14

Random Set Theory for Target Tracking and Identification

Ronald Mahler
Lockheed Martin

0-8493-2379-7/01/$0.00+$1.50

The subject of this chapter is finite-set statistics (FISST), which is described extensively in *Mathematics of Data Fusion*,[1] and summarized in the recent article, "An Introduction to Multisource-Multitarget Statistics and its Applications."[2] FISST provides a fully unified, scientifically defensible, probabilistic foundation for the following aspects of multisource, multitarget, multiplatform data fusion: (1) multisource integration (i.e., detection, identification, and tracking) based on Bayesian filtering and estimation;[3-7] (2) sensor management using control theory;[5,8,9] (3) performance evaluation using information theory;[10-13] (4) expert systems theory (e.g., fuzzy logic, the Dempster-Shafer theory of evidence, and rule-based inference);[14-16] (5) distributed fusion;[17] and (5) aspects of situation/threat assessment.[18]

The core of FISST is a multisource-multitarget differential and integral calculus based on the fact that belief-mass functions are the multisensor-multitarget counterparts of probability-mass functions. One purpose of this calculus is to enable signal processing engineers to directly generalize conventional, engineering-friendly statistical reasoning to multisensor, multitarget, multi-evidence applications. A second purpose is to extend Bayesian (and other probabilistic) methodologies so that they are capable of dealing with (1) imperfectly characterized data and sensor models and (2) true sensor models and true target models for multisource-multitarget problems. One consequence is that FISST encompasses certain expert-system approaches that are often described as "heuristic" — fuzzy logic, the Dempster-Shafer theory of evidence, and rule-based inference — as special cases of a single probabilistic paradigm.

FISST has attracted a great deal of positive attention — and two distinct forms of criticism — since it was first described in 1994. "An Introduction to Multisource-Multitarget Statistics and its Applications"[2] was written as a response to both reactions, and this chapter is essentially a distillation of that monograph.

On the one hand, many engineering researchers have complained that *Mathematics of Data Fusion* is difficult to understand. "An Introduction to Multisource-Multitarget Statistics and its Applications" explains that if the foundations outlined in that book are taken for granted, FISST can — by design — be reduced to a relatively simple "Statistics 101" formalism suitable for real applications using real data. Some applications currently being investigated at the applied-research level are[2]

- Multisource datalink fusion and target identification,[19]
- Naval passive-acoustic anti-submarine fusion and target identification,[20]
- Air-to-ground target identification using SAR,[21]
- Scientific performance evaluation of multisource-multitarget data fusion algorithms,[13,22,23]
- Unified detection and tracking using true multitarget likelihood functions and approximate nonlinear filters,[24]
- Joint tracking, pose estimation, and target identification using HRRR.[25]

In recent years, FISST has been the target of a few attacks that can be summarized by the following statement: "The multisource-multitarget engineering problems addressed by FISST actually require nothing more complicated than Bayes' rule, which means that FISST is of mere theoretical interest at best and, at worst, is nothing more than pointless mathematical obfuscation." "Multisource-Multitarget Statistics" demonstrates that this assertion is extraordinary — less in the ignorance that it displays regarding FISST than in the ignorance that it displays regarding Bayes' rule. Venturing away from standard applications addressed by standard textbooks and applying Bayesian approaches in a "cookbook" fashion (rather than using a FISST approach) can create severe difficulties. This chapter highlights these difficulties and shows how and why FISST resolves them.

The chapter is organized as follows. Section 14.1 introduces the basic practical issues underlying Bayesian tracking and identification. Section 14.2 summarizes the basic statistical foundations of single-sensor, single-target tracking and identification. Section 14.3 introduces the concept of a multisource-multitarget measurement model; multitarget motion models are likewise introduced in Section 14.4. The mathematical core of the approach — the FISST multisource-multitarget integral and differential calculus — is summarized in Section 14.5. The basic concepts of multisource-multitarget statistics — especially those of true multitarget measurement models and true multitarget motion models — are described in Section 14.6. Optimal-Bayes and Robust-Bayes multisource-multitarget fusion, detection, tracking, and identification (including sensor management) are described in Sections 14.7 and 14.8, respectively. Conclusions are presented in Section 14.9.

14.1 Introduction

This section describes the problems that FISST is meant to address and summarizes the FISST approach for addressing them. The section is organized as follows. Sections 14.1.1 and 14.1.2 describe some basic engineering issues in single-target and multitarget Bayesian inference — the "Bayesian Iceberg." The FISST approach is summarized in in Section 14.1.3, and Section 14.1.4 shows why this approach is necessary if the "Bayesian Iceberg" is to be confronted successfully.

14.1.1 The "Bayesian Iceberg": Models, Optimality, Computability

Recursive Bayesian nonlinear filtering and estimation has become the most accepted standard for developing algorithms that are optimal, theoretically defensible, and practical. However, the success of any optimal-Bayes technique hinges on a number of subtle issues. In recent years, shortcomings in conventional Bayesian thinking — and corresponding desires to expand the conceptual scope of Bayesian methods to deal with them — have become evident. The purpose of this section is to explain why this is the case. Our jumping-off point is Bayes' rule:

$$f_{posterior}\left(\mathbf{x}\middle|\mathbf{z}\right)=\frac{f\left(\mathbf{x}\middle|\mathbf{z}\right)f_{prior}\left(\mathbf{x}\right)}{f\left(\mathbf{z}\right)} \tag{14.1}$$

where:

$\mathbf{x}$ denotes the unknown quantities of interest (e.g., position, velocity, and target class)

$f_{prior}(\mathbf{x})$, the prior distribution, encapsulates our previous knowledge about $\mathbf{x}$

$\mathbf{z}$ represents new data

$f(\mathbf{x}|\mathbf{z})$, the likelihood function, describes the generation of data

$f_{posterior}(\mathbf{x}|\mathbf{z})$, the posterior distribution, encapsulates our current knowledge about $\mathbf{x}$

$f(\mathbf{z})$ is the normalization constant

If time has expired before collection of the new information, $\mathbf{z}$, then $f_{prior}(\mathbf{x})$ cannot be immediately used in Bayes' rule. It must first be extrapolated to a new prior $f^{+}_{prior}(\mathbf{x})$ that accounts for the uncertainties caused by possible interim target motion. This extrapolation is accomplished using either the Markov time-prediction integral[26]

$$f^{+}_{prior}\left(\mathbf{x}\right)=\int f^{+}\left(\mathbf{x}\middle|\mathbf{y}\right)f_{prior}\left(\mathbf{y}\right)d\mathbf{y} \tag{14.2}$$

or solution of the Fokker-Planck equation (FPE).[27] In Equation 14.2, the density $f^{+}(\mathbf{x}|\mathbf{y})$ is the Markov transition density that describes the likelihood that the target will have state $\mathbf{x}$ if it previously had state $\mathbf{y}$.

The density, $f_{posterior}(\mathbf{x}|\mathbf{z})$, contains all relevant information about the unknown state variables (i.e., position, velocity, and target identity) contained in $\mathbf{x}$. One could, in principle, plot graphs or contour maps of $f_{posterior}(\mathbf{x}|\mathbf{z})$ in real time. However, this presupposes the existence of a human operator trained to interpret them. Since current operational reality in most data fusion and tracking applications consists of fewer and fewer operators overwhelmed by more and more data, full automation is a necessity. To render the information in $f_{posterior}(\mathbf{x}|\mathbf{z})$ available for fully automated real-time applications, we need a Bayes-optimal state estimator that extracts an estimate $\hat{\mathbf{x}}$ of the actual target state from the posterior. The maximum *a posteriori* (MAP) and expected *a posteriori* (EAP) estimators

$$\hat{\mathbf{x}}^{MAP} = \arg\max_{x} f_{posterior}(\mathbf{x}|\mathbf{z}) \qquad \hat{\mathbf{x}}^{EAP} = \int x f_{posterior}(\mathbf{x}|\mathbf{z})d(\mathbf{x}) \tag{14.3}$$

are the most familiar Bayes-optimal state estimators. (Here, $\hat{\mathbf{x}} = \arg\max_x f(\mathbf{x})$ means that $f(\hat{\mathbf{x}})$ is the largest possible value of $f(\mathbf{x})$.)

The current near-sacrosanct status of the Bayesian approach is due in large part to the fact that it leads to provably optimal algorithms within a simple conceptual framework — one that imposes no great mathematical demands on its practitioners. Since engineering mathematics is a tool and not an end in itself, this simplicity is a great strength — but also a great weakness. Recent years have seen the emergence of a "cookbook Bayesian" viewpoint, which seems to consist of the belief that it is possible to appear deeply authoritative about any engineering research and development problem — and at the same time avoid careful thinking — by (1) writing down Bayes' rule, (2) uttering the sacred incantation "Bayes-optimal" (usually without a clue as to what this phrase actually means), (3) declaring victory, and then (4) portraying complacency toward possible unexpected difficulties as a sign of intellectual superiority. What such a belief reveals is a failure to grasp the fact that in and of itself, Bayes' rule is nearly content-free — its proof requires barely a single line. Its power derives from the fact that it is merely the visible tip of a conceptual iceberg, the existence of which tends to be forgotten precisely because the rest of the iceberg is taken for granted. In particular, both the optimality and the simplicity of the Bayesian framework can be taken for granted only within the confines of standard applications addressed by standard textbooks. When one ventures out of these confines one must exercise proper engineering prudence — which includes verifying that standard textbook assumptions still apply.

14.1.1.1 The Bayesian Iceberg: Sensor Models

Bayes' rule exploits, to the best possible advantage, the high-fidelity knowledge about the sensor contained in the likelihood function $f(\mathbf{z}|\mathbf{x})$. If $f(\mathbf{z}|\mathbf{x})$ is too imperfectly understood, then an algorithm will "waste" a certain amount N_{sens} of data trying (and perhaps failing) to overcome the mismatch between model and reality. Many forms of data (e.g., generated by tracking radars) are well enough characterized that $f(\mathbf{z}|\mathbf{x})$ can be constructed with sufficient fidelity. Other kinds of data (e.g., synthetic aperture radar (SAR) images or high range resolution radar (HRRR) range profiles) are proving to be so difficult to simulate that there is no assurance that sufficiently high fidelity will ever be achieved, particularly in real-time operation. This is the point at which "cookbook Bayesian" complacency enters the scene: we jot down Bayes' rule and declare victory. In so doing, we have either failed to understand that there is a potential problem — that our algorithm is Bayes-optimal with respect to an imaginary sensor unless we have the provably true likelihood $f(\mathbf{z}|\mathbf{x})$ — or we are playing a shell game. That is, we are avoiding the real algorithmic issue (what to do when likelihoods cannot be sufficiently well characterized) and instead implicitly "passing the buck" to the data simulation community.

Finally, there are types of data — features extracted from signatures, English-language statements received over link, and rules drawn from knowledge bases — that are so ambiguous (i.e., poorly understood from a statistical point of view) that probabilistic approaches are not obviously applicable. Rather than seeing this as a gap in Bayesian inference that needs to be filled, the "cookbook Bayesian" viewpoint sidesteps the problem and frowns upon those who attempt to fill the gap with heuristic approaches such as fuzzy logic.

Even if $f(\mathbf{z}|\mathbf{x})$ can be determined with sufficient fidelity, multitarget problems present a new challenge. We will "waste" data — or worse — unless we find the corresponding provably true multitarget likelihood — the specific function $f(\mathbf{z}_1,\ldots,\mathbf{z}_m|\mathbf{x}_1,\ldots,\mathbf{x}_n)$ that describes, with the same high fidelity as $f(\mathbf{z}|\mathbf{x})$, the likelihood that the sensor will collect observations $\mathbf{z}_1,\ldots,\mathbf{z}_m$ (m random) given the presence of targets with states $\mathbf{x}_1,\ldots\mathbf{x}_n$ (n random). Again, "cookbook Bayesian" complacency overlooks the problem — that our boast of "Bayes-optimality" is hollow unless we can construct the provably true $f(\mathbf{z}_1,\ldots,\mathbf{z}_m|\mathbf{x}_1,\ldots\mathbf{x}_n)$ — or encourages us to play another shell game, constructing a heuristic multitarget likelihood and implicitly assuming that it is true.

14.1.1.2 The Bayesian Iceberg: Motion Models

Much of what has been said about likelihoods $f(\mathbf{z}|\mathbf{x})$ applies with equal force to Markov densities $f^+(\mathbf{x}|\mathbf{y})$. The more accurately that $f^+(\mathbf{x}|\mathbf{y})$ models target motion, the more effectively Bayes' rule will do its job. Otherwise, a certain amount N_{targ} of data must be expended in overcoming poor motion-model selection. Once again, however, what does one do in the multitarget situation if $f^+(\mathbf{x}|\mathbf{y})$ is truly accurate? We must find the provably true multitarget Markov transition density — i.e., the specific function $f(\mathbf{x}_1,\ldots,\mathbf{x}_n|\mathbf{y}_1,\ldots\mathbf{y}_{n'})$ that describes, with the same high fidelity as $f^+(\mathbf{x}|\mathbf{y})$ how likely it is that a group of targets that previously were in states $\mathbf{y}_1,\ldots\mathbf{y}_{n'}$ (n' random) will now be found in states $\mathbf{x}_1,\ldots\mathbf{x}_n$ (n random). The "cookbook Bayesian" viewpoint encourages us to simply assume that

$$f(\mathbf{x}_1,\ldots,\mathbf{x}_n|\mathbf{y}_1,\ldots,\mathbf{y}_{n'}) = f^+(\mathbf{x}_1|\mathbf{y}_1)\ldots f^+(\mathbf{x}_n|\mathbf{y}_{n'})$$

meaning, in particular, that the number of targets is constant and target motions are uncorrelated. However, in real-world scenarios, targets can appear (e.g., MIRVs and decoys emerging from a ballistic missile re-entry vehicle) or disappear (e.g., aircraft that drop beneath radar coverage) in correlated ways. Consequently, multitarget filters that assume uncorrelated motion and/or constant target number may perform poorly against dynamic multitarget environments, for the same reason that single-target trackers that assume straight-line motion may perform poorly against maneuvering targets. In either case, data is "wasted" in trying to overcome — successfully or otherwise — the effects of motion-model mismatch.

14.1.1.3 The Bayesian Iceberg: Output Generation

When we are faced with the problem of extracting an "answer" from the posterior distribution, complacency may encourage us to blindly copy state estimators from textbooks — e.g. the MAP and EAP estimators of Equation 14.3 — or define ad hoc ones. Great care must be exercised in the selection of a state estimator, however. If the state estimator has unrecognized inefficiencies, then a certain amount, N_{est}, of data will be unnecessarily "wasted" in trying to overcome them, though not necessarily with success. For example, the EAP estimator plays an important role in theory, but often produces erratic and inaccurate solutions when the posterior is multimodal.

Another example involves joint-estimation applications in which the state $\mathbf{x}$ has the form $\mathbf{x} = (\mathbf{u}, \mathbf{v})$, where $\mathbf{u}$, $\mathbf{v}$ are two different kinds of state variables — e.g. kinematic state variables $\mathbf{u}$ versus target-identity state variables $\mathbf{v}$. In this case, we may be tempted to treat $\mathbf{u}, \mathbf{v}$ as nuisance parameters and compute separate marginal-MAP estimates:

$$\hat{\mathbf{u}}^{MAP} = \arg\max_{\mathbf{u}} \int f_{posterior}\left(\mathbf{u},\mathbf{v}\middle|\mathbf{z}\right)d\mathbf{v} \qquad \hat{\mathbf{v}}^{MAP} = \arg\max_{\mathbf{v}} \int f_{posterior}\left(\mathbf{u},\mathbf{v}\middle|\mathbf{z}\right)d\mathbf{u} \tag{14.4}$$

Because integration loses information about the state variable being regarded as a nuisance parameter, estimators of this type can converge more slowly than a joint estimator. They will also produce noisy, unstable solutions when $\mathbf{u}$, $\mathbf{v}$ are correlated and the signal-to-noise ratio is not large.[28]

In the multitarget case, the dangers of taking state estimation for granted become even more acute. For example, we may fail to notice that the multitarget versions of the standard MAP and EAP estimators are not even defined, let alone provably optimal. The source of this failure is, moreover, not some abstruse point in theoretical statistics. Rather, it is a "cookbook Bayesian" carelessness that encourages us to overlook unexpected difficulties caused by a familiar part of everyday engineering practice: *units of measurement.*

14.1.1.4 The Bayesian Iceberg: Formal Optimality

The failure of the standard Bayes-optimal state estimators in the multitarget case has far-reaching consequences for *optimality.* One of the reasons that the "cookbook Bayesian" viewpoint has become so prevalent is that — because well-known textbooks have already done the work for us — we can, without expended effort or cost, proclaim the "optimality" of whatever algorithms we propose. A key point that is often overlooked, however, is that many of the classical Bayesian optimality results depend on certain seemingly esoteric mathematical concerns. In perhaps ninety-five percent of all engineering applications — which is to say, standard applications covered by the standard textbooks — these concerns can be safely ignored. Danger awaits the "cookbook Bayesian" viewpoint in the other five percent. Such is the case with multitarget applications. Because the standard Bayes-optimal state estimators fail in the multitarget case, we must construct new multitarget state estimators and prove that they are well behaved. This means that words like "topology" and "measurable" can no longer be swept under the rug.

14.1.1.5 The Bayesian Iceberg: Computability

If the simplicity of Equations 14.1 and 14.3 cannot be taken for granted in so far as modeling and optimality are concerned, then such is even more the case when it comes to computational tractability.[29] The prediction integral and Bayes normalization constant must be computed using numerical integration, and — since an infinite number of parameters are required to characterize $f_{posterior}(\mathbf{x}|\mathbf{y})$ in general — approximation is unavoidable. A "cookbook Bayesian" viewpoint may tempt us to apply blindly (deceptively) easy-to-understand textbook techniques that seem to promise high (i.e., $O(N)$ or $O(N \log N)$) computational efficiencies per data-collection cycle. Naive approximations, however, create the same difficulties as model-mismatch problems. An algorithm must "waste" a certain amount N_{appx} of data overcoming — or failing to overcome — accumulation of approximation error, numerical instability, etc.

One example is the use of central finite-difference schemes to solve the Fokker-Planck equation (FPE) for f^{+}_{prior}. In filtering problems, the convection term of the FPE dominates the diffusion term. Under such circumstances, central differencing results in loss of probability mass (as well as creation of negative probability), not only at the boundaries but throughout the region of interest, often resulting in poor solutions and numerical instability. This fact has long been well known in the computational fluid dynamics community.[30] Not only has this problem been cited as one of "Seven Deadly Sins of Numerical Computation,"[31] it is such a well-known error that it is cited as such in *Numerical Recipes in C.*[32]

14.1.1.6 The Bayesian Iceberg: Robustness

The engineering issues addressed in the previous sections (measurement-model mismatch, motion-model mismatch, inaccurate or slowly-convergent state estimators, accumulation of approximation error, numerical instability) can be collectively described as problems of robustness — i.e., the "brittleness" that Bayesian approaches can exhibit when reality deviates too greatly from assumption. One might be tempted to argue that, in practical application, these difficulties can be overcome by simple *brute force* — i.e., by assuming that the data-rate is high enough to permit a large number of computational cycles per unit time. In this case — or so the argument goes — the algorithm will override its internal inefficiencies because the total amount N_{data} of data that is collected is much larger than the amount $N_{ineffic} \triangleq N_{sens}+N_{targ}+N_{est}+N_{apps}$ of data required to overcome those inefficiencies.

If this were the case, there would be few tracking and target identification problems left to solve. Most current challenging problems are challenging *either because data rates are not sufficiently high or because brute force computation cannot be accomplished in real time.* This unpleasant reality is particularly evident in target-classification problems. An optimal-Bayes target identification algorithm that is forced to use less-than-high-fidelity sensor models has a tendency to be *highly confident about incorrect identifications.*

This is because it treats its imperfect target models as though they are perfect, thus potentially interpreting a weak or spurious match (between data and target model) as a high-likelihood match. Because data is either too sparse or too low-quality, not enough information is available to compensate for avoidable algorithm inefficiencies. The brute force approach is likely to fail even when data rates are large. In tracking applications, for example, Bayes filtering algorithms will usually be *barely tractable for real-time operation even if they are ideal* (i.e., $N_{ineffic} = 0$ and even if they have $O(N)$ computational efficiency.)[2] This is because N (which typically has the form $N = N_0^d$ where d is the dimensionality of the problem) is *already very large* — which means that the algorithm designer must use all possible means to reduce computations, not increase them. In such situations, brute force means the same thing as *non-real-time operation*. Such difficulties will be even more pronounced in multitarget situations where d will be quite large.

14.1.2 Why Multisource, Multitarget, Multi-Evidence Problems Are Tricky

Given the technical community's increased understanding of the actual complexity of real signature and other kinds of data, it is no longer credible to invoke Bayesian filtering and estimation as a cookbook panacea. Likewise, given the technical community's increased understanding of the actual complexity of multitarget problems, it is not credible to propose yet another ad hoc multitarget tracking approach. One needs systematic and fully probabilistic methodologies for

- Modeling uncertainty in poorly characterized likelihoods
- Modeling ambiguous data and likelihoods for such data
- Constructing multisource likelihoods for ambiguous data
- Efficiently fusing data from all sources (ambiguous or otherwise)
- Constructing provably true (as opposed to heuristic) multisource-multitarget likelihood functions from the underlying sensor models of the sensors
- Constructing provably true (as opposed to heuristic) multitarget Markov densities, from the underlying motion models of the targets, that account for target motion correlations and changes in target number
- Constructing stable, efficient, and provably optimal multitarget state estimators that address the failure of the classical Bayes-optimal estimators — in particular, that simultaneously determine target number, target kinematics, and target identity without resort to operator intervention or optimal report-to-track association.

14.1.3 Finite-Set Statistics (FISST)

One of the major goals of finite-set statistics (FISST) is to address the "Bayesian Iceberg" issues described in Sections 14.1.1 and 14.1.2. FISST deals with imperfectly characterized data and/or measurement models by extending Bayesian approaches in such a way that they are robust with respect to these ambiguities. This *robust-Bayes* methodology is described more fully in Section 14.1.7. FISST deals with the difficulties associated with multisource and/or multitarget problems by *directly* extending engineering-friendly single-sensor, single-target statistical calculus to the multisensor-multitarget realm. This *optimal-Bayes* methodology is described in Sections 14.3 through 14.7. Finally, FISST provides mathematical tools that may help address the formidable computational difficulties associated with multisource-multitarget filtering (whether optimal or robust). Some of these ideas are discussed in Section 14.7.3.

The basic approach is as follows. A suite of known sensors transmits, to a central data fusion site, the observations they collect regarding targets whose number, positions, velocities, identities, threat states, etc. are unknown. Then

1. Reconceptualize the sensor suite as a single sensor (a "global sensor").
2. Reconceptualize the target set as a single target (a "global target") with multitarget state $X = \{\mathbf{x}_1,\ldots,\mathbf{x}_n\}$ (a "global state").

3. Reconceptualize the set $Z = \{\mathbf{z}_1,\ldots,\mathbf{z}_m\}$ of observations, collected by the sensor suite at approximately the same time, as a single measurement (a "global measurement") of the "global target" observed by the "global sensor."
4. Represent statistically ill-characterized ("ambiguous") data as random closed subsets Θ of (multisource) observation space (see Section 14.8). Thus, in general, $Z = \{\mathbf{z}_1,\ldots,\mathbf{z}_m, \Theta_1,\ldots,\Theta_{m'}\}$.
5. Just as single-sensor, single-target data can be modeled using a measurement model $Z = h(\mathbf{x}, \mathbf{W})$, model multitarget multisensor data using a multisensor-multitarget measurement model — a randomly varying finite set $\Sigma = T(X) \cup C(X)$ (See Section 14.3.).
6. Just as single-target motion can be modeled using a motion model $X_{k+1} = \Phi_k(\mathbf{x}_k,\mathbf{V}_k)$, model the motion of multitarget systems using a multitarget motion model — a randomly varying finite set $\Gamma_{k+1} = \Phi_k(X_k,V_k) \cup B_k(X_k)$. (See Section 14.4.)

Given this, we can *mathematically reformulate multisensor, multitarget estimation problems as single-sensor, single-target problems.* The basis of this reformulation is the concept of belief-mass. Belief-mass functions are nonadditive generalizations of probability-mass functions. (Nevertheless, they are not heuristic: they are equivalent to probability-mass functions on certain abstract topological spaces.[1]) That is,

- Just as the probability-mass function $p(S|\mathbf{x}) = \Pr(\mathbf{Z} \in S)$ is used to describe the generation of conventional data, $\mathbf{z}$, use belief-mass functions of the general form $\rho(\Theta|\mathbf{x}) = \Pr(\Theta \subseteq \Sigma|\mathbf{x})$ to describe the generation of ambiguous data Θ. (See Section 14.8.)
- Just as the probability-mass function $p(S|\mathbf{x}) = \Pr(\mathbf{Z} \in S)$ of a single-sensor, single-target measurement model, $\mathbf{Z}$, is used to describe the statistics of ordinary data, use the belief-mass function $\beta(S|X) = \Pr(\Sigma \subseteq S)$ of a multisource-multitarget measurement model Σ to describe the statistics of multisource-multitarget data.
- Just as the probability-mass function $p_{k+1|k}(S|\mathbf{x}_k) = \Pr(\mathbf{X}_k \in S)$ of a single-target motion model $\mathbf{X}_k$ is used to describe the statistics of single-target motion, use the belief-mass function $\beta_{k+1|k}(S|X_k) = \Pr(\Gamma_{k+1} \subseteq S)$ of a multitarget motion model Γ_{k+1} to describe multitarget motion.

The FISST multisensor-multitarget differential and integral calculus is what transforms these mathematical abstractions into a form that can be used in practice.

- Just as the likelihood function, $f(\mathbf{z}|\mathbf{x})$, can be derived from $p(S|\mathbf{x})$ via differentiation, so the true multitarget likelihood function $f(Z|X)$ can be derived from $\beta(S|X)$ using a generalized differentiation operator called the *set derivative.* (See Section 14.6.1.)
- Just as the Markov transition density $f_{k+1|k}(\mathbf{x}_{k+1}|\mathbf{x}_k)$ can be derived from $p_{k+1|k}(S|\mathbf{x}_k)$ via differentiation, so the true multitarget Markov transition density index can be derived from $\beta_{k+1|k}(S|X_k)$ via set differentiation.
- Just as $f(\mathbf{z}|\mathbf{x})$ and $p(S|\mathbf{x})$ are related by $p(S|\mathbf{x}) = \int_S f(\mathbf{z}|\mathbf{x})d\mathbf{x}$, so $f(Z|X)$ and $\beta(Z|X)$ are related by $\beta(Z|X) = \int_S f(Z|X)\delta Z$ where the integral is now a multisource-multitarget *set integral.*

Accordingly, let $Z^{(k)} = \{Z_1,\ldots,Z_k\}$ be a time sequence of multisource-multitarget observations. This enables true *multitarget posterior distributions* $f_{k|k}(X|Z^{(k)})$ to be created from the true multisource, multitarget likelihood (see Section 14.6.4):

$$f_{k|k}\left(\varnothing\middle|Z^{(k)}\right) = \text{posterior likelihood that no targets are present}$$

$$f_{k|k}\left(\left\{\mathbf{x}_1,\ldots,\mathbf{x}_n\right\}\middle|Z^{(k)}\right) = \text{posterior likelihood of } n \text{ targets with states } \mathbf{x}_1,\ldots,\mathbf{x}_n$$

From these distributions, simultaneous, provably optimal estimates of target number, kinematics, and identity can be computed without resorting to the optimal report-to-track assignment characteristic of

TABLE 14.1 Mathematical Parallels between Single-Sensor, Single-Target Statistics and Multisensor, Multitarget Statistics

Random Vector, Z	Finite Random Set, Σ
Sensor, ⊙	Global sensor, ⊙*
Target, ⊗	Global target, ⊗*
Observation, $\mathbf{z}$	Observation-set, Z
Parameter, $\mathbf{x}$	Parameter-set, X
Sensor model, $\mathbf{z} = h(\mathbf{x}, \mathbf{W})$	Multitarget sensor model, $Z = T(X) \cup C(X)$
Motion model, $\mathbf{x}_{k+1} = \mathbf{\Phi}_k(\mathbf{x}_k, \mathbf{V}_k)$	Multitarget, multi-motion, $X_{k+1} = \mathbf{\Phi}_k(X_k, \mathbf{V}_k) \cup B_k(X_k)$
Differentiation, $\frac{dp}{dz}$	Set differentiation, $\frac{\delta\beta}{\delta Z}$
Integration, $\int_s f(\mathbf{z}\|\mathbf{x})d\mathbf{z}$	Set integration, $\int_s f(Z\|X)\delta Z$
Probability-mass function, $p(S\|\mathbf{x})$	Belief-mass function, $\beta(S\|X)$
Likelihood function, $f(\mathbf{z}\|\mathbf{x})$	Multitarget likelihood function, $f(Z\|X)$
Posterior density, $f_{k\|k}(\mathbf{x}\|Z^k)$	Multitarget posterior, $f_{k\|k}(X\|Z^{(k)})$
Markov densities, $f_{k+1\|k}(\mathbf{x}_{k+1}\|\mathbf{x}_k)$	Multitarget Markov densities, $f_{k+1\|k}(X_{k+1}\|X_k)$

multihypothesis approaches (see Section 14.6.7). Finally, these fundamentals enable both optimal-Bayes and robust-Bayes multisensor-multitarget data fusion, detection, tracking, and identification (see Sections 14.7.1 and 14.8).

Table 14.1 summarizes the direct mathematical parallels between the world of single-sensor, single-target statistics and the world of multisensor, multitarget statistics. This parallelism is so close that general statistical methodologies can, with a bit of prudence, be directly translated from the single-sensor, single-target case to the multisensor-multitarget case. That is, the table can be thought of as a dictionary that establishes a direct correspondence between the words and grammar in the random-vector language and cognate words and grammar of the random-set language. Consequently, any "sentence" (any concept or algorithm) phrased in the random-vector language can, in principle, be directly "translated" into a corresponding sentence (corresponding concept or algorithm) in the random-set language. This process can be encapsulated as a general methodology for attacking multisource-multitarget data fusion problems:

> Almost-Parallel Worlds Principle (APWOP) — Nearly any concept or algorithm phrased in random-vector language can, in principle, be directly translated into a corresponding concept or algorithm in the random-set language.[4,5,10]

The phrase "almost-parallel" refers to the fact that the correspondence between dictionaries is not precisely one-to-one. For example, vectors can be added and subtracted, whereas finite sets cannot. Nevertheless, the parallelism is complete enough that, provided some care is taken, 100 years of accumulated knowledge about single-sensor, single-target statistics can be directly brought to bear on multisensor-multitarget problems.

The following simple example has been used to illustrate the APWOP since 1994. The performance of a multitarget data fusion algorithm can be measured by constructing information-based measures of effectiveness, as shown in the following multitarget generalization of the Kullback-Leibler cross-entropy.[1,4,10]

$$\begin{array}{ccc} \text{Single-sensor, single-target} & & \text{Multisensor, multitarget} \\ K(f,g) \triangleq \int f(\mathbf{x}) \log\left(\dfrac{f(\mathbf{x})}{g(\mathbf{x})}\right) d\mathbf{x} & \rightarrow & K(f,g) \triangleq \int f(X) \log\left(\dfrac{f(X)}{g(X)}\right) \delta X \end{array} \tag{14.5}$$

Here the APWOP replaces conventional statistical concepts with their FISST multisensor, mulitarget counterparts. The ordinary densities f,g and the ordinary integral on the left are replaced by the multitarget densities f,g and the set integral (Section 14.5.3) on the right.

14.1.4 Why Random Sets?

Random set theory was systematically formulated by Mathéron in the mid-1970s.[33] Its centrality as a unifying foundation for expert systems theory and ill-characterized evidence has become increasingly apparent since the mid-1980s.[34-38] Its centrality as a unifying foundation for data fusion applications has been promoted since the late-1970s by I.R. Goodman. The basic relationships between random set theory and the Dempster-Shafer theory of evidence were established by Shafer,[39] Nguyen,[40] and others.[41] The basic relationships between random set theory and fuzzy logic can be attributed to Goodman,[34,42] Orlov,[43] and Hohle.[44] Mahler developed relationships between random set theory and rule-based evidence.[45,46] Mori, Chong, et al. first proposed random set theory as a potential foundation for multitarget detection, tracking, and identification (although within a multihypothesis framework).[1,47] (Since 1995, Mori has published a number of very interesting papers based on random set ideas.[48-51])

FISST, whose fundamental ideas were codified in 1993 and 1994, builds upon this existing body of research by showing that random set theory provides a unified framework for both expert systems theory and multisensor-multitarget fusion detection, tracking, identification, sensor management, and performance estimation. FISST is unique in that it provides, under a single probabilistic paradigm, a unified and relatively simple and familiar statistical calculus for addressing all of the "Bayesian Iceberg" problems described earlier.

14.2 Basic Statistics for Tracking and Identification

This section summarizes those aspects of conventional statistics that are most pertinent to tracking and target identification. The foundation of applied tracking and identification — the recursive Bayesian nonlinear filtering equations — are described in Section 14.2.1. The procedure for constructing provably true sensor likelihood functions from sensor models, and provably true Markov transition densities from target motion models, is described in Sections 14.2.2 and 14.2.3, respectively. Bayes-optimal state estimation is reviewed in Section 14.2.4. In all four sections, data "vectors" have the form $y = (y_1,\ldots, y_n, w_1,\ldots, w_n)$ where $y_1,\ldots, y_n$ are continuous variables and $w_1,\ldots,w_n$ are discrete variables.[1] Integrals of functions of such variables involve both summations and continuous integrals.

14.2.1 Bayes Recursive Filtering

Most signal processing engineers are familiar with the Kalman filtering equations. Less well known is the fact that the Kalman filter is a special case of the Bayesian discrete-time recursive nonlinear filter.[26,27,52] This more general filter is nothing more than Equations 14.6, 14.7, and 14.8 applied recursively:

$$f_{k+1|k}\left(\mathbf{x}_{k+1} \middle| Z^k\right) = \int f_{k+1|k}\left(\mathbf{x}_{k+1} \middle| \mathbf{x}_k\right) f_{k|k}\left(\mathbf{x}_k \middle| Z^k\right) d\mathbf{x}_k \tag{14.6}$$

$$f_{k+1|k+1}\left(\mathbf{x}_{k+1} \middle| Z^{k+1}\right) = \frac{f\left(\mathbf{z}_{k+1} \middle| \mathbf{x}_{k+1}\right) f_{k+1|k}\left(\mathbf{x}_{k+1} \middle| Z^k\right)}{f\left(\mathbf{z}_{k+1} \middle| Z^k\right)} \tag{14.7}$$

$$\hat{\mathbf{x}}_{k|k}^{MAP} = \arg\max_{\mathbf{x}} f_{k|k}\left(\mathbf{x}|Z^k\right) \qquad \hat{\mathbf{x}}_{k|k}^{EAP} = \int \mathbf{x}\, f_{k|k}\left(\mathbf{x}|Z^k\right)d\mathbf{x} \tag{14.8}$$

where

$f(\mathbf{z}_{k+1}|Z^k) = \int f(\mathbf{z}_{k+1}|\mathbf{y}_{k+1}) f_{k+1|k}(\mathbf{y}_{k+1}|Z^k)d\mathbf{y}_{k+1}$ is the Bayes normalization constant

$f(\mathbf{z}|\mathbf{x})$ is the sensor likelihood function

$f_{k+1|k}(\mathbf{x}_{k+1}|Z^k)$ is the target Markov transition density

$f_{k|k}(\mathbf{x}_k|Z^k)$ is the posterior distribution conditioned on the data-stream $Z^k = \{\mathbf{z}_1,\ldots,\mathbf{z}_k\}$

$f_{k+1|k}(\mathbf{x}_{k+1}|Z^k)$ is the time-prediction of the posterior $f_{k|k}(\mathbf{x}_k|Z^k)$ to time-step $k+1$

The practical success of Equations 14.6 and 14.7 relies on the ability to construct effectively the likelihood function, $f(\mathbf{z}|\mathbf{x})$, and the Markov transition density, $f_{k+1|k}(\mathbf{x}_{k+1}|\mathbf{x}_k)$. Although likelihood functions sometimes are constructed via direct statistical analysis of data, more typically they are constructed from sensor measurement models. Markov densities typically are constructed from target motion models. In either case, differential and integral calculus must be used, as shown in the following two sections.

14.2.2 Constructing Likelihood Functions from Sensor Models

Suppose that a target with randomly varying state $\mathbf{X}$ is interrogated by a sensor that generates observations of the form $\mathbf{Z} \triangleq \mathbf{Z}|_{\mathbf{X}=\mathbf{x}} \triangleq \mathbf{h}(\mathbf{x}) + \mathbf{W}$ (where $\mathbf{W}$ is a zero-mean random noise vector with density $f_{\mathbf{w}}(\mathbf{w})$) but does not generate missed detections or false alarms. The statistical behavior of $\mathbf{Z}$ is characterized by its *likelihood function*, $f(\mathbf{z}|\mathbf{x}) \triangleq f_{\mathbf{Z}|\mathbf{X}}(\mathbf{z}|\mathbf{x})$, which describes the likelihood that the sensor will collect measurement $\mathbf{z}$ given that the target has state $\mathbf{x}$. How is this likelihood function computed? Begin with the *probability mass function* of the sensor model: $p(S|\mathbf{x}) \triangleq p(\mathbf{z}|\mathbf{x}) = pr(\mathbf{Z} \in S)$. This is the total probability that the random observation $\mathbf{Z}$ will be found in any given region S if the target has state $\mathbf{x}$. The total probability mass, $p(S|\mathbf{x})$, in a region S is the sum of all of the likelihoods in that region: $p(S|\mathbf{x}) = \int_S f(\mathbf{y}|\mathbf{x})d\mathbf{y}$. So,

$$p\left(E_{\mathbf{z}}|\mathbf{x}\right) = \int_{E_{\mathbf{z}}} f\left(\mathbf{y}|\mathbf{x}\right)d\mathbf{y} \cong f\left(\mathbf{z}|\mathbf{x}\right)\cdot\lambda\left(E_{\mathbf{z}}\right)$$

when $E_{\mathbf{z}}$ is some very small region surrounding the point $\mathbf{z}$ with (hyper) volume $V = \lambda(E_{\mathbf{z}})$. (For example, $E_{\mathbf{z}} = B_{\varepsilon,\mathbf{z}}$ is a hyperball of radius ε centered at $\mathbf{z}$.) So,

$$\frac{p\left(E_{\mathbf{z}}|\mathbf{x}\right)}{\lambda\left(E_{\mathbf{z}}\right)} \cong f\left(\mathbf{z}|\mathbf{x}\right)$$

where the smaller the value of $\lambda(E_{\mathbf{z}})$, the more accurate the approximation. Stated differently, the likelihood function, $f(\mathbf{z}|\mathbf{x})$, can be constructed from the probability measure, $p(S|\mathbf{x})$, via the limiting process

$$f\left(\mathbf{z}|\mathbf{x}\right) = \frac{\delta p}{\delta \mathbf{z}} \triangleq \lim_{\lambda(E_{\mathbf{z}})\searrow 0} \frac{p\left(E_{\mathbf{z}}|\mathbf{x}\right)}{\lambda\left(E_{\mathbf{z}}\right)} \tag{14.9}$$

The resulting equations

$$p\left(S|\mathbf{x}\right) = \int_S \frac{\delta p}{\delta \mathbf{z}}d\mathbf{z} \qquad \frac{\delta}{\delta \mathbf{z}}\int_S f\left(\mathbf{y}|\mathbf{x}\right)d\mathbf{y} = f\left(\mathbf{z}|\mathbf{x}\right) \tag{14.10}$$

are the relationships that show that $f(\mathbf{z}|\mathbf{x}) \triangleq \delta p|\delta \mathbf{z}$ is the provably true likelihood function (i.e., the density function that faithfully describes the measurement model $\mathbf{Z} = \mathbf{h}(\mathbf{x}, \mathbf{W})$). For this particular problem, the "true" likelihood is, therefore:

$$f\left(\mathbf{z}\middle|\mathbf{x}\right) = \lim_{\varepsilon \searrow 0} \frac{\Pr\left(\mathbf{Z} \in B_{\varepsilon,\mathbf{z}}\right)}{\lambda\left(B_{\varepsilon,\mathbf{z}}\right)} = \lim_{\varepsilon \searrow 0} \frac{p_{\mathbf{w}}\left(B_{\varepsilon,\mathbf{z}-h(\mathbf{x})}\right)}{\lambda\left(B_{\varepsilon,\mathbf{z}-h(\mathbf{x})}\right)} = f\mathbf{w}\left(\mathbf{z} - \mathbf{h}\left(\mathbf{x}\right)\right)$$

14.2.3 Constructing Markov Densities from Motion Models

Suppose that, between the k^{th} and $(k+1)^{\text{st}}$ measurement collection times, the motion of the target is best modeled by an equation of the form $\mathbf{X}_{k+1} = \Phi_k(\mathbf{x}_k) + \mathbf{V}_k$ where $\mathbf{V}_k$ is a zero-mean random vector with density $f\mathbf{v}_k(\mathbf{v})$. That is, if the target had state $\mathbf{x}_k$ at time-step k, then it will have state $\Phi_k(\mathbf{x}_k)$ at time-step $k+1$ — except possible error in this belief is accounted for by appending the random variation $\mathbf{V}_k$. How would $f_{k+1|k}(\mathbf{X}_{k+1}|\mathbf{X}_k)$ be constructed? This situation parallels that of Section 14.2.2. The probability mass function $p_{k+1}(S|\mathbf{x}_k) \triangleq \Pr(\mathbf{X}_{k+1} \in S)$ is the total probability that the target will be found in region S at time-step $k+1$, given that it had state $\mathbf{x}_k$ at time-step k. So,

$$f_{k+1|k}\left(\mathbf{x}_{k+1}\middle|\mathbf{x}_k\right) = \lim_{\varepsilon \searrow 0} \frac{p_{k+1}\left(B_{\varepsilon,\mathbf{x}_{k+1}}\middle|\mathbf{x}_k\right)}{\lambda\left(B_{\varepsilon,\mathbf{x}_{k+1}}\right)} = f\mathbf{V}_k\left(x_{k+1} - \Phi_k\left(\mathbf{x}_k\right)\right)$$

is the true Markov density associated with the motion model $\mathbf{X}_{k+1} = \Phi_k(\mathbf{X}_k) + \mathbf{V}_k$ more generally, the equations

$$p_{k+1|k}\left(S\middle|\mathbf{x}_k\right) = \int_S \frac{\delta p_{k+1|k}}{\delta \mathbf{x}_{k+1}} d\mathbf{x}_{k+1}, \quad \frac{\delta}{\delta \mathbf{x}_{k+1}} \int_S f_{k+1|k}\left(\mathbf{x}_{k+1}\middle|\mathbf{x}_k\right) d\mathbf{x}_{k+1} = f_{k+1|k}\left(\mathbf{x}_{k+1}\middle|\mathbf{x}_k\right)$$

are the relationships that show that $f_{k+1|k}(\mathbf{x}_{k+1}|\mathbf{x}_k) \triangleq \delta p_{k+1|k}/\delta \mathbf{x}_{k+1}$ is the provably true Markov density — i.e., the density function that faithfully describes the motion model $\mathbf{X}_{k+1} = \Phi_k(\mathbf{x}_k, \mathbf{V}_k)$.

14.2.4 Optimal State Estimators

An estimator of the state $\mathbf{x}$ is any family $\hat{\mathbf{x}}(\mathbf{z}_1,\ldots,\mathbf{z}_m)$ of state-valued functions of the (static) measurements $\mathbf{z}_1,\ldots,\mathbf{z}_m$. "Good" state estimators $\hat{\mathbf{x}}$ should be Bayes-optimal in the sense that, in comparison to all other possible estimators, they minimize the Bayes risk

$$R_C\left(\hat{\mathbf{x}}, m\right) = \int C\left(\mathbf{x}, \hat{\mathbf{x}}\left(\mathbf{z}_1,\ldots,\mathbf{z}_m\right)\right) f\left(\mathbf{z}_1,\ldots,\mathbf{z}_m\middle|\mathbf{x}\right) f\left(\mathbf{x}\right) d\mathbf{x}\, d\mathbf{z}_1 \cdots d\mathbf{z}_m$$

for some specified cost (i.e., objective) function $C(\mathbf{x}, \mathbf{y})$ defined on states $\mathbf{x}$, $\mathbf{y}$.[53] Secondly, they should be statistically consistent in the sense that $\hat{\mathbf{x}}(\mathbf{z}_1,\ldots,\mathbf{z}_m)$ converges to the actual target state as $m \to \infty$. Other properties (e.g., asymptotically unbiased, rapidly convergent, stably convergent, etc.) are desirable as well. The most common "good" Bayes state estimators are the maximum *a posteriori* (MAP) and expected *a posteriori* (EAP) estimators described earlier.

14.3 Multitarget Sensor Models

In the single-target case, probabilistic approaches to tracking and identification (and Bayesian approaches in particular) depend on the ability to construct sensor models and likelihood functions that model the data-generation process with enough fidelity to ensure optimal performance. As argued in Section 14.1.1, multitarget tracking and identification have suffered from the lack of a statistical calculus of the kind that is readily available in the single-target case. The purpose of this section is to show how to construct FISST multitarget measurement models. These models, and the FISST statistical calculus to be introduced in Section 14.5, will be integrated in Section 14.6.1 to construct true multitarget likelihood functions.

The following sections illustrate the process of constructing multitarget measurement models for the following successively more realistic situations: (1) multitarget measurement models with no missed detections and no false alarms/clutter; (2) multitarget measurement models with missed detections; (3) multitarget measurement models with missed detections and false alarms or clutter; and (4) multitarget measurement models for the multiple-sensor case. The problem of constructing single-target and multitarget measurement models for data that is ambiguous is discussed in Section 14.8.

14.3.1 Case I: No Missed Detections, No False Alarms

Suppose that two targets with states $\mathbf{x}_1$ and $\mathbf{x}_2$ are interrogated by a single sensor that generates observations of the form $\mathbf{Z} = h(\mathbf{x}) + \mathbf{W}_1$, where $\mathbf{W}_1$ is a random noise vector with density $f_{\mathbf{w}}(\mathbf{w})$. Assume also that there are no missed detections or false alarms, and that observations within a scan are independent. Then the *multitarget measurement* is the *randomly varying two-element observation set.*

$$\Sigma = \left\{\mathbf{Z}_1, \mathbf{Z}_2\right\} = \left\{h\left(\mathbf{x}_1\right) + \mathbf{W}_1, h\left(\mathbf{x}_2\right) + \mathbf{W}_2\right\} \tag{14.11}$$

where $\mathbf{W}_1$, $\mathbf{W}_2$ are independent random vectors with density $f_{\mathbf{w}}(\mathbf{w})$. We assume that individual targets produce unique observations only for the sake of clarity. Clearly, we could just as easily produce models for other kinds of sensors — for example, sensors that detect only superpositions of the signals produced by multiple targets. One such measurement model is

$$\Sigma = \left\{\mathbf{Z}\right\} = \left\{h_1\left(\mathbf{x}_1\right) + h_2\left(x_2\right) + \mathbf{W}\right\}$$

14.3.2 Case II: Missed Detections

Suppose that the sensor of Section 14.3.1 has a probability of detection $\mathrm{p_D} < 1$. In this case, observations can have not only the form $Z = \{\mathbf{z}_1, \mathbf{z}_2\}$, but also $Z = \{\mathbf{z}\}$ or $Z = \emptyset$ (missed detection). The more complex observation model $\Sigma = T_1 \cup T_2$ is needed, which has T_1, T_2 observation sets with the following properties: (a) $T_i = \emptyset$ (i.e., missed detection) with probability $1 - p_D$ and (b) T_i is nonempty with probability $\mathrm{p_D}$, in which case, $T_i = \{Z_i\}$. If the sensor has a specific field of view, then $p_D = p_D\,(\mathbf{x}, \mathbf{y})$ will be a function of both the target state $\mathbf{x}$ and the state $\mathbf{y}$ of the sensor.

14.3.3 Case III: Missed Detection and False Alarms

Suppose the sensor of Section 14.3.2 has probability of false alarm $p_{FA} \neq 0$. Then we need an observation model of the form

$$\Sigma = \overset{\text{target}}{\overleftarrow{T}} \cup \overset{\text{clutter}}{\overleftarrow{C}} = T_1 \cup T_2 \cup C$$

where C models false alarms and/or (possibly state-dependent) point clutter. As a simple example, C could have the form $C = C_1 \cup \ldots \cup C_m$, where each C_j is a clutter generator — meaning that there is a probability, p_{FA}, that C_j will be nonempty (i.e., generator of a clutter-observation) — in which case $C = \{C_j\}$ where C_i is some random noise vector with density $f_{C_i}(\mathbf{z})$. (Note: $\Sigma = T_1 \cup C$ models the single-target case.)

14.3.4 Case IV: Multiple Sensors

In this case, observations will have the form $\mathbf{z}^{[s]} \triangleq (\mathbf{z}, s)$ where the integer tag s identifies which sensor originated the measurement. A two-sensor multitarget measurement will have the form $\Sigma = \Sigma^{[1]} \cup \Sigma^{[2]}$ where $\Sigma^{[s]}$ for $s = 1, 2$ is the random multitarget measurement-set collected by the sensor with tag s and can have any of the forms previously described.

14.4 Multitarget Motion Models

This section shows how to construct multitarget motion models in relation to the construction of single-target motion models. These models, combined with the FISST calculus (Section 14.5), enables the construction of true multitarget Markov densities (Section 14.6.2). In the single-target case, the construction of Markov densities from motion models strongly parallels the construction of likelihood functions from sensor measurement models. In like fashion, the construction of multitarget Markov densities strongly resembles the construction of multisensor-multitarget likelihood functions.

This section illustrates the process of constructing multitarget motion models by considering the following increasingly more realistic situations: (1) multitarget motion models assuming that target number does not change; (2) multitarget motion models assuming that target number can decrease; and (3) multitarget motion models assuming that target number can decrease or increase.

14.4.1 Case I: Target Number Does Not Change

Assume that the states of individual targets have the form $\mathbf{x} = (\mathbf{y}, c)$ where $\mathbf{y}$ is the kinematic state and c is the target type. Assume that each target type has an associated motion model $\mathbf{Y}_{c,k+1} = \mathbf{\Phi}_{c,k}(\mathbf{y}_k) + \mathbf{W}_{c,k}$. Define

$$\mathbf{X}_{k+1,\mathbf{x}} = \Phi_{k+1}\left(\mathbf{x}, \mathbf{W}_k\right) \triangleq \left(\Phi_{c,k}\left(\mathbf{y}\right) + \mathbf{W}_{c,k}, c\right)$$

where $\mathbf{W}_k$ denotes the family of random vectors $\mathbf{W}_{c,k}$.

To model a multitarget system in which two targets never enter or leave the scenario, the obvious multitarget extension of the single-target motion model would be $\Gamma_{k+1} = \Phi_k(X_k, W_k)$, where Γ_{k+1} is the randomly varying parameter set at time step $k + 1$. That is, for the cases $X = \emptyset$, $X = \{\mathbf{x}\}$, *or* $X = \{\mathbf{x}_1, \mathbf{x}_2\}$, respectively, the multitarget state transitions are:

$$\Gamma_{k+1} = \emptyset$$

$$\Gamma_{k+1} \triangleq \left\{\mathbf{X}_{k+1,\mathbf{x}}\right\} = \left\{\Phi_k\left(\mathbf{x}, \mathbf{W}_k\right)\right\}$$

$$\Gamma_{k+1} \triangleq \left\{\mathbf{X}_{k+1,\mathbf{x}_1}, \mathbf{X}_{k+1,\mathbf{x}_2}\right\} = \left\{\Phi_k\left(\mathbf{x}_1, \mathbf{W}_k\right), \Phi_k\left(\mathbf{x}_2, \mathbf{W}_k\right)\right\}$$

14.4.2 Case II: Target Number Can Decrease

Modeling scenarios in which target number can decrease (but not increase) is analogous to modeling multitarget observations with missed detections. Suppose that no more than two targets are possible, but that one or more of them can vanish from the scene. One possible motion model would be $\Gamma_{k+1|k} = \mathbf{\Phi}(X_{k|k}, \mathbf{W}_k)$ where, for the cases $X = \emptyset$, $X = \{\mathbf{x}\}$, *or* $X = \{\mathbf{x}_1, \mathbf{x}_2\}$, respectively,

$$\Gamma_{k+1} = \emptyset, \ \Gamma_{k+1} = T_{k,x}, \ \Gamma_{k+1} = T_{k,x_1} \cup T_{k,x_2}$$

where $T_{k,x}$ is a track-set with the following properties: (a) $T_{k,x} \neq \emptyset$ with probability p_v, in which case $T_{k,x} = \{\mathbf{X}_{k+1,x}\}$, and (b) $T_{k,x} = \emptyset$ (i.e., target disappearance), with probability $1 - p_v$. In other words, if no targets are present in the scene, this will continue to be the case. If, however, there is one target in the scene, then either this target will persist (with probability p_v) or it will vanish (with probability $1 - p_v$). If there are two targets in the scene, then each will either persist or vanish in the same manner. In general, one would model $\Phi_k(\{\mathbf{x}_1,\ldots,\mathbf{x}_n\}) = T_{k,x_1} \cup \ldots \cup Tk,\mathbf{x}_n$.

14.4.3 Case III: Target Number Can Increase and Decrease

Modeling scenarios in which target number can decrease and/or increase is analogous to modeling multitarget observations with missed detections and clutter. In this case, one possible model is

$$\Phi_k\left(\left\{\mathbf{x}_1,\ldots,\mathbf{x}_n\right\}\right) = T_{k,\mathbf{x}_1} \cup \ldots \cup T_{k,\mathbf{x}_n} \cup B_k$$

where B_k is the set of birth targets (i.e, targets that have entered the scene).

14.5 The FISST Multisource-Multitarget Calculus

This section introduces the mathematical core of FISST — the FISST multitarget integral and differential calculus. That is, it shows that the belief-mass function $\beta(S)$ of a multitarget sensor or motion model plays the same role in multisensor-multitarget statistics that the probability-mass function $p(S)$ plays in single-target statistics. The integral $\int_s f(\mathbf{z})d\mathbf{z}$ and derivative $dp/d\mathbf{z}$ — which can be computed using elementary calculus — are the mathematical basis of conventional single-sensor, single-target statistics. We will show that the basis of multisensor-multitarget statistics is a multitarget integral $\int_s f(\mathbf{z})\delta\mathbf{z}$ and a multitarget derivative $\delta\beta/\delta Z$ that can also be computed using "turn-the-crank" calculus rules. In particular we will show that, using the FISST calculus,

- True multisensor-multitarget likelihood functions can be constructed from the measurement models of the individual sensors, and
- True multitarget Markov transition densities can be constructed from the motion models of the individual targets.

Section 14.5.1 defines the belief-mass function of a multitarget sensor measurement model and Section 14.5.2 defines the belief-mass function of a multitarget motion model. The FISST multitarget integral and differential calculus is introduced in Section 14.5.3. Section 14.5.4 lists some of the more useful rules for using this calculus.

14.5.1 The Belief-Mass Function of a Sensor Model

Just as the statistical behavior of a random observation vector Z is characterized by its probability mass function $p(S|\mathbf{x}) = \Pr(\mathbf{Z} \in S)$, the statistical behavior of the random observation-set Σ is characterized by its *belief-mass function*:[1]

$$\beta(S|X) \triangleq \beta_{\Sigma|\Gamma}(S|X) \triangleq \Pr(\Sigma \subseteq S)$$

where Γ is the random multitarget state. The belief mass is the total probability that all observations in a sensor (or multisensor) scan will be found in any given region S, if targets have multitarget state X.

For example, if $X = \{\mathbf{x}\}$ and $\Sigma = \{\mathbf{Z}\}$ where $\mathbf{Z}$ is a random vector, then

$$\beta(S|X) = \Pr(\Sigma \subseteq S) = \Pr(Z \in S) = p(S|\mathbf{x})$$

In other words, the belief mass of a *random vector* is equal to its probability mass. On the other hand, for a single-target, missed-detection model $\Sigma = T_1$,

$$\begin{aligned}\beta(S|X) &= \Pr(T_1 \subseteq S) = \Pr(T_1 = \emptyset) + \Pr(T_1 \neq \emptyset, Z \in S) \\ &= \Pr(T_1 = \emptyset) + \Pr(T_1 \neq \emptyset)\Pr(Z \in S) = 1 - p_D + p_D\, p_Z(S|\mathbf{x})\end{aligned} \tag{14.12}$$

and for the two-target missed-detection model $\Sigma = \{T_1 \cup T_2\}$ (Section 14.3.2),

$$\beta(S|X) = \Pr(T_1 \subseteq S)\Pr(T_2 \subseteq S) = \left[1 - p_D + p_D p(S|\mathbf{x}_1)\right]\left[1 - p_D + p_D\, p(S|\mathbf{x}_2)\right]$$

where $p\ (S|\mathbf{x}) \triangleq Pr\ (T_i \subseteq S | T_i \neq \emptyset)$ and $X = \{\mathbf{x}_1, \mathbf{x}_2\}$. Setting $p_D = 1$ yields

$$\beta(S|X) = p(S|\mathbf{x}_1)\, p(S|\mathbf{x}_2) \tag{14.13}$$

which is the belief-mass function for the model $\Sigma = \{\mathbf{Z}_1, \mathbf{Z}_2\}$ of Equation 14.11. Suppose next that two sensors with identifying tags $s = 1,2$ collect observation-sets $\Sigma = \Sigma^{[1]} \cup \Sigma^{[2]}$. The corresponding belief-mass function has the form $\beta_\Sigma\ (S^{[1]} \cup S^{[2]}|X) = \Pr(\Sigma^{[1]} \subseteq S^{[1]}, \Sigma^{[2]} \subseteq S^{[2]})$ where $S^{[1]}$, $S^{[2]}$ are (measurable) subsets of the measurement spaces of the respective sensors. If the two sensors are independent then the belief-mass function has the form

$$\beta_\Sigma\left(S^{[1]} \cup S^{[2]} | X\right) = \beta_{\Sigma^{[1]}}\left(S^{[1]} | X\right)\beta_{\Sigma^{[2]}}\left(S^{[2]} | X\right) \tag{14.14}$$

14.5.2 The Belief-Mass Function of a Motion Model

In single-target problems, the statistics of a motion model $\mathbf{X}_{k+1} = \mathbf{\Phi}_k(X_k, \mathbf{W}_k)$ are described by the probability-mass function $p_{\mathbf{X}k+1}\ (S|\mathbf{x}_k) = \Pr(\mathbf{X}_{k+1} \in S)$, which is the probability that the target-state will be found in the region S if it previously had state $\mathbf{x}_k$. Similarly, suppose that $\Gamma_{k+1} = \mathbf{\Phi}_k\ (X_k, \mathbf{W}_k)$ is a multitarget motion model (Section 14.4). The statistics of the finitely varying random state-set Γ_{k+1} can be described by its belief-mass function:

$$\beta_{\Gamma_{k+1}}\left(S|X_k\right) \triangleq \Pr\left(\Gamma_{k+1} \subseteq S\right)$$

This is the total probability of finding all targets in region S at time-step $k+1$ if, in time-step k, they had multitarget state $X_k = \{\mathbf{x}_{k,1}, \ldots, \mathbf{x}_{k,n(k)}\}$.

For example, the belief-mass function for the multitarget motion model of Section 14.4.1 is $\beta_{k+1|k}(S\,|\,\mathbf{x}_k, \mathbf{y}_k) = \Pr(\mathbf{\Phi}_k(\mathbf{x}_k) + V_k \in S)\Pr(\mathbf{\Phi}_k(\mathbf{y}_k) + \mathbf{W}_k \in S) = p\mathbf{W}_k(S\,|\,\mathbf{x}_k)\,p\mathbf{W}_k(S\,|\,\mathbf{y}_k)$. This is entirely analogous to Equation 14.13, the belief-mass function for the multitarget measurement model of Section 14.3.1.

14.5.3 The Set Integral and Set Derivative

Equation 14.9 showed that single-target likelihood can be computed from probability-mass functions using an operator $\delta/\delta\mathbf{z}$ inverse to the integral. Multisensor-multitarget likelihoods can be constructed from belief-mass functions in a similar manner.