1

INTRODUCTION

1.1 INTRODUCTION

In this chapter we motivate the philosophy of Bayesian processing from a probabilistic perspective. We show the coupling between model-based signal processing (*MBSP*) incorporating the *a priori* knowledge of the underlying processes and the Bayesian framework for specifying the distribution required to develop the processors. The idea of the sampling approach evolving from Monte Carlo (*MC*) and Markov chain Monte Carlo (*MCMC*) methods is introduced as a powerful methodology for simulating the behavior of complex dynamic processes and extracting the embedded information required. The main idea is to present the proper perspective for the subsequent chapters and construct a solid foundation for solving signal processing problems.

1.2 BAYESIAN SIGNAL PROCESSING

The development of Bayesian signal processing has evolved in a manner proportional to the evolution of high performance/high throughput computers. This evolution has led from theoretically appealing methods to pragmatic implementations capable of providing reasonable solutions for nonlinear and highly multi-modal (multiple distribution peaks) problems. In order to fully comprehend the Bayesian perspective, especially for signal processing applications, we must be able to separate our thinking and in a sense think more abstractly about probability distributions without worrying about how these representations can be "applied" to realistic processing problems. Our motivation is to first present the Bayesian approach from a statistical viewpoint and then couple it to useful signal processing implementations following the well-known model-based approach [1, 2]. Here we show that when we constrain the Bayesian

Bayesian Signal Processing. By James V. Candy

distributions in estimation to Markovian representations using primarily state–space models, we can construct sequential processors capable of "pseudo real-time" operations that are easily be utilized in many physical applications. Bayes' rule provides the foundation of all Bayesian estimation techniques. We show how it can be used to both theoretically develop processing techniques based on a specific distribution (e.g., Poisson, Gaussian, etc.) and then investigate properties of such processors relative to some of the most well-known approaches discussed throughout texts in the field.

Bayesian signal processing is concerned with the estimation of the underlying probability distribution of a random signal in order to perform statistical inferences [3]. These inferences enable the extraction of the signal from noisy uncertain measurement data. For instance, consider the problem of extracting the random variate, say X, from the noisy data, Y. The Bayesian approach is to first *estimate* the underlying conditional probability distribution, $\Pr(X|Y)$, and then perform the associated inferences to extract $\hat{X}$, that is,

$$\hat{\Pr}(X|Y) \Rightarrow \hat{X} = \arg\max_{X} \hat{\Pr}(X|Y)$$

where the caret, $\hat{X}$ denotes an estimate of X. This concept of estimating the underlying distribution and using it to extract a signal estimate provides the foundation of Bayesian signal processing developed in this text.

Let us investigate this idea in more detail. We start with the previous problem of trying to estimate the random parameter, X, from noisy data $Y = y$. Then the associated conditional distribution $\Pr(X|Y = y)$ is called the *posterior distribution* because the estimate is conditioned "after (*post*) the measurements" have been acquired. Estimators based on this *a posteriori* distribution are usually called *Bayesian* because they are constructed from *Bayes' rule*, since $\Pr(X|Y)$ is difficult to obtain directly. That is,

$$\Pr(X|Y) = \frac{\Pr(Y|X) \times \Pr(X)}{\Pr(Y)} \tag{1.1}$$

where $\Pr(X)$ is called the *prior distribution* (before measurement), $\Pr(Y|X)$ is called the *likelihood* (more likely to be true) and $\Pr(Y)$ is called the *evidence* (scales the posterior to assure its integral is unity). Bayesian methods view the sought after parameter as random possessing a "known" *a priori* distribution. As measurements are made, the *prior* is transformed to the *posterior distribution* function adjusting the parameter estimates. Thus, the result of increasing the number of measurements is to improve the *a posteriori* distribution resulting in a sharper peak closer to the true parameter as shown in Fig. 1.1.

When the variates of interest are *dynamic*, then they are functions of time and therefore, $X_t \to X$ and $Y_t \to Y$. Bayes' rule for the joint dynamic distribution is

$$\Pr(X_t|Y_t) = \frac{\Pr(Y_t|X_t) \times \Pr(X_t)}{\Pr(Y_t)} \tag{1.2}$$

In Bayesian theory, the *posterior* defined by $\Pr(X_t|Y_t)$ is decomposed in terms of the *prior* $\Pr(X_t)$, its *likelihood* $\Pr(Y_t|X_t)$ and the *evidence* or normalizing factor,

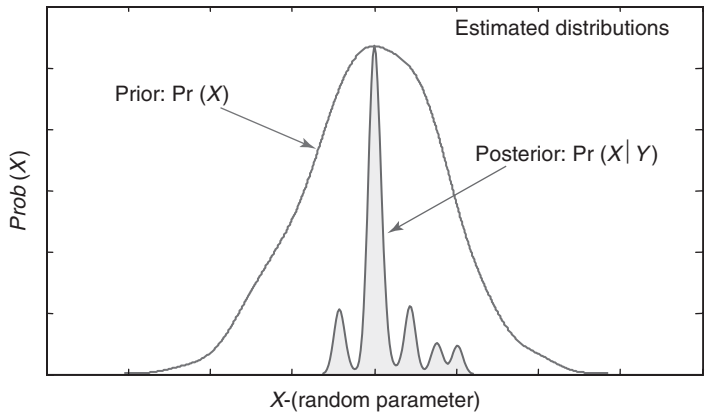

FIGURE 1.1 Bayesian estimation of the random variate X transforming the prior, $\Pr(X)$ to the posterior, $\Pr(X|Y)$ using Bayes' rule.

$\Pr(Y_t)$. Bayesian signal processing in this dynamic case follows the identical path, that is,

$$\hat{\Pr}(X_t|Y_t) \Rightarrow \hat{X}_t = \arg\max_{X_t} \hat{\Pr}(X_t|Y_t)$$

So we begin to see the versatility of the Bayesian approach to random signal processing. Once the posterior distribution is determined, then all statistical inferences or estimates are made. For instance, suppose we would like to obtain the prediction distribution. Then it can be obtained as

$$\Pr(X_{t+1}|Y_t) = \int \Pr(X_{t+1}|X_t, Y_t) \times \Pr(X_t|Y_t)\, dX_t$$

and a point estimate might be the conditional mean of this distribution, that is,

$$E\{X_{t+1}|Y_t\} = \int X_{t+1} \Pr(X_{t+1}|Y_t)\, dX_{t+1}$$

This relation shows how information that can be estimated from the extracted distribution is applied in the estimation context by performing statistical inferences.

Again, even though the Bayesian signal processing concept is simple, conceptually, the real problem to be addressed is that of evaluating the integrals which is very difficult because they are only analytically tractable for a small class of priors and likelihood distributions. The large dimensionality of the integrals cause numerical integration techniques to break down, which leads to the approximations we discuss subsequently for stabilization. Next let us consider the various approaches taken to solve the probability distribution estimation problems using non-parametric or parametric representations. This will eventually lead to the model-based approach [4].

1.3 SIMULATION-BASED APPROACH TO BAYESIAN PROCESSING

The *simulation-based approach* to Bayesian processing is founded on Monte Carlo (*MC*) methods that are stochastic computational techniques capable of efficiently simulating highly complex systems. Historically motivated by games of chance and encouraged by the development of the first electronic computer (ENIAC), the *MC* approach was conceived by Ulam (1945), developed by Ulam, Metropolis and von Neumann (1947) and coined by Metropolis (1949) [5–9]. The method evolved in the mid-1940s during the Manhattan project by scientists investigating calculations for atomic weapon designs [10]. It evolved further from such areas as computational physics, biology, chemistry, mathematics, engineering, materials and finance to name a few. *Monte Carlo* methods offer an alternative approach to solving classical numerical integration and optimization problems. Inherently, as the dimensionality of the problem increases classical methods are prone to failure while *MC* methods tend to increase their efficiency by reducing the error—an extremely attractive property. For example, in the case of classical grid-based numerical integration or optimization problems as the number of grid points increase along with the number of problem defining vector components, there is an accompanying exponential increase in computational time [10–15]. The stochastic *MC* approach of selecting random samples and averaging over a large number of points actually reduces the computational error by the Law of Large Numbers irrespective of the problem dimensionality. It utilizes Markov chain theory as its underlying foundation establishing the concept that through random sampling the resulting "empirical" distribution converges to the desired posterior called the stationary or invariant distribution of the chain. Markov chain Monte Carlo (*MCMC*) techniques are based on sampling from probability distributions based on a Markov chain, which is a stochastic system governed by a transition probability, having the desired posterior distribution as its invariant distribution. Under certain assumptions the chain converges to the desired posterior through proper random sampling as the number of samples become large—a crucial property (see [10] for details). Thus, the Monte Carlo approach has evolved over a long time period and is well understood by scientists and statisticians, but it must evolve even further to be useful for signal processors to become an effective tool in their problem solving repertoire.

Perhaps the best way to visualize the *MC* methods follows directly from the example of Frenkel [11]. Suppose that a reasonable estimate of the depth of the Mississippi river is required. Using numerical quadrature techniques the integrand value is measured at prespecified grid points. We also note that the grid points may not be in regions of interest and, in fact, the integrand may vanish as shown in Fig. 1.2. On the other hand, the surveyor is in the Mississippi and performing a (random) walk within the river measuring the depth of the river directly. In this sampling approach measurements are accepted as long as the surveyor is in the river and rejected if outside. Here the "average" depth is simply the sample average of the measurements much the same as a sampling technique might perform. So we see that a refinement of the brute force integration approach is to use random points or samples that "most likely" come from regions of high contribution to the integral rather than from low regions.

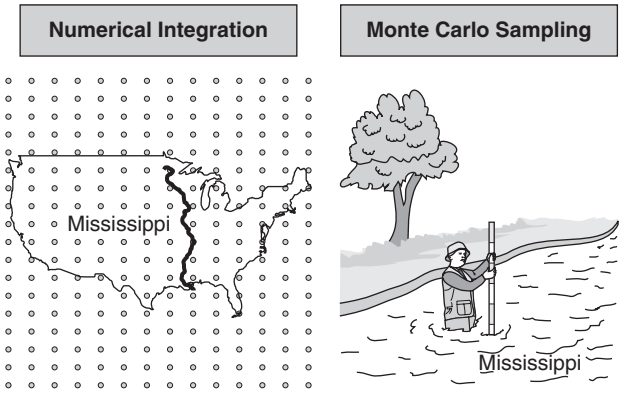

FIGURE 1.2 Monte Carlo sampling compared with numerical grid based integration for depth of Mississippi estimation.

Modern *MC* techniques such as in numerical integration seek to select random samples in high regions of concentration of the integrand by drawing samples from a proposed function very similar to the integrand. These methods lead to the well-known importance sampling approaches (see Chapter 3). Besides numerical integration problems that are very important in statistical signal processing for extracting signals/parameters of interest, numerical optimization techniques (e.g., genetic algorithms, simulated annealing, etc.) benefit directly from sampling technology. This important discovery has evolved ever since and become even more important with the recent development of high speed/high throughput computers.

Consider the following simple example of estimating the area of a circle to illustrate the *MC* approach.

Example 1.1

Define a sample space bounded by a square circumscribing (same center) a circle of radius r. Draw uniform random samples say $z := (X, Y)$ such that $z \sim \mathcal{U}(-r, +r)$; therefore, the number of random samples drawn from within the circle of radius r to the number of total samples drawn (bounded by the square) defines the probability

$$\Pr(Z = z) = \frac{\text{No. circle samples}}{\text{Total No. of (square) samples}}$$

From geometry we know that the probability is simply the ratio of the two areas (circle-to-square), that is,

$$\Pr(Z = z) = \frac{\pi r^2}{4r^2} = \pi/4$$

Let $r = 1$, then a simple computer code can be written that:

- Draws the X,Y-coordinates from $z \sim \mathcal{U}(-1, +1)$;
- Calculates the range function, $\rho = \sqrt{X^2 + Y^2}$;
- Counts the number of samples that are less than or equal to ρ;
- Estimates the probability, $\hat{\Pr}(Z = z)$.

The area is determined by multiplying the estimated probability by the area of the square. The resulting sample scatter plot is shown in Fig. 1.3 for a 10,000 sample realization resulting in $\pi \approx 3.130$. As the number of samples increase the estimate of the area (π) gets better and better demonstrating the *MC* approach. △△△

In signal processing, we are usually interested in some statistical measure of a random signal or parameter usually expressed in terms of its moments [16–23]. For example, suppose we have some signal function, say $f(X)$, with respect to some underlying probabilistic distribution, $\Pr(X)$. Then a typical measure to seek is its performance "on the average" which is characterized by the expectation

$$E_X\{f(X)\} = \int f(X) \Pr(X)\, dX \tag{1.3}$$

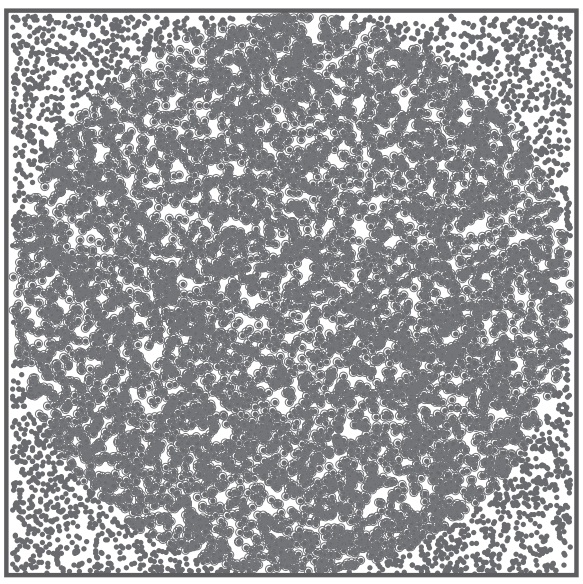

FIGURE 1.3 Area of a circle of unit radius using a Monte Carlo approach (area is estimated as 3.130 using 10,000 samples).

Instead of attempting to use direct numerical integration techniques, stochastic sampling techniques or *Monte Carlo integration* is an alternative. As mentioned, the *key idea* embedded in the *MC* approach is to represent the required distribution as a set of random *samples* rather than a specific analytic function (e.g., Gaussian). As the number of samples becomes large, they provide an equivalent (empirical) representation of the distribution enabling moments to be estimated directly (inference).

Monte Carlo integration draws samples from the required distribution and then forms sample averages to approximate the sought after distributions. That is, *MC* integration evaluates integrals by drawing samples, $\{X(i)\}$ from the designated distribution $\Pr(X)$. Assuming perfect sampling, this produces the estimated or *empirical distribution* given by

$$\hat{\Pr}(X) \approx \frac{1}{N}\sum_{i=1}^{N}\delta(X - X(i))$$

which is a probability mass distribution with weights, $\frac{1}{N}$ and random variable or sample, $X(i)$. Substituting the empirical distribution into the integral gives

$$E_X\{f(X)\} = \int f(X)\,\hat{\Pr}(X)\,dX \approx \frac{1}{N}\sum_{i=1}^{N} f(X(i)) \equiv \bar{f} \tag{1.4}$$

which follows directly from the sifting property of the delta or impulse function. Here $\bar{f}$ is said to be a *Monte Carlo estimate* of $E_X\{f(X)\}$.

As stated previously, scientists (Ulam, von Neumann, Metropolis, Fermi, Teller, etc. [7]) created statistical sampling-based or equivalently *simulation-based* methods for solving problems efficiently (e.g., neutron diffusion or eigenvalues of the Schrodinger relation). The *MC* approach to problem solving is a class of stochastic computations to simulate the dynamics of physical or mathematical systems capturing their inherent uncertainties. The *MC method* is a powerful means for generating random samples used in estimating conditional and marginal probability distributions required for statistical estimation and therefore signal processing. It offers an alternative numerical approach to find solutions to mathematical problems that cannot easily be solved by integral calculus or other numerical methods. As mentioned, the efficiency of the *MC* method increases (relative to other approaches) as the problem dimensionality increases. It is useful for investigating systems with a large number of degrees of freedom (e.g., energy transport, materials, cells, genetics) especially for systems with input uncertainty [5].

These concepts have recently evolved to the signal processing area and are of high interest in nonlinear estimation problems especially in model-based signal processing applications [16] as discussed next.

1.4 BAYESIAN MODEL-BASED SIGNAL PROCESSING

The estimation of probability distributions required to implement Bayesian processors is at the heart of this approach. How are these distributions obtained from data or simulations? Nonparametric methods of distribution estimation ranging from simple histogram estimators to sophisticated kernel smoothing techniques rooted in classification theory [3] offer reasonable approaches when data are available. However, these approaches usually do not take advantage of prior knowledge about the underlying physical phenomenology generating the data. An alternative is to parameterize the required distributions by prior knowledge of their actual form (e.g., exponential, Poisson, etc.) and fit their parameters from data using optimization techniques [3]. Perhaps the ideal realization is the parameterization of the evolution dynamics associated with the physical phenomenology using underlying mathematical representation of the process combined with the data samples. This idea provides the essence of the model-based approach to signal processing which (as we shall see) when combined with the Bayesian processors provide a formidable tool to attack a wide variety of complex processing problems in a unified manner. An alternative view of the underlying processing problem is to decompose it into a set of steps that capture the strategic essence of the processing scheme. Inherently, we believe that the more *a priori* knowledge about the measurement and its underlying phenomenology we can incorporate into the processor, the better we can expect the processor to perform—as long as the information that is included is correct! One strategy called the model-based approach provides the essence of model-based signal processing [1].

Simply stated, the *model-based approach* is "incorporating mathematical models of both physical phenomenology and the measurement process (including noise) into the processor to extract the desired information." This approach provides a mechanism to incorporate knowledge of the underlying physics or dynamics in the form of mathematical process models along with measurement system models and accompanying noise as well as model uncertainties directly into the resulting processor. In this way the model-based processor (*MBP*) enables the interpretation of results directly in terms of the problem physics. It is actually a modeler's tool enabling the incorporation of any *a priori* information about the problem to extract the desired information. The fidelity of the model incorporated into the processor determines the complexity of the model-based processor with the ultimate goal of increasing the inherent signal-to-noise ratio (*SNR*). These models can range from simple, implicit, non-physical representation of the measurement data such as the Fourier or wavelet transforms to parametric black-box models used for data prediction, to lumped mathematical representation characterized by ordinary differential equations, to distributed representations characterized by partial differential equation models to capture the underlying physics of the process under investigation. The dominating factor of which model is the most appropriate is usually determined by how severe the measurements are contaminated with noise and the underlying uncertainties. If the *SNR* of the measurements is high, then simple non-physical techniques can be used to extract the desired information; however, for low *SNR* measurements more and more of the physics and instrumentation must be incorporated for the extraction. For instance, consider the example of

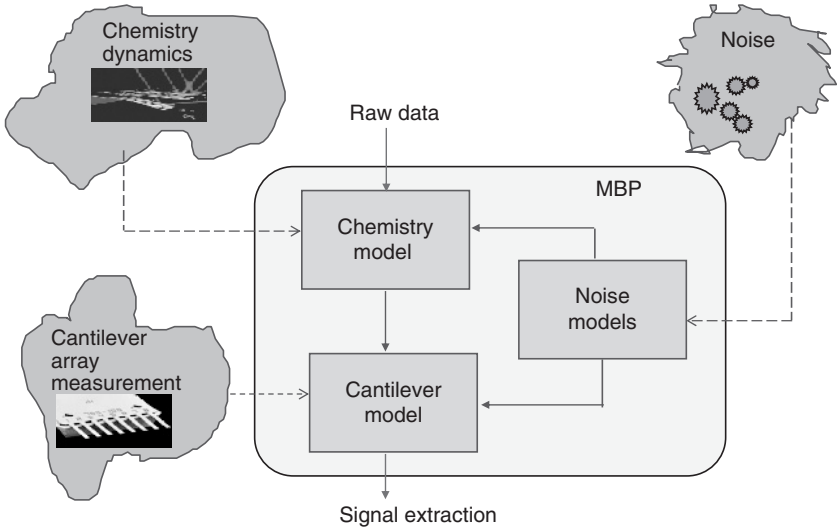

FIGURE 1.4 Model-based approach to signal processing: process (chemistry and physics), measurement (microcantilever sensor array) and noise (Gaussian) representations.

detecting the presence of a particular species in a test solution using a microcantilever sensor measurement system [4].

Example 1.2

The model-based processing problem is characterized in Fig. 1.4 representing the process of estimating the presence of a particular species of material in solution using the multichannel microcantilever sensor system. Here the microcantilever sensor is pre-conditioned by depositing attractor material on its levers to attract molecules of the target species. Once calibrated, the test solution flows along the levers with the target molecules attracted and deposited on each "tuned" microcantilever creating a deflection that is proportional to the concentration. This deflection is measured using a laser interferometric technique and digitized for processing. The process model is derived directly from the fluidics, while the measurement model evolves from the dynamics of the microcantilever structure. The resulting processor is depicted in Fig. 1.5, where we note the mathematical models of both the process dynamics and microcantilever measurement system. Since parameters, Θ, of the model are unknown *a priori* calibration data is used to estimate them directly and then they are employed in the *MBP* to provide the enhanced signal estimate shown in the figure. Even though nonlinear and non-Gaussian, the processor appears to yield reasonable estimates. See Sec. 10.3 [4] for details. $\triangle\triangle\triangle$

The above example demonstrates that incorporating reasonable mathematical models of the underlying phenomenology can lead to improved processing capability;

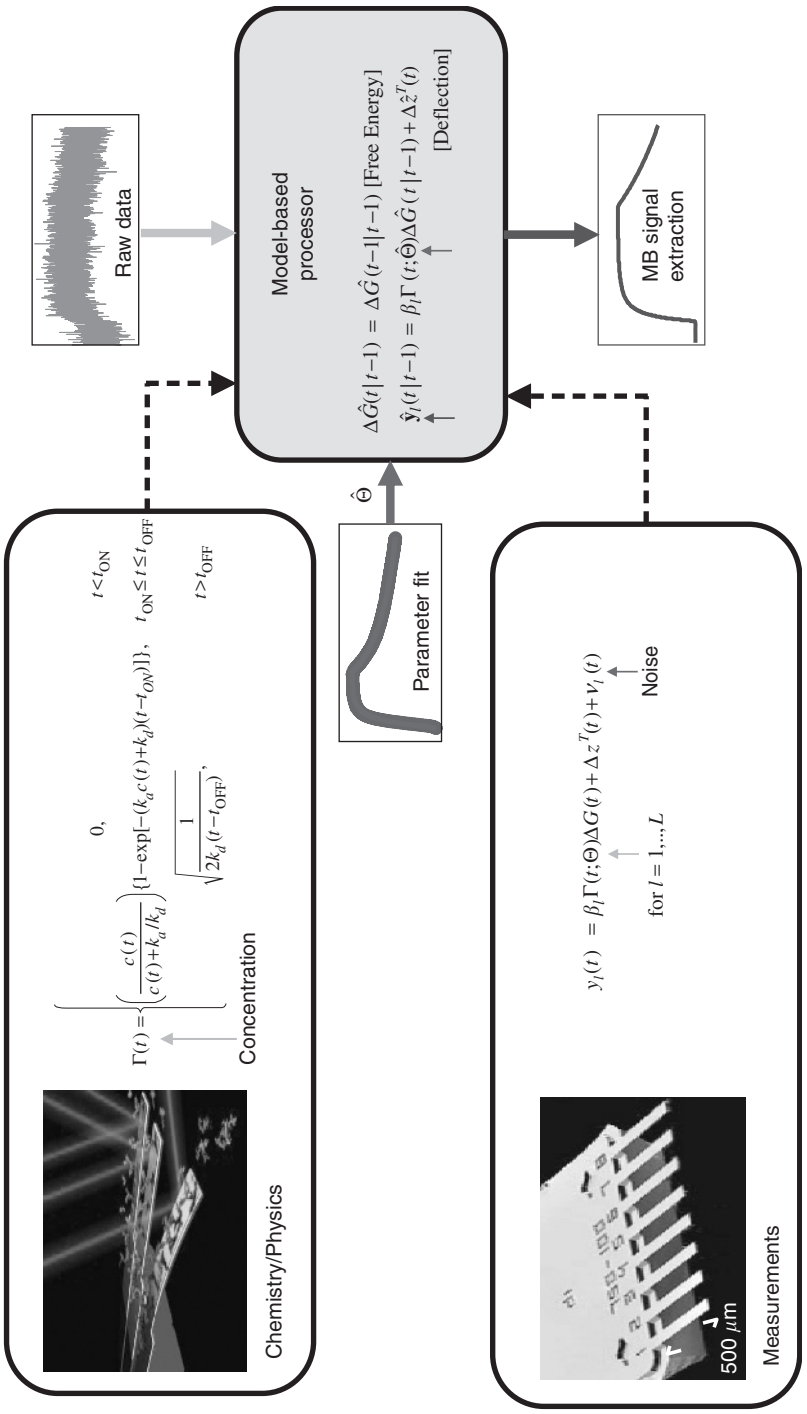

FIGURE 1.5 Model-based processor representation of species detection problem: process (concentration model), measurement (microcantilever sensor array), raw data, parameter estimator (coefficients) and model-based processor (enhancement).

however, even further advantages can be realized by combining the *MBP* concepts in conjunction with Bayesian constructs to generalize solutions.

Combining Bayesian and model-based signal processing can be considered a parametric representation of the required distributions using mathematical models of the underlying physical phenomenology and measurement (sensor) system. Certainly, if we assume the distribution is Gaussian and we further constrain the processes to be Markovian (only depending on the previous sample), then the multivariate Gaussian can be completely characterized using state–space models resulting in the well-known Kalman filter in the linear model case [2].

Since we are primarily concerned with pseudo real-time techniques in this text, we introduce the notion of a recursive form leading to the idea of sequential processing techniques. That is, we investigate "recursive" or equivalently "sequential" solutions to the estimation problem. Recursive estimation techniques evolved quite naturally during the advent of the digital computer in the late fifties, since both are sequential processes. It is important to realize the recursive solution is identical to the batch solution after it converges, so there is *no gain* in estimator performance properties; however, the number of computations is significantly less than the equivalent batch technique. It is also important to realize that the recursive approach provides the underlying theoretical and pragmatic basis of all *adaptive estimation* techniques; thus, they are important in their own right [2]!

Many processors can be placed in a recursive form with various subtleties emerging in the calculation of the current estimate ($\hat{X}_{old}$). The standard technique employed is based on correcting or updating the current estimate as a new measurement data sample becomes available. The estimates generally take the *recursive form*:

$$\hat{X}_{new} = \hat{X}_{old} + KE_{new} \tag{1.5}$$

where

$$E_{new} = Y - \hat{Y}_{old} = Y - C\hat{X}_{old}$$

Here we see that the new estimate is obtained by correcting the old estimate with a K-weighted error. The error term E_{new} is the new information or innovation—the difference between the actual and the predicted measurement ($\hat{Y}_{old}$) based on the old estimate ($\hat{X}_{old}$). The computation of the weight matrix K depends on the criterion used (e.g., mean-squared error, absolute error, etc.).

Consider the following example, which shows how to recursively estimate the sample mean.

Example 1.3

The sample mean estimator can easily be put in recursive form. The estimator is given by

$$\hat{X}(N) = \frac{1}{N}\sum_{t=1}^{N} y(t)$$

Extracting the N^{th} term from the sum, we obtain

$$\hat{X}(N) = \frac{1}{N}y(N) + \frac{1}{N}\sum_{t=1}^{N-1} y(t)$$

Identify $\hat{X}(N-1)$ from the last term,

$$\hat{X}(N) = \frac{1}{N}y(N) + \frac{N-1}{N}\hat{X}(N-1)$$

The recursive form is given by

$$\underbrace{\hat{X}(N)}_{NEW} = \underbrace{\hat{X}(N-1)}_{OLD} + \underbrace{\frac{1}{N}}_{WT}\underbrace{[y(N) - \hat{X}(N-1)]}_{ERROR}$$

This procedure to develop the "recursive form" is very important and can be applied to a multitude of processors. Note the steps in determining the form:

1. Remove the N^{th}-term from the summation;
2. Identify the previous estimate in terms of the $N-1$ remaining terms; and
3. Perform the algebra to determine the gain factor and place the estimator in the recursive form of Eq. 1.5 for a *scalar* measurement. △△△

1.5 NOTATION AND TERMINOLOGY

The notation used throughout this text is standard in the literature. Where necessary, vectors are represented by boldface, lowercase, **x**, and matrices by boldface, uppercase, **A**. We denote the real part of a signal by *Re x* and its imaginary part by *Im x*. We define the notation $\underline{N}$ to be a shorthand way of writing $1, 2, \ldots, N$. It will be used in matrices, $A(\underline{N})$ to mean there are N-columns of A. As mentioned previously, estimators are annotated by the caret, such as $\hat{x}$. We also define partial derivatives at the component level by $\frac{\partial}{\partial\theta_i}$, the N_θ-gradient vector by ∇_θ and higher order partials by ∇_θ^2.

The most difficult notational problem will be with the "time" indices. Since this text is predominantly discrete-time, we will use the usual time symbol, t to mean a discrete-time index, that is, $t \in \mathcal{I}$ for $\mathcal{I}$ the set of integers. However, and hopefully not too confusing, t will also be used for continuous-time, that is, $t \in \mathcal{R}$ for $\mathcal{R}$ the set of real numbers denoting the continuum. When used as a continuous-time variable, $t \in \mathcal{R}$ it will be represented as a subscript to distinguish it, that is, $x_{\mathbf{t}}$. This approach of choosing $t \in \mathcal{I}$ primarily follows the system identification literature and for the ease of recognizing discrete-time variable in transform relations (e.g., discrete Fourier transform). The rule-of-thumb is therefore to "interpret t as a discrete-time index

unless noted by a subscript as continuous in the text." With this in mind we will define a variety of discrete estimator notations as $\hat{x}(t|t-1)$ to mean the estimate at time (discrete) t based upon all of the previous data up to $t-1$. We will define these symbols prior to their use within the text to assure no misunderstanding of its meaning.

With a slight abuse of notation, we will use the terminology distribution of X, $\Pr(X)$ in general, so as not to have to differentiate between density for continuous random variables or processes and mass for discrete variates. It will be obvious from the context which is meant. In some cases, we will be required to make the distinction between cumulative distribution function (*CDF*) and density (*PDF*) or mass (*PMF*) functions. Here we use the uppercase notation, $\mathrm{P}_X(x)$ for the *CDF* and lower case $\mathrm{p}_X(x)$ for the *PDF* or *PMF*.

Subsequently we will also need to express a discrete *PMF* as a continuous *PDF* using impulse or delta functions as "samplers" much the same as in signal processing when we assume there exists an impulse sampler that leads to the well-known Nyquist sampling theorem [2]. Thus, corresponding to a discrete *PMF* we can define a continuous *PDF* through the concept of an *impulse sampler*, that is, given a discrete *PMF* defined by

$$\mathrm{p}_X(x) \approx \mathrm{p}(X = x_i) = \sum_i p_i \, \delta(x - x_i) \tag{1.6}$$

then we define the *equivalent continuous PDF* as $\mathrm{p}_X(x)$. Moments follow from the usual definitions associated with a continuous *PDF*, for instance, consider the definition of the expectation or mean. Substituting the equivalent *PDF* and utilizing the sifting property of the impulse function gives

$$E\{x\} = \int_{-\infty}^{\infty} x \, \mathrm{p}_X(x) \, dx = \int_{-\infty}^{\infty} x \left(\sum_i p_i \, \delta(x - x_i) \right) dx = \sum_i x_i p_i \tag{1.7}$$

which is precisely the mean of the discrete *PMF*.

Also, as mentioned, we will use the symbol $\sim$ to mean "distributed according to" as in $x \sim \mathcal{N}(m, v)$ defining the random variable x as Gaussian distributed with mean m and variance v. We may also use the extended notation: $\mathcal{N}(x : m, v)$ to include the random variable x as well. When *sampling* we use the *non-conventional* right arrow "action" notation $\rightarrow$ to mean "draw a sample from" a particular distribution such as $x_i \rightarrow \Pr(x)$—this again will be clear from the context. When *resampling*, that is, replacing samples with new ones we use the "block" right arrow such as $x_j \Rightarrow x_i$ meaning new sample x_j replaces current sample x_i.

Finally in a discrete (finite) probabilistic representation, we define a purely discrete variate as $x_k(t) := \Pr(x(t) = \mathcal{X}_k)$ meaning that x can only take on values (integers) k from a known set $\mathcal{X} = \{\mathcal{X}_1, \ldots, \mathcal{X}_k, \ldots, \mathcal{X}_N\}$ at time t. We also use the symbol, $\triangle\triangle\triangle$ to mark the end of an example.

MATLAB NOTES

MATLAB is command oriented vector-matrix package with a simple yet effective command language featuring a wide variety of embedded *C* language constructs making it ideal for signal processing applications and graphics. All of the algorithms we have applied to the examples and problems in this text are *MATLAB*-based in solution ranging from simple simulations to complex applications. We will develop these notes primarily as a summary to point out to the reader many of the existing commands that already perform the signal processing operations discussed in the presented chapter and throughout the text.

REFERENCES

1. J. Candy, *Signal Processing: The Model-Based Approach* (New York: McGraw-Hill, 1986).
2. J. Candy, *Model-Based Signal Processing* (Hoboken, NJ: Wiley/IEEE Press, 2006).
3. R. Duda, P. Hart and D. Stork, *Pattern Classification* (Hoboken, NJ: Wiley/IEEE Press, 2001).
4. J. Tringe, D. Clague, J. Candy and C. Lee, "Model-based signal processing of multichannel cantilever arrays," *IEEE J. Micromech. Syst.*, **15**, 5, 1371–1391, 2006.
5. S. Ulam, R. Richtmyer and J. von Neumann, "Statistical methods in neutron diffusion," *Los Alamos Scientific Laboratory Report*, **LAMS-551**, 1947.
6. N. Metropolis and S. Ulam, "The Monte Carlo method," *J. American Stat. Assoc.*, **44**, 335–341, 1949.
7. N. Metropolis, A. Rosenbluth, M. Rosenbluth, A. Teller and E. Teller, "Equations of state calculations by fast computing," *J. Chemical Physics*, **21**, 6, 1087–1091, 1953.
8. W. Hastings, "Monte Carlo sampling methods using Markov chains and their applications," *Biometrika*, **57**, 1, 97–109, 1970.
9. N. Metropolis, "The beginning of the Monte Carlo method," *Los Alamos Science*, Special Issue, 125–130, 1987.
10. J. Liu, *Monte Carlo Strategies in Scientific Computing* (New York: Springer-Verlag, 2001).
11. D. Frenkel, "Introduction to Monte Carlo methods," in *Computational Soft Matter: From Synthetic Polymers to Proteins*, N. Attig, K. Binder, H. Grubmuller and K. Kremer (Eds.) J. von Neumann Instit. for Computing, Julich, NIC Series, Vol. 23, pp. 29–60, 2004.
12. C. Robert and G. Casella, *Monte Carlo Statistical Methods* (New York: Springer, 1999).
13. M. Tanner, *Tools for Statistical Inference: Methods for the Exploration of Posterior Distributions and Likelihood Functions, 2nd Ed.* (New York: Springer-Verlag, 1993).
14. J. Ruanaidh and W. Fitzgerald, *Numerical Bayesian Methods Applied to Signal Processing* (New York: Springer-Verlag, 1996).
15. W. Gilks, S. Richardson and D. Spiegelhalter, *Markov Chain Monte Carlo in Practice* (New York: Chapman & Hall/CRC Press, 1996).
16. A. Doucet, N. de Freitas and N. Gordon, *Sequential Monte Carlo Methods in Practice* (New York: Springer-Verlag, 2001).

17. B. Ristic, S. Arulampalam and N. Gordon, *Beyond the Kalman Filter: Particle Filters for Tracking Applications* (Boston: Artech House, 2004).
18. O. Cappe, E. Moulines and T. Ryden, *Inference in Hidden Markov Models* (New York: Springer-Verlag, 2005).
19. S. Godsill and P. Djuric, "Special Issue: Monte Carlo methods for statistical signal processing." *IEEE Trans. Signal Proc.*, **50**, 173–499, 2002.
20. P. Djuric, J. Kotecha, J. Zhang, Y. Huang, T. Ghirmai, M. Bugallo and J. Miguez, "Particle Filtering," *IEEE Signal Proc. Mag.*, **20**, 5, 19–38, 2003.
21. S. Haykin and N. de Freitas, "Special Issue: Sequential state estimation: from Kalman filters to particle filters." *Proc. IEEE*, **92**, 3, 399–574, 2004.
22. A. Doucet and X. Wang, "Monte Carlo methods for signal processing," *IEEE Signal Proc. Mag.*, **24**, 5, 152–170, 2005.
23. J. Candy, "Bootstrap particle filtering for passive synthetic aperture in an uncertain ocean environment." *IEEE Signal Proc. Mag.*, **24**, 4, 73–85, 2007.

PROBLEMS

1.1 Estimate the number of times a needle when dropped between two parallel lines intersects a line. One was to accomplish this is experimentally by setting up the experiment and doing it—this is the famous Buffon's needle experiment performed in 1725.

(*a*) Set up the experiment and perform the measurements for 100 samples. Estimate the underlying probabilities.

(*b*) Analyze the experiment using a "closed form" approach.

(*c*) How do your answers compare?

Note that this is one of the first Monte Carlo approaches to problem solving.

1.2 Suppose we have three loaded dice with the following six "face" probabilities (each):

$$D1 = \left\{\frac{1}{12}, \frac{1}{6}, \frac{1}{12}, \frac{1}{3}, \frac{1}{6}, \frac{1}{6}\right\}$$

$$D2 = \left\{\frac{1}{6}, \frac{1}{6}, \frac{1}{6}, \frac{1}{12}, \frac{1}{12}, \frac{1}{3}\right\}$$

$$D3 = \left\{\frac{1}{6}, \frac{1}{6}, \frac{1}{6}, \frac{1}{12}, \frac{1}{12}, \frac{1}{3}\right\}$$

Applying Bayes' rule, answer the following questions:

(*a*) Selecting a die at random from the three, what is the probability of rolling a 6?

(*b*) What is the probability that die two ($D = D2$) was selected, if a six ($R = 6$) is rolled with the chosen die?

1.3 A binary communication transmitter (T) sends either a 0 or a 1 through a channel to a receiver (R) with the following probabilities for each as:

$$\begin{aligned} \Pr(T_1) &= 0.6 \quad \Pr(T_0) &&= 0.4 \\ \Pr(R_1|T_1) &= 0.9 \quad \Pr(R_0|T_1) &&= 0.1 \\ \Pr(R_1|T_0) &= 0.1 \quad \Pr(R_0|T_0) &&= 0.9 \end{aligned}$$

(a) What is the probability that R_1 is received?

(b) What is the probability that R_0 is received?

(c) What is the probability that the true transmitted signal was a 1, when a 1 was received?

(d) What is the probability that the true transmitted signal was a 0, when a 0 was received?

(e) What is the probability that the true transmitted signal was a 1, when a 0 was received?

(f) What is the probability that the true transmitted signal was a 0, when a 1 was received?

(g) Draw a probabilistic directed graph with nodes being the transmitters and receivers and links being the corresponding prior and conditional probabilities?

1.4 We are asked to estimate the displacement of large vehicles (semi-trailers) when parked on the shoulder of a freeway and subjected to wind gusts created by passing vehicles. We measure the displacement of the vehicle by placing an accelerometer on the trailer. The accelerometer has inherent inaccuracies which is modeled as

$$y = K_a x + n$$

with y, x, n the measured and actual displacement and white measurement noise of variance R_{nn} and K_a the instrument gain. The dynamics of the vehicle can be modeled by a simple mass-spring-damper.

(a) Construct and identify the measurement model of this system.

(b) Construct and identify the process model and model-based estimator for this problem.

1.5 Think of measuring the temperature of a liquid in a beaker heated by a burner. Suppose we use a thermometer immersed in the liquid and periodically observe the temperature and record it.

(a) Construct a measurement model assuming that the thermometer is linearly related to the temperature, that is, $y(t) = k\,\triangle T(t)$. Also model the uncertainty of the visual measurement as a random sequence $v(t)$ with variance R_{vv}.

(*b*) Suppose we model the heat transferred to the liquid from the burner as

$$Q(t) = CA\ \triangle T(t)$$

where C is the coefficient of thermal conductivity, A is the cross-sectional area, and $\triangle T(t)$ is the temperature gradient with assumed random uncertainty $w(t)$ and variance R_{ww}. Using this process model and the models developed above, identify the model-based processor representation.

1.6 We are given an *RLC* series circuit driven by a noisy voltage source $V_{in}(t)$ and we use a measurement instrument that linearly amplifies by K and measures the corresponding output voltage. We know that the input voltage is contaminated by and additive noise source, $w(t)$ with covariance, R_{ww} and the measured output voltage is similarly contaminated with noise source, $v(t)$ with R_{vv}.

(*a*) Determine the model for the measured output voltage, $V_{out}(t)$ (measurement model).

(*b*) Determine a model for the circuit (process model).

(*c*) Identify the general model-based processor structures. In each scheme, specify the models for the process, measurement and noise.

1.7 A communications satellite is placed into orbit and must be maneuvered using thrusters to orientate its antennas. Restricting the problem to the single axis perpendicular to the page, the equations of motion are

$$J\frac{d^2\theta}{dt^2} = T_c + T_d$$

where J is the moment of inertia of the satellite about its center of mass, T_c is the thruster control torque, T_d is the disturbance torque, and θ is the angle of the satellite axis with respect to the inertial reference (no angular acceleration) A. Develop signal and noise models for this problem and identify each model-based processor component.

1.8 Consider a process described by a set of linear differential equations

$$\frac{d^2c}{dt^2} + \frac{dc}{dt} + c = Km$$

The process is to be controlled by a proportional-integral-derivative (PID) control law governed by the equation

$$m = K_p\left(e + \frac{1}{T_i}\int e\,dt + T_d\frac{de}{dt}\right)$$

and the controller reference signal r is given by

$$r = e + c$$

Suppose the reference is subjected to a disturbance signal and the measurement sensor, which is contaminated with additive noise, measures the "square" of the output. Develop the model-based signal and noise models for this problem.

1.9 The elevation of a tracking telescope is controlled by a DC motor. It has a moment of inertia J and damping B due to friction, the equation of motion is given by

$$J\frac{d^2\theta}{dt^2} + B\frac{d\theta}{dt} = T_m + T_d$$

where T_m and T_d are the motor and disturbance torques and θ is the elevation angle. Assume a sensor transforms the telescope elevation into a proportional voltage that is contaminated with noise. Develop the signal and noise models for the telescope and identify all of the model-based processor components.

1.10 Suppose we have a two-measurement system given by

$$y = \begin{bmatrix} 3 \\ 4 \end{bmatrix} + v$$

where $R_{vv} = \text{diag}[1,\ 0.1]$.

(*a*) What is the batch least-squares estimate ($W = I$) of the parameter x, if $y = [7 \quad 21]'$?

(*b*) What is the batch weighted least-squares estimate of the parameter x with W selected for minimum variance estimation?

1.11 Calculate the batch and sequential least-squares estimate of the parameter vector x based on two measurements $y(1)$ and $y(2)$ where

$$y(1) = C(1)x + v(1) = \begin{bmatrix} 2 \\ 1 \end{bmatrix}$$

$$y(2) = c'x + v(2) = 4$$

$$C = \begin{bmatrix} 1 & 1 \\ 0 & 1 \end{bmatrix}, \quad c'(1) = [1 \quad 2], \quad W = I$$