. The second criterion was the Maximum a posteriori (MAP) criterion which included the prior probability. Again as the a priori probability for each model has some contributions from all the models in the same subset, the a posteriori probability also has mixed terms. Since more information is used in this decision rule the tracking performance was improved. The simulation results showed that the performances for all the four designs of SFIMM algorithm were acceptable, although design No. 4 which used subset type No. 2 and the MAP decision rule had the best performance. The tracking accuracy was only slightly worse than that of an IMM algorithm with thirteen subfilters and took approximately only one quarter of the computation time.

The variable structure IMM estimator is formulated in terms of graph-theoretic concepts (GIMM estimator).

Definitions:

1. A digraph (or directed graph) D is an ordered pair of disjoint sets (VER(D), E(D)) such that E is a subset of the set of ordered pairs of elements of VER. A set of modes with possible mode switching forms a digraph. Edges in this digraph can be weighted using the transition probability. Associated with such a digraph, there is a one-to-one correspondence adjacency matrix with (i, j)-th element defined as

$$A_{ij} = \begin{cases} P_{ij} & \text{if there exists } e(ver_i, ver_j) \\ 0 & \text{otherwise} \end{cases} \tag{3.4}$$

   which leads to a stochastic digraph if

$$\sum_j P_{ij} = 1 \qquad \forall i \tag{3.5}$$

   where $e(ver_i, ver_j)$ is the edge from vertex ver_i to vertex ver_j.

2. A digraph D is said to be strongly connected if every two vertices of D are mutually reachable (via some directed paths). A Markov chain is ergodic if and only if its associated digraph is a strongly connected stochastic digraph. Three essential ingredients of a IMM algorithm can be identified: a) a set of single-model algorithms (e.g., Kalman filters), each matched to a particular mode; b) a mechanism of obtaining the underlying digraph at each time that defines the graphical relationship between modes; c) and

a fusion rule that provides the overall results based on the mode-matched algorithms and the digraph. This underlying digraph of an IMM algorithm is called supporting digraph. Note that there may be many different supporting digraphs associated with a mode set and thus it is much better to use digraphs rather than mode sets to describe the IMM algorithms.

3. A fixed digraph (or fixed structure) IMM algorithm is an IMM algorithm with identical supporting digraphs at all times; otherwise it is said to be a time-varying digraph (or variable structure) algorithm. An adaptive digraph IMM algorithm is an algorithm in which the time-varying digraph is obtained via adaptation. A fixed mode-set IMM algorithm is one that all its supporting digraphs at different times must have the same set of modes. An IMM algorithm is said to be switchable (or dynamic) if its supporting digraphs need not consist of only isolated modes (i.e., modes that are not linked to any other modes). Otherwise, it is said to be nonswitchable (or static). If all the supporting digraphs are strongly connected, then the algorithm is said to be strongly switchable. Li and Bar-Shalom [LI 92a] consider only the strongly switchable algorithms, which include almost all effective practical algorithms. In fact, the supporting digraphs of a practical IMM algorithm are usually symmetric in the sense that $e(ver_i, ver_j)$ exists if and only if $e(ver_j, ver_i)$ exists [LI 92a]. The algorithms that obtain the supporting digraph adaptively belong to the class of Recursive Adaptive Digraph (RAD) algorithms.

<u>Summary of the RAD Approach (one cycle):</u>

Adaptation of digraph: Determine the digraph $D(k)$ for time k, based on $D(k-1), X(k-1), P(k-1), \mu(k-1)$, and measurement $z(k)$. Here

$$X(k-1) = \{\hat{x}_m(k-1|k-1) : m \in D(k-1)\} \tag{3.6}$$

$$P(k-1) = \{P_m(k-1|k-1) : m \in D(k-1)\} \tag{3.7}$$

$$\mu(k-1) = \left\{P\{m|Z^k\} : m \in D(k-1)\right\} \tag{3.8}$$

where $\hat{x}_m(k-1|k-1)$ denotes the state estimate obtained by the filter matched to mode m; $P_m(k-1|k-1)$ is the associated covariance.

Mode-matched estimation: For each mode in $D(k)$, make the one-step estimation under the assumption that this mode is in effect at time k.

Mode probability calculation: For each mode in $D(k)$, calculate the probability that this mode is in effect at time k.

Estimation fusion: Provide the overall estimate and its covariance.

Only the first step will be discussed here since the other steps are identical to those in the fixed structure approach (see the steps Interaction, Filtering and Combination of the base IMM algorithm).

Definition: If $VER' \subset VER(D)$ then $D' = (VER', E')$ is said to be the *subdigraph* of D *induced* by VER', denoted by $D[VER']$, if E' consists of all edges of D with both end vertices in VER'. A *minimal set* is such a set that any of its proper subset does not preserve the property of interest. A minimal set with smallest cardinality is called a *minimum* set. Maximal and maximum sets are defined similarly. A cut-set of a strongly connected digraph D is a minimal set of edges of D such that their removal makes D not strongly connected. *Normalization* of a weighted digraph is the procedure of properly scaling all weights in the digraph such that (3.5) is satisfied. In the sequel, let D always be the *total digraph*, obtained by normalizing the union of the supporting digraphs at all times of the IMM algorithm under consideration. The *strongly adjacent* set of vertex ver_i in D is defined as

$$B(ver_i) = \{ver_j \in VER(D) : A_{ij} \neq 0 \text{ and } A_{ji} \neq 0\} \tag{3.9}$$

3.4.1 Active digraph (AD) algorithm

In this algorithm, at time k some unlikely modes in the current (active) digraph $D(k)$ are discarded whereas some other modes in D, depending on their graphical relationship to the modes in $D(k)$, are possibly activated and added to $D(k)$. All modes can be divided into three classes: unlikely, significant and dominant modes. Suppose that the active digraph at time $k-1$ is $D(k-1)$. Let B_i be the strongly adjacent set of a dominant mode m_i and SET_{max} be a maximum set, of unlikely modes such that the set of edges incident with SET_{max} contains no cut-set of $D(k-1)$.

Summary of the AD algorithm (one cycle):

1. Identify each mode in $D(k-1)$ to be unlikely, significant, or dominant.
2. Choose $Y = (VER(D(k-1) - SET_{max}) \cup B$, where $VER(D(k-1)$ is the mode set of $D(k-1)$, SET_{max} was defined above, and B is the union of the strongly adjacent sets of the dominant modes of $D(k-1)$.
3. Obtain $D(k)$ by normalizing $D(Y)$, the subdigraph of D induced by Y.
4. Execute Steps 2 - 4 of the RAD approach using digraph $D(k)$.

Remarks: (1) The algorithm reduces to have a fixed structure if none of the modes are deemed either unlikely or dominant. (2) B may be empty.

3.4.2 Digraph switching (DS) algorithm

Another way of adapting the supporting digraph is to switch between a number of pre-determined digraphs according to a certain logic. Each of these digraphs is a graphical representation of a specific group of closely related operational modes. A family of digraphs $D^L \triangleq \{D_1, D_2, \ldots, D_L\}$ is called a *strong cover* of D if the following is satisfied:

1. Every D_i in D^L is a strongly connected subdigraph of D
2. The mode set of D is covered by the mode sets of D_i, $i = 1, 2, \ldots, L$, i.e., $VER(D) \subset \cup VER(D_i)$. ($VER(D^L)$ need not be a partition of $VER(D)$)
3. The algorithm reduces to the fixed structure algorithms if every D_i consists of exactly one mode of the total digraph D.

In the DS algorithm, a strong cover of the total digraph D is set up first. In the context of IMM algorithms this strong cover can, in most cases, be obtained quite easily from the physical meaning of the modes. Each member of this family is then normalized to become a stochastic digraph. The resulting family of digraphs then act as the range of the time-varying supporting digraph of the IMM algorithm. That is,

$$D(k) \in D^L = \{D_1, D_2, \ldots, D_L\} \tag{3.10}$$

Each digraph in D^L contains a number of modes:

$$VER(D_i) = \{m_{i1}, m_{i2}, \ldots, m_{iN_i}\} \quad i = 1, 2, \ldots, L \tag{3.11}$$

which are those members of D that are "close" to each other. The meaning of the closeness is best explained via the examples in [AVERBUCH 91a, LI 92a].

In the DS approach, two kinds of switching logics can be used. A typical logic for a *hard switching* is as follows. If the mode probability vector is in a specified switching region, defined by a certain combination of thresholds, then certain switching occurs. Assume the digraph sequence to be a Markov chain. Its transition matrix thus determines the logic for *soft switching* and then the fixed structure IMM algorithms can be applied to it. As such, the fixed structure IMM approach is applied in two levels: mode (lower) level and digraph (higher) level.

3.4.3 Adaptive Grid (AG) algorithm

A third way of obtaining the supporting digraphs uses an adaptive grid of the parameters that characterize the possible modes. In this algorithm, an initial coarse grid is set up first and then an adaptive grid is obtained recursively according to an adaptation scheme, which can be obtained based on the current estimates and/or mode probabilities [LI 92a]. It suggests the use of variable structures as a direction of improving the performance with modest computational requirements, as contrasted to the existing efforts of developing better implementable versions of the optimal fixed structure algorithms.

3.5 IMM Algorithm with Correlated Measurement Noise

In tracking airborne or missile targets using noisy radar data, the measurement noise is significantly correlated when the measurement frequency is high. In the paper [GUU91], a simple decorrelation process is proposed to enhance the IMM algorithm to track a maneuvering target with correlated measurement noise. It is assumed the simplest form of hybrid system (2.9) - (2.11) and that the noise can be modeled as a first-order Markov process given by

$$w_j(k+1) = \lambda w_j(k-1) + w_j(k) \tag{3.12}$$

where λ is the correlation coefficient ($0 \leq \lambda \leq 1$) between successive measurements. To decorrelate the measurement noise a new measurement $z_a(k)$, denoted as "artificial measurement" in [ROGERS 87], is generated. Let $\bar{\lambda}$ be the estimated value of noise-correlation, then the measurement equation can be rederived as follows.

$$z_a(k) = z(k) - \bar{\lambda} z(k-1) \tag{3.13}$$

$$\begin{aligned}
&= (H_j x(k) + w_j(k)) - \bar{\lambda}(H_j x(k-1) + w_j(k-1)) \\
&= \left[H_j x(k) - \bar{\lambda} H_j (F_j)^{-1}(x(k) - \Gamma_j v_j(k-1)\right] + \left[w_j(k) - \bar{\lambda} w_j(k-1)\right] \\
&= \left[H_j - \bar{\lambda} H_j (F_j)^{-1}\right] x(k) + \bar{\lambda} H_j (F_j)^{-1} \Gamma_j v_j(k-1) \\
&\quad + \left[(\lambda - \bar{\lambda}) w_j(k-1) + w_j(k)\right]
\end{aligned} \tag{3.14}$$

Let

$$\hat{H}_j = H_j - \bar{\lambda} H_j (F_j)^{-1} \tag{3.15}$$

$$\hat{\Gamma}_j = \bar{\lambda} H_j (F_j)^{-1} \Gamma_j \tag{3.16}$$

$$\hat{w}_j(k) = \hat{\Gamma}_j v_j(k-1) + (\lambda - \bar{\lambda}) w_j(k-1) + w_j(k) \tag{3.17}$$

Then

$$z_a(k) = \hat{H}_j x(k) + \hat{w}_j(k) \tag{3.18}$$

If $\bar{\lambda} \approx \lambda$ the new measurement noise $\hat{w}_j(k)$ would be white, but it is correlated with the process noise $v_j(k-1)$. By reformulating the dynamic equation (2.9) properly, the process noise can be made to be uncorrelated with new measurement noise. In most practical systems, this procedure can be omitted with little degradation in performance since the term $\hat{\Gamma}_j v_j(k-1)$ is usually small. Thus, the baseline IMM algorithm can be applied to the case with correlated measurement noise by the following substitutions:

$$H_j \to \hat{H}_j \qquad w_j(k) \to \hat{w}_j(k) \qquad z(k) \to z_a(k) \text{ for } j = 1, 2, \ldots, N_f$$

The results of computer simulations [GUU91] indicated that the decorrelation process may improve system performance significantly, especially in velocity and acceleration estimations. These large improvements in velocity and acceleration are particularly useful in situation when the target is suddenly accelerated by the pilot or missile guidance program.

3.6 Multisensor IMM Tracking Algorithms

In some military and civilian applications, the data are collected by a network of sensors distributed over a large geographic region. In such distributed sensor networks (DSN), because of considerations such as reliability, survability, and communication bandwidth, centralized processing is either undesirable or infeasible. Instead, the sensors supply data to a set of local processors/nodes which are connected by a communication network. The nodes process the local sensor data and exchange processing results with other nodes. The key problem in multitarget tracking is data association, i.e. the association of the measurements from single or multiple sensors to the same targets. Many different approaches have been proposed for data association in a centralized framework [CHANG 86], in which all the sensor measurements are transmitted to a centralized site where they are processed. One such approach is JPDAF and PDAF. In these algorithms the data association problem is handled by computing the posterior probabilities of the measurement associations with targets in clutter. These probabilities are used to update a set of Gaussian distributions representing (suboptimal) Bayesian estimates of the target locations. Several distributed tracking algorithms for DSN have been discussed in [CHANG 86]. The paper [CHONG 85] presents a distributed tracking scheme based on the JPDAF where both data association and distributed estimation are considered. In this scheme, each node first performs the tracking functions with the JPDAF using the local measurements and sends the processed results to other nodes. The receiving node then fuses the information from other nodes with its local information to arrive at a better estimate (the sequential fusion). The paper [CHANG 86] presents a JPDAF in DSN structure, where each local node processes the local sensor measurements and sends the local estimates periodically to the global processor (fusing node, the parallel fusion). After processing the results, the global processor sends back the global estimates to each local node.

3.6.1 Sequential Multisensor IMMPDAF Algorithm

The sequential multisensor (MS) IMMPDAF algorithm is presented in [HOULES 89]. This algorithm assumes the simplest form of hybrid system (2.9) - (2.11); each mode-matched filter is a standard PDAF. The sensors (two sensors) are in the same location, assumed, for simplicity, synchronized and with the same sampling rate. Track initialization is assumed to have been made. At each scan, a validation gate, centered around the predicted measurement of the target, is set up for each sensor for the next scan.
Summary of the sequential MS IMMPDAF algorithm (one cycle):

1. Mixing of the estimates from the previous time (see section 3.1)

2. Predicted states and predicted measurements

3. Measurement validation from sensor 1: The validation region has to be taken the same for each of the models as the largest of them (see [HOULES 89]).

4. Estimation with sensor 1 measurements in each filter: At the end of the estimation, one computes a new "predicted" measurement for sensor 2, similarly and in step 2 (see [HOULES 89]).

5. Measurement validation from sensor 2: this step is similar to step 3.

6. Estimation with sensor 2 measurements in each filter: This step is similar to step 4.

7. Updating of the model probabilities: (see Filtering in section 3.1).

8. Combination of the model-conditioned estimates: (see Combination in section 3.1).

3.6.2 Parallel Multisensor IMMPDAF Algorithm

The parallel MS IMMPDAF algorithm is presented in [CHANG 87, CHANG 89]. This algorithm assumes the simplest form of hybrid system (2.9) - (2.11) too; each mode-matched filter is a PDAF. The two-node scenario similar to that given in [CHANG 86] is considered, where each node processes the local measurements from its own sensor and sends the local estimates to the fusion processor periodically. The fusion processor then sends back the processed results after each communication time. Lossless communication is assumed and the information communicated is the sufficient statistics. For each local node, the centralized IMMPDAF algorithm, where all measurements are sent to and processed with one processor, is described similar to base IMM algorithm with a standard PDAF. The goal is to compute the conditional state distribution given the local accumulated measurements. With the local conditional pdf of node i, we can derive the fusion algorithm for the base IMM filter. The information needed to be communicated from local nodes to the fusion node consists of: (a) the model probabilities; (b) the association event probabilities; and (c) the corresponding pdfs (mean and covariance for Gaussian case). Three different scenarios were tested. First, each sensor tracks the target independently using IMM algorithm from section 3.1. Second, measurements from sensors were first compressed [WILLNER 76] into one, then the standard IMM algorithm, as in the first case, was applied. Finally the *distributed* (parallel MS IMMPDAF) case was simulated. Simulation results show that the performance of the distributed case depends explicitly on the amount of information being communicated between nodes. The results of centralized processing scheme represent the upper bound of the performance. When two nodes communicate every scan, the results of the distributed case are the same as the centralized case (but has the advantages of increased reliability), which confirms their theoretical equivalence. When two nodes communicate every two scans, the results of the distributed case are better than that of the decentralized case, which shows the advantage of interchanging information.

3.7 Square-Root Algorithms for IMM Filters

Theoretically, the KF gives the unbiased, minimum variance estimate of the state vector of a linear dynamic system, distributed by additive white noise when measurements of the state vector are linear, but corrupted white noise. Such performance is hardly ever realized in actual practice since the information required to construct the KF is only approximately known. The noise parameters and models may be based upon only relatively few data points, computer round-off errors may be significant, and the system model may not be adequate. The conventional KF algorithm is not numerically robust due to round-off errors and ill-conditioning problems. Numerical studies in the past [BIERMAN 77a, LEONDES 73] have shown that the Kalman algorithm is so sensitive to computer round-off errors that numerical accuracy can degrade to the point where results cease to be meaningful. Typical problems include the loss of positive-definiteness of the covariance matrix resulting from numerical errors such as finite computer word length and cancellation errors due to the subtraction term in the Kalman covariance update. This type of error may be manifested as a covariance matrix computed erroneously as having negative eigenvalues. In a significant class of filtering problems, propagation of the error covariance matrix, by means of the KF equations, results in a matrix which is not positive semidefinite - a theoretical impossibility. This may occur when (1) the covariance matrix is rapidly reduced by processing very accurate measurements, (2) a linear combination of state vector components is known with great precision, while other combinations are essentially unobservable. The source of trouble in both cases is the numerical computation of ill-conditioned quantities in finite word length. These errors can cause filter divergence or computations that are so erroneous as to be meaningless. In order to overcome these difficulties, square-root formulations of the Kalman filtering have been proposed since they improve the numerical conditioning (robustness) of the problem and achieve twice the effective precision given by conventional Kalman algorithms [BIERMAN 77b, KENEFIC 90, ZEITZ 93]. One class of algorithms that yields satisfactory results uses a square root factorization of the covariance matrix. Square root factorizations generally have less dynamic range associated with their calculations, and are less susceptible to round-off error. Another advantage of the square root representation is that the covaraince matrix does not need to be calculated directly. Instead, it is represented in terms of its square root matrices. This factored form cannot represent a matrix with negative eigenvalues. There are some disadvantages to factored forms of the KF. One significant disadvantage is that they normally require more computations, and therefore more computer time, than the conventional KF. The Potter covariance square root, the Carlson covariance square root, and Bierman's-Thornton's U-D filter are three of the more common factorization methods for discrete-time Kalman filtering. Of these factored forms, the U-D filter offers a high degree of numerical precision along with a reasonable increase in the required number of mathematical operations. Because of these properties, the U-D filter is a standard for comparison in the remainder of this work. The U-D filter is a factorization of the covariance matrix in the form UDU^T, where U is an upper triangular unit matrix (it has ones along the main diagonal) and D is a diagonal matrix. The efficient L-D (or U-D) factorization algorithms for IMM and IMMPDAF are presented in [RAGHAVAN 92, RAGHAVAN 93]. The system dynamics are modeled by the equations (2.9) - (2.10); it is assumed that the model jump process is a Markov process with known transition probabilities p_{ij} (see (2.11)). The new version of the square-root PDAF is approximately two times faster than Kenefic's algorithm [KENEFIC 90].

3.7.1 Square-Root Filter

Summary of the square-root covariance updating for the IMM filter (one cycle):

Interaction mixing - Starting with L-D factors of the model conditioned covariances of time $k-1$

$$P_j(k-1|k-1) = L_j(k-1|k-1)D_j(k-1|k-1)L_j(k-1|k-1)^T \quad j = 1, \ldots, N_f$$

we compute the L-D factors of the mixed initial covariances for each of the filters. Note that the factor $\mu_{i|j}(k-1|k-1)$ can be absorbed into $D_j(k-1|k-1)$. To compute the factorization of $P_{0j}(k-1|k-1)$ it is applied a rank-one correction to factors of each $P_j(k-1|k-1)$ and the resulting sum of N_f L-D factored terms are combined to form a single L-D factored term [RAGHAVAN 92, RAGHAVAN 93].

Filtering - The state estimates and the factored covariances computed in the interaction mixing step form inputs to filter matched to each of the N_f models. The output of this stage are the model conditioned estimates (see section 3.1 (filtering)), L-D factored covariances for time k [RAGHAVAN 92, RAGHAVAN 93], likelihood functions

$$\Lambda_j(k) = \frac{1}{(2\pi)^{\frac{m'}{2}} \left[\prod_{i=1}^{m'} a_{ij}\right]^{\frac{1}{2}}} e^{-\frac{1}{2}\sum_{i=1}^{m'} \frac{(\rho_{ij})^2}{a_{ij}}}$$

where m' is the number of measurements originated from the object in track at time k, $\{\rho_{ij}\}$ and $\{a_{ij}\}$ are the sequential innovation and sequential innovation covariance of the i-th scalar component of measurement vector for the filter matched to the j-th model (these are computed during the covariance, update recursion of the filter matched to the j-th model). The likelihoods are used to update the model probabilities (see section 3.1 (Filtering)).

Overall state and covariance estimates - The overall state estimate is computed as usual. The covariance mixture equation is very similar in structure to the Interaction mixture equation of Interaction mixing step. Therefore, the strategy followed here is the same. Let

$$P_j(k|k) = L_j(k|k)D_j(k|k)L_j(k|k)^T \quad i = 1, 2, \ldots, N_f$$

Note that the factor $\mu_j(k)$ can be absorbed into $D_j(k|k)$. To compute the factorization of $P(k|k)$, it is needed to apply a rank-one correction to factors of each of $P_j(k|k)$ and the resulting sum of N_f L-D factored terms must be combined to form a single L-D factored term. The rank-one correction can be done using the generalized L-D rank-one correction algorithm given in [RAGHAVAN 93]. The merging of N_f factored terms can be accomplished by recursively applying the algorithm which is also given in [RAGHAVAN 93].

3.7.2 Square Root IMMPDAF

The algorithm described above for the square-root IMM remains essentially the same except for the following modifications.

Model conditioned filtering- Since the standard KF are replaced by PDAFs, the covariance prediction and update recursions for the square-root PDAF, given in [RAGHAVAN 92, RAGHAVAN 93] are used for the N_f filters. The output of the stage are the model conditioned estimates and L-D factored covariances for time k.

Computation of likelihoods - The likelihoods can be expressed as

$$\Lambda_j(k) = d(k)\left[b'(k)(2\pi)^{\frac{m'}{2}}\left(\prod_{i=1}^{m'} a_{ij}\right)^{\frac{1}{2}} + \sum_{k=1}^{m'_k} e^{-\frac{1}{2}\sum_{i=1}^{m'}\frac{(\rho_{ijk})^2}{a_{ij}}}\right]$$

where

$$b'(k) = m'_k V_k^{-1}\frac{1-P_D P_G}{P_D P_G}$$

$$d(k) = V_k^{-m'_k+1}\frac{P_D P_G}{m'_k}$$

where V_k is the volume of the common validation region at time k, m'_k is the number of validated measurements at time k, P_D is the probability of detection, P_G is the probability that the target measurement falls in the validation region, $\{\rho_{ijk}\}$ and $\{a_{ij}\}$ are the sequential innovation and its variance corresponding to the $i-th$ scalar component of $k-th$ validated measurement vector for the filter matched to $j-$th model (these are computed during the covariance update recursionof PDAF matched to $j-th$ model). The rest of the algorithm is the same as that of the square-root IMM filter.

4 Comparison of the main MM Algorithms

Two maneuvering target tracking techniques are compared in the note [BARSHALOM 89a]. The first, called input estimation, models the maneuver as constant unknown input, estimates its magnitude and onset time and then corrects accordingly the state estimate. The second models the maneuver as a switching of the target state model, where the various state models can be of different dimension and driven by process noises of different intensities, and estimates the state according to the IMM algorithm. While the first requires around 20 parallel filters, it is shown how the later, implemented in the form of the IMM, performs equally or better with 2 or 3 filters.

The paper [AVERBUCH 91a] investigates and compares the performance of the IMM and Viterbi algorithms, applied to the problem of radar tracking of maneuvering target. It assumes a kinematic model in which the acceleration during a maneuver is almost constant and its mean value is one of a finite set of accelerations. The perturbation upon this constant acceleration is modeled as white zero-mean Gaussian noise. It considers two different tracks and twelve models, each being assigned a specific acceleration value ranging from 0 to 4g (= 40m/s^2). In the first track the acceleration level of the target almost matches the acceleration level of one of the system models. In the second track, there is a significant difference between the acceleration of the target and the acceleration level of the closest model. It is shown that the performance depends strongly on the parameter $\eta = \frac{|\Delta\bar{a}|(\Delta\tau^2)}{\sigma_r}$, in which $|\Delta\bar{a}|$ the maximal difference between the acceleration levels of any two models, $\Delta\tau$ is the time interval between measurements and σ_r is the standard deviation of the measurement error. In general, when this parameter is relatively large, both algorithms perform well. When η is relatively small, the Viterbi algorithm is better. However, during the beginning of a track and after the start of a maneuver the IMM algorithm provides better estimations.

The combination of the IMM approach with PDAF and JPDAF methods are discussed extensively in [BARSHALOM 89b, BARSHALOM 90b, BARSHALOM 92b, BLOM 92, DUFOUR 91, DUFOUR 92, HOULES 89, LERRO 93]. These recursive algorithms have fixed computational and memory requirements and can initiate tracks, maintain tracks in the presence of maneuvers and terminate tracks if warranted. The algorithms are useful for situations of low signal-to-noise ratios, where the detection threshold has to be set low to detect the targets, this leading to a high rate of false alarms for which logic-based techniques are not adequate. These algorithms, which yield model probabilities, provide the true target probability for each track under consideration. Thus, these algorithms can assess their own performance; that is, they can be called intelligent.

5 Parallel Implementation of IMM Algorithms

The IMM algorithm basically consists of a group of filters which run in parallel, and a global computation process which collects the results of the filters and produces the output estimation. However, in real applications, processing time may become impractical. Multiprocessors seem to offer the throughput needed for such applications. In the paper [AVERBUCH 91b] the authors investigate the performance of IMM estimator, when implemented on a general purpose shared-memory and a shared-bus MIMD (Multiple Instruction Multiple Data) multiprocessor. The implementation of this algorithm on the parallel processor is based on the concept of one processor acting as a manager which performs the global computations, and all other processors acting as workers. Both the manager and the workers are implemented as local tasks. The assignments of the manager during each time slice are:

- getting new measurements
- generating the output based on results computed by the other processors
- synchronizing between processors
- updating some variables which are part of the EKF and are relevant to all processors

Each time slice consists of a large number of computations, most of which are an EKF step performed independently for each possible current model. The input of each such filter is an estimation of the system state vector along with its covariance for the previous time step and a new measurement. The output is an updated estimation. This computational load can be divided among the processors almost without additional overhead by assigning each processor a subgroup of the models.

The computational complexity (both the time and space (memory) complexity) as well as the speed-up and efficiency are examined in detail. The time complexity is evaluated using four categories:

- elementary arithmetic operations $(+, -, \cdot, /)$
- complex arithmetic operations (exp ( ), arctan ( ))
- accesses to local memory

- access to shared memory

The space complexity is evaluated using two categories: shared memory and local memory. The basic unit in calculations is the space needed to store a floating variable which is 4 bytes.

It is shown that the IMM algorithm was proven to be very suitable to parallel processing. Efficiency with P processors, when P is relatively small, is about $(1 - \frac{1}{P})$. It is expected to obtain relatively high efficiency, even for large P, especially when considering many filters and large state and measurement vectors.

In the study reported by [ATHERTON 94] the objective was to implement the IMM algorithm in a parallel structure (on a network of four transputers). The experimental work has been done on transputer development system (TDS) using the occam programming language. Three different architectures for parallel implementation of the IMM algorithm were examined, each with different task partitions for the manager and the workers. A speedup of approximately 2.7 was achieved for the 13 filter IMM implementation. This was further improved to 3.2 by using a time-slipping calculation method which was shown experimentally to have a negligible effect on the average tracking accuracy.

6 Nonsimulation Performance Prediction Techniques for IMM

Performance prediction of IMM algorithms only via costly and time-consuming Monte-Carlo simulations does not provide good insight into the problems. Unfortunately, none of the existing nonsimulation techniques [FORTMANN85, LI 91, MORI 92, ROGERS 91] can be applied to hybrid state estimation schemes based on MM, such as the IMM algorithm. Error bounding techniques (see e.g. [DAUM 90, EZZINE 88, KERR 89, WASHBURN 85]) can provide at most only bounds. In the paper [LI 92b] an effective hybrid approach to the nonsimulation performance prediction of IMM systems is developed. In view of the fact that the performance of a MM scheme is scenario dependent, this approach is based on a stochastic performance measure (i.e., conditional expectation of the error covariance) of hybrid nature in the sense that it is continuous-valued matrix function of a discrete-valued sequence - the system mode sequence, which constitutes the scenario of the problem of interest. This hybrid approach is obtained by using a conditional expectation operation, through which the randomness of the error covariance due to the uncertainties in the continuous subspace of the hybrid space is averaged out whereas the information of the system mode jumping is retained. To calculate efficiently the performance measure, an off-line recursion has been obtained based on conditional successive one-step-ahead-predictions. The capability of the approach in predicting quantitatively the average performance of the algorithm was illustrated via two important examples: a generic maneuvering target tracking problem in Air Traffic Control and a non-stationary noise identification problem. The predicted performance given by this approach agrees remarkably well with the simulated one, which verifies the good accuracy of the approach and therefore it can be used as an efficient tool for design, sensitivity analysis, and so on.

The philosophy of this conditional expectation based approach is quite general and can be applied to many other algorithms for hybrid systems. For example, the performance of the STDM-based IMM estimator [CAMPO 88, CAMPO 91] can be evaluated using this approach with suitable modifications. The approach can be readily extended to the variable structure IMM estimator (GIMM estimator) [LI 92a], square-root IMM algorithm [RAGHAVAN 92, RAGHAVAN 93], correlated measurement noise IMM estimator [GUU91], IMBM and IMAM algorithms [BLAIR 92].

7 Using an IMM Estimator for Phased Array Radar

The output of the IMM algorithm was used to compute the time of the next measurement, or radar dwell, to maintain a given level of tracking performance. By including maneuvering and non-maneuvering models in the IMM algorithm, the sample times automatically decreased during maneuvers to maintain the desired level of performance. Thus, using the output of the IMM algorithm gives measurement or dwell times that automatically reflect target maneuvers, target range, missed detections, and weak and strong signal returns. Also, the probability of the non-maneuvering mode within the IMM algorithm was used to adjust the energy of the wave form. By increasing the energy during maneuvers, the accuracy of the state estimates was improved significantly. The use of the IMM algorithm for computing measurement times for a phased array radar is illustrated by Blair's works [BLAIR 94a, BLAIR 94c], where the average sample was about four seconds for an aircraft in a holding pattern and about two seconds for a manned aircraft performing 7-g maneuvers. A revisit technique that responds to target maneuvers and avoids the decision-directed approach to both tracking and sample rate adjustment can be accomplished using the IMM algorithm. Since the output error covariance of the IMM algorithm better reflects the uncertainty in target motion during a maneuver than that of single model filters, the predicted error covariance of the IMM algorithm can be used to compute the time for the next measurement, so that a given level of the performance is maintained during a maneuver.

The design and performance of an IMM estimator for a given benchmark problem [BLAIR 94b] of highly maneuvering targets are presented in [DAEIPOUR 94]. The design parameters of the IMM algorithm as well as an adaptive sampling policy are described. It is shown that the IMM filter with adaptive sampling policy has a better performance in terms of the average track loss, averaged wells per run and noise reduction in non-maneuvering periods of time than the best KF. The design is done based on the first four benchmark trajectories. The evaluations re done for these four plus two additional trajectories. In the work the authors considered three modes: the first mode is a nearly constant velocity motion (second order per coordinate with a low level white noise acceleration). In the second mode, the target dynamic model has a relatively higher level of noise. This mode corresponds to an on-going maneuver. Finally, the third mode of operation, which is a Wiener process acceleration model, corresponds to the situations that the target starts/terminates a maneuver - large acceleration increments. The tracking algorithm is an IMM estimator with three model corresponding to the above three modes of the dynamic system. The evolution of the hybrid system among the modes is modeled as a first order Markov chain in each mode. The covariance of the noise, contributing to the evolution of the state in each mode, id determined based on the maximum acceleration corresponding to that mode. The tracking algorithm also selects the revisit (dwell) time for the target. The interval to the next dwell is selected from a predetermined set of values as the largest interval, such that the prediction of the measurement errors are smaller than a certain selected threshold related to the beam width. Beam pointing control is done based on the prediction of the target coordinates at the selected revisit time and pointing the beam to this direction. The range gate of the radar is centered at the predicted range of the target at the revisit time. The pointing accuracy (how much the target is off the beam center) is accounted for in the next measurement since it affects the SNR (signal-to-noise ratio) and the measurement accuracies depend on the SNR.

The probability transition matrix of the first order Markov chain is designed based on the expected sojourn time in each mode of the hybrid system [BARSHALOM 93]. In [DAEIPOUR 94] a predetermined set of sampling intervals is used, with the selection done based on the prediction of the radar angle innovation standard deviations compared to the radar beam width. At each step time, the estimator predicts the innovation covariance for the largest possible sampling interval. The uncertainty, caused by the fluctuations of the target cross-section, is also considered in the calculations of the variances. Then, the standard deviations of both angle innovations are compared with a fraction of the antenna beamwidth. If any of the azimuthal or elevation angle innovation standard deviations exceeds this threshold, the candidate sampling interval is rejected and the test is repeated for the next largest sampling interval. Sampling for initialization is done with the maximum allowable rate (10Hz). This rate lasts for a total of four steps to assure a fast decrease of the initial errors. After the sampling interval is determined, the position of the target is predicted and used to point the beam.

The simulation results of the work reveal that the IMM estimator in conjunction with the adaptive sampling policy yields satisfactory results in terms of track loss percentage and average dwells per run. This comes from the maneuver adaptability of the IMM algorithm. A single model KF with the same adaptive sampling policy does not yield satisfactory results. Even in comparison with the KF with 1s constant sampling, the IMM with the average sampling interval of about 2s has less track loss. Another advantage of the IMM algorithm is the noise reduction in the state estimate, especially, in the velocity components of the state.

8 IMM Algorithms for Target Tracking with Glint Noise and Noise Identification

The optimality of the KF is based on the assumption of having zero-mean white Gaussian noises. If this assumption is violated, the KF is no longer the optimal filter. The non-Gaussian noise arising in a radar system is known as a glint noise. The glint noise distribution is long tailed and seriously affects the tracking performance. Filtering in non-Gaussian environments has been studied by many researchers. Robust preprocessing methods for KF have been proposed in [HEWER 87, MASRELIEZ 77]. In the method of [MASRELIEZ 75] a nonlinear score function was introduced which yields a corrective term in the state estimate equation, while retaining the computationally appealing recursive structure of the KF. In this method it is assumed that either the process noise or the measurement noise is non-Gaussian bot not both. However, since it requires a numerical convolution operation to evaluate the score function , this limits the practicality of this method.

New algorithms were developed in [WU 94]. These algorithms incorporate the nonlinear Masreliez filter [MASRELIEZ 77] into the IMM method. They substitute the KF in the IMM algorithm with the nonlinear Masreliez filter and modify the estimate of model probabilities. Simulations show that these new algorithms not only can filter the glint noise efficiently, but also respond quickly to maneuvering.Although Wu has used the score function method with combination of IMM approach, he has avoided the convolution operation and simplified the evaluation of the score function by applying a normal expansion for the distribution of the measurement prediction. However, the approximate spherical model [MOOSE 75] used to decouple the state components is not a good approximation [DAEIPOUR 95].

The application of the IMM approach to the target tracking problem, when the measurements are perturbed by glint noise, is considered also in [DAEIPOUR 95]. A good approximation for a glint noise with heavy tails is a mixture of a Gaussian noise with a moderate variance and a Laplacian distributed noise, with high variance and low occurrence probability. Therefore, for this problem the authors considered an IMM algorithm with two filters. One filter is matched to the dynamic system with Gaussian measurement noise and the other is matched to the same dynamic system but with a high variance Laplacian distributed noise. With target motion described n Cartesian coordinates and measurements in the radar coordinates, the former filter is an EKF. Even though one can obtain a nonlinear filter that accounts explicitly for the Laplacian distributed measurement noise,it s expensive due to the need to perform numerical integration in real time. The results of this work show that a KF with moment-matched measurement noise variance, perform close to the optimum. Therefore, the filter matched to the second model is also an EKF. The authors showed that this method performs better than the score function method and it is also robust to the variations of the glint noise parameters.

In [LI 94] an approach to noise identification is proposed which uses hybrid system models with a suitable estimation scheme. This approach is valid for non-stationary noise with rapidly or slowly varying statistics as well as stationary noise. The IMM algorithm was incorporated in the approach. The proposed approach is powerful in the sense that it can be used to identify simultaneously various noise parameters in linear and nonlinear systems. It is also flexible in terms of model design and version selection. In addition, the approach fits naturally into the state estimation since it is in the framework of hybrid system estimation.

9 Conclusion

While no tracking algorithms can work in an arbitrarily dense multitarget-multisensor environment, the recently developed IMM techniques described in this paper extend the range in which one can achieve reliable tracking performance. Compromises between computational requirements and performance will be necessary in many cases, especially where the data rate is high and the available computing power is limited, as in airborne systems. The rate is high and the available computing power is limited, as in airborne systems. The assumption made in the various algorithms discussed here will probably be modified according to the needs of the specific applications.

A number of problems that are yet open can be seen at this time. One such problem is the treatment of multisensor group tracking of maneuvering targets in clutter [BLACKMAN 86, BLAIR 92]. At this point it is not known whether filtering techniques can be combined with any of the methods cited above. The problem of applying new techniques to phased-array radar, with special consideration to the necessary scanning patterns, should also be continued [BROOKNER 91, BLAIR 92, VonKEUK 93, DAEIPOUR 94, BLAIR 94b].. The performance of the IMM estimator can be further improved by fine-tuning of the design parameters.

The IMM procedure was established on a solid theoretical basis and proved to be quite appropriate for the maneuvering target tracking problem [VACHER 92]. The investigations documented in the literature indicate that the IMM algorithm is the superior technique for multisensor-multi-target tracking and data association.

The recent activity in the area of IMM methods indicates that this emerging body of knowledge experiences a rapid transition to becoming relevant to engineering practice by solving real problems.

Acknowledgment: The first author wish to thank Dr. S. Rogers, Dr. S. Levin and Dr. B. Golubev from Elta Corp., Israel, for introducing the problem and continuing support through all the stages of the research.

References

[ALOUANI 91a] A.T. Alouani and W.D. Blair, Use of a Kinematic Constraint in Tracking Constant Speed, Maneuvering Targets, Proc. 30th Conf. Decision and Control, Brighton, England, Dec. 1991, 2055-1058.

[ALOUANI 91b] A.T. Alouani, P. Xia, T.R. Rice, and W.D. Blair, A Two-Stage Kalman Estimator For State Estimation in the Presence of Random Bias and for Tracking Manuevering Targets, Proc. 30th IEEE Conf. Decision and Control, Brighton, England, Dec. 1991, 1059 - 1062.

[ATHERTON 94] D.P. Atherton and H-J. Lin, Parallel Implementation of IMM Tracking Algorithm using Transputers, IEEE Proc. Radar Sonar Navig., Vol. 141, No. 6, Dec. 1994, 325-332.

[AVERBUCH 91a] A. Averbuch, S. Itzikowitz, and T. Kapon, Radar Target Tracking—Viterbi versus IMM, IEEE Trans. Aerosp. Electronic Systems, Vol. AES-27, No. 3, May 1991, 550-563.

[AVERBUCH 91b] A. Averbuch, S. Itzikowitz, and T. Kapon, Parallel Implementation of Multiple Model Tracking Algorithms, IEEE Trans. Parallel and Distributed Systems, Vol. 2, No. 2, April 1991, 242-251.

[BARSHALOM 88] Y. Bar-Shalom and T.E. Fortmann, “Tracking and Data Association”, Academic Press, New York, FL, 1988.

[BARSHALOM 89a] Y. Bar-Shalom, K.C. Chang, and H.A.P. Blom, Tracking a Maneuvering Target Using Input Estimation Versus the Interacting Multiple Model Algorithm, IEEE Trans. Aerosp. Electronic Systems, Vol. AES-25, No. 2, Mar. 1989, 296-300.

[BARSHALOM 89b] Y. Bar-Shalom, Recursive Tracking Algorithms: From the Kalman Filter to Intelligent Trackers for Cluttered Environment, Proc. 1989 IEEE Intern'l. Conf. on Control and Applications, Jerusalem, 1989.

[BARSHALOM 90a] Y. Bar-Shalom (Ed.), "Multitarget-Multisensor Tracking: Advanced Applications", Artech House, Norwood, MA, 1990.

[BARSHALOM 90b] Y. Bar-Shalom, K.C. Chang, and H.A.P. Blom, Automatic Track formation in Clutter with a Recursive Algorithm, in [BARSHALOM 90a], 25 - 42.

[BARSHALOM 92a] Y. Bar-Shalom (Ed.), "Multitarget-Multisensor Tracking: Applications and Advances (Vol. II)," Artech House, Norwood, MA, 1992.

[BARSHALOM 92b] Y. Bar-Shalom, K.C. Chang, and M.A.P. Blom, Tracking Splitting Targets in Clutter by Using an Interacting Multiple Model Joint Probabilistic Data Association Filter, in [BARSHALOM 92a], 93-110.

[BARSHALOM 93] Y. Bar-Shalom, X. Li, "Estimation and Tracking: Principles, Techniques, and Software, Artech House, 1993.

[BARSHALOM 95] Y. Bar-Shalom,X.R. Li, "Multitarget-Multisensor Tracking: Principles and Techniques", Storrs, Ct (06269-3157): YBS Publishing, 1995.

[BIERMAN 77a] G.J. Bierman and C.L. Thornton, Numerical comparison of Kalman Filter algorithms: Orbit determination case study, Automatica, 13, Jan. 1977, 23-27.

[BIERMAN 77b] G.J. Bierman, "Factorization Methods for Discrete Sequential Estimation", Academic Press, New York, 1977.

[BLACKMAN 86] S.S. Blackman, "Multiple Target Tracking with Radar Applications", Arech House, Norwood, MA, 1986.

[BLACKMAN 92] S.S. Blackman, Multitarget Tracking with an Agile Beam Radar, in [BARSHALOM 92a], 237-269.

[BLAIR 92] W.D. Blair and G.A. Watson, Interacting Multiple Bias Model Algorithm With Application To Tracking Maneuvering Targets, Proc. 31st Conf. Decision and Control, Tucson, Arizona, Dec. 1992, 3790-3795.

[BLAIR 94a] W. D. Blair, Toward the Integration of Tracking and Signal Processing for Phased Array Radar, Signal and Data Processing for Small Targets, Orlando, April 1994.

[BLAIR 94b] W. D. Blair, G. A. Watson, and S. A. Hoffman, Benchmark Problem for Beam Pointing Control of Phased Array Radar Against Maneuvering Targets, Proc. 1994 American Control Conf., Baltimore, Maryland, June 1994.

[BLAIR 94c] W. D. Blair and G. A. Watson, IMM Algorithm for Solution to Benchmark Problem for Tracking Maneuvering Targets, Proc. of Acquisition, Tracking, and Pointing, SPIE Orlando 94, April 1994.

[BLOM 84] H.A. Blom, An Efficient filter for Abruptly Changing Systems, Proc. 23rd IEEE Conf. Decision and Control, Las Vegas, NV, Dec. 1984, 656-658.

[BLOM 85] H.A. Blom, An Efficient Decision-Making-Free Filter For Processes With Abrupt Changes, Proc. IFAC, Symp. IFAG, Idenification and System Parameter Estimation, 1985, 631-636.

[BLOM 88] H.A.P. Blom and Y. Bar-Shalom, The Interacting Multiple Model Algorithm for Systems with Markovian Switching Coefficients, IEEE Trans. Automat. Control, Vol. AC-33, No. 8, Aug. 1988, 780-783.

[BLOM 90] H.A.P Blom and Y. Bar-Shalom, Time-Reversion of a Hybrid State Stochastic Difference System with a Jump-Linear Smoothing Application, IEEE Trans. Information Theory, Vol. 36, No. 4, July 1990, 836-847.

[BLOM 91] H.A. Blom, Hybrid State Estimation For Systems with Semi-Markov Switching Coefficients, Proc. Europ. Control Conf., Grenoble, July 1991.

[BLOM 92] H.A.P. Blom, R.A. Hogendoorn, and B.A. Van Doorn, Design of a Multisensor Tracking System for Advanced Air Traffic Control, in [BAR-SHALOM 92a], 31-63.

[BOGLER 87] P.L. Bogler, Tracking a Maneuvering Target Using Input Estimation, IEEE Trans. Aerosp. Electronic systems, Vol. AES-23, No. 3, May 1987, 298-310.

[BOGLER 90] P.L. Bogler, "Radar Principles with Applications to Tracking Systems", J. Wiley & Sons, New York, 1990.

[BROOKNER 91] E. Brookner (Ed.), "Practical Phased Array Antenna Systems", Artech House, Boston, 1991.

[CAMPO 88] L. Campo, P. Mookerjee, and Y. Bar-Shalom, Failure Detection Via Recursive Estimation for a Class of Semi-Markov Switching Systems, Proc. 27th IEEE Conf. Decision and Control, Austin, Texas, Dec. 1988, 1966-1971.

[CAMPO 90] L. Campo and Y. Bar-Shalom, Control of Discrete-Time Hybrid Stochastic Systems, Proc. 1990 American Control Conf., Boston, June 19990, 7 - 12.

[CAMPO 91] L. Campo, P. Mookerjee, and Y. Bar-Shalom, State Estimation for Systems with Sojourn-Time-Dependent Markov Model Switching, IEEE Trans. Automat. Control, Vol. AC-36, No. 2, Feb. 1991, 238-243.

[CHANG 78] C.B. Chang and M. Athans, State Estimation for Discrete System with Switching Parameters, IEEE Trans. Aerosp. Electronic Systems, Vol. AES-14, May 1978, 418-425.

[CHANG 86] K.C. Chang, C.Y. Chong, and Y. Bar-Shalom, Joint Probabilistic Data Association in Distributed Sensor Networks, IEEE Trans. Automat. Control, Vol. AC-31, No. 10, Oct. 1986, 889-897.

[CHANG 87] K.C. Chang and Y. Bar-Shalom, Distributed Multiple Model Estimation, Proc. American Control Conf., 1987, 797-802.

[CHANG 89] K.C. Chang and Y. Bar-Shalom, Distributed Adaptive Estimation with Probabilistic Data Association, Automatica, Vol. 25, No. 3, 1989, 359-369.

[CHONG 85] C.H. Chong, S. Mori, and K.C. Chang, Information Fusion in Distributed Sensor Networks, Proc. American Contrrol Conf. 1985, 830 - 835.

[DAEIPOUR 94] E. Daeipour, Y. Bar-Shalom, and X. Li, Adaptive Beam Pointing Control of a Phased Array Radar Using an IMM Estimator, Proc. 1994 American Control Conf., Baltimore, Maryland, June 1994.

[DAEIPOUR 95] E. Daeipour and Y. Bar-Shalom, An Interacting Multiple Model Approach for Target Tracking with Glint Noise, IEEE Trans. Aerosp. Electronic Systems, Vol. AES-31, No. 2, April 1995.

[DAUM 90] F.E. Daum, Bounds on Performance for Multiple Target Tracking, IEEE Trans. Automat. Control, Vol. AC-35, No. 4, Apr. 1990, 443-446.

[DUFOUR 91] F. Dufour and M. Mariton, Tracking a 3D Maneuvering Target with Passive Sensors, IEEE Trans. Aerosp. Electronic systems, Vol. 27, No. 4, July 1991, 725-738.

[DUFOUR 92] F. Dufour and M. Mariton, Passive Sensor Data Fusion and Maneuvering Target Tracking, in [BARSHALOM 92a], 65-92.

[EZZINE 88] J. Ezzine and A.H. Haddad, Error bounds in the Averaging of Hybrid Systems, Proc. 27th IEEE Conf. Decision and Control, Austin, TX, Dec. 1988, 1787-1791.

[FARINA 80] A. Farina and S. Pardini, Survey of Radar Data-Processing Techniques in Air-Traffic-Control and Surveillance systems, IEE Proc., Vol. 127, Pt.F, No. 3, June 1980, 190 - 204.

[FORTMANN85] T.E. Fortmann, Y. Bar-Shalom, M. Scheffe, and S. Gelfand, Detection Thresholds for Tracking in Clutter - A Connection Between Estimation and Signal Processing, IEEE Trans. Automat. Control, Vol. AC-30, No. 5, Mar. 1985, 221-229.

[GUU91] J.A. Guu and C.H. Wei, Maneuvering Target Tracking Using IMM Method at High Measurement Frequency, IEEE Trans. Aerosp. Electronic Systems, Vol. AES-27, No. 3, May 1991, 514-519.

[HEWER 87] G. A. Hewer, R. D. Martin, and J. Seh, Robust preprocessing for Kalman filtering of glint noise, IEEE Trans. Aerosp. Electronic Systems, Vol. AES-23, No. 1, Jan. 1987, 120-128.

[HOULES 89] A. Houles and Y. Bar-Shalom, Multisensor Tracking of a Maneuvering Target in Clutter, IEEE Trans. Aerosp. Electronic Systems, Vol. AES-25, No. 2, Mar. 1989, 176-189.

[KENEFIC 90] R. Kenefic, Active Sonar Application of a U-D square Root PDAF, IEEE Trans. Aerosp. Electronic Systems, Vol. AES-26, No. 5, Sept. 1990, 850-857.

[KERR 89] T.H. Kerr, Status of CR-Like Lower Bounds for Nonlinear Filtering, IEEE Trans. Aerosp. Electronic Systems, Vol. AES-25, No. 5, Sept. 1989, 590-601.

[LEONDES 73] C.T. Leondes and J. O. Pearson, Kalman filtering of systems with parameter uncertainties - a survey, Int. J. Control, Vol. 17, No. 4, 1973, 785 - 801.

[LERRO 93] D. Lerro and Y. Bar-Shalom, Interacting Multiple Model Tracking with Target Amplitude Feature, IEEE Trans. Aerosp. Electronic systems, Vol. 29, No. 2, Apr. 1993, 494-508.

[LI 91] X.R.Li and Y. Bar-Shalom, Stability Evaluation and Track Life of he PDAF for Tracking in Clutter, IEEE Trans. Automat. Control, Vol. AC-36, No. 5, May 1991, 588-602.

[LI 92a] X.R.Li and Y. Bar-Shalom, Mode-Set Adaptation in Multiple-Model Estimators for Hybrid Systems, Proc. 1992 American Control Conf., Chicago, IL. June 1992, 1794-1799; to appear in IEEE T-AC.

[LI 92b] X.R. Li and Y. Bar-Shalom, Performance Prediction of the Interacting Multiple Model Algorithm, Proc. 1992 American Control Conf., Chicago, IL, June 1992, 2109-2113; IEEE T-AES 7-93.

[LI 93] X.R. Li and Y. Bar-Shalom, Design of an Interacting Multiple Model Algorithm for Air Traffic Control Tracking, IEEE Trans. Control Systems Technology, Vol. 1, No. 3, Sept. 1993, 186-194.

[LI 94] X.R. Li and Y. Bar-Shalom, A Recursive Multiple Model Approach to Noise Identification, IEEE Trans. Aerospace Electronic Systems, Vol. 30, No. 3, July 1994, 671-684.

[LIN 93] H-J. Lin and D.P. Atherton, An Investigation of the SFIMM Algorithm for Tracking Maneuvering Targets, Proc. 32nd Conf. Decision and Control, San Antonio, Texas, Dec. 1993, 930-935.

[LU 92] M. Lu, X. Qiao, and G. Chen, A Parallel Square-Root Algorithm for Modified Extended Kalman Filter, IEEE Trans. Aerosp. Electronic Systems, Vol. AES-28, No. 1, Jan. 1992, 153-163.

[MARITON 92] M. Mariton and D. Sworder, Maneuvering Target Tracking: Imaging and Non-Imaging Sensors, in "Control and Dynamic Systems: Advances in Theory and Applications", Ed.: C.T. Leondes, Academic Press, New York 1992, 483-516.

[MASRELIEZ 75] C. J. Masreliez, Approximate non-Gaussian filtering with linear state and observation relations, IEEE Trans. Automat. Control, Feb. 1975, 107-110.

[MASRELIEZ 77] C.J. Masreliez and R.D. Martin, Robust Bayesian Estimation for the Linear Model and Robustifying the Kalman Filter, IEEE Trans. Automatic Control, 1977, 361-371.

[MAYBECK 90] P.S. Maybeck and M.R. Schore, Robustness of a Moving-Bank Multiple Model Adaptive Algorithm for Control of a Flexible Spacestructure, Proc. National Aerosp. and Electronics Conf. NAECON 90, Dayton, 1990, 368-374.

[MILLNERT 80] M. Millnert, An Approach to Recursive Identification of Abruptly Changing Systems, Proc. 19th IEEE Conf. on Decision and Control, 1980, 1026.

[MOOKERJEE 87] P. Mookerjee, L. Campo, and Y. Bar-Shalom, Estimation in Systems with a Semi-Markov Switching Model, Proc. 26th Conf. Decision and Control, Los Angeles, CA, Dec. 1987, 332-334.

[MOOSE 75] R.L. Moose, An Adaptive State Estimation Solution to the Maneuvering Target Problem, IEEE Trans. Aut. Control, AC-20, 1975, 359.

[MOOSE 79] R. L. Moose, H. F. Vanlandingham, and D. H. McCabe, Modeling and Estimation for Tracking maneuvering targets IEEE Trans. Aerosp. Electronic Systems, Vol. AES-15, No. 3, May 1979, 448-455.

[MORI 92] S. Mori, K.C. Chang, and C.Y. Chong, Performance Analysis of Optimal Data Association - With Applications to Multiple Target Tracking, in [BARSHALOM 92a], 183-235.

[MUNIR 94] A. Munir and D.P. Atherton, Maneuvering Target Tracking Using An Adaptive Interacting Multiple Model Algorithm, Proc. 1994 American Control Conf., Baltimore, Maryland, June 1994, 1324-1328.

[NAGY 90] P. Nagy and S. Bier, Intelligent Internetted Sensor Management Systems for Tactical Aircraftr, Proc. National Aerosp. and Electronics Conf. NAECON 90, Dayton, 1990, 321-327.

[PATTIPATI 83] K.R. Pattipati and N.R. Sandell, A Unified View of State Estimation in Switching Environments, Proc. 1983 American Control Conf., 1983, 458-465.

[RAGHAVAN 92] V. Raghavan, K.R. Pattipati, and Y. Bar-Shalom, Efficient Square-root Algorithms for PDA, IMM and IMMPDA Filters, Proc. 31st Conf. Decision and Control, Tucson, Arizona, Dec. 1992, 3680-3685.

[RAGHAVAN 93] V. Raghavan, K.R. Pattipati and Y. Bar-Shalom, Efficient L-D Factorization Algorithms for PDA, IMM, and IMMPDA Filters, IEEE Trans. Aerosp. Electronic Systems, Vol. 29, No. 4, Oct. 1993, 1297-1310.

[RAISBECK 83] G. Raisbeck, Viewing the Issue, IEEE Journal of Oceanic Engineering, Special Issue on Positioning, Localization and Tracking, Vol. OE-8, No. 3, July 1983, 110-112.

[ROGERS 87] S.R. Rogers, Alpha-Beta Filter with Correlated Measurement Noise, IEEE Trans. Aerosp. Electronic Systems, Vol. AES-23, No. 4, July 1, 1987, 592-594.

[ROGERS 91] S.R. Rogers, Diffusion Analysis of Track Loss in Clutter, IEEE Trans. Aerosp. Electronic Systems, Vol. AES-2, No. 2, Mar. 1991, 380 - 387.

[SUNG 87] T.Y. Sung, Y.H. Hu, Parallel VLSI Implementation of the Kalman Filter, IEEE Trans. Aerosp. Electronic Systems, Vol. AES-23, No. 2, Mar. 1987, 215-224.

[TUGNAIT 82] J.K. Tugnait, Detection and Estimation for Abruptly Changing Systems, Automatica, Vol. 18, No. 5, 1982, 607-615.

[VACHER 92] P. Vacher, I. Barret, and M. Gauvrit, Design of a Tracking Algorithm for an Advanced ATC System, in [BARSHALOM 92a], pp. 1 - 29.

[VonKEUK 93] G.VonKeuk and S.S. Blackman, On Phased-Array Radar Tracking and Parameter Control, IEEE Trans. Aerosp. Electronic systems, Vol. 29, No. 1, Jan. 1993, 186-194.

[WALTZ 90] E. Waltz and J. Llinas, "Multisensor Data Fusion", Artech House, Norwood, MA, 1990.

[WASHBURN 85] R.B. Wahsburn, T.G. Allen, and D. Teneketzis, Performance Analysis for Hybrid State Estimation Problems, Proc. 1985 American Control Conf., Boston, MA, June 1985, 1047-1053.

[WILLNER 76] D. Willner, C.B. Chang, and K.P. Dunn, Kalman Filter Algorithms for a Multi-Sensor System, Proc. 15th IEEE Conf. Decision and Control, Dec. 1976, 570 - 574.

[WU 94] W.R. Wu and P.P. Cheng, A Nonlinear IMM Algorithm for Maneuvering Target Tracking, IEEE Trans. Aerospace Electronic Systems, Vol. 30, No. 3, July 1994, 875-886.

[ZEITZ 93] F.H. Zeitz and P.S. Maybeck, An Alternate Algorithm for Discrete-Time Filtering, IEEE Trans. Aerosp. Electronic Systems, Vol. 29, No. 4, Oct. 1993, 1123 - 1136.