

1972

Connection between sampled-data digital filter and Kalman filter

Gong Yoon Oak
Iowa State University

Follow this and additional works at: <https://lib.dr.iastate.edu/rtd>

 Part of the [Electrical and Electronics Commons](#)

Recommended Citation

Oak, Gong Yoon, "Connection between sampled-data digital filter and Kalman filter " (1972). *Retrospective Theses and Dissertations*. 5270.

<https://lib.dr.iastate.edu/rtd/5270>

This Dissertation is brought to you for free and open access by the Iowa State University Capstones, Theses and Dissertations at Iowa State University Digital Repository. It has been accepted for inclusion in Retrospective Theses and Dissertations by an authorized administrator of Iowa State University Digital Repository. For more information, please contact digirep@iastate.edu.

INFORMATION TO USERS

This dissertation was produced from a microfilm copy of the original document. While the most advanced technological means to photograph and reproduce this document have been used, the quality is heavily dependent upon the quality of the original submitted.

The following explanation of techniques is provided to help you understand markings or patterns which may appear on this reproduction.

1. The sign or "target" for pages apparently lacking from the document photographed is "Missing Page(s)". If it was possible to obtain the missing page(s) or section, they are spliced into the film along with adjacent pages. This may have necessitated cutting thru an image and duplicating adjacent pages to insure you complete continuity.
2. When an image on the film is obliterated with a large round black mark, it is an indication that the photographer suspected that the copy may have moved during exposure and thus cause a blurred image. You will find a good image of the page in the adjacent frame.
3. When a map, drawing or chart, etc., was part of the material being photographed the photographer followed a definite method in "sectioning" the material. It is customary to begin photoing at the upper left hand corner of a large sheet and to continue photoing from left to right in equal sections with a small overlap. If necessary, sectioning is continued again — beginning below the first row and continuing on until complete.
4. The majority of users indicate that the textual content is of greatest value, however, a somewhat higher quality reproduction could be made from "photographs" if essential to the understanding of the dissertation. Silver prints of "photographs" may be ordered at additional charge by writing the Order Department, giving the catalog number, title, author and specific pages you wish reproduced.

University Microfilms

300 North Zeeb Road
Ann Arbor, Michigan 48106

A Xerox Education Company

72-26,935

OAK, Gong Yoon, 1942-
CONNECTION BETWEEN SAMPLED-DATA DIGITAL FILTER
AND KALMAN FILTER.

Iowa State University, Ph.D., 1972
Engineering, electrical

University Microfilms, A XEROX Company , Ann Arbor, Michigan

Connection between sampled-data digital filter
and Kalman filter

by

Gong Yoon Oak

A Dissertation Submitted to the
Graduate Faculty in Partial Fulfillment of
The Requirements for the Degree of
DOCTOR OF PHILOSOPHY

Subject: Electrical Engineering

Approved:

Signature was redacted for privacy.

In Charge of Major Work

Signature was redacted for privacy.

For the Major Department

Signature was redacted for privacy.

For ~~the~~ ~~of~~ Graduate College

Iowa State University
Ames, Iowa

1972

PLEASE NOTE:

Some pages may have
indistinct print.

Filmed as received.

University Microfilms, A Xerox Education Company

TABLE OF CONTENTS

	Page
I. INTRODUCTION	1
II. KALMAN-BUCY FILTER AND DISCRETE KALMAN FILTER	8
III. CLASSICAL DESIGN OF SAMPLED-DATA DIGITAL FILTER	21
IV. EQUIVALENT KALMAN-BUCY FILTER	43
V. DISCRETE KALMAN-BUCY DERIVED FILTER	61
VI. COMPARISON OF DISCRETE KALMAN-BUCY DERIVED FILTER AND z -TRANSFORM DERIVED FILTER	77
VII. CONCLUSIONS	119
VIII. LITERATURE CITED	124
IX. ACKNOWLEDGEMENTS	127
X. APPENDIX A	128
XI. APPENDIX B	131
XII. APPENDIX C	140
XIII. APPENDIX D	146

I. INTRODUCTION

In control and communication systems, it is often desirable to insert a network that will freely pass currents of one band of frequencies but that will greatly attenuate currents of frequencies outside this band. Such selective networks are called filters. This shaping of amplitude or phase with respect to frequencies is based on electrical properties of inductances, capacitances, and resistances. These lead to a description of the performance of a network by a set of linear differential equations. By contrast, a set of linear difference equations is used to describe a discrete linear system. These equations are realized in a special or a general purpose digital computer.

The justification for the use of digital methods is based on the advantages of inherent stability and accuracy, on greater variety of digital filters which can be built, and on simple components needed for time-varying parameters, all of which are problem areas on analog circuits. In addition, the rapid advances in integrated-circuit technology coupled with increasingly sophisticated signal filtering have made filtering by digital techniques more and more feasible.

Linear digital filter theory is largely the results of James, Nichols and Phillips (14), Truxal (35), Ragazzini and Franklin (32), and Blackman (1). Boxer and Thaler (3), Kaiser (17), and Carney (7) are some who have made initial steps towards the development of design techniques from the point of view of frequency selectivity.

Most of digital filter design techniques are formulated in the frequency domain. Rader and Gold (31), Rader (30), Golden (10), and

Otens (26) have summarized these frequency-domain methods. A major distinction can be made between functions of sampled-data digital filter and digital filter. The sampled-data digital filter operates on signals that have been quantized in time by a sampling device, whereas the digital filter requires data that have been quantized in both time and amplitude, and then encoded into binary form for digital processing.

The design of a sampled-data digital filter often begins with a requirement to find a digital approximation for some continuous filter. Various methods have been described for mapping a given analog filter into a digital form. Transforms that permit formulation of digital filter transfer function from an analog filter transfer function are called z-transforms or sampled-data transforms. Three special transforms that find the greatest applications are the impulse invariant z-transform, the bilinear z-transform, and the matched z-transform. But, unfortunately, not all of these techniques produce filters that have the same degree of stability as their analog counterparts.

Burrs and Parks (6) have studied a time-domain design technique which has a desired impulse response over a specified interval. The description of the time-domain technique is in terms of the z-transform transfer function with undetermined coefficients. This synthesis method is to find those unknown coefficients such that the impulse response of the transfer function approximates in some sense a given continuous impulse response. Several design procedures that require only linear calculations are given for approximate realization of digital filter transfer function.

In the analog filters, the realization of a given filter transfer function is a difficult problem that has received considerable attention. For the sampled-data digital filter, the digital computer realization is almost trivial. But, for an accurate digital filter implementation, selection of the implementation scheme is important. The effects of different transformations, coefficient word length, computational word length, sampling rate have to be predicted and verified for the realization of the digital filter after the sampled-data filter has been selected. The problems caused by quantization of the sampled-data digital filter have been discussed by Gold and Rader (9), and Knowles and Olcayto (22).

Digital filter implementation has been confined primarily to computer programs for simulation works or for processing relatively small amounts of data, usually not in real time. Conveniently, the digital computer provides printouts of all coefficients as well as frequency responses of amplitude and phase, and transient response. However, the rapid development of the integrated-circuit technology and especially the potential for large-scale integration of digital circuits now makes it possible for the digital filter to be constructed from a small set of relatively simple digital circuits.

Jackson, Kaiser and McDonald (13), and White and Mitsutomi (37) show that the digital filter may be implemented inexpensively by a building block method using integrated-circuit chips. Adders, multipliers, shift registers, address logic, control logic, and timing logic are contained in a few chips, and convert the mathematics into circuits. This special digital filter is optimized for the applications. As an

example, a recursive digital filter design technique is developed for airborne applications that enables the filter designer to minimize computer word-length requirements, and, at the same time, meet essential practical constraints.

When filters are considered from the point of view of optimization, performance criteria have to be introduced. The criteria should be relevant to the purpose of the filter, such as the separation of signals from noise, and quantitative evaluation of the filter against the selected performance criteria should be analytically tractable. Then the filter is synthesized according to the chosen criteria, such as the minimum mean-square error.

The initial significant contribution to this problem was made by Wiener (38). He synthesized the optimum linear, minimum mean-square-error filter for the case where the measurement and signal are continuous time and stationary random processes. The optimization procedure always leads to a Wiener-Hopf integral equation of the first kind which is difficult to solve in most cases. Bode and Shannon (2) solved Wiener filter problem in a more general form than that of James, Nichols and Phillips (14), and their solution is essentially an analysis from a frequency-domain viewpoint of the same problem Wiener considered in time domain.

Kalman and Bucy (19), (21) then solved the same problem in an entirely different way. They assumed that the vector signal process could be characterized as the state variables of a linear dynamical

system excited by uncorrelated noise. The measurement process is then assumed to be a linear transform of the state vector, corrupted by a vector noise process. The "Kalman filter," which includes both Kalman-Bucy continuous filter and discrete Kalman filter, is the result of the pioneering works by them. The optimality of the Kalman filter requires two assumptions: the linearity of the dynamical system, and the complete knowledge of the a priori Gaussian statistics of the random processes involved.

Kalman (20) also proved that the covariance of the error between the actual state and filter's estimate of the state converges to an equilibrium state irrespective of the initial state when the dynamical system is uniformly completely controllable and uniformly completely observable. This stability is investigated further for the case where there is insufficient knowledge of a priori information by Nishmura (25), Sorenson (34), Griffin and Sage (12), and Price (29). Schlee, Standish and Toda (33) investigated the problems of divergence in the Kalman filter implementation.

The primary intent of this dissertation is to develop a systematic design approach for digital implementation of any continuous filter whose transfer function is given as a ratio of two polynomials in the complex frequency variable s . When the continuous filter is given in terms of a weighting function or a fundamental differential equation describing the filter, the system transfer function of the continuous filter is obtained from them without any problem for one is just the transform of the other.

The development begins with an equivalent Kalman-Bucy filter, which is composed of a fixed-gain scalar-measurement Kalman-Bucy filter and an additional output equation. The state representation of the denominator of the continuous filter transfer function leads to the fixed-gain Kalman-Bucy filter and that of the numerator becomes the output equation. Even if both filters have been started from different design viewpoints (for example, the continuous filter is for frequency selectivity of signals, and the Kalman-Bucy filter is for state estimation of signals), they satisfy the same purpose for the separation of signals from noise. Since the equivalent Kalman-Bucy filter behaves the same as the classical continuous filter, not only in the steady state but also in the transient state, the gain in the equivalent Kalman-Bucy filter has to be fixed. Being an optimum filter, the equivalent Kalman-Bucy filter needs more information about signal and noise. As a result of this, the linear dynamical system, system noise, and measurement noise characteristics have to be imagined and added to the continuous filter assignments. The input signal of the equivalent Kalman-Bucy filter is supposed to be generated by those imagined stochastic process models. The fixed-gain Kalman-Bucy filter of the equivalent Kalman-Bucy filter becomes an optimum filter to these imagined models. Noise statistics in the stochastic signal and measurement process models are assumed to be simple so that the matrix Riccati-type steady-state covariance equation is represented as a set of simple first-order linear differential equations.

The next step is the discretization of the derived fixed-gain Kalman-Bucy filter and the output equation, which are equivalent to the given continuous filter. The advantage of this approach is that the conversion to discrete form is quite a routine process. The derived fixed-gain Kalman-Bucy filter is discretized to be a form of discrete Kalman filter, and the output equation is not changed except for the discretization of continuous time. Since the gain is also fixed after discretization, the discretized equivalent Kalman-Bucy filter can be simplified, which is called the discrete Kalman-Bucy derived filter.

To compare the resulting discrete Kalman-Bucy derived filter to various z-transform derived filters, the amount of digital hardware requirements is estimated, and the fidelity of each in terms of frequency response is derived.

As an aid to understanding the development of the new filter design approach, a tutorial survey of Kalman-Bucy filter and discrete Kalman filter is presented in the following chapter.

Before describing the new approach of the filter design, the classical design methods are reviewed in the preceding chapter. The the purpose of comparison to the new filter method, the z-transform approaches are summarized.

II. KALMAN-BUCY FILTER AND DISCRETE KALMAN FILTER

The Kalman filter is an optimal filter that estimates the states of a linear dynamic system from measurement data which are linearly related to the states. When this system is driven by white Gaussian noises, and when a set of noisy sequential measurements are processed, the Kalman filter provides the minimum variance, unbiased estimate of the system state. The basic feature of this filter consists of two parts, the state estimator and the gain computation.

The state estimator is updated by a sequence of measurements so as to minimize its variance, or equivalently in Gaussian statistics, to maximize the conditional probability density of the current states after having a set of measurements. In this estimator, the a priori estimation error from the updated measurement is weighted by a gain factor which is a solution of the gain computation.

The error covariance matrix associated with the gain computation is used as the statistical description of the error in the estimate. The covariance matrix equation is of the Riccati-type nonlinear equation. Since this statistical measure is independent of the measurement data, it can be computed from the system matrices and a priori statistics. Thus, this covariance matrix can be examined before applying the filter to an actual realization of a physical system in order to determine whether the expected response is satisfactory.

This filter design assumes a priori knowledge of the initial states and their variances as well as complete knowledge of system

dynamics and noise statistics. Then the estimation procedure is expressed as a set of recursion relationships.

In this chapter, the statistical models of the signal and the noise processes considered in this dissertation will first be defined. Then the "Kalman filter," including both Kalman-Bucy continuous filter and discrete Kalman filter, will be reviewed to show how the minimum mean-square-error estimates can be obtained. In the succeeding discussion, the error covariance matrix will be examined with the intent of establishing conditions that must be satisfied to assure stability of the filters. The concepts of observability and controllability play a key role for this stability.

A. Kalman-Bucy Filter

The statistical model of the signal process is assumed to be described by the continuous, linear differential equation

$$\dot{\underline{\xi}}(t) = F(t)\underline{\xi}(t) + G(t)\underline{u}(t) \quad (2-1)$$

where

$\underline{\xi}(t)$ is a k_{ξ} vector of states with

$$E\{\underline{\xi}(0)\} = \underline{0},$$

$\underline{u}(t)$ is a k_u vector stochastic input of the signal process with

$$E\{\underline{u}(t)\} = \underline{0} \quad \text{for all } t,$$

$$E\{\underline{u}(t)\underline{u}(\tau)'\} = Q(t)\delta(t-\tau) \quad \text{for all } t \text{ and } \tau.$$

Here $Q(t)$ is a $k_u \times k_u$ symmetric nonnegative-definite

matrix, $\delta(t)$ is a Dirac delta function, $E\{*\}$ is an expectation value of $*$, and $\underline{u}(t)'$ is a transpose of $\underline{u}(t)$. $F(t)$ is a $k_{\xi} \times k_{\xi}$ system matrix, and $G(t)$ is a $k_{\xi} \times k_u$ input matrix.

The statistical model of the measurement process is assumed to be described by the equation

$$\underline{y}(t) = H(t)\underline{\xi}(t) + \underline{v}(t) \quad (2-2)$$

where

$\underline{y}(t)$ is a k_y measurement vector,

$\underline{v}(t)$ is a k_y measurement noise vector with

$$E\{\underline{v}(t)\} = \underline{0} \quad \text{for all } t,$$

$$E\{\underline{v}(t)\underline{v}(\tau)'\} = R(t)\delta(t-\tau) \quad \text{for all } t \text{ and } \tau.$$

Here $R(t)$ is a $k_y \times k_y$ symmetric positive-definite matrix, and

$H(t)$ is a $k_y \times k_{\xi}$ measurement matrix.

It is also assumed that the process noise $\underline{u}(t)$ and the measurement noise $\underline{v}(t)$ are uncorrelated. That is

$$E\{\underline{u}(t)\underline{v}(\tau)'\} = 0 \quad \text{for all } t \text{ and } \tau. \quad (2-3)$$

Then an estimate of $\underline{\xi}(t)$, given the measurement $y(\tau)$ where τ is $0 \leq \tau \leq s$, is denoted by $\hat{\underline{\xi}}(t|s)$. When $t > s$, it denotes a prediction, $t = s$, it denotes a filtering, and $t < s$, it denotes a smoothing.

The optimum estimator $\hat{\underline{\xi}}(t|s)$ of $\underline{\xi}(t)$ which minimizes

$$E\{\|\hat{\underline{\xi}}(t|s) - \underline{\xi}(t)\|^2\} \quad (2-4)$$

is given as

$$\hat{\underline{\xi}}(t|s) = E\{\underline{\xi}(t) | \underline{y}(\tau), 0 \leq \tau \leq s\} \quad (2-5)$$

where $E\{\cdot | *\}$ is a conditional expectation of \cdot given $*$.

The Kalman-Bucy algorithm, with given initial conditions $\hat{\underline{\xi}}(0|0)$ and $P(0)$, which estimates the state vector $\underline{\xi}(t)$, given measurements $\underline{y}(\tau)$, $0 \leq \tau \leq t$, is obtained as follows.

$$\dot{\hat{\underline{\xi}}}(t|t) = F(t)\hat{\underline{\xi}}(t|t) + K(t)[\underline{y}(t) - H(t)\hat{\underline{\xi}}(t|t)] \quad (2-6)$$

$$K(t) = P(t|t)H(t)'R^{-1}(t) \quad (2-7)$$

$$\begin{aligned} \dot{P}(t|t) = & F(t)P(t|t) + P(t|t)F(t)' - P(t|t)H(t)'R^{-1}(t)H(t)P(t|t) \\ & + G(t)Q(t)G(t)' \end{aligned} \quad (2-8)$$

where

$K(t)$ is a $k_{\underline{\xi}} \times k_y$ gain matrix for incorporating $\underline{y}(t)$ into the estimate of $\underline{\xi}(t)$, and

$P(t|t)$ is a $k_{\underline{\xi}} \times k_{\underline{\xi}}$ symmetric covariance matrix which is the covariance of the error in estimating $\underline{\xi}(t)$ based on the knowledge of $\underline{y}(\tau)$, $0 \leq \tau \leq t$.

The matrix block diagram of the Kalman-Bucy filter is shown in Figure 2.1.

B. Discrete Kalman Filter

The statistical model of the signal process is assumed to be described by the discrete, linear vector difference equation, with a uniform sampling interval T ,

$$\underline{\xi}_{n+1} = F_n \underline{\xi}_n + G_n u_{n-n} \quad (2-9)$$

where

$\underline{\xi}_n$ is a k_ξ vector of states with

$$E\{\underline{\xi}_0\} = \underline{0},$$

u_n is a k_u vector stochastic input to the signal process with

$$E\{u_n\} = \underline{0}, \quad \text{for every } n,$$

$$E\{u_m u_n'\} = \begin{cases} Q_n & \text{if } n = m, \\ 0 & \text{if } n \neq m. \end{cases}$$

Here Q_n is a $k_u \times k_u$ symmetric nonnegative-definite matrix,

F_n is a $k_\xi \times k_\xi$ system matrix, and

G_n is a $k_\xi \times k_u$ input matrix.

The statistical model of the measurement process is assumed to be described by the equation

$$y_n = H_n \underline{\xi}_n + v_n \quad (2-10)$$

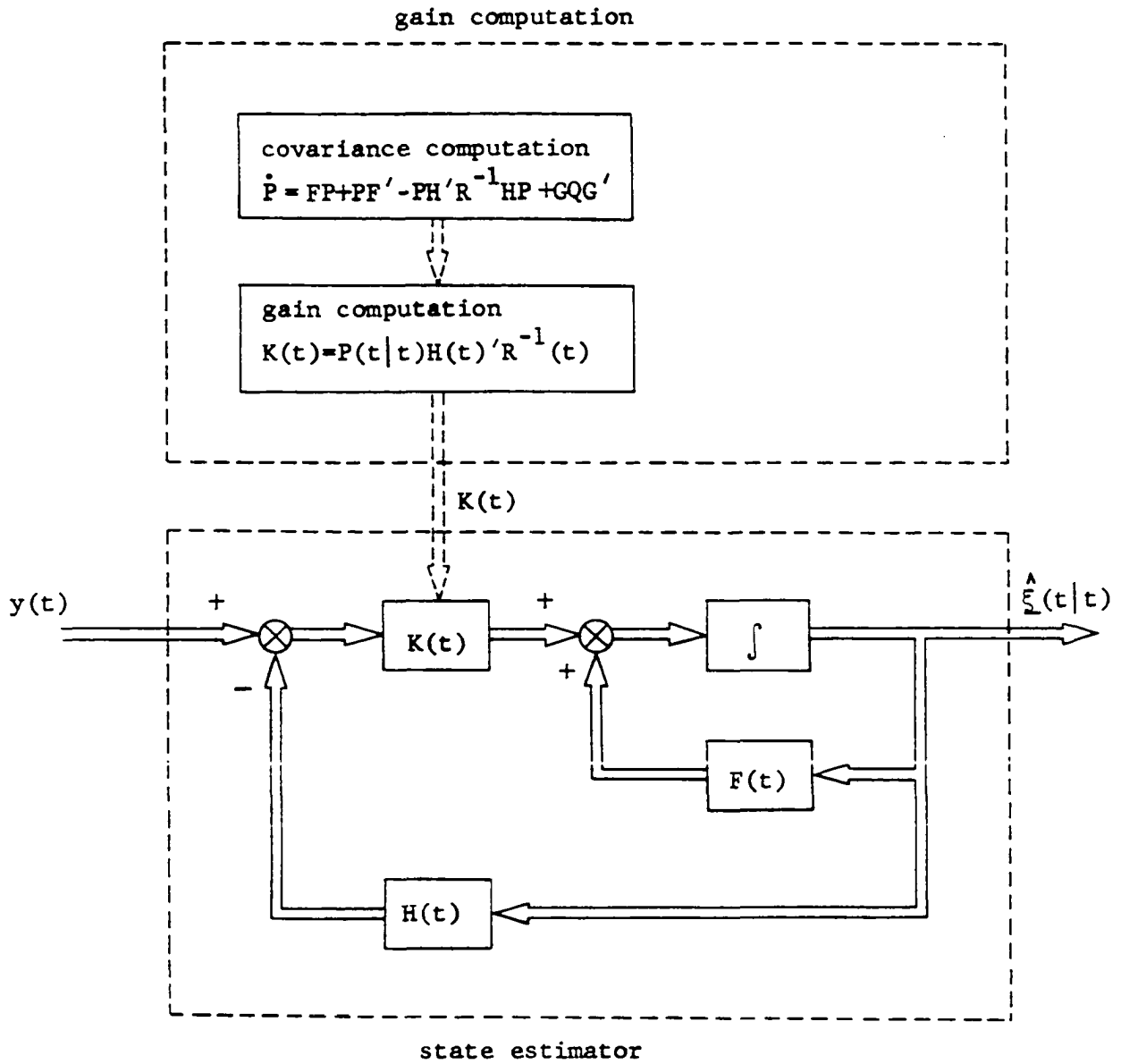


Figure 2.1. A block diagram of Kalman-Bucy filter

where

\underline{y}_n is a k_y measurement vector at time $t = nT$,

\underline{v}_n is a k_y measurement noise vector with

$$E\{\underline{v}_n\} = \underline{0}, \quad \text{for every } n,$$

$$E\{\underline{v}_m \underline{v}_n'\} = \begin{cases} R_n & \text{if } n = m, \\ 0 & \text{if } n \neq m. \end{cases}$$

Here R_n is a $k_y \times k_y$ symmetric nonnegative-definite matrix,
and

H_n is a $k_y \times k_x$ measurement matrix.

It is also assumed that the process noise \underline{u}_n and the measurement noise \underline{v}_n are uncorrelated. That is

$$E\{\underline{u}_m \underline{v}_n'\} = 0 \quad \text{for every } m \text{ and } n. \quad (2-11)$$

Then an estimate of ξ_n given observations \underline{y}_i from $i = 0$ up to m is denoted by $\hat{\xi}_n|_m$. The notation with regard to prediction, filtering, and smoothing is same here as that used for the continuous filter.

Now the optimum estimator $\hat{\xi}_n|_m$ of ξ_n which minimizes

$$E\{\|\hat{\xi}_n|_m - \xi_n\|^2\} \quad (2-12)$$

is

$$\hat{\xi}_n|_m = E\{\xi_n | \underline{y}_i, i = 0, 1, 2, \dots, m\}. \quad (2-13)$$

Then, with a priori information on initial conditions $\hat{\xi}_0|_{-1}$ and $P_0|_{-1}$, the optimal estimator $\hat{\xi}_n|_n$ of the state vector ξ_n is given by the discrete Kalman filter algorithm which is recursively defined as follows:

- 1) optimum gain

$$K_n = P_{n|n-1} H_n' [H_n P_{n|n-1} H_n' + R_n]^{-1} \quad (2-14)$$

- 2) a posteriori estimate

$$\hat{\xi}_n|_n = \hat{\xi}_n|_{n-1} + K_n [y_n - H_n \hat{\xi}_n|_{n-1}] \quad (2-15)$$

- 3) a posteriori error covariance

$$P_{n|n} = [I - K_n H_n] P_{n|n-1} \quad (2-16)$$

- 4) a priori estimate

$$\hat{\xi}_{n+1}|_n = F_n \hat{\xi}_n|_n \quad (2-17)$$

- 5) a priori error covariance

$$P_{n+1}|_n = F_n P_{n|n} F_n' + G_n Q_n G_n' \quad (2-18)$$

where

K_n is a $k_y \times k_\xi$ gain matrix for incorporating y_n into the estimate of ξ_n ,

$P_{n|n}$ is a $k_\xi \times k_\xi$ a posteriori covariance matrix that is the covariance of the error in estimating ξ_n based on the knowledge of $y_0, y_1, \dots, y_{n-1}, y_n$, and

$P_{n+1|n}$ is a $k_\xi \times k_\xi$ a priori covariance matrix which is the covariance of the error in estimating ξ_{n+1} based on the knowledge of $y_0, y_1, \dots, y_{n-1}, y_n$.

The matrix block diagram of the discrete Kalman filter is shown in Figure 2.2.

As it can be seen from the recursive equations above, the gain matrix K_n , the a posteriori covariance matrix $P_{n|n}$, and the a priori covariance matrix $P_{n+1|n}$ can be obtained for all possible n irrespective of any measurements. The models of the dynamic system, which are the signal process and the measurement process, have to be known completely. It can be seen that the discrete Kalman filter is a recursive estimator; hence it processes the measurements as they are generated in real time without any growing memory problem. Thus, it is easy to implement on a digital computer for on-line estimation.

C. Stability of Kalman Filter

The search for conditions under which optimality implies various forms of stability is the central problem of filtering. A uniform asymptotic stability of the optimum filter is an indispensable requirement.

A theorem in Kalman (20) denotes that if the dynamic system (2-1) and (2-2) is uniformly completely observable and uniformly completely controllable, then the corresponding Kalman-Bucy filter is uniformly asymptotically stable. When this system is time fixed, then uniform conditions among the above can be omitted.

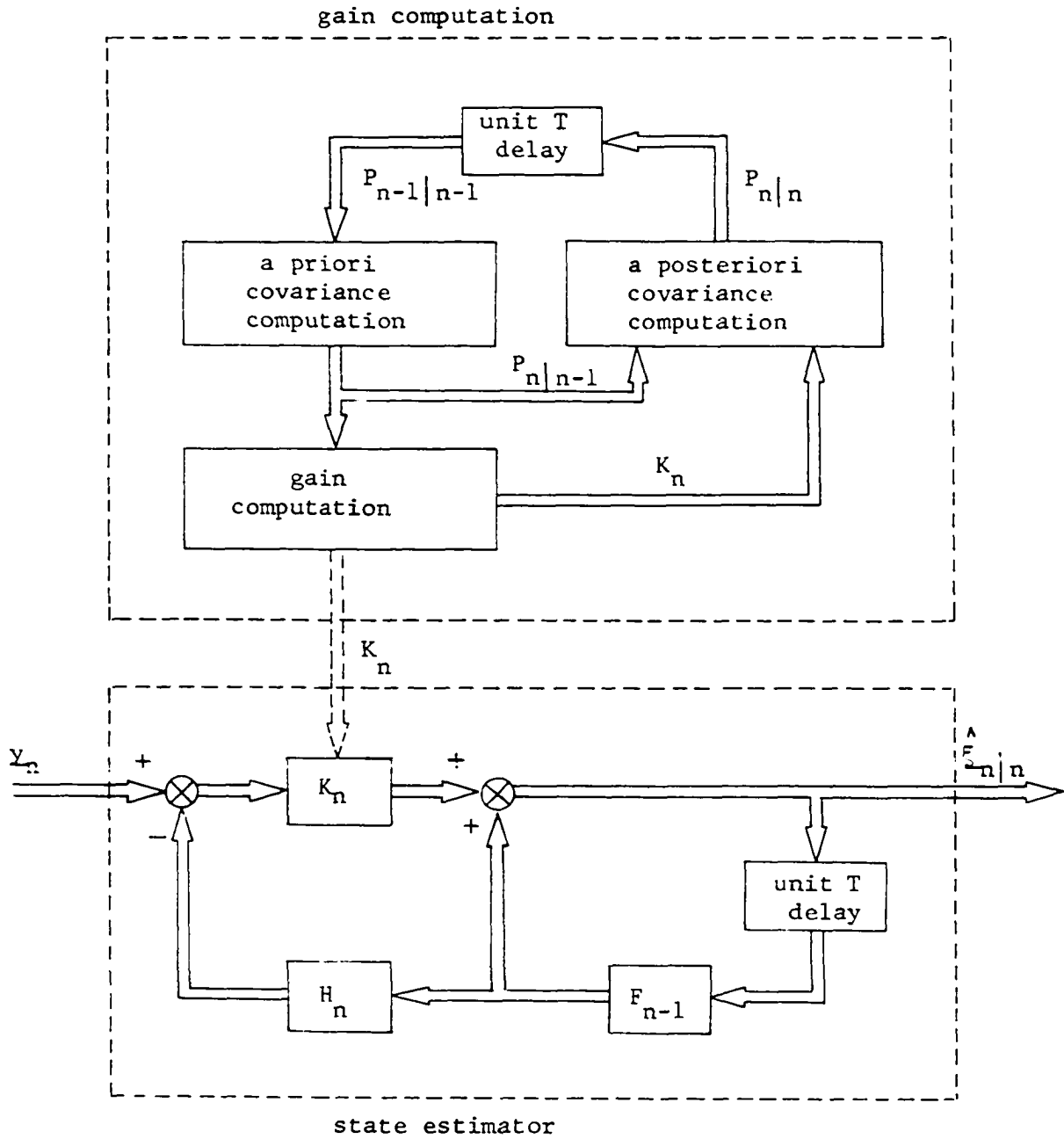


Figure 2.2. A block diagram of discrete Kalman filter

Sorenson (34) discusses the conditions in the discrete Kalman filter. In his work, a decomposition property is exhibited that permits the derivation of upper and lower bounds upon the covariance matrix.

In this section, conditions for a uniform asymptotic stability only are shown.

Completely observable matrices are defined first by

$$O(t_0, t) = \int_{t_0}^t \phi(\lambda, t)' H(\lambda)' R^{-1}(\lambda) H(\lambda) \phi(\lambda, t) d\lambda \quad (2-19)$$

for some $t > t_0$ in a continuous system, and

$$O_{m,n} = \sum_{i=m}^n \phi(m-1, i)' \underline{H}_i' R_i^{-1} \underline{H}_i \phi(m-1, i) \quad (2-20)$$

for some $n > m$ in a discrete system,

where $\phi(\lambda, t)$ is a system transition matrix and

$\phi(i, m)$ is a discrete notation for $\phi(iT, mT)$.

If these $O(t_0, t)$ and $O_{m,n}$ are positive definite, then the corresponding filters are called completely observable.

Completely controllable matrices are defined as

$$C(t_0, t) = \int_{t_0}^t \phi(t, \lambda) G(\lambda) Q(\lambda) G(\lambda)' \phi(t, \lambda)' d\lambda \quad (2-21)$$

for some $t > t_0$ in a continuous system, and

$$C_{m,n} = \sum_{i=m}^n \phi(i, m-1) G_i Q_i G_i' \phi(i, m-1)' \quad (2-22)$$

for some $n > m$ in a discrete system.

If these $C(t_0, t)$ and $C_{m, n}$ are positive definite, then the corresponding filters are called completely controllable.

A filter is said to be uniformly completely observable if there exist fixed positive constants a_0 , b_0 , and τ_0 so that

$$0 < a_0 I \leq O(t - \tau_0, t) \leq b_0 I \quad (2-23)$$

for all t in a continuous system, or if there exist fixed positive constants a_0 , b_0 , and n_0 so that

$$0 < a_0 I \leq O_{n - n_0, n} \leq b_0 I \quad (2-24)$$

for all n in a discrete filter.

And a filter is called uniformly completely controllable if there exist fixed positive constants a_c , b_c , and τ_c so that

$$0 < a_c I \leq C(t - \tau_c, t) \leq b_c I \quad (2-25)$$

for all t in a continuous filter, or if there exist fixed positive constants a_c , b_c , and n_c so that

$$0 < a_c I \leq C_{n - n_c, n} \leq b_c I \quad (2-26)$$

for all n in a discrete system.

Now suppose the dynamical system, (2-1) and (2-2) in the continuous case or (2-9) and (2-10) in the discrete case, is uniformly completely observable and uniformly completely controllable. Then the optimal filter, (2-6) to (2-8) in the continuous case or (2-14) to (2-18) in the discrete case, is called uniformly asymptotically stable.

The uniform asymptotic stability of the optimal filter is an indispensable requirement in the filter design. If the optimal filter does not satisfy this condition, then a bounded input may result in an unbounded output and hence a small bias error can ruin the performance of the filter.

III. CLASSICAL DESIGN OF SAMPLED-DATA DIGITAL FILTER

The synthesis of a sampled-data digital filter can be approached from two directions: time-domain design, and frequency-domain design. Time-domain design technique begins with unit impulse response requirement and finds coefficients of the sampled-data filter transfer function such that its impulse response function approaches, in some sense, the given output function over a finite interval of time. The use of the frequency-domain approach in filter theory and design is well known to most filter designers. This method uses the z-transform calculus. Since much information is available on continuous filter design, a useful approach to digital filter design involves finding a set of difference equations having a system function with delay operators which resembles the known analog system function.

A technique for doing this is the impulse invariant approach. By this, it is meant that the discrete response to an impulse function of the derived digital filter will be the samples of the continuous impulse response of the given analog filter. Another technique uses conformal mapping to transform a sampled-data digital filter design problem into a continuous filter design problem, for which a vast body of design literature exists. This technique is referred to as a bilinear transform although other transformations have sometimes been used. Finally, a technique referred to as a matched transform makes use of poles and zeros matched to those of the corresponding continuous filter.

When the same filtering requirements can be adequately met by various digital filters, the choice among them depends on the speed and size of execution of a computer program which performs difference equations. An important factor in this speed is the number of multiplications. Some digital filters are able to meet essentially the same requirements as others with substantially fewer multiplications per output sample, and these are to be preferred. It is important to stress that speed of execution is the main limiting factor in the realization of digital filters.

A. General Considerations

The digital filter development is based upon the behavior of the linear analog filter. The linear analog filter is characterized by its impulse response, $w(\tau, t)$, which describes the output of the filter for all $t \geq 0$ in response to an impulse input applied at time τ . The output of the filter, $x(t)$, in response to some arbitrary input, $y(t)$, is given by the convolution integral

$$x(t) = \int_0^t w(\lambda, t)y(\lambda)d\lambda \quad (3-1)$$

where zero initial conditions are assumed.

Now suppose the interest is in the output of this filter only at regularly spaced time intervals, T seconds apart. Then the sampled output at the end of n th sampling interval is

$$\begin{aligned}
 x(nT) &= \int_0^{nT} \omega(\lambda, nT)y(\lambda)d\lambda \\
 &= \sum_{m=0}^{n-1} \int_{mT}^{(m+1)T} \omega(\lambda, nT)y(\lambda)d\lambda.
 \end{aligned} \tag{3-2}$$

As a notation, the inside term of the integral is represented as $g(\lambda)$. Then this $g(\lambda)$ and all its derivatives will be assumed to exist in an interval from mT to $(m+1)T$ including $\lambda = mT$. Therefore, $g(\lambda)$ can be expressed as a Taylor series, in the form

$$g(\lambda) = \sum_{i=0}^{\infty} \frac{g^{(i)}(mT)}{i!} (\lambda - mT)^i. \tag{3-3}$$

Then the equation (3-2) becomes

$$x(nT) = \sum_{m=0}^{n-1} \sum_{i=0}^{\infty} \frac{1}{i!} \left\{ \frac{\partial^i}{\partial \lambda^i} \omega(\lambda, nT)y(\lambda) \right\}_{\lambda=mT} \int_{mT}^{(m+1)T} (\lambda - mT)^i d\lambda. \tag{3-4}$$

First, suppose that T is sufficiently small so that there is negligible error in representing equation (3-4) by the first term in its Taylor's series expansion as

$$x(nT) = T \sum_{m=0}^{n-1} \omega(mT, nT)y(mT). \tag{3-5}$$

The approximation error is given by the sum of the higher-order terms from Taylor's series expansion as

$$\Delta x(nT) = \sum_{m=0}^{n-1} \sum_{i=1}^{\infty} \frac{T^{i+1}}{(i+1)!} \left\{ \frac{\partial^i}{\partial \tau^i} \omega(\tau, nT)y(\tau) \right\}_{\tau=mT}. \tag{3-6}$$

It can be noticed that as the sampling rate becomes extremely fast, the error expansion given by equation (3-6) approaches zero, and therefore the performance of the sampled processor approaches that of the continuous processor. Reflecting on the significance of equation (3-5), it can be seen that the output of the linear analog network at discrete-time instants can be approximated by a series of nonrecursive calculations on samples of the input signal as illustrated in Figure 3.1. The nature of the sampled-data digital filter is clear though complicated. The gain applied to the filter input $y(t)$ is required to change in each feed-forward path with each successive sample of the input signal. As n , that is time, increases, the required length of the sampled-data digital filter increases. As n approaches infinity, so does the filter size. Each word time delay is a one word memory whose capacity in bits is determined by the number of bits of accuracy used to represent $y(nT)$. If $y(nT)$ is scaled in amplitude and quantized so as to be represented by ℓ_w bits, then the each unit time delay block is an ℓ_w -bits memory. If the length of the filter input $y(nT)$ sequence is limited to m samples, that is $y(T)$, $y(2T)$, ..., $y(mT)$, the length of the memory requirement is correspondingly reduced by $\ell_w \times m$ bits.

Alternately, if, after some time mT , the amplitude of the weighting function falls and remains below some level of interest σ , that is

$$\left| w[(n-i)T, nT] \right|_{i > m} \leq \sigma, \quad (3-7)$$

the length of the filter may be truncated to a finite length, in this notation, after the storage of m samples. It is noticed that, for each

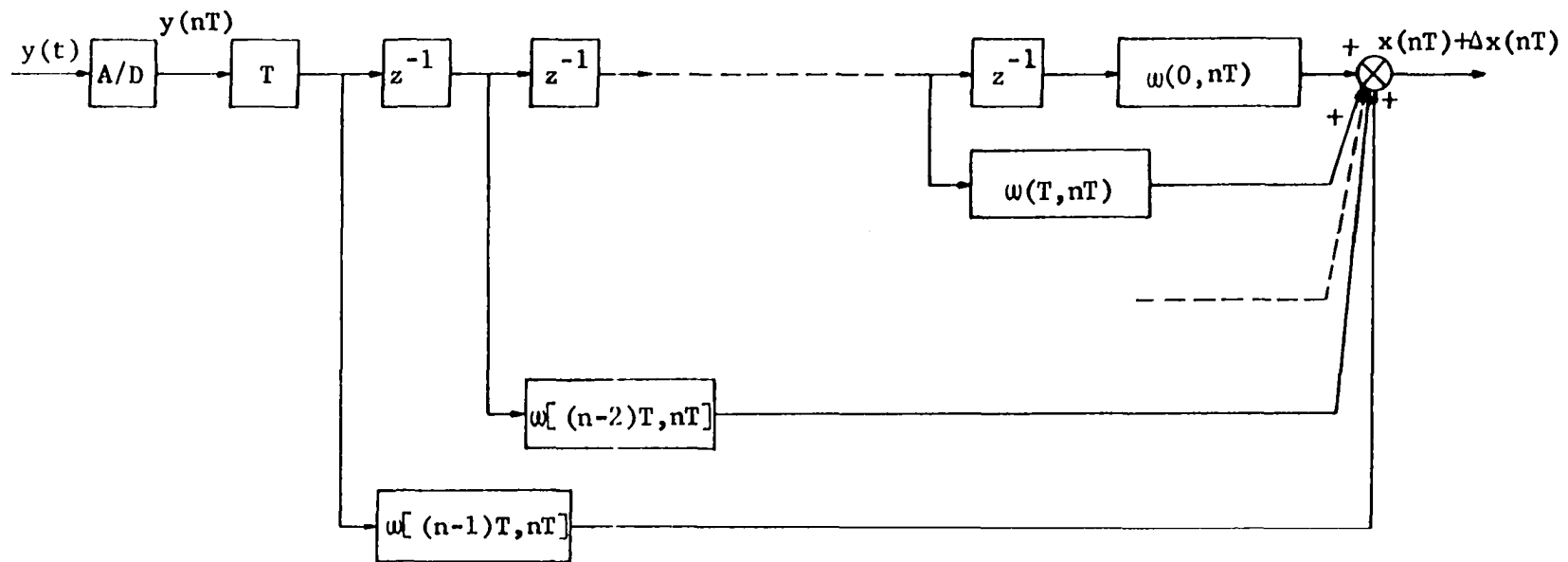


Figure 3.1. Nonrecursive time-varying sampled-data digital filter where $\omega(mT, nT)$, $m = 0, 1, \dots, n-1$, are time-varying gains

word time of delay required in this filter, one multiplication is also required. Again, in general, there are variable coefficient multipliers. For obvious reasons, this is often called both a taped delay-line filter, or for untruncated case, a growing memory filter, because the memory requirement increases linearly with the number of samples taken of the input signal.

If the continuous filter is time invariant, then, referring to Brown and Nilsson (5), the weighting function with the age variable τ becomes $\omega(\tau, t) = \omega(t - \tau)$. In which case the convolution integral and the convolution summation becomes

$$\begin{aligned} x(t) &= \int_0^t \omega(t - \lambda) y(\lambda) d\lambda \\ &= \int_0^t \omega(\lambda) y(t - \lambda) d\lambda \end{aligned} \quad (3-8)$$

and

$$\begin{aligned} x(nT) &= T \sum_{m=0}^{n-1} \omega(nT - mT) y(mT) \\ &= T \sum_{m=0}^{n-1} \omega(mT) y(nT - mT), \end{aligned} \quad (3-9)$$

respectively. A fixed-parameter digital structure for a truncated impulse response is illustrated in Figure 3.2.

Three alternate procedures will be discussed for developing filter input-output relations for various z-transforms as frequency-domain methods. Sampled-data digital filter transfer function will be implemented by digital techniques, starting with fixed-parameter continuous filter impulse response function in the following.

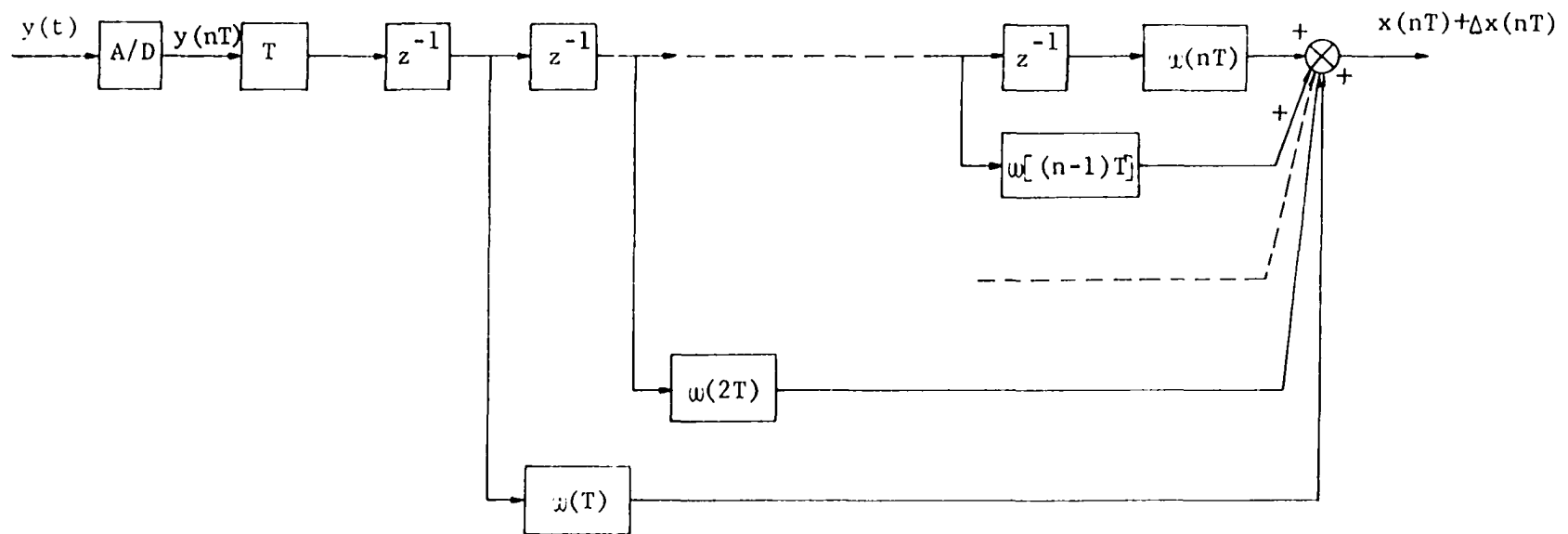


Figure 3.2. Nonrecursive time-fixed sampled-data digital filter structure where $\omega(mT)$, $m = 0, 1, 2, \dots, n$, are fixed gains

B. Impulse Invariant z-transform

Consider an impulse response function of a continuous filter in time-invariant form as $w(\tau)$. The transfer function of this continuous filter is given by

$$W(s) = \mathcal{L}\{w(\tau)\}. \quad (3-10)$$

Now $w(\tau)$ is to be sampled at regular intervals, T seconds. This sampled waveform can be thought of as a series of impulse functions, each of magnitude $w(nT)$ at time $\tau = nT$. The sampled waveform can be represented by

$$w^*(\tau) = \sum_{n=0}^{\infty} w(nT)\delta(\tau-nT) \quad (3-11)$$

which represents impulse modulation of $w(\tau)$. In the frequency domain, this sampled impulse response is simply the Laplace transform of equation (3-11)

$$W^*(s) = \sum_{n=0}^{\infty} w(nT)e^{-nTs} \quad (3-12)$$

where $*$ represents the sampled-data form of $W(s)$.

The first form of the z-transform can be defined from using the unit delay operation

$$z^{-1} \triangleq e^{-sT} \quad (3-13)$$

as

$$W(z) = \sum_{n=0}^{\infty} w(nT)z^{-n}. \quad (3-14)$$

The notation on the left-hand side of the equation (3-14) is not strictly correct, but it conforms to general usage.

The second form of the z-transform can be defined by performing convolution integral of $W(s)$ with the impulse response train. Assume for simplicity in this form that $W(s)$ is a proper rational function with all poles in the left-half s-plane. Since the impulse response train is

$$\sum_{n=0}^{\infty} e^{-nTs} = \frac{1}{1 - e^{-sT}}, \quad (3-15)$$

the second representation for $W^*(s)$ is obtained as follows.

$$W^*(s) = \frac{1}{2\pi j} \int_C \frac{W(s-\lambda)}{1 - e^{-\lambda T}} d\lambda \quad (3-16)$$

where the contour C of the integration extends from the bottom to the top of the complex plane to the left of the singularities of the impulse train and to the right of the singularities of $W(\lambda)$. Then the contour of integration can be closed to the right yielding

$$W^*(s) = \frac{1}{T} \sum_{n=-\infty}^{\infty} W\left(s + j\frac{2\pi n}{T}\right). \quad (3-17)$$

See Lindorff (24) for this derivation using a Fourier transform. From the above it is apparent that $W^*(s)$ is periodic in

$$\omega_0 = 2\pi f_0 = \frac{2\pi}{T}, \quad (3-18)$$

which is called the sampling frequency. The consequence is that the frequency responses of sampled and continuous filters are the same only if $W(s)$ is band limited, that is

$$W(j\omega) = 0 \quad \text{for } |\omega| > \frac{\omega_0}{2}. \quad (3-19)$$

The third important representation of $W^*(s)$ is obtained by choosing different contour of integration. When the contour in equation (3-16) is closed to left and contains all poles of $W(s)$ in the left-half s -plane, then the equation (3-16) becomes

$$W^*(s) = \sum \left\{ \text{residues} \frac{W(\lambda)}{1 - e^{-(s-\lambda)T}} \right\}_{\text{poles of } W(\lambda)} . \quad (3-20)$$

As $W(s)$ is assumed to be a proper rational function, the given transfer function $W(s)$ is written in partial fraction form, with only first-order poles and complex-conjugate second-order poles, such as

$$W(s) = \sum_{i=1}^{m_1} \frac{b_i}{s + a_i} + \sum_{i=1}^{m_2} \frac{(s+c_i)e_i + d_i f_i}{(s+c_i)^2 + d_i^2} . \quad (3-21)$$

Then each term in the above equation (3-21) is represented according to the equation (3-20) with the gain compensation factor T as follows.

$$\frac{b_i}{s + a_i} \longrightarrow \frac{b_i T}{1 - e^{-a_i T} z^{-1}} \quad (3-22)$$

$$\frac{(s+c_i)e_i + d_i f_i}{(s+c_i)^2 + d_i^2} \longrightarrow \frac{e_i T - T e^{-c_i T} (e_i \cos d_i T - f_i \sin d_i T) z^{-1}}{1 - 2e^{-c_i T} (\cos d_i T) z^{-1} + e^{-2c_i T} z^{-2}} \quad (3-23)$$

where T is a sampling interval and $z^{-1} = e^{-sT}$ is a unit delay operator.

If following notations are used,

$$A_i = e^{-a_i T} ,$$

$$B_i = b_i T ,$$

$$\begin{aligned}
C_i &= 2e^{-c_i T} \cos d_i T, \\
D_i &= e^{-2c_i T}, \\
E_i &= e_i T, \text{ and} \\
F_i &= T e^{-c_i T} (e_i \cos d_i T - f_i \sin d_i T), \tag{3-24}
\end{aligned}$$

then the complete sampled-data digital filter transfer function of equation (3-21) becomes

$$W(z) = \sum_{i=1}^{m_1} \frac{B_i}{1 - A_i z^{-1}} + \sum_{i=1}^{m_2} \frac{E_i - F_i z^{-1}}{1 - C_i z^{-1} + D_i z^{-2}}. \tag{3-25}$$

This function is realized in a block diagram form in Figure 3.3.

In general, this representation gives excellent results when applied to all-pole low-pass and band-pass filters. Some applications show that the impulse invariant transform should be used only suitable band-limited function. Fortunately the design of band-stop and high-pass sampled-data digital filters can be accomplished without resorting to a band-limiting low-pass filters by using either the bilinear z-transform or the matched z-transform.

C. Bilinear z-transform

The design specification of many filter transfer functions requires the realization of a given response characteristic in the frequency domain, but does not demand any particular impulse or step response characteristic. Specification of frequency response only permits the bilinear z-transform to be used to realize sampled-data digital filters that have relatively

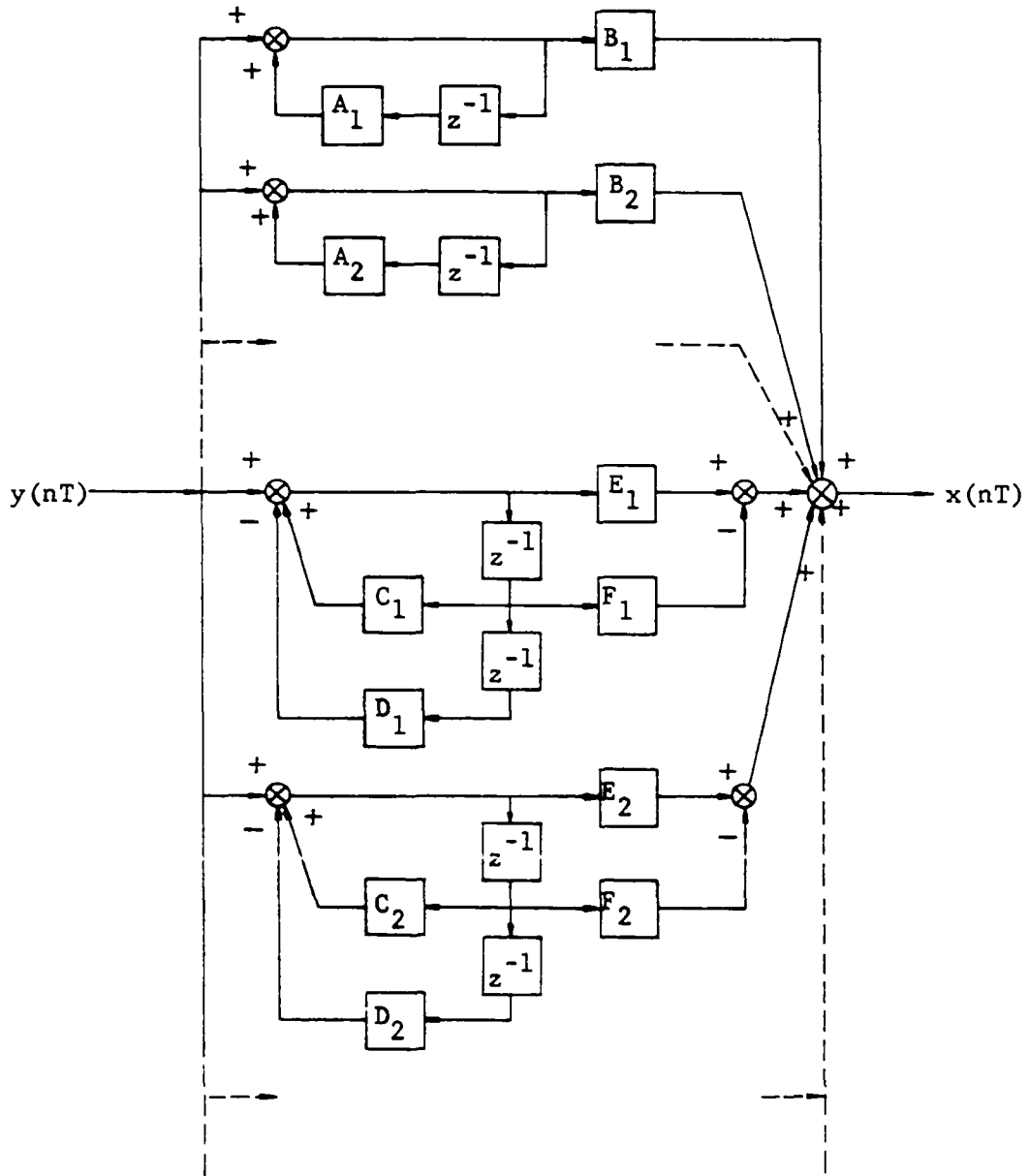


Figure 3.3. Realization of sampled-data digital filter transfer function in parallel form

constant magnitude pass-band and stop-band characteristic. This transformation, being an algebraic one, may be applied to either partial fraction expansion representation or the rational fraction form. This transformation is

$$s \longrightarrow \frac{2}{T} \tanh \frac{sT}{2} = \frac{2}{T} \frac{1 - e^{-sT}}{1 + e^{-sT}} = \frac{2}{T} \frac{1 - z^{-1}}{1 + z^{-1}} . \quad (3-26)$$

The bilinear z-transform yields the following relations for real and complex functions.

a. real pole in partial fraction expansion

$$\frac{b_i}{s + a_i} \longrightarrow \frac{B_i (1 + z^{-1})}{1 - A_i z^{-1}} \quad (3-27)$$

where

$$A_i = \frac{1 - \frac{a_i T}{2}}{1 + \frac{a_i T}{2}}$$

and

$$B_i = \frac{b_i T}{2(1 + \frac{a_i T}{2})} .$$

b. complex pole in partial fraction expansion

$$\frac{(s + c_i)e_i + d_i f_i}{(s + c_i)^2 + d_i^2} \longrightarrow \frac{(1 + z^{-1})(E_i - F_i z^{-1})}{1 - C_i z^{-1} + D_i z^{-2}} \quad (3-28)$$

where

$$C_i = \frac{2}{\Delta} \left\{ 1 - \left(\frac{c_i T}{2}\right)^2 - \left(\frac{d_i T}{2}\right)^2 \right\}$$

$$D_i = \frac{1}{\Delta} \left\{ \left(1 - \frac{c_i T}{2}\right)^2 + \left(\frac{d_i T}{2}\right)^2 \right\}$$

$$E_i = \frac{T}{2\Delta} \left\{ \left(1 + \frac{c_i T}{2}\right) e_i + \frac{d_i T}{2} f_i \right\}$$

$$F_i = \frac{T}{2\Delta} \left\{ \left(1 - \frac{c_i T}{2}\right) e_i - \frac{d_i T}{2} f_i \right\}$$

and
$$\Delta = \left(1 + \frac{c_i T}{2}\right)^2 + \left(\frac{d_i T}{2}\right)^2 .$$

Since the term $(1 + z^{-1})$ is common, it may be factored out of the fractions may be reduced to proper forms plus a constant term, as

$$\frac{B_i (1 + z^{-1})}{1 - A_i z^{-1}} = -\frac{B_i}{A_i} + \frac{B_i (1 + \frac{1}{A_i})}{1 - A_i z^{-1}} \quad (3-29)$$

and

$$\frac{(1 + z^{-1})(E_i - F_i z^{-1})}{1 - C_i z^{-1} + D_i z^{-2}} = -\frac{F_i}{D_i} + \frac{(E_i + \frac{F_i}{D_i}) + \{E_i - F_i (1 + \frac{C_i}{D_i})\} z^{-1}}{1 - C_i z^{-1} + D_i z^{-2}} . \quad (3-30)$$

After combining all of these terms, the sampled-data digital filter transfer function may now be put in the form given by equation (3-25) which is added by a constant term.

c. real factor in rational form

$$s + a_i \longrightarrow \frac{G_i (1 - A_i z^{-1})}{1 + z^{-1}} \quad (3-31)$$

where
$$A_i = \frac{1 - \frac{a_i T}{2}}{1 + \frac{a_i T}{2}}$$

and
$$G_i = \frac{2}{T} \left(1 + \frac{a_i T}{2}\right) .$$

d. complex factor in rational form

$$(s + c_i)^2 + d_i^2 \longrightarrow \frac{H_i(1 - C_i z^{-1} + D_i z^{-2})}{(1 + z^{-1})^2}, \quad (3-32)$$

where $C_i = \frac{2}{\Delta} \left\{ 1 - \left(\frac{c_i T}{2}\right)^2 - \left(\frac{d_i T}{2}\right)^2 \right\}$

$$D_i = \frac{1}{\Delta} \left\{ \left(1 - \frac{c_i T}{2}\right)^2 + \left(\frac{d_i T}{2}\right)^2 \right\}$$

$$H_i = \left(\frac{2}{T}\right)^2 \left\{ \left(1 + \frac{c_i T}{2}\right)^2 + \left(\frac{d_i T}{2}\right)^2 \right\}$$

and $\Delta = \left(1 + \frac{c_i T}{2}\right)^2 + \left(\frac{d_i T}{2}\right)^2.$

Therefore, the resulting sampled-data digital filter transfer function takes the following form, if the continuous filter transfer function is given as

$$W(s) = \frac{\prod_{\ell=1}^{m_2} (s + b_\ell) \prod_{n=1}^{m_4} \{(s + e_n)^2 + f_n^2\}}{\prod_{k=1}^{m_1} (s + a_k) \prod_{m=1}^{m_3} \{(s + c_m)^2 + d_m^2\}}, \quad (3-33)$$

then

$$W(z) = G_d \frac{\prod_{\ell=1}^{m_2} (1 - B_\ell z^{-1}) \prod_{n=1}^{m_4} (1 - E_n z^{-1} + F_n z^{-2})}{\prod_{k=1}^{m_1} (1 - A_k z^{-1}) \prod_{m=1}^{m_3} (1 - C_m z^{-1} + D_m z^{-2})} (1 + z^{-1})^{m_5}, \quad (3-34)$$

where $m_5 = (m_1 - m_2) + 2(m_3 - m_4)$ and G_d is gain change in the transformation. Realization of the individual first-order and second-order terms is similar to the recursive structure and is illustrated in Figure 3.4. A block diagram showing the factored rational sampled-data digital filter transfer function is shown in Figure 3.5.

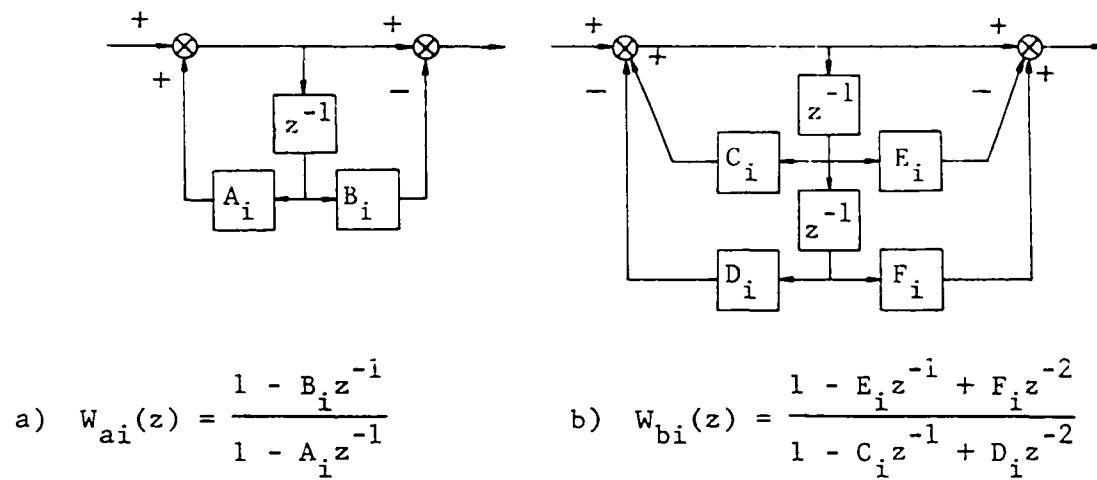


Figure 3.4. Realization of first-order and second-order terms

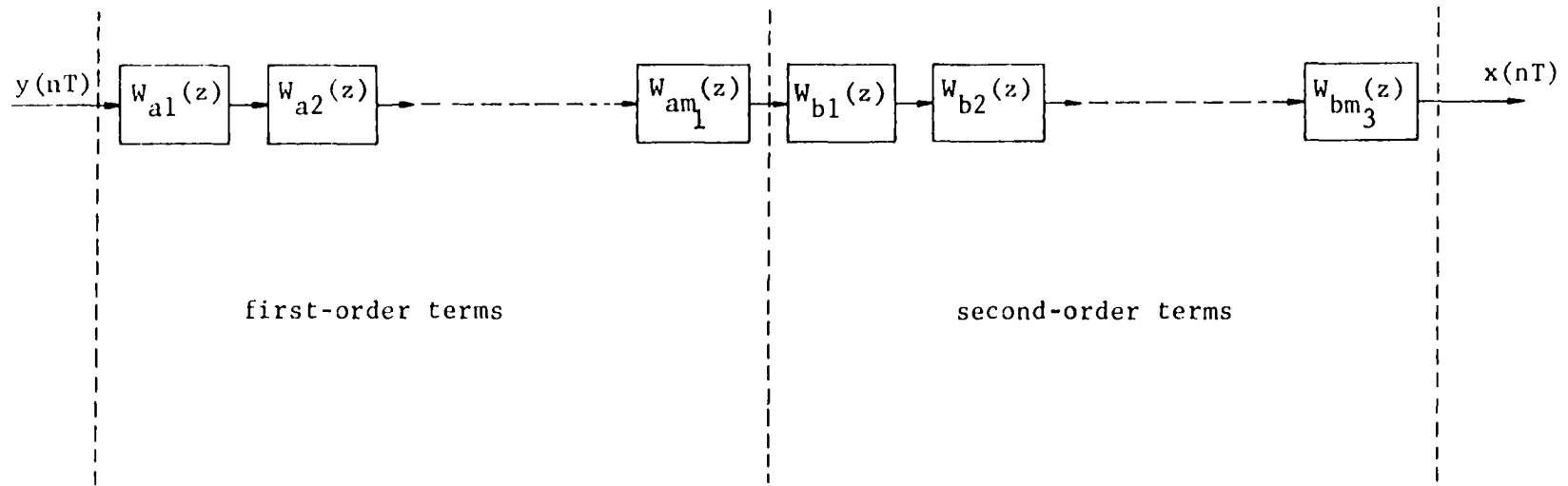


Figure 3.5. Realization of sampled-data digital filter transfer function in cascade form

Because the frequency axis is distorted by the bilinear z-transform, care must be taken when designing filters with critical frequencies near the half-sampling frequency. The critical frequencies will map according to the relationship

$$f_c = \frac{1}{\pi T} \tan \pi f_d T \quad (3-35)$$

where f_c is a critical frequency in the continuous design and f_d is a critical frequency in the discrete design. This relationship is plotted in Figure 3.6. Fortunately, most continuous filters to be approximated are often initially designed with the aid of a frequency band transformation. It is, therefore, a simple matter to change the critical frequencies in the band transformation to yield a prewarped continuous design. This new design will yield a sampled-data digital filter with the desired critical frequencies when the bilinear z-transform is used.

A disadvantage of the bilinear z-transform is the frequency scale distortion it introduces into the digital filter magnitude response characteristic. For narrow band-width filters and even with prewarping or predistortion, the bilinear z-transform may not yield a sampled-data digital filter with the desired magnitude response characteristic. This leaves the matched z-transform as the third alternative for realizing a sampled-data digital filter from a continuous design.

D. Matched z-transform

The matched z-transform is another useful transform for designing sampled-data digital filters from continuous filters. This transformation

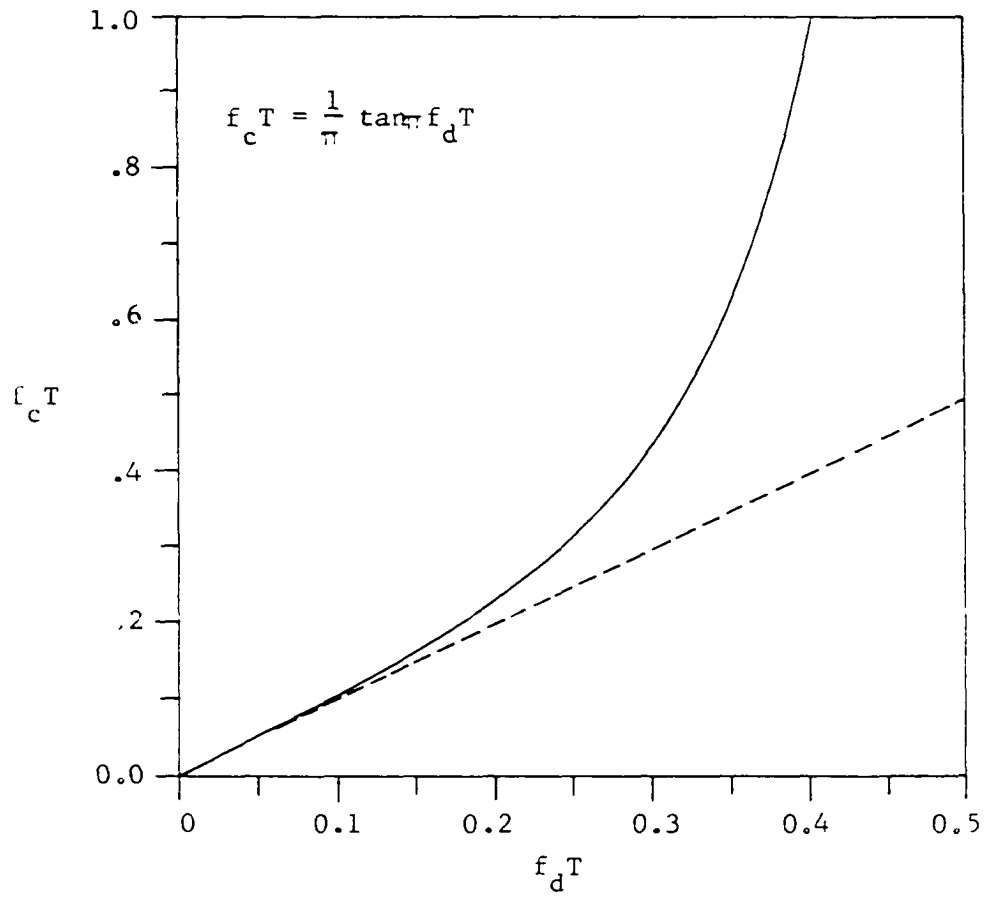


Figure 3.6. Nonlinear frequency scale imposed by bilinear z-transform

generates a digital filter transfer function with poles and zeros matched to those of continuous filter transfer function. The mapping transforms for the poles and zeros of the continuous filter transfer function given by

$$s \longrightarrow e^{sT} = z. \quad (3-36)$$

Real poles and zeros are transformed according to

$$s + a_i \longrightarrow \frac{1 - e^{-a_i T} z^{-1}}{T}. \quad (3-37)$$

While complex poles and zeros in the second-order filter yield terms of the form

$$(s + \alpha_i)^2 + \beta_i^2 \longrightarrow \frac{1 - 2e^{-\alpha_i T} (\cos \beta_i T) z^{-1} + e^{-2\alpha_i T} z^{-2}}{T^2}. \quad (3-38)$$

The transfer function for the sampled-data digital filter in general form of the matched z-transform which is z-transform of equation (3-33) becomes

$$W(z) = \frac{\prod_{\ell=1}^{m_2} (1 - B_\ell z^{-1}) \prod_{n=1}^{m_4} (1 - E_n z^{-1} + F_n z^{-2})}{\prod_{k=1}^{m_1} (1 - A_k z^{-1}) \prod_{m=1}^{m_3} (1 - C_m z^{-1} + D_m z^{-2})} T^{m_5} \quad (3-39)$$

where

$$A_k = e^{-a_k T},$$

$$B_\ell = e^{-b_\ell T},$$

$$C_m = 2e^{-c_m T} \cos d_m T,$$

$$D_m = e^{-2d_m T},$$

$$E_n = 2e^{-e_n T} \cos f_n T,$$

$$F_n = e^{-e_n T}$$

and T^{m_5} is chosen to adjust the insertion level gain factor with

$$m_5 = (m_1 - m_2) + 2(m_3 - m_4).$$

Note that the poles of the filter transfer function are the same as those derived by the impulse invariant z-transform. However the zeros of the matched z-transform derived filter transfer function will not usually correspond to those of the impulse invariant z-transform. The result of this fact is that the matched z-transform may be used to obtain useful designs for high-pass and band-stop filters. There are, however, some designs for which the matched z-transform derived filter does not give satisfactory results without employing some modification. These designs are all-pole low-pass and band-pass functions. A simple modification consisting of the insertion of zeros at the half-sampling frequency will correct an unsatisfactory design. The modification requires multiplication of the matched z-transform derived filter by the terms of the form $(1 + z^{-1})^m$, with m which is equal to the half-sampling frequency zeros desired.

In the preceding discussions, basic criteria are implied for determining that a sampled-data digital filter design is an accurate representation of a continuous design. These criteria simply require that the frequency and time response characteristics of the filter design yield the desired results.

Various methods have been demonstrated for transforming a given analog filter into a sampled-data digital filter form. As it has been seen, not

all of these techniques produce good results as desired criteria. The digital filter, as a final goal of the sampled-data digital filter, whose characteristics are determined by the structural form and accuracy specifications of the coefficients in terms of the sampled-data digital filter transfer function, does not yield the good samples at the discrete time of the continuous filter. Hence a study must be conducted to achieve a better sampled-data digital filter.

IV. EQUIVALENT KALMAN-BUCY FILTER

The basic idea behind the new approach to the sampled-data digital filter design is that a Kalman-Bucy filter can be discretized with much accuracy so that the discretized Kalman-Bucy filter has the form of the discrete Kalman filter algorithm. When the sampling interval is small, this discretization is simplified to be a routine matter.

The problem in this chapter is to develop an equivalent Kalman-Bucy filter from a given continuous filter. One difficulty of intuitive ideas is that a Kalman-Bucy filter is based on state estimation, but a classical filter is based on frequency selectivity. It can be assumed that the best estimate of states in the mean-square sense gives the best frequency distribution since the states correspond to the signal which is selected in frequency. In other words, once the states of the filter are estimated as functions of time, the filter output in frequency distribution can be calculated.

The new method begins with a continuous filter transfer function $W(s,t)$ in general form which is given as a ratio of two polynomials as the following equation (4-1):

$$W(s,t) = \frac{N(s,t)}{D(s,t)} \cdot \quad (4-1)$$

Here $N(s,t) = \beta_{k-1}(t)s^{k-1} + \beta_{k-2}(t)s^{k-2} + \dots + \beta_1(t)s + \beta_0(t)$

and $D(s,t) = s^k + \alpha_{k-1}(t)s^{k-1} + \alpha_{k-2}(t)s^{k-2} + \dots + \alpha_1(t)s + \alpha_0(t),$

where $W(s,t)$ is a Laplace transform of a continuous classical filter impulse response $w(\tau,t)$ with respect to the age variable τ . Either this transfer function $W(s,t)$ or the impulse response function $w(\tau,t)$ is assumed to be given at the beginning of the filter design.

A single input and a single output of this filter are represented as $y(t)$ and $x(t)$, respectively, which are scalar functions of time. This output of the continuous filter is supposed to be very close to the original signal process, if this can be imagined, according to the classical design.

It should be noticed that the state representation for the given classical filter needs to be carefully selected for having appropriate Kalman-Bucy equivalent filter correspondence. A block diagram corresponding to $W(s,t)$ that will be used for the new method is shown in Figure 4.1. As shown in the figure, the states are defined uniquely such that a differentiated state is a next state except the last state $\hat{\xi}_k(t|t)$. Note that the number of states to be defined is the order of the denominator of the given transfer function in equation (4-1). In this state-variable representation, the output $x(t)$ is a linear combination of the states as indicated by the coefficients of the numerator of the continuous filter transfer function.

In a matrix representation of these state variables, it can be seen that the denominator of the transfer function determines the system matrix, and the numerator reflects eventually the output equation, depending on what random variables are actually observed in the situation at hand. In matrix equation form, the state and the output equations are

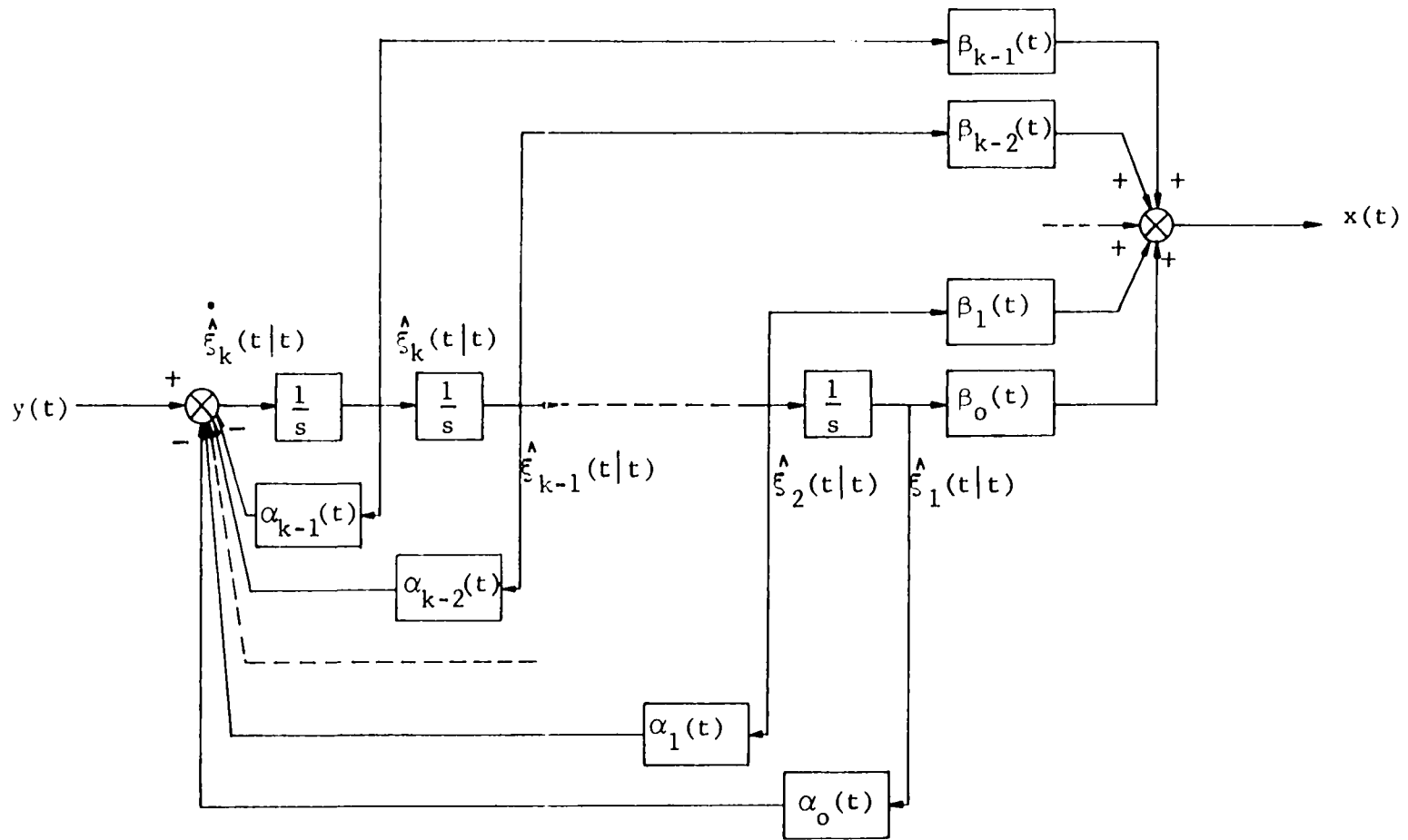


Figure 4.1. A general block diagram of a classical filter and its state representation

$$\dot{\hat{\underline{x}}}(t|t) = A(t)\hat{\underline{x}}(t|t) + \underline{B}y(t), \quad (4-2)$$

and

$$x(t) = \underline{M}(t)\hat{\underline{x}}(t|t), \quad (4-3)$$

where

$\hat{\underline{x}}(t|t)$ is a $k \times 1$ column vector representing each state, i.e.,

$$\hat{\underline{x}}(t|t) = [\hat{\xi}_1(t|t)\hat{\xi}_2(t|t) \dots \hat{\xi}_{k-1}(t|t)\hat{\xi}_k(t|t)]', \quad (4-4)$$

$A(t)$ is a $k \times k$ system matrix determined by the denominator of the filter transfer function such that

$$A(t) = \begin{bmatrix} 0 & 1 & 0 & \dots & 0 & 0 \\ 0 & 0 & 1 & \dots & 0 & 0 \\ \vdots & \vdots & \vdots & & \vdots & \vdots \\ \vdots & \vdots & \vdots & & \vdots & \vdots \\ 0 & 0 & 0 & \dots & 0 & 1 \\ -\alpha_0(t) & -\alpha_1(t) & -\alpha_2(t) & \dots & -\alpha_{k-2}(t) & -\alpha_{k-1}(t) \end{bmatrix}, \quad (4-5)$$

\underline{B} is a $k \times 1$ constant column vector which has an only unit element of k th element, i.e.,

$$\underline{B} = [0 \ 0 \ 0 \ \dots \ 0 \ 1]', \quad (4-6)$$

and

$\underline{M}(t)$ is a $1 \times k$ row vector such that it consists of the coefficients of the numerator of the filter transfer function as

$$\underline{M}(t) = [\beta_0(t)\beta_1(t)\beta_2(t) \dots \beta_{k-2}(t)\beta_{k-1}(t)]. \quad (4-7)$$

It will be noted here that $\hat{\underline{x}}(t|t)$ is defined using only the denominator of the continuous filter transfer function. The output $x(t)$ is a weighted sum of each element of $\hat{\underline{x}}(t|t)$.

New continuous filter equations which are equivalent to the equations (4-2) and (4-3), and which have the Kalman-Bucy filter structure and the output equation, are assumed and are as follows:

$$\dot{\hat{\underline{x}}}(t|t) = F(t)\hat{\underline{x}}(t|t) + \underline{K}(t)[y(t) - \underline{H}(t)\hat{\underline{x}}(t|t)] \quad (4-8)$$

$$x(t) = \underline{M}(t)\hat{\underline{x}}(t|t). \quad (4-9)$$

Here

$$\underline{K}(t) = P(t|t)\underline{H}(t)'r^{-1}(t) \quad (4-10)$$

and

$$\begin{aligned} \dot{P}(t|t) = & F(t)P(t|t) + P(t|t)F(t)' \\ & - P(t|t)\underline{H}(t)'r^{-1}(t)\underline{H}(t)P(t|t) + Q(t) \end{aligned} \quad (4-11)$$

where $F(t)$, $\underline{H}(t)$, $P(0|0)$, $P(t|t)$, $Q(t)$ and $r(t)$ are to be chosen according to equations (4-2) and (4-3); that is, the given continuous filter transfer function (4-1).

An initial attempt at the equivalent Kalman-Bucy filter will be made on the transient basis to that of the continuous classical filter. Since the equation (4-8) is designed to have the same transient behavior as the equation (4-2), the equivalent Kalman-Bucy filter initial gain should be fixed as

$$\underline{K}(0) = \underline{B}. \quad (4-12)$$

It has been proved in Kalman (20) that every solution of the covariance equation (2-8) which has a nonnegative-definite initial value converges uniformly to its equilibrium state when the stochastic process models (2-1) and (2-2) are uniformly completely observable and uniformly completely controllable. Hence the stochastic process models used for this new filter have to be uniformly completely observable and uniformly completely controllable. Then the fixed initial value $P(0|0)$ in equation (4-11) means a fixed covariance solution $P(t|t)$, and, therefore, a fixed initial value of the covariance equation means a fixed initial gain $\underline{K}(0)$.

Hence an initial covariance value $P(0|0)$ has to be fixed such that

$$\underline{B} = P(0|0)\underline{H}(0)'r^{-1}(0), \quad (4-13)$$

and this $P(0|0)$, satisfying the above equation, is an initial value of the covariance equation (4-11).

The second step in the design of the equivalent Kalman-Bucy filter is to assure steady-state equivalency to the given continuous classical filter. That is, in the steady state, the equivalent Kalman-Bucy filter has to perform the same roles as those of continuous filter. This second condition requires

$$F(t) - \underline{K}(t)\underline{H}(t) = A(t),$$

and

$$\underline{K}(t) = \underline{B}, \quad (4-14)$$

here

$$\underline{K}(t) = \bar{P}(t|t)\underline{H}(t)'r^{-1}(t),$$

where $\bar{P}(t|t)$ is an equilibrium state of the covariance equation (4-11). This $\bar{P}(t|t)$ is defined, for a given initial state $P(0|0)$ at time t_0 , as

$$\bar{P}(t|t) = \lim_{t_0 \rightarrow -\infty} P(t|t; P(0|0), t_0), \quad (4-15)$$

if this limit exists for all t .

This $\bar{P}(t|t)$ has to be solved in terms of $A(t)$, $Q(t)$, and $r(t)$, first, and then $F(t)$ and $\underline{H}(t)$ can be obtained algebraically using equations (4-14). The ways of solving this equilibrium covariance matrix are different depending on whether $A(t)$ is time-fixed or time-varying matrix. This implies $D(s,t)$ in equation (4-1) is time fixed or time varying. As can be seen from equations (4-14), if $A(t)$ is time fixed, $F(t)$ and $\underline{H}(t)$ can be chosen as time-invariant matrices. In the case of $F(t)$ and $\underline{H}(t)$ being time invariant, the stochastic process model described by (2-1) and (2-2) is called a fixed-parameter system. The covariance equation (4-11) becomes $k(k+1)/2$ algebraic equations in the steady-state fixed-parameter system, and $\bar{P}(t|t)$ is solved for easily. In the case that $F(t)$ and $\underline{H}(t)$ are not time fixed by a time-varying $A(t)$ matrix, the stochastic process model, (2-1) and (2-2), is known as the time-varying parameter system. In this system, the covariance equation (4-11) becomes $k(k+1)/2$ first-order differential equations even in the steady state, and the equilibrium solution of the covariance equation is complex, but can be solved. These problems will be discussed in subsequent sections.

Notice should be taken that an attempt has been made to choose the gain $\underline{K}(t)$ of the equivalent Kalman-Bucy filter to be identical to the vector \underline{B} in both transient and steady states as in equations (4-12)

and (4-14), and, therefore, the parameters as $Q(t)$ and $r(t)$ do not affect this gain directly.

Also it should be noted that the output row vector $\underline{M}(t)$ obtained from $N(s,t)$ of the equation (4-1) is only connected to the output equation (4-9) of the equivalent Kalman-Bucy filter. Hence the numerator $N(s,t)$ of the given classical filter transfer function can be freely time fixed or time varying without causing any problems in the new filter design.

A. Fixed-parameter System

In a fixed-parameter system, which is a fixed-coefficient denominator $D(s,t)$ of a classical filter transfer function (4-1), the corresponding equivalent Kalman-Bucy filter coefficients will become time fixed as can be seen from equations (4-8) through (4-11) and (4-14). The fixed-parameter equivalent Kalman-Bucy filter will be considered in this section as a fixed-parameter denominator of the given classical filter.

Kalman (20) has discussed very precisely the fixed-parameter case, and it is not a problem to have an equilibrium state of the covariance equation once the stochastic process models are given. If the fixed-parameter stochastic process models that generate a filter input process are completely observable and completely controllable, then it has been proved that every solution of the covariance equation (4-11) which has a nonnegative initial value $P(0|0)$ tends uniformly to a constant matrix in the limit as t goes to infinity. Moreover, this matrix is a unique positive-definite equilibrium state of the covariance equation.

Hence, if it is assumed that the stochastic process models which can be imagined from the equivalent Kalman-Bucy filter are completely observable

and completely controllable, then $\dot{\bar{P}}(t|t)$ is zero in its steady state, and the covariance equation (4-11) becomes

$$\bar{F}\bar{P} + \bar{P}\bar{F}' - \bar{P}\bar{H}'r^{-1}\bar{H}\bar{P} + Q = 0 \quad (4-16)$$

where Q and r are undetermined constants at this point, and $\bar{P}(t|t)$ is written as \bar{P} since it is constant in the steady state.

Now the problem considered here is how to choose \bar{P} in terms of the given A , some constant scalar r , and a constant $k \times k$ matrix Q . From equations (4-14)

$$F = A + \underline{B}r\underline{B}'\bar{P}^{-1}, \quad (4-17)$$

and

$$\underline{H} = r\underline{B}'\bar{P}^{-1}. \quad (4-18)$$

Since \underline{B} has one unit element from equation (4-6), the above equations can be written as

$$F = A + \begin{bmatrix} 0 & \dots & \dots & 0 & 0 \\ \cdot & & & \cdot & \cdot \\ \cdot & & & \cdot & \cdot \\ \cdot & & & \cdot & \cdot \\ 0 & \dots & \dots & 0 & 0 \\ 0 & \dots & \dots & 0 & r \end{bmatrix} \bar{P}^{-1}, \quad (4-19)$$

$k \times k$

and

$$\underline{H} = \begin{bmatrix} 0 & 0 & \dots & \dots & 0 & r \end{bmatrix} \bar{P}^{-1}. \quad (4-20)$$

$1 \times k$

Substituting these two equations into the equation (4-16), then the steady-state covariance equation becomes

$$\bar{A}\bar{P} + \bar{P}\bar{A}' + \begin{bmatrix} 0 & \dots & 0 & 0 \\ \cdot & & \cdot & \cdot \\ \vdots & & \vdots & \vdots \\ \cdot & & \cdot & \cdot \\ 0 & \dots & 0 & 0 \\ 0 & \dots & 0 & r \end{bmatrix} + Q = 0 \quad (4-21)$$

$k \times k$

Observing equations (4-19), (4-20) and (4-21), it can be seen that once Q and r are chosen, then \bar{P} will be obtained from equation (4-21), its inverse \bar{P}^{-1} can be computed, and then F and \underline{H} are obtained from equations (4-19) and (4-20), respectively.

B. Time-varying Parameter System

When the denominator of the classical continuous filter is given with time-varying coefficients, the state equation (4-2) has a time-varying system matrix $A(t)$. Since the system matrix determines the dynamics of the filter, the equivalent Kalman-Bucy filter depends on this. The time-variable system matrix makes a time-varying fixed-gain Kalman-Bucy filter necessary. The output matrix $\underline{M}(t)$ reflected by the numerator of the continuous filter transfer function (4-1) remains the same in the output equation of the equivalent Kalman-Bucy filter, and, hence, this output matrix does not affect time-variable dynamics of the equivalent Kalman-Bucy filter.

If the time-varying parameters of the filter equation are given, the representation procedure is more complex. The stochastic process models

that generate the filter input process need to be uniformly completely observable and uniformly completely controllable so that the equilibrium solution of the covariance equation (4-11) is unique and exists.

If this is assumed, the steady-state covariance equation becomes

$$\begin{aligned} \dot{\bar{P}}(t|t) = & F(t)\bar{P}(t|t) + \bar{P}(t|t)F(t)' \\ & - \bar{P}(t|t)\underline{H}(t)'r^{-1}(t)\underline{H}(t)\bar{P}(t|t) + Q(t) \end{aligned} \quad (4-22)$$

where $Q(t)$ and $r(t)$ are undetermined but will be chosen such that this covariance equation becomes simple. $F(t)$ and $\underline{H}(t)$ are related to $\bar{P}(t|t)$ as follows:

$$F(t) = A(t) + \underline{B}r(t)\underline{B}'\bar{P}^{-1}(t|t) \quad (4-23)$$

and

$$\underline{H}(t) = r(t)\underline{B}'\bar{P}^{-1}(t|t) \quad (4-24)$$

from equations (4-14). Substituting above equations into the equation (4-22), the steady-state covariance equation becomes

$$\dot{\bar{P}}(t|t) - A(t)\bar{P}(t|t) - \bar{P}(t|t)A(t)' - \begin{bmatrix} 0 & \dots & 0 & 0 \\ \cdot & & \cdot & \cdot \\ \vdots & & \vdots & \vdots \\ 0 & \dots & 0 & 0 \\ 0 & \dots & 0 & r(t) \end{bmatrix}_{k \times k} - Q(t) = 0 \quad (4-25)$$

If this equation is compared to the matrix Riccati equation (2-8), it will be seen that the quadratic term in the matrix Riccati equation has disappeared. Hence the covariance equation (4-25) is not a Riccati-type

nonlinear matrix equation any more, but this equation represents simply $k(k+1)/2$ first-order differential equations. It is possible, therefore, to solve this covariance equation (4-25), and it assumed to have been solved in terms of $A(t)$, $Q(t)$ and $r(t)$. Then parameters $F(t)$ and $\underline{H}(t)$ are found by using this result, and equations (4-23) and (4-24) become

$$F(t) = A(t) + \begin{bmatrix} 0 & \dots & \dots & 0 & 0 \\ \cdot & & & \cdot & \cdot \\ \cdot & & & \cdot & \cdot \\ \cdot & & & \cdot & \cdot \\ 0 & \dots & \dots & 0 & 0 \\ 0 & \dots & \dots & 0 & r(t) \end{bmatrix} \bar{P}^{-1}(t|t) \quad (4-26)$$

$k \times k$

and

$$\underline{H}(t) = [0 \ 0 \ \dots \ 0 \ r(t)]_{1 \times k} \bar{P}^{-1}(t|t). \quad (4-27)$$

In summary, the parameters $F(t)$ and $\underline{H}(t)$ of the new filter are obtained by equations (4-26) and (4-27), respectively, where the inverse matrix of the steady-state covariance matrix is found from equation (4-25).

C. Development of Stochastic Process Models

Once the parameters $F(t)$, $\underline{H}(t)$ and $\underline{M}(t)$ of the equivalent Kalman-Bucy filter are obtained from the given classical filter transfer function, it is possible to imagine that an equivalent Kalman-Bucy filter input $y^*(t)$ is generated by stochastic process models. This input $y^*(t)$ may not coincide to the supposed original filter input $y(t)$, but this $y^*(t)$ will give ideas about how the equivalent Kalman-Bucy filter has been designed and optimized.

Stochastic process models of the signal process and measurement process are

$$\dot{\underline{\xi}}(t) = F(t)\underline{\xi}(t) + \underline{u}(t) \quad (4-28)$$

and

$$y^*(t) = \underline{H}(t)\underline{\xi}(t) + v(t), \quad (4-29)$$

respectively, where

$$E\{\underline{u}(t)\underline{u}(\tau)'\} = Q(t)\delta(t-\tau) \quad \text{for every } t \text{ and } \tau,$$

$$E\{v(t)v(\tau)\} = r(t)\delta(t-\tau) \quad \text{for every } t \text{ and } \tau,$$

$$E\{\underline{u}(t)\} = \underline{0} \quad \text{for all } t,$$

$$E\{v(t)\} = 0 \quad \text{for all } t,$$

$$E\{\underline{\xi}(0)\} = \underline{0},$$

$$E\{\underline{\xi}(0)\underline{u}(t)'\} = 0 \quad \text{for all } t,$$

$$E\{\underline{\xi}(0)v(t)\} = \underline{0} \quad \text{for all } t,$$

$$E\{\underline{u}(t)v(\tau)\} = \underline{0} \quad \text{for all } t \text{ and } \tau,$$

$y^*(t)$ and $v(t)$ are scalar functions.

The equivalent Kalman-Bucy filter is an optimal filter whose signal process and measurement process are given by above equations, and is an equivalent filter to the given continuous filter transfer function (4-1) which is represented by equations (4-2) and (4-3). This equivalent Kalman-Bucy filter has a fixed-gain scalar-measurement Kalman-Bucy filter and an associated output equation. With an initial $\hat{\underline{\xi}}(0|0)$, the filter equations are

$$\dot{\hat{\underline{x}}}(t|t) = F(t)\hat{\underline{x}}(t|t) + \underline{K}(t)[y^*(t) - \underline{H}(t)\hat{\underline{x}}(t|t)] \quad (4-30)$$

and

$$\underline{x}(t) = \underline{M}(t)\hat{\underline{x}}(t|t), \quad (4-31)$$

here

$$\underline{K}(t) = \bar{P}(t|t)\underline{H}(t)'r^{-1}(t), \quad (4-32)$$

$$\begin{aligned} \dot{\bar{P}}(t|t) = & F(t)\bar{P}(t|t) + \bar{P}(t|t)F(t)' \\ & - \bar{P}(t|t)\underline{H}(t)'r^{-1}(t)\underline{H}(t)\bar{P}(t|t) + Q(t) \end{aligned} \quad (4-33)$$

and $\bar{P}(0|0)$ satisfies

$$\underline{B} = \bar{P}(0|0)\underline{H}(0)'r^{-1}(t), \quad (4-34)$$

where parameters $F(t)$, $\underline{H}(t)$, $\underline{M}(t)$ and $\bar{P}(t|t)$ are given by equations (4-7), (4-19), (4-20), (4-21), (4-25), (4-26) and (4-27). A block diagram of these stochastic process models and the equivalent Kalman-Bucy filter are shown in Figure 4.2.

Here the imagined stochastic process models (4-28) and (4-29) have to be uniformly completely observable and uniformly completely controllable in the time-varying parameter system, and completely observable and completely controllable in the time-fixed parameter system. In other words, it has to be kept in mind that $\bar{P}(t|t)$ in terms of $A(t)$, $Q(t)$ and $r(t)$, and hence $F(t)$ and $\underline{H}(t)$ must be chosen properly for these conditions.

The equivalence of the new filter to the given filter will now be shown. Since the optimum gain $\underline{K}(t)$ is designed to be the same as \underline{B} in

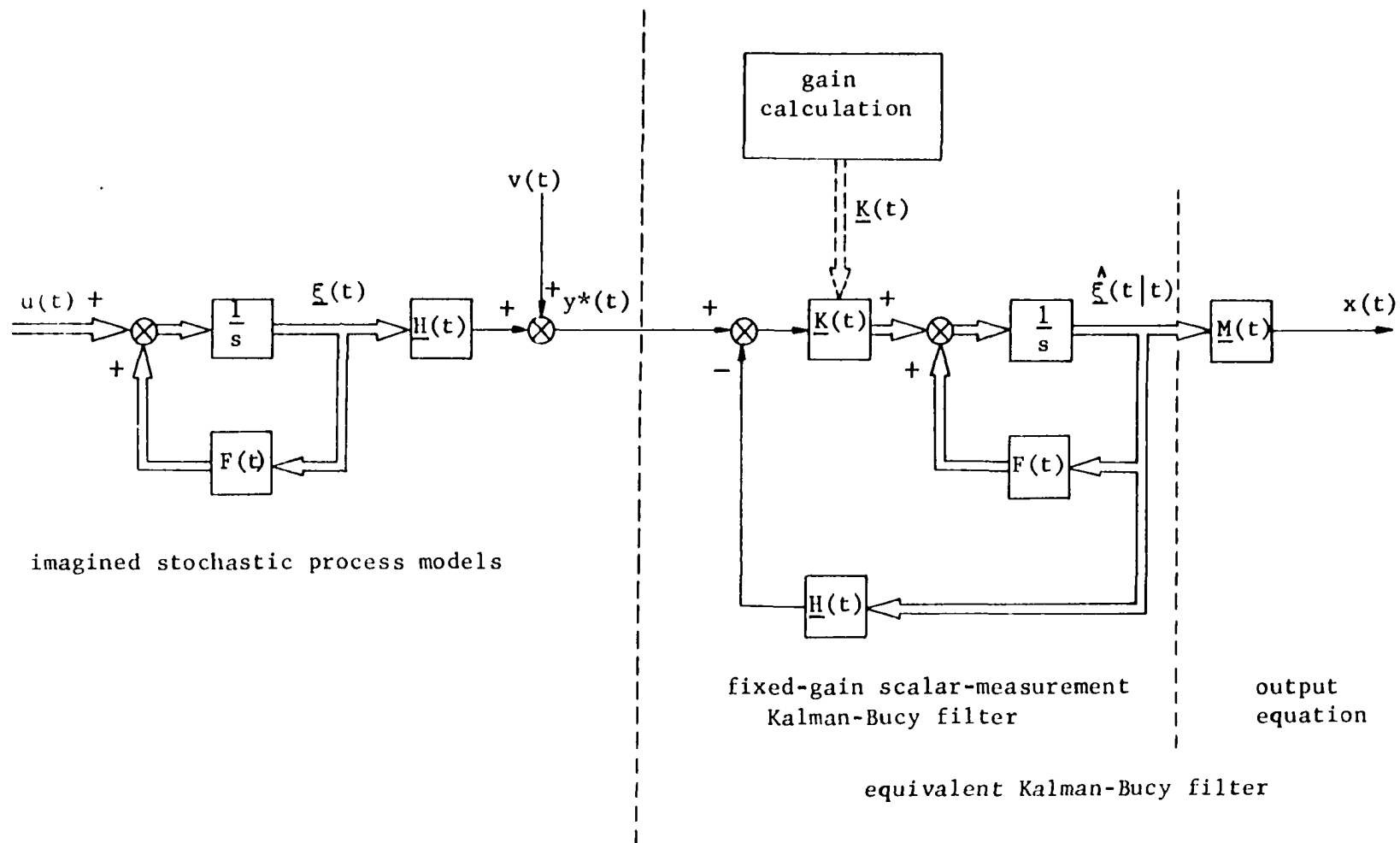


Figure 4.2. Imagined stochastic process models and the equivalent Kalman-Bucy filter

equation (4-6), the new filter equations are

$$\begin{aligned} \dot{\hat{\underline{x}}}(t|t) = & \{A(t) + \begin{bmatrix} 0 & \dots & 0 & 0 \\ \vdots & & \vdots & \vdots \\ 0 & \dots & 0 & 0 \\ 0 & \dots & 0 & r(t) \end{bmatrix} \bar{P}^{-1}(t|t)\} \hat{\underline{x}}(t|t) \\ & + \underline{B}\{y(t) - [0 \ \dots \ 0 \ r(t)]_{1 \times k} \bar{P}^{-1}(t|t) \hat{\underline{x}}(t|t)\} \\ = & A(t) \hat{\underline{x}}(t|t) + \underline{B}y(t). \end{aligned} \quad (4-35)$$

This is equivalent to equation (4-2), and the output equation (4-31) in the new filter is equivalent to equation (4-3). Therefore it can be said that the equivalent Kalman-Bucy filter is identical to the given continuous filter.

Moreover, the equation (4-35) shows that, whatever nonnegative-definite $Q(t)$ and positive $r(t)$ are chosen, this equivalence remains the same. It is because the gain $\underline{K}(t)$ is equal to the \underline{B} vector, and, therefore, $Q(t)$ and $r(t)$ do not affect the gain directly, but they only affect the covariance matrix $\bar{P}(t|t)$, and the equation (4-32) forces $\underline{K}(t)$ to be the same as \underline{B} . Hence the best way of choosing $Q(t)$ and $r(t)$ is such that the covariance equation (4-33) becomes as simple as possible. So $Q(t)$ and $r(t)$ are chosen in this dissertation as

$$Q(t) = \begin{bmatrix} 0 & \dots & 0 & 0 \\ \vdots & & \vdots & \vdots \\ 0 & \dots & 0 & 0 \\ 0 & \dots & 0 & 1 \end{bmatrix}_{k \times k}, \quad (4-36)$$

and

$$r(t) = 1, \quad (4-37)$$

respectively.

When these assumptions are made, the covariance equation (4-21) in the time-fixed system becomes

$$A\bar{P} + \bar{P}A' + \begin{bmatrix} 0 & \dots & 0 & 0 \\ \vdots & & \vdots & \vdots \\ 0 & \dots & 0 & 0 \\ 0 & \dots & 0 & 2 \end{bmatrix}_{k \times k} = 0, \quad (4-38)$$

and the equation (4-25) in the time-varying parameter system becomes

$$\dot{\bar{P}}(t|t) - A(t)\bar{P}(t|t) - \bar{P}(t|t)A(t)' - \begin{bmatrix} 0 & \dots & 0 & 0 \\ \vdots & & \vdots & \vdots \\ 0 & \dots & 0 & 0 \\ 0 & \dots & 0 & 2 \end{bmatrix}_{k \times k} = 0. \quad (4-39)$$

In this way, the measurement noise process $v(t)$ has unity white noise and the system input process noise $\underline{u}(t)$ has unity white noise at the k th element only.

These simple noise processes are added to the given classical filter, but these processes do not generate the same filter input $y(t)$ which is assumed to be at the classical filter input. Hence, what is available as an input is an imagined input $y^*(t)$, and this is not equal to $y(t)$ in most cases. By chance, if $y^*(t)$ is the same as $y(t)$, the derived fixed-gain Kalman-Bucy filter becomes an optimum filter for this input $y(t)$, and the equivalent Kalman-Bucy filter is the best filter in the sense of least mean-square error. Therefore, more information is needed about the noises for the new filter to be an optimal filter.

Many papers about identification techniques of the Kalman filter describe the selection of unknown noise information, and it means a good possibility exists for obtaining correct noise information. But this problem will not be dealt with in this dissertation.

With simple $Q(t)$ and $r(t)$, the equivalent Kalman-Bucy filter becomes a suboptimal filter for most inputs. Since this filter is equivalent to the given classical filter identically, the suboptimality depends on the given classical filter.

V. DISCRETE KALMAN-BUCY DERIVED FILTER

In the previous chapter, the derivation for obtaining the continuous equivalent Kalman-Bucy filter from a continuous classical filter transfer function was developed. It has been noted also that the equivalent Kalman-Bucy filter is a suboptimal filter for the given measurement. The suboptimal filter can also be thought of as an optimal filter having noise information about $Q(t)$ and $r(t)$ correctly known.

In this chapter, the equivalent Kalman-Bucy filter is going to be discretized. The fixed-gain Kalman-Bucy filter is discretized to be a fixed-gain discrete Kalman filter. The method used here is to discretize the stochastic process models first, so that the corresponding discrete models are obtained. Since the discrete stochastic process models are approximated from the continuous models, the corresponding discrete Kalman filter which is an optimal filter for these discrete models is identical to a discretized Kalman-Bucy filter. This approach can be said to be the same as discretizing the Kalman-Bucy filter directly. The output equation is discretized by simply sampling at discrete time.

After the equivalent Kalman-Bucy filter is discretized, the discretization procedure is simplified. For a small sampling interval, this discretization becomes a routine procedure such that an approximated discrete equivalent Kalman-Bucy filter is obtained which will be the new sampled-data digital filter as desired. This systematic design approach, called the discrete Kalman-Bucy derived filter, will be summarized at the end.

A. Discretization of Signal Process

Continuous stochastic process models of equations (4-28) and (4-29) are to be represented by discrete models as

$$\underline{\xi}_{n+1} = F_n \underline{\xi}_n + \underline{u}_n, \quad (5-1)$$

and

$$y_n^* = H_n \underline{\xi}_n + v_n, \quad (5-2)$$

where n is a discrete time at $t = nT$, and F_n , H_n , $\underline{\xi}_n$, \underline{u}_n , v_n and y_n^* are defined such that the solution of this discrete process is identical with the solution of the continuous process at time t in a statistical sense.

Also the output x_n of the discrete filter will be represented as

$$x_n = \frac{M}{n} \hat{\underline{\xi}}_n | n, \quad (5-3)$$

so that the discrete output x_n has approximately the same value as that of the continuous filter output $x(t)$ at time $t = nT$.

The signal process will be discretized in this section and the measurement process in the next section. The output equation is going to be discretized in the following section.

The solution of the linear differential equation (4-28) is, for $t \geq 0$,

$$\underline{\xi}(t) = \phi(t,0) \underline{\xi}(0) + \int_0^t \phi(t,\lambda) \underline{u}(\lambda) d\lambda, \quad (5-4)$$

where $\phi(t,\lambda)$ is the state transition matrix of the system described by the equation (4-28).

Suppose the solution is obtained at time $t = nT$; then equation (5-4) can be written as

$$\underline{x}(nT) = \phi(nT, 0)\underline{x}(0) + \int_0^{nT} \phi(nT, \lambda)\underline{u}(\lambda)d\lambda, \quad (5-5)$$

and at time $t = (n+1)T$

$$\begin{aligned} \underline{x}[(n+1)T] &= \phi[(n+1)T, 0]\underline{x}(0) + \int_0^{(n+1)T} \phi[(n+1)T, \lambda]\underline{u}(\lambda)d\lambda \\ &= \phi[(n+1)T, nT] \left\{ \phi(nT, 0)\underline{x}(0) + \int_0^{nT} \phi(nT, \lambda)\underline{u}(\lambda)d\lambda \right\} \\ &= \phi[(n+1)T, nT] \left\{ \phi(nT, 0)\underline{x}(0) + \int_0^{nT} \phi(nT, \lambda)\underline{u}(\lambda)d\lambda \right\} \\ &\quad + \phi[(n+1)T, nT] \int_{nT}^{(n+1)T} \phi(nT, \lambda)\underline{u}(\lambda)d\lambda \\ &= \phi[(n+1)T, nT]\underline{x}(nT) + \phi[(n+1)T, nT] \int_{nT}^{(n+1)T} \phi(nT, \lambda)\underline{u}(\lambda)d\lambda. \end{aligned} \quad (5-6)$$

Having n as nT and $(n+1)$ as $(n+1)T$, then the signal process model is digitalized from equation (4-28) as

$$\underline{x}_{n+1} = F_n \underline{x}_n + \underline{u}_n, \quad (5-7)$$

where

$$F_n = \phi[(n+1)T, nT] \quad (5-8)$$

and

$$\underline{u}_n = \phi[(n+1)T, nT] \int_{nT}^{(n+1)T} \phi(nT, \lambda)\underline{u}(\lambda)d\lambda. \quad (5-9)$$

Hence the noise covariance $Q_n = E\{\underline{u}_{n-n} \underline{u}_{n-n}'\}$ becomes

$$Q_n = \phi[(n+1)T, nT] \left\{ \int_{nT}^{(n+1)T} \phi(nT, \lambda) Q(\lambda) \phi(nT, \lambda)' d\lambda \right\} \phi[(n+1)T, nT]'. \quad (5-10)$$

Therefore the discrete signal process model is derived to have parameters F_n and Q_n as given by equations (5-8) and (5-10), respectively.

B. Discretization of Measurement Process

Now the measurement process of stochastic process models is going to be discretized. Here it will be assumed the discrete measurement process y_n^* in equation (5-2) is an average of the continuous process $y^*(t)$ over the small interval of time T , then the discrete measurement y_n^* is given, at time $t = nT$, as

$$\begin{aligned} y_n^* &= \frac{1}{T} \int_{nT}^{(n+1)T} [\underline{H}(\lambda) \underline{\xi}(\lambda) + v(\lambda)] d\lambda \\ &= \underline{H}_{n-n} \underline{\xi}_{n-n} + \frac{1}{T} \int_{nT}^{(n+1)T} v(\lambda) d\lambda, \end{aligned} \quad (5-11)$$

where the discrete measurement vector \underline{H}_n is defined as

$$\underline{H}_n = \frac{1}{T} \int_{nT}^{(n+1)T} \underline{H}(\lambda) d\lambda. \quad (5-12)$$

Again, for the small interval T , this discrete measurement vector \underline{H}_n has approximately the same value as the continuous measurement vector $\underline{H}(t)$ at time $t = nT$.

And the discrete measurement noise v_n will be defined as

$$v_n = \frac{1}{T} \int_{nT}^{(n+1)T} v(\lambda) d\lambda, \quad (5-13)$$

then the discrete measurement noise covariance is derived as

$$\begin{aligned} r_n &= E\{v_n^2\} \\ &= \frac{1}{T^2} \int_{nT}^{(n+1)T} \int_{nT}^{(n+1)T} E\{v(\lambda)v(\sigma)\} d\lambda d\sigma \\ &= \frac{1}{T^2} \int_{nT}^{(n+1)T} \int_{nT}^{(n+1)T} r(\lambda) \delta(\sigma-\lambda) d\lambda d\sigma \\ &= \frac{1}{T} r(nT). \end{aligned} \quad (5-14)$$

In summary, the parameters \underline{H}_n and r_n of the discrete measurement process models are obtained from equations (5-12) and (5-14), respectively.

C. Discretization of Output Equation

In this section, the output equation (4-31) is going to be discretized. As can be seen from the derivation (5-6), the discrete states are samples of the continuous states at time $t = nT$. Hence the statistical properties of the continuous states $\underline{x}(t)$ and discrete states \underline{x}_n are identical as the sampling interval T becomes small. The fixed-gain Kalman-Bucy filter (4-30) is an optimal filter estimating the continuous states $\underline{x}(t)$; and the fixed-gain discrete Kalman filter, which can be built from the results of previous sections, is an optimal filter of \underline{x}_n . Therefore the estimated

states in both continuous and discrete are same in the statistical sense.

Comparing output equations (4-31) and (5-3), the discrete estimates $\hat{\xi}_n|n$ are samples of $\hat{\xi}(t|t)$ at time $t = nT$, and the discrete output x_n is needed as a sample of $x(t)$ at time $t = nT$. Therefore the discrete output matrix \underline{M}_n has the relationship with the continuous output matrix $\underline{M}(t)$ as

$$\underline{M}_n = \underline{M}(nT), \quad (5-15)$$

where $t = nT$.

D. Discrete Kalman-Bucy Derived Filter

Now, in summarizing previous results, a discretized equivalent Kalman-Bucy filter is finally achieved as follows. The discrete state estimation and the discrete output equations are

$$\hat{\xi}_n|n = \hat{\xi}_n|n-1 + \frac{K}{n} [y_n - \frac{H}{n} \hat{\xi}_n|n-1], \quad (5-16)$$

and

$$x_n = \frac{M}{n} \hat{\xi}_n|n, \quad (5-17)$$

where

$$\frac{K}{n} = \bar{P}_{n|n-1} \frac{H'}{n} [\frac{H}{n} \bar{P}_{n|n-1} \frac{H'}{n} + r_n]^{-1} \quad (5-18)$$

$$\bar{P}_{n|n} = [I - \frac{K}{n} \frac{H}{n}] \bar{P}_{n|n-1}, \quad (5-19)$$

$$\hat{\xi}_{n+1}|n = F_n \hat{\xi}_n|n, \quad (5-20)$$

and

$$\bar{P}_{n+1|n} = F_n \bar{P}_{n|n} F_n' + Q_n, \quad (5-21)$$

with

$$\hat{\underline{x}}_0|_{-1} = \underline{0}, \quad (5-22)$$

and

$$\bar{P}_0|_{-1} = \bar{P}(0|0). \quad (5-23)$$

The above F_n , H_n and M_n are obtained from equations (5-8), (5-12) and (5-15), respectively. Noise covariances Q_n and r_n are found from equations (5-10) and (5-14), respectively. A block diagram representation is shown in Figure 5.1.

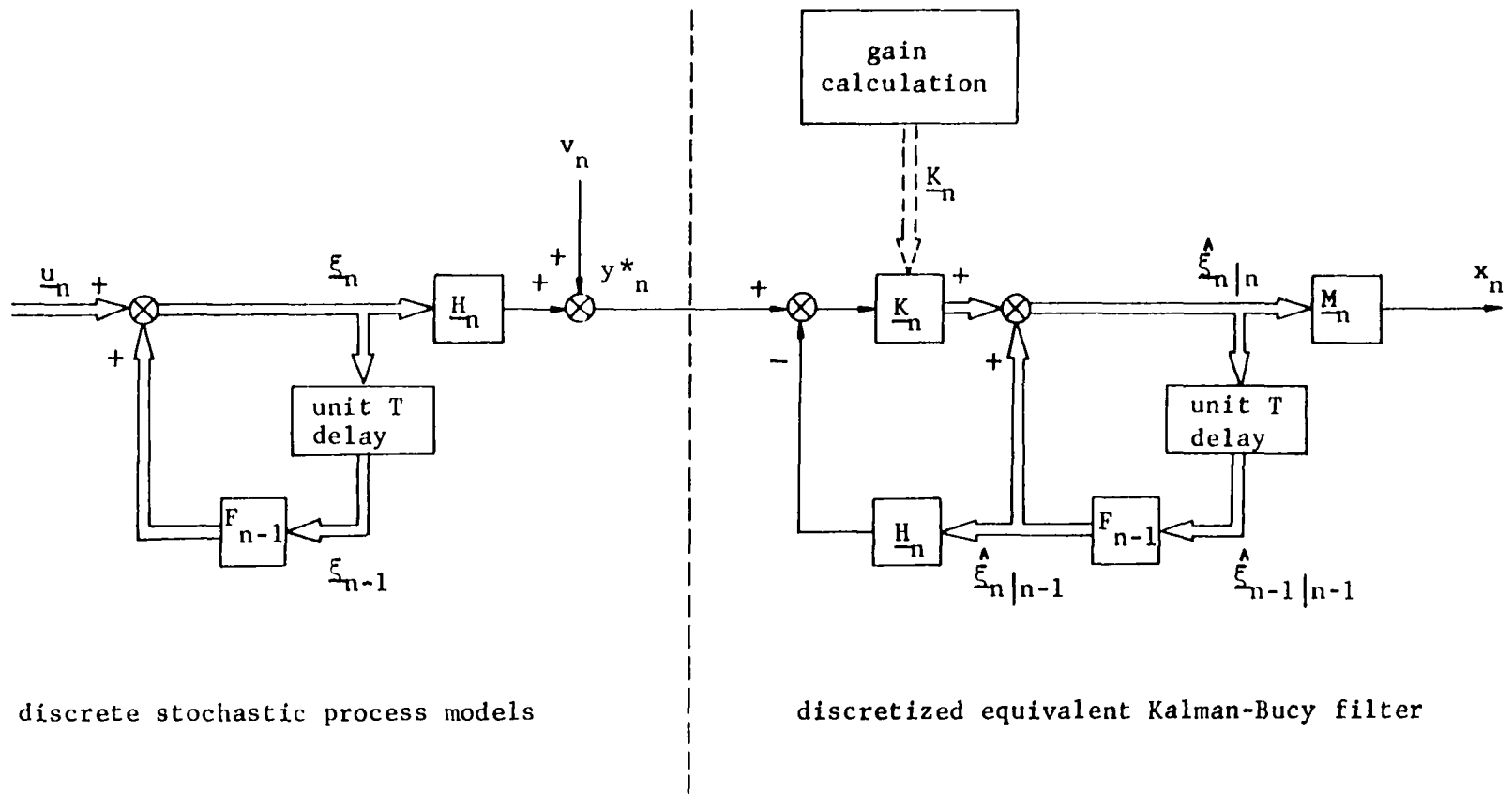
As described so far, the discretization procedure is straight forward as developed previously. However there is difficulty obtaining the state transition matrix of the system model (4-28), which is simply denoted as the equation (5-8). When the system matrix $F(t)$ is solved and $F(t)$ satisfies the commutative condition

$$F(t_1)F(t_2) = F(t_2)F(t_1) \quad (5-24)$$

for all t_1 and t_2 , then the state transition matrix is simply given by

$$\phi(t, \tau) = \exp \int_{\tau}^t F(\lambda) d\lambda. \quad (5-25)$$

Two trivial cases where the commutative condition is valid are those in which $F(t)$ is a time-fixed matrix, and those in which $F(t)$ is a diagonal matrix. The later case is not satisfied in this new filter derivation except when the system (4-28) is scalar. It is because the system matrix $F(t)$ is given as a companion form as can be seen from the equation (4-26).



discrete stochastic process models

discretized equivalent Kalman-Bucy filter

Figure 5.1. Discrete stochastic process models and the discretized equivalent Kalman-Bucy filter

The scalar state transition matrix is obtained easily with a scalar $F(t)$ from equation (5-25). For the former case, the state transition matrix is obtained without any trouble by the equation (5-25), too.

When the system matrix $F(t)$ does not satisfy the commutative condition (5-24), the state transition matrix can be obtained by a method known as the Peano-Baker method. This method is described in Appendix A. In this general case, the state transition matrix is represented by a matrizant according to this method:

$$\phi(t, \tau) = \Lambda(F). \quad (5-26)$$

As is seen in equations (5-25) and (5-26), it is a little complex obtaining the discrete system matrix even it is denoted as the equation (5-8). An approximate procedure will be able to alleviate this problem. This simplification begins with the system matrix F_n first, and then the discretized equivalent Kalman-Bucy filter equations (5-16) through (5-23) are approximated such that those eight equations are reduced to three equations.

If it is assumed that the sampling interval T is small, the system matrix is simplified as

$$F_n = I + TF(nT), \quad (5-27)$$

where only the unit matrix term and first-order term in T are taken from both equations (5-25) and (5-26). The higher-order terms in T are supposed to approach zero as the sampling interval becomes small.

By substituting this approximation (5-27) into equation (5-10), the noise covariance Q_n is simplified also. This follows:

$$Q_n = TQ(nT). \quad (5-28)$$

Another simplification begins with the realization of the discrete gain in terms of the continuous gain. First the a priori covariance $\bar{P}_{n+1|n}$ in the discrete filter is formulated in connection with the continuous covariance matrix. Substituting equation (5-19) into the equation (5-21), this becomes

$$\begin{aligned} \bar{P}_{n+1|n} = & F_n \bar{P}_{n|n-1} - F_n \bar{P}_{n|n-1} \underline{H}'_n [\underline{H}_n \bar{P}_{n|n-1} \underline{H}'_n + r_n]^{-1} \underline{H}_n \bar{P}_{n|n-1} F'_n \\ & + Q_n. \end{aligned} \quad (5-29)$$

Here it is assumed that r_n is much bigger than the term $\underline{H}_n \bar{P}_{n|n-1} \underline{H}'_n$, for a small sampling interval T , and then the bracket term in the above equation becomes

$$[\underline{H}_n \bar{P}_{n|n-1} \underline{H}'_n + r_n]^{-1} \approx r_n^{-1}. \quad (5-30)$$

Then equation (5-29), after using equations (5-12), (5-14), (5-27) and (5-28), becomes

$$\begin{aligned} \bar{P}_{n+1|n} = & \bar{P}_{n|n-1} + TF(nT)\bar{P}_{n|n-1} + T\bar{P}_{n|n-1}F(nT)' \\ & - T\bar{P}_{n|n-1}\underline{H}(nT)'r^{-1}(nT)\underline{H}(nT)\bar{P}_{n|n-1} + TQ(nT). \end{aligned} \quad (5-31)$$

This has been rearranged to give

$$\begin{aligned} \frac{\bar{P}_{n+1|n} - \bar{P}_{n|n-1}}{T} &= F(nT)\bar{P}_{n|n-1} + \bar{P}_{n|n-1}F(nT)' \\ &\quad - \bar{P}_{n|n-1}\underline{H}(nT)'r^{-1}(nT)\underline{H}(nT)\bar{P}_{n|n-1} + Q(nT). \end{aligned} \quad (5-32)$$

For the small T again, this equation (5-32) is an approximation of the continuous covariance equation (4-33). Hence it can be said that the a priori covariance matrix has remained the same as the continuous covariance matrix $\bar{P}(t|t)$ during the discretization, and, therefore, it can be written as

$$\bar{P}_{n+1|n} = \bar{P}(t|t) \quad (5-33)$$

for the time $t = nT$. With this result, the approximated equation (5-30) is also checked again. By substituting \underline{H}_n by the equations (4-27), (5-12) and r_n by equation (5-14) as

$$\begin{aligned} \underline{H}_n \bar{P}_{n|n-1} \underline{H}_n' + r_n &= \begin{bmatrix} 0 & \dots & 0 & r(nT) \end{bmatrix}_{1 \times k} \bar{P}^{-1}[(n-1)T|(n-1)T] \begin{bmatrix} 0 \\ \vdots \\ 0 \\ r(nT) \end{bmatrix}_{k \times 1} \\ &\quad + \frac{1}{T} r(nT) \\ &= r(nT) \bar{P}_{kk}^*[(n-1)T|(n-1)T] + \frac{1}{T} r(nT), \end{aligned} \quad (5-34)$$

where $\bar{P}_{kk}^*[(n-1)T|(n-1)T]$ is a $k \times k$ th element of $\bar{P}^{-1}[(n-1)T|(n-1)T]$. As the sampling interval T is small, the dominating term in the above equation is r_n .

Finally, the discrete gain is simplified. From the equations (5-18) and (5-30)

$$\underline{K}_n = \bar{P}_n |_{n-1} \underline{H}'_n r_n^{-1}. \quad (5-35)$$

Substituting equations (5-12), (5-14) and (5-33) into this relationship, this becomes

$$\underline{K}_n = T \bar{P}(nT) |_{nT} \underline{H}(nT) r^{-1}(nT). \quad (5-36)$$

Comparing this equation to the continuous gain (4-32), it can be seen that

$$\underline{K}_n = T \underline{K}(nT) \quad (5-37)$$

where the continuous time $t = nT$.

Since an equivalent Kalman-Bucy filter as given by equations (4-30) and (4-31) has a fixed gain which is equal to \underline{B} , the corresponding discrete gain is as follows.

$$\underline{K}_n = T \underline{B}. \quad (5-38)$$

Once the gain in equation (5-16) is fixed as determined by equation (5-38), it is not necessary to go through the recursive covariance algorithm in equations (5-18), (5-19), (5-21) and (5-23).

The rest of the filter equations are summarized as follows:

$$\hat{\underline{x}}_n |_{n-1} = [I - \frac{K_n \underline{H}}{n-1}] F_{n-1} \hat{\underline{x}}_{n-1} |_{n-1} + \frac{K_n}{n-1} y_n, \quad (5-39)$$

and

$$\underline{x}_n = \frac{M}{-n} \sum_{\ell=0}^{\Lambda} \underline{z}_{n-\ell}, \quad (5-40)$$

where

$$\underline{K}_n = \begin{bmatrix} 0 \\ 0 \\ \vdots \\ \vdots \\ 0 \\ T \end{bmatrix} \cdot \quad (5-41)$$

$k \times 1$

Equations (5-39) and (5-40) are the final form of the new sampled-data digital filter corresponding to the continuous filter given by the equation (4-1). A block diagram for this new filter is shown in Figure 5.2.

The stability of the new filter will now be checked with uniformly completely observable and uniformly completely controllable conditions. From equation (5-27), the state transition matrix of the new filter is

$$\phi(m,i) = \begin{cases} \prod_{\ell=i}^{m-1} [I + TF(\ell T)] & \text{for } m > i, \\ I & \text{for } i = m. \end{cases} \quad (5-42)$$

The observability matrix $O_{m,n}$ is obtained from equation (2-20) as

$$O_{m,n} = \sum_{i=m}^n \left\{ \prod_{\ell=m-1}^{i-1} [I + TF(\ell T)]^{-1} \right\}' \frac{H_i}{i-1} \left\{ \prod_{\ell=m-1}^{i-1} [I + TF(\ell T)]^{-1} \right\}, \quad (5-43)$$

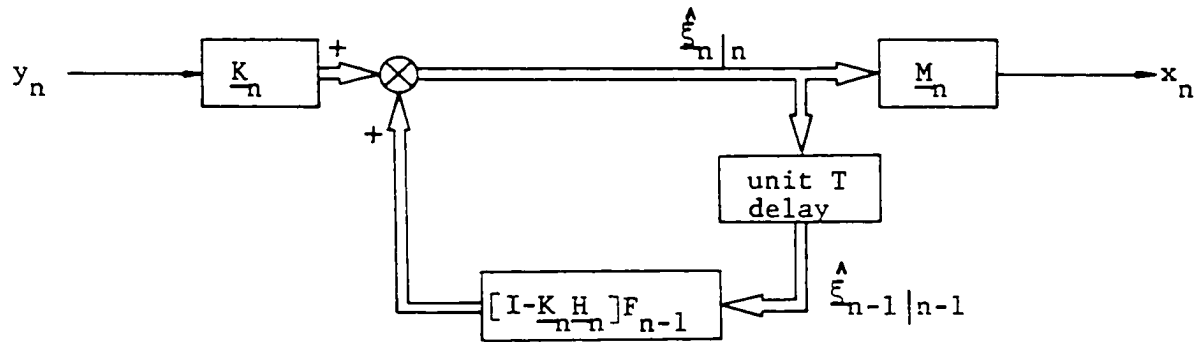


Figure 5.2. A general block diagram of the discrete Kalman-Bucy derived filter

and from equation (2-22) the controllability matrix $C_{m,n}$ is

$$C_{m,n} = \sum_{i=m}^n \left\{ \prod_{\ell=m-1}^{i-1} [I + TF(\ell T)] \right\} \left\{ \prod_{\ell=m-1}^{i-1} [I + TF(\ell T)] \right\}', \quad (5-44)$$

for some $n \geq m$.

As can be seen from above two equations, both the controllability matrix and the observability matrix are positive definite and finite for a small T , and for all possible n . Hence the stability of the discrete Kalman-Bucy derived filter is guaranteed if the given continuous filter is stable.

The design procedure for obtaining this new sampled-data digital filter can be summarized as follows:

- 1) The classical continuous filter transfer function is given in general form of equation (4-1).
- 2) The first step is the state representation of the classical filter and the determination of $A(t)$ and $\underline{M}(t)$ from equations (4-5) and (4-7), respectively.
- 3) The second step is to solve the steady-state covariance equation using equation (4-38) or (4-39) and to find its inverse matrix.
- 4) The third step is to determine the continuous parameters $F(t)$ and $\underline{H}(t)$ using the inverse covariance matrix $\bar{P}^{-1}(t|t)$ and equations (4-19) and (4-20) in the time-fixed parameter system, or (4-26) and (4-27) in the time-varying parameter system.
- 5) The fourth step is to determine the discrete parameters F_n , \underline{H}_n and \underline{M}_n where F_n is obtained from the equation (5-27), \underline{H}_n from equation (5-12), and \underline{M}_n from equation (5-15).

- 6) The result is the new sampled-data digital filter called the discrete Kalman-Bucy derived filter. It is specified by equations (5-39) and (5-40).

VI. COMPARISON OF DISCRETE KALMAN-BUCY DERIVED FILTER AND z-TRANSFORM DERIVED FILTER

A new sampled-data digital filter thus obtained from a classical filter will be called the discrete Kalman-Bucy derived filter. There are three alternate z-transforms which are known best. In this chapter, comparisons will be made between the discrete Kalman-Bucy derived filter and z-transform derived filters.

When a sampled-data digital filter is realized with digital arithmetic elements, considerations of the computer size, speed, and accuracy of the implementation are necessary to assess the performance of the filter. The first two considerations may be compared by the number of additions and multiplications contained in the sampled-data digital filter. This implies that adders and multipliers in the hardware of the digital filter contribute mostly the digital filter size and speed. Especially the multipliers consume most of the computation time in the digital implementation. The third consideration is distinguished by the fidelity of the sampled-data digital filter as compared to the frequency response of the original classical continuous filter.

A. Hardware Requirements

In an analog system, the realization of a given system transfer function is a difficult problem that has received considerable attention; but for a sampled-data system, the implementation of difference equation is almost trivial. Coefficient word length, computational word length and sampling interval with the sampled-data digital filter equations have to

be decided first. Filter accuracy problems are brought about mostly by the computational word length rather than the filter coefficient word length. It will be assumed in this dissertation that the word lengths are long enough so that they will not cause truncation errors. Since the computational time delay through the filter is defined as the time required to compute the present output $x(nT)$ after the present filter input $y(nT)$ has been sampled, the sampling interval T has to be chosen longer than the computational time delay.

After the word lengths and the sampling interval are determined, the design considerations left are hardware considerations. They are compositions of serial-parallel and combinational arrangements of integrated-circuit chips. Serial-parallel multiplier, digital filter shift register, time-varying adaptive elements, and clock and word timing register chips have been made feasible by the rapid advances in integrated-circuit technology. The logic configurations for these chips are discussed by White and Mitsutomi (37). As an aid to understanding these chips, the shift register, the serial full adder, and the serial-parallel multiplier chips are discussed in Appendix B.

White and Mitsutomi (37) developed their own chips, each measuring between 100 and 200 millimeters across length and width, which house a large number of circuits. For example, the serial-parallel multiplier chip contains 650 transistors. The shift register chip contains 1,200 transistors which are designed for eight-bit plus sign-bit precision. According to their configurations, the bit time of multiplication input data can be between 600 nanoseconds and 100 microseconds. The multiplica-

tional delay time can be calculated in this case. Since the longest word generated at the multiplier output gate is the same as the sum of coefficient word lengths, the multiplicand coefficient word length plus the multiplier coefficient word length, the multiplicative delay time is computed as follows:

$$\tau_m = (\ell_a + \ell_b)\tau_b, \quad (6-1)$$

where

τ_m is a multiplicative delay time per channel used in the serial-parallel multiplier,

ℓ_a is a multiplicand coefficient word length,

ℓ_b is a multiplier coefficient word length,

and τ_b is a computer bit time in the multiplier.

This multiplicative delay time is equal to two word times when lengths of the multiplicand and the multiplier words are same, i.e., $\ell_a = \ell_b = \ell_w$.

A computational time delay for one data sample is represented as, therefore,

$$\tau_d > k_c(\ell_a + \ell_b)\tau_b, \quad (6-2)$$

where k_c is a number of channels in the digital processing. From equation (6-2), it can be seen that the number of multiplications controls the computer speed as well as the computer size, mostly.

Now the design of the discrete Kalman-Bucy derived filter is investigated. First, elements of $F(t)$ and $\underline{H}(t)$ in equations (4-26) and (4-27) are defined as, in general case,

$$F(t) = \begin{bmatrix} 0 & 1 & 0 & \cdot & \cdot & \cdot & \cdot & 0 & 0 \\ 0 & 0 & 1 & \cdot & \cdot & \cdot & \cdot & 0 & 0 \\ \cdot & \cdot & \cdot & & & & & \cdot & \cdot \\ \cdot & \cdot & \cdot & & & & & \cdot & \cdot \\ 0 & 0 & 0 & \cdot & \cdot & \cdot & \cdot & 0 & 1 \\ f_1(t) & f_2(t) & f_3(t) & \cdot & \cdot & \cdot & f_{k-1}(t) & f_k(t) \end{bmatrix}_{k \times k}, \quad (6-3)$$

and

$$\underline{H}(t) = [h_1(t) \quad h_2(t) \quad h_3(t) \quad \cdot \quad \cdot \quad \cdot \quad h_{k-1}(t) \quad h_k(t)]_{1 \times k}, \quad (6-4)$$

respectively.

The reason that $F(t)$ is represented as a companion matrix form is that $A(t)$ in the equation (4-5) is given as a companion matrix form and the second term in the equation (4-26) has nonzero elements only in the k th row. The equation (6-3) becomes

$$F_n = \begin{bmatrix} 1 & T & 0 & \cdot & \cdot & \cdot & \cdot & 0 & 0 \\ 0 & 1 & T & \cdot & \cdot & \cdot & \cdot & 0 & 0 \\ 0 & 0 & 1 & \cdot & \cdot & \cdot & \cdot & 0 & 0 \\ \cdot & \cdot & \cdot & & & & & \cdot & \cdot \\ \cdot & \cdot & \cdot & & & & & \cdot & \cdot \\ 0 & 0 & 0 & \cdot & \cdot & \cdot & \cdot & T & 0 \\ 0 & 0 & 0 & \cdot & \cdot & \cdot & \cdot & 1 & T \\ Tf_1(nT) & Tf_2(nT) & Tf_3(nT) & \cdot & \cdot & \cdot & Tf_{k-1}(nT) & 1 + Tf_k(nT) \end{bmatrix}_{k \times k} \quad (6-5)$$

by using the equation (5-27). Hence the coefficient matrix of the first term in the equation (5-39) is

$$[I - \frac{KH}{n-n}]F_{n-1} = \begin{bmatrix} 1 & T & 0 & \dots & 0 & 0 \\ 0 & 1 & T & \dots & 0 & 0 \\ 0 & 0 & 1 & \dots & 0 & 0 \\ \vdots & \vdots & \vdots & \ddots & \vdots & \vdots \\ 0 & 0 & 0 & \dots & T & 0 \\ 0 & 0 & 0 & \dots & 1 & T \\ \theta_1(nT) & \theta_2(nT) & \theta_3(nT) & \dots & \theta_{k-1}(nT) & \theta_k(nT) \end{bmatrix} \quad (6-6)$$

where

$$\theta_1(nT) = -Th_1(nT) + T\{1 - Th_k(nT)\}f_1[(n-1)T]$$

$$\theta_i(nT) = -T^2h_{i-1}(nT) - Th_i(nT) + T\{1 - Th_k(nT)\}f_i[(n-1)T]$$

$$\text{for } i = 2, 3, \dots, k-2, k-1,$$

$$\text{and } \theta_k(nT) = -T^2h_{k-1}(nT) + \{1 - Th_k(nT)\}\{1 + Tf_k[(n-1)T]\}.$$

One of the discrete Kalman-Bucy derived filter equations (5-39) and (5-40) becomes

$$\begin{bmatrix} \hat{S}_1 \\ \hat{S}_2 \\ \vdots \\ \hat{S}_{k-1} \\ \hat{S}_k \end{bmatrix}_{n|n} = \begin{bmatrix} 1 & T & \dots & 0 \\ 0 & 1 & \dots & 0 \\ \vdots & \vdots & \ddots & \vdots \\ 0 & 0 & \dots & T \\ \theta_1(nT) & \theta_2(nT) & \dots & \theta_k(nT) \end{bmatrix}_{k \times k} \begin{bmatrix} \hat{S}_1 \\ \hat{S}_2 \\ \vdots \\ \hat{S}_{k-1} \\ \hat{S}_k \end{bmatrix}_{n-1|n-1} + \begin{bmatrix} 0 \\ 0 \\ \vdots \\ 0 \\ T \end{bmatrix}_{k \times 1} y_n \quad (6-7)$$

What can be seen from the above equation is that, for this recursive estimation, $(2k - 1)$ additions and $2k$ multiplications are needed. Of the $2k$ multiplications, k are constant multiplications by T . Moreover it can be seen that this estimator needs k memories for the previous states

$$\hat{\xi}_{n-1|n-1}^A$$

The output equation (5-40) has k multiplications and $(k - 1)$ additions as can be seen from

$$x_n = [\beta_0(nT) \ \beta_1(nT) \ \dots \ \beta_{k-2}(nT) \ \beta_{k-1}(nT)]_{1 \times k} \begin{bmatrix} \hat{\xi}_1^A \\ \hat{\xi}_2^A \\ \vdots \\ \hat{\xi}_{k-1}^A \\ \hat{\xi}_k^A \end{bmatrix}_{n|n} \quad (6-8)$$

A generalized block diagram of the new filter is shown in Figure 6.1.

Consequently, $(3k - 2)$ additions, $3k$ multiplications, and k memories for each storage are needed. The k multiplications of those are done by constant multiplication T .

The number of channels which delay the computation speed is two multiplication channels and $(k + 1)$ addition channels. The two multiplication channels are multiplications of T and y_n , and $\beta_{k-1}(nT)$ and $\hat{\xi}_k^A(n|n)$, as can be shown in Figure 6.1. The rest of the multiplications $(3k - 2)$ are estimated during or before these two multiplications. If the digital filter word length, l_w , is given to be $l_w = l_a = l_b$, and τ_a is

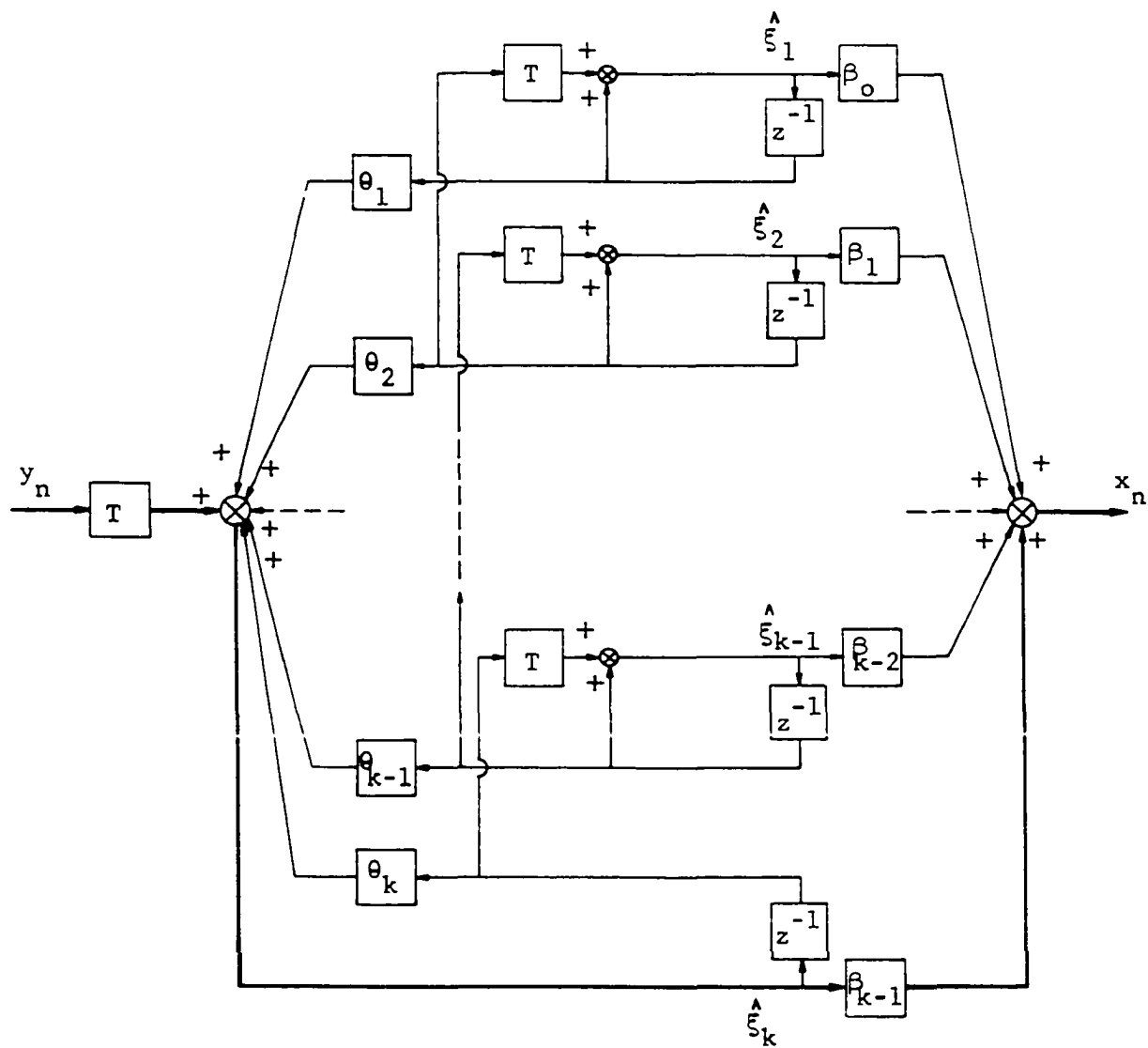


Figure 6.1. A block diagram of the discrete
Kalman-Bucy derived filter

computer bit time in the adder, the computational delay time becomes

$$\tau_d = (k + 1)l_w \tau_a + 4l_w \tau_b. \quad (6-9)$$

However as can be seen from equations (6-7) and (6-8), two equations may be solved in terms of x_n such that the combined discrete filter equation is simpler than its original equation which is represented according to states given in the first step. It is because the state defined in the first step need not remain same. In this way, the form of Kalman-Bucy filter equations or the form of discrete Kalman filter form is changed and a simpler discrete filter equation is obtained. In this case the hardware requirements are less and the computational time delay becomes shorter than before.

Setting the delay operator z^{-1} as $\hat{\xi}_{n-1|n-1} = z^{-1} \hat{\xi}_n|n$, equations (6-7) and (6-8) are equivalent to the simple equation noted by $W_N(z)$, i.e.,

$$W_N(z) = \frac{T \left\{ \sum_{i=1}^k \beta_{i-1}(nT) (Tz^{-1})^{k-i} (1-z^{-1})^{i-1} \right\}}{(1-z^{-1})^{k-1} - \left\{ \sum_{i=1}^k \theta_i(nT) (Tz^{-1})^{k-i} (1-z^{-1})^{i-1} \right\} z^{-1}}. \quad (6-10)$$

This derivation is explained in Appendix C. The denominator of the $W_N(z)$ is a k th-order polynomial in z^{-1} and the order of the numerator is less than k . Now suppose the above equation (6-10) is solved and rearranged in order of z^{-1} , then this can be written as

$$W_N(z) = \frac{\sum_{i=0}^{k-1} B_i(nT) z^{-i}}{1 - \sum_{i=1}^k A_i(nT) z^{-i}}. \quad (6-11)$$

Figure 6.2 shows the block diagram this simplified notation.

As can be seen from these, the new filter in this arrangement needs $2k$ multiplications and $(2k - 1)$ additions. There are k shift registers needed for memories. Moreover, there is only one channel which delays the multiplicative time. It is B_0 with the same state in forward loop. Additional channels are $(k + 1)$. The computational time delay, therefore, can be represented as

$$\tau_d = (k + 1)\ell_w \tau_a + 2\ell_w \tau_b. \quad (6-12)$$

These configurations will be compared to the z -transform derived filters using same serial machine. When the given transfer function is solved in partial fraction form as the equation (3-21), the corresponding sampled-data digital filter transfer function is given as the equation (3-25) and the block diagram is shown as Figure 3.3. Here $m_1 + 2m_2$ will correspond to k . The equation (3-25) shows m_1 times two multiplications and one addition, and m_2 times four multiplications and three additions in each term. Hence the number of multiplications are $(2m_1 + 4m_2) = 2k$, and the number of additions are $(2m_1 + 4m_2 - 1) = 2k - 1$ including $(m_1 + m_2 - 1)$ additions at the output gate.

In this case, the number of multiplications and the number of additions remain the same as those of the new filter. The number of shift registers needed for memories are $m_1 + 2m_2$ which are also the same as those of the discrete Kalman-Bucy derived filter. The number of channels in this method are one multiplicative channel and $(m_1 + m_2 + 1)$ additional channels. The computational time delay is

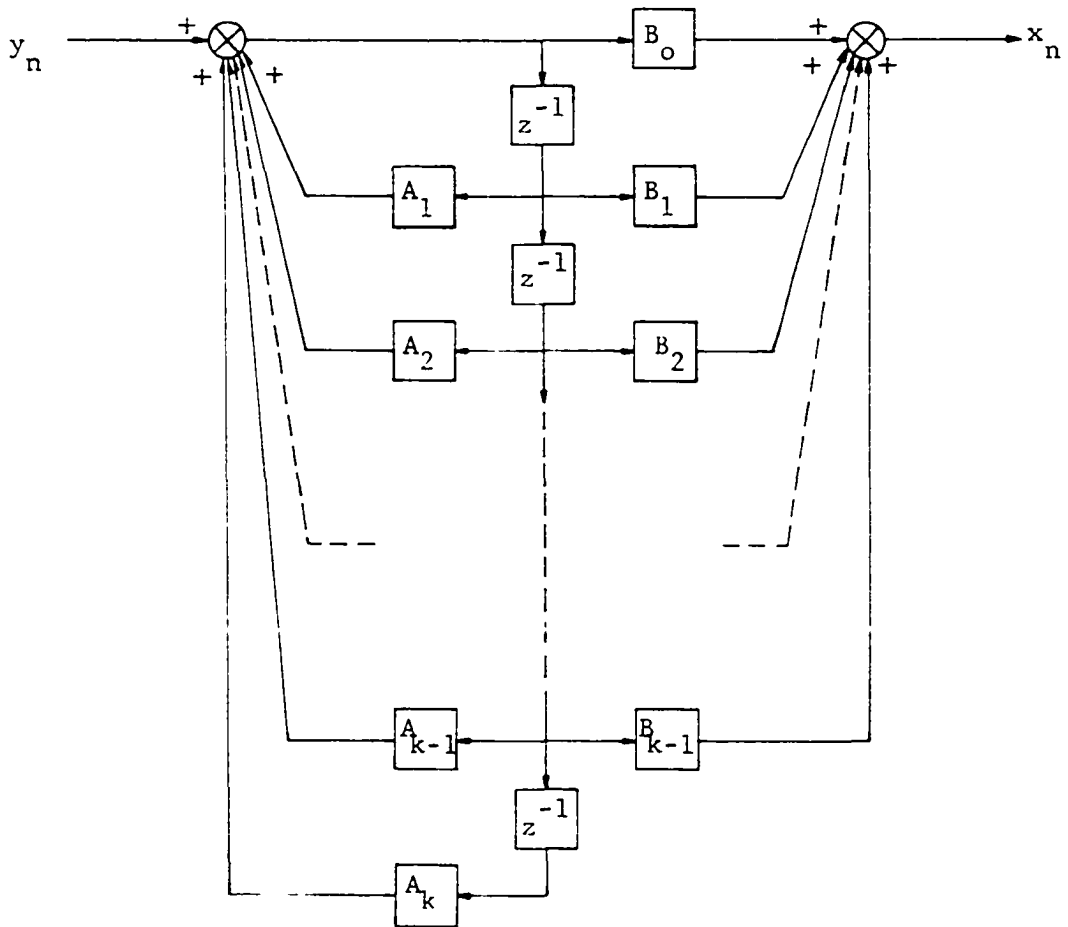


Figure 6.2. A block diagram of the new filter

$$\tau_d = (m_1 + m_2 + 1)\ell_w \tau_a + 2\ell_w \tau_b. \quad (6-13)$$

And when the given transfer function is factored and represented as the equation (3-33), the sampled-data digital filter transfer function becomes as the equation (3-34) in the bilinear z-transform digital filter, and the equation (3-39) in the matched z-transform digital filter. In this case, $m_1 + 2m_3$ corresponds to k and $m_2 + 2m_4$ corresponds to $k - 1$, comparing to equation (4-1). From equation (3-34) and Figure 3.4, it is evident that the digital implementation of the equation (3-34) needs $2m_1 + 4m_3 = 2k$ multiplications and $(2m_1 + 4m_3) = 2k$ additions. This implies that the bilinear z-transform derived filter has one more addition and the same multiplication as those of the new filter. Here the one multiplication is added for the total gain level factor. And as is seen from Figure 3.5, the computational time delay is longer than that of the new filter as

$$\tau_d = (m_1 + k)\ell_w \tau_a + 2\ell_w \tau_b. \quad (6-14)$$

It is because there are two additional and one multiplicative channel in each first-order digital filter element and there are two additional and one multiplicative channels in each second-order digital filter. But each multiplication is performed before the input has been sampled and during the gain multiplication. Since the serial machine is used, each multiplicative channel has two word times, the computational time delay becomes as equation (6-14). The memories needed for storing previous states are k which are the same as those in the new filter.

The matched z-transform derived filter has $2k$ multiplications and $2k - 1$ additions which are the same number as the bilinear z-transform derived filter with the exceptions of one less addition. This is due to the term of T^m in equation (3-39). Hence the computational time delay for this matched z-transform derived filter becomes one less additional time delay than the bilinear z-transform derived filter such that

$$\tau_d = (m_1 + k - 1)\ell_w \tau_a + 2\ell_w \tau_b. \quad (6-15)$$

As results indicated, the discrete Kalman-Bucy derived filter will have the same number of multiplications and additions as the partial fractional digital filters (impulse invariant z-transform derived filter and bilinear z-transform derived filter) or factored matched z-transform derived filter, and less numbers than factored bilinear z-transform derived digital filter. And the computational time delay is about the same as the impulse invariant z-transform derived filter, and is smaller than those of the others. These will be discussed again with examples.

B. Fidelity of Discrete Kalman-Bucy Derived Filter

Now the accuracy of the discrete Kalman-Bucy derived filter is compared to its original continuous filter and z-transform derived filters. Frequency response functions are employed for this fidelity purpose.

To obtain the amplitude response with respect to frequency in a continuous filter, the Laplace transform might be applied to the filter response function and its magnitude gives the frequency response of this

filter in usual manner. However, an immediate difficulty is encountered when this idea is applied to the discrete filter. If the filter response function is given in discrete form, this ends up as an equation with delay operators. As a common method, a w -transform is used. But this gives only a continuous correspondence to the discrete filter transfer function, and this can not be ideal as the frequency becomes high when compared to the sampling frequency.

The difficulty just described can be obviated by defining a frequency response in amplitude as

$$|\text{transfer function}| = \left| \frac{\text{fundamental of output } x(t)}{\text{amplitude of input } y(t)} \right|. \quad (6-16)$$

When a sinusoidal function is an input of the continuous filter, this definition gives a clear concept of the frequency response. Consequently, in the discrete filter, the frequency response can be evaluated in this manner. When a low-range frequency response (inside the quarter-sampling frequency) is of interest, the w -transform method gives an algebraic representation which will be described soon.

To convert the continuous filter transfer function to the frequency response function, it is only necessary to make a substitution of variable. The reason for this is that, if the transfer function is known, the steady-state response to a sinusoidal driving function is described by the transfer function where s is replaced by $j\omega$. The frequency response function of equation (4-1) is, therefore,

$$W(j\omega, t) = \frac{N(j\omega, t)}{D(j\omega, t)} \quad . \quad (6-17)$$

Equation (6-17) will give the original continuous filter amplitude which will be compared with the corresponding sampled-data digital filters. The identity of the phase which is the phase associated with the continuous filter will not be considered in this dissertation.

In order for the amplitude response to be applicable to the discrete Kalman-Bucy derived filter, it is necessary to write the digital filter transfer function $W_N(z)$ in the form of a frequency response function by substituting

$$z = \frac{1 + w}{1 - w} \quad (6-18)$$

into the function for $W_N(z)$ which will be represented by $W_N(w)$. Since the imaginary value of w is mapped from $s = j\omega$ in the zero strip, the frequency response function can be derived from $W_N(w)$ by letting w take on imaginary values, and, thus $w = j\sigma$, the amplitude response is evidently applicable to $W_N(j\sigma)$ by direct analogy to the familiar application in the realm of continuous filters.

The next step in developing the frequency response by use of the w -transform is to relate the dimensionless frequency σ to the dimensional frequency ω . From relationships (6-18) and $w = j\sigma$, it follows that

$$j\sigma = \frac{e^{j\omega T} - 1}{e^{j\omega T} + 1} \quad (6-19)$$

By use of the identities, $\tanh(j\omega T/2) = (e^{j\omega T} - 1)/(e^{j\omega T} + 1) = j \tan(\omega T/2)$, the expression for the dimensionless frequency becomes $\sigma = \tan(\omega T/2)$. Using $T = 2\pi/\omega_0$, where ω_0 is a sampling frequency, an alternate form is obtained:

$$\sigma = \tan \frac{\omega}{\omega_0} \pi . \quad (6-20)$$

As can be seen from the above equation, the full range of σ , $-\infty < \sigma < \infty$, corresponds to the ω range $-\omega_0/2 < \omega < \omega_0/2$.

Amplitude response of the equation (6-17), which will have a frequency response function derived by a w -transform method, can be compared to that of the equation (6-16) by using a numerical example. Obviously both responses will be about the same within the quarter-sampling frequency when equation (6-16) is processed by the full word-length machine. Those responses will be discussed in the following subsection.

C. Examples

Four examples are presented to illustrate the use of the discrete Kalman-Bucy derived filter design technique. It will be assumed that continuous time-fixed parameter filters: the first-order filter, the second-order filter and the third-order filter, are given, and also that a first-order time-varying parameter filter is given. Those examples are of very limited practical interest, but they are useful in conveying insight into the sampled-data digital filter design methods.

First-order filter with a real pole and second-order filter with complex conjugate poles which are time fixed become elements of z -transform derived filters as given by equation (3-21) when the continuous filter transfer function can be given in partial fraction form. The third-order time-fixed parameter filter has to be rearranged so that it

is composed of the first-order filter with a real pole and the second-order filter with complex conjugate poles. But the discrete Kalman-Bucy derived filter design method does not need to do this.

Comparisons are made with regard to two aspects. The first is concerned with the amount of digital hardware required for the discrete Kalman-Bucy derived filter and z-transform derived filters. The second aspect is the fidelity of each in frequency response of amplitude.

For the purpose of demonstrating an amplitude response of the new filter design method to the time-varying parameter continuous filter, a simple first-order time-varying parameter filter is considered in the first example. This is called a finite-time averaging filter. As it will be seen in Example 1, this finite-time averaging filter is an exceptional case of the time-varying denominator in the transfer function being the fixed-parameter new filter. Even though there is not any time-varying parameter in the denominator of this averaging filter, the time-varying covariance matrix is obtained. This is due to the only single s term in the denominator.

Since the developed procedure works on the time-varying parameter system matrix $A(t)$, the k th-order continuous filter also can be transformed into a sampled-data digital filter form. In this case, the only problem occurred is in the second step which solves the covariance equation (4-39). A digital computer will be introduced to solve these $k(k + 1)/2$ first-order linear differential equations and will give the equilibrium covariance matrix $\bar{P}(t|t)$ in function of time. However this dissertation will be concerned only with the simple case.

1. Example 1¹

A finite-time averaging filter, which weights all of the past input data from t back to zero uniformly, will be considered in this example. The block diagram and the desired weighting function are shown in Figure 6.3 and Figure 6.4, respectively. The mathematical expression for this weighting function is

$$\omega(\tau, t) = \begin{cases} \frac{1}{t} & \text{for } 0 \leq \tau \leq t, \\ 0 & \text{for } \tau < 0 \text{ or } \tau > t, \end{cases} \quad (6-21)$$

and the Laplace transform with respect to the age variable τ is

$$W(s, t) = \begin{cases} \frac{1}{t} \frac{1}{s} & \text{for } 0 \leq \tau \leq t, \\ 0 & \text{for } \tau < 0 \text{ or } \tau > t. \end{cases} \quad (6-22)$$

Problem in this example is to find the corresponding discrete Kalman-Bucy derived filter.

According to the developed steps, the discrete parameters are estimated as follows:

$$1) \quad a(t) = 0$$

$$m(t) = \frac{1}{t}$$

$$2) \quad \dot{\bar{p}}(t|t) - a(t)\bar{p}(t|t) - \bar{p}(t|t)a(t) - 2 = 0$$

$$\text{therefore } \bar{p}(t|t) = 2t$$

¹This example is taken from Brown and Nilsson (5).

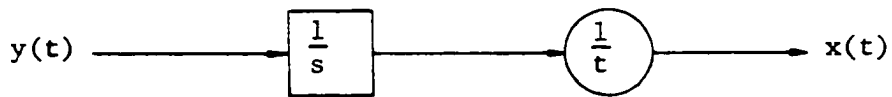


Figure 6.3. Finite-time averaging filter

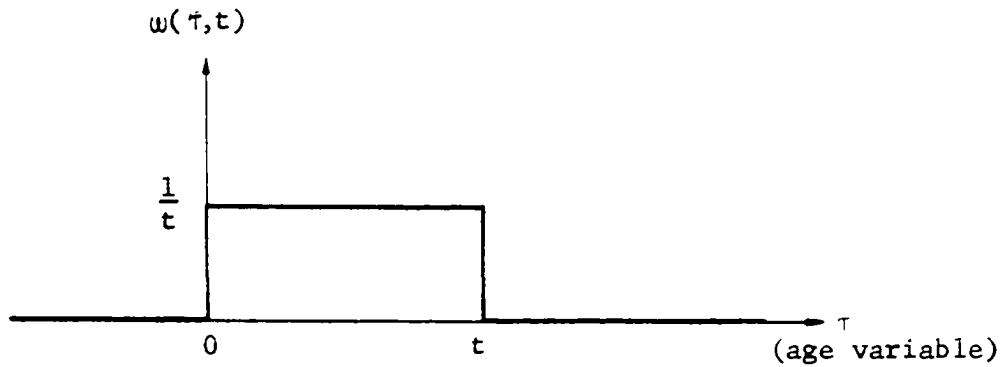


Figure 6.4. Weighting function for the finite-time averaging filter .

$$\text{hence } \bar{p}^{-1}(t|t) = \frac{1}{2t} \quad \text{for } t > 0$$

$$3) \quad f(t) = a(t) + \bar{p}^{-1}(t|t) = \frac{1}{2t}$$

$$h(t) = \bar{p}^{-1}(t|t) = \frac{1}{2t}$$

$$4) \quad f_n = 1 + Tf(nT) = 1 + \frac{1}{2n}$$

$$h_n = \frac{1}{2nT}$$

$$m_n = \frac{1}{nT} \quad .$$

Therefore the discrete Kalman-Bucy derived filter has the block diagram in Figure 6.5 and equations as follows:

$$\hat{\xi}_n|n = [1 - (\frac{1}{2n})^2] \hat{\xi}_{n-1}|n-1 + Ty_n, \quad (6-23)$$

and

$$x_n = \frac{1}{nT} \hat{\xi}_n|n, \quad (6-24)$$

where

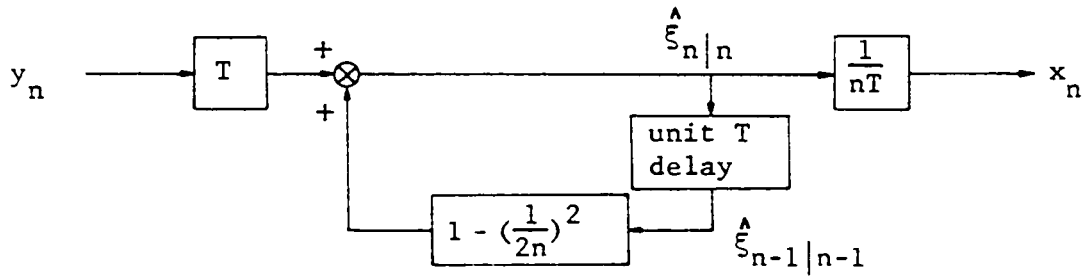
$$\hat{\xi}_0|0 = 0.$$

Equations (6-23) and (6-24) are combined as

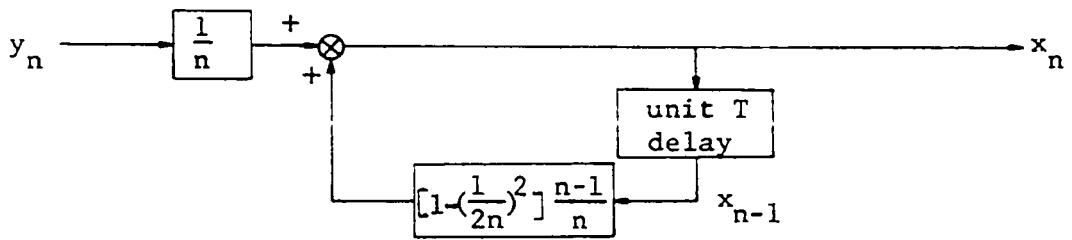
$$x_n = [1 - (\frac{1}{2n})^2] \frac{n-1}{n} x_{n-1} + \frac{1}{n} y_n, \quad (6-25)$$

and $x_0 = 0$.

Now this discrete Kalman-Bucy derived filter of the finite-time averaging filter will be compared to the continuous finite-time averaging



a) A discrete Kalman-Bucy derived filter for Example 1



b) A simplified block diagram of a

Figure 6.5. A finite-time averaging filter

filter. The impulse response function of the original filter is the inverse Laplace transform of the equation (6-22), and it is given by $1/t$. The response to a unit step input is a unit step function. The impulse response function of the new sampled-data digital filter (6-25) is obtained with $y_1 = 1/T$ and $y_i = 0$ for $i = 2, 3, \dots$. The unit step response function is obtained by $y_i = 1$ for $i = 1, 2, \dots$. They are plotted in Figure 6.6 and Figure 6.7.

As can be seen from the equation (6-25), the realization of this time-varying parameter filter needs two time varying adaptive elements, which will track the time-varying coefficients, two multipliers and one adder. And one memory is needed for storing the previous state.

It will be noticed that z-transform derived filters can not be applied to this time-varying parameter filter.

2. Example 2

A first-order real pole filter which has time-fixed parameters α_0 and β_0 will be discussed. As can be seen from equations (3-21) and (3-33), this first-order filter is a kind of basic filter in z-transform derived filters. The transfer function of the continuous first-order filter is

$$W(s) = \frac{\beta_0}{s + \alpha_0} . \quad (6-26)$$

From equations (3-22) and (3-37), the impulse invariant z-transform derived filter and the matched z-transform derived filter have the sampled-data digital filter transfer function

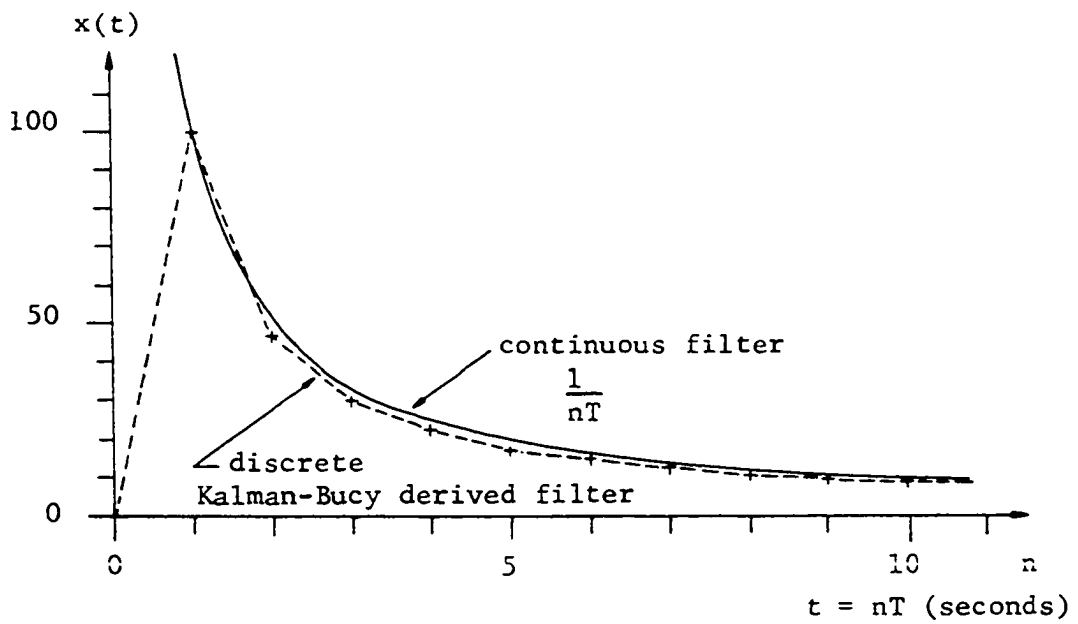


Figure 6.6. Impulse responses of the finite-time averaging filter when $T = 0.01$ second

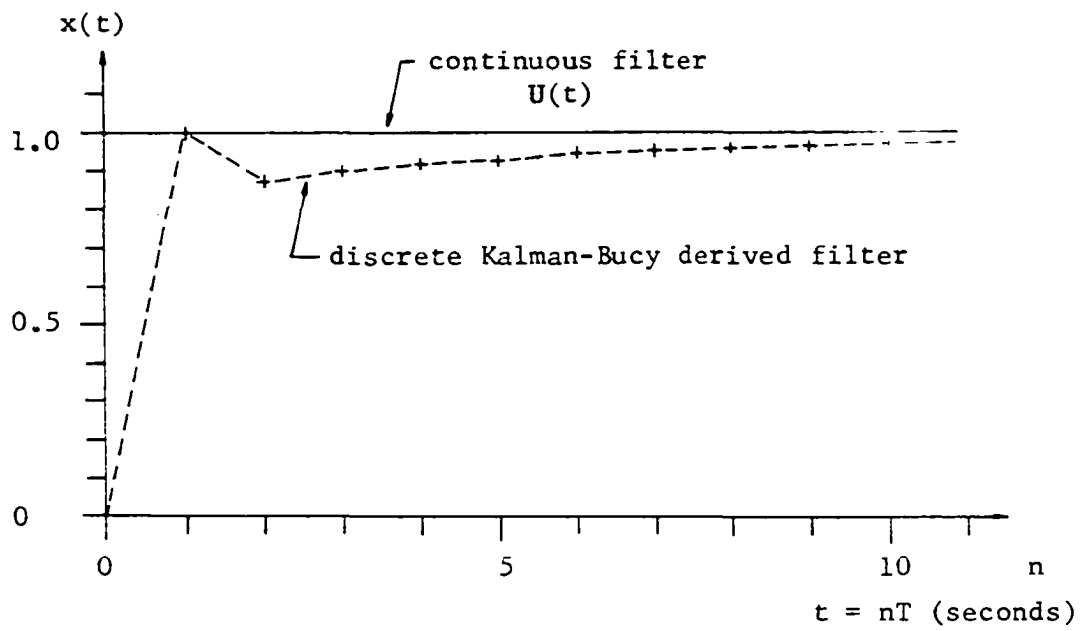


Figure 6.7. Unit step responses of the finite-time averaging filter when $T = 0.01$ second

$$W_I(z) = W_M(z) = \frac{\beta_o T}{1 - e^{-\alpha_o T} z^{-1}} \quad (6-27)$$

The bilinear z-transform derived filter corresponding to the equation (6-26) is, from the equation (3-27),

$$W_B(z) = \frac{\frac{\beta_o T}{2(1 + \frac{\alpha_o T}{2})} (1 + z^{-1})}{1 - \frac{1 - \frac{\alpha_o T}{2}}{1 + \frac{\alpha_o T}{2}} z^{-1}} \quad (6-28)$$

The discrete Kalman-Bucy derived filter for the given continuous filter (6-26) is obtained by the following steps.

$$1) \quad a = -\alpha_o$$

$$m = v_o$$

$$2) \quad a\bar{p}(t|t) + \bar{p}(t|t)a' + 2 = 0$$

$$\text{therefore } \bar{p}(t|t) = \frac{1}{\alpha_o}$$

$$\text{hence } \bar{p}^{-1}(t|t) = \alpha_o$$

$$3) \quad \Gamma = a + \bar{p}^{-1}(t|t) = 0$$

$$h = \bar{p}^{-1}(t|t) = \alpha_o$$

$$4) \quad l_n = 1 + T\Gamma = 1$$

$$h_n = v_o$$

$$m_n = \beta_o$$

Therefore the new filter equations are

$$\hat{\xi}_n|n = (1 - T\alpha_o)\hat{\xi}_{n-1}|n-1 + Ty_n, \quad (6-29)$$

and

$$x_n = \beta_o \hat{\xi}_n|n, \quad (6-30)$$

where $\hat{\xi}_0|0 = 0$.

Substituting x_n into the first equation, a simplified equation results as

$$x_n = (1 - \alpha_o T)x_{n-1} + \beta_o Ty_n. \quad (6-31)$$

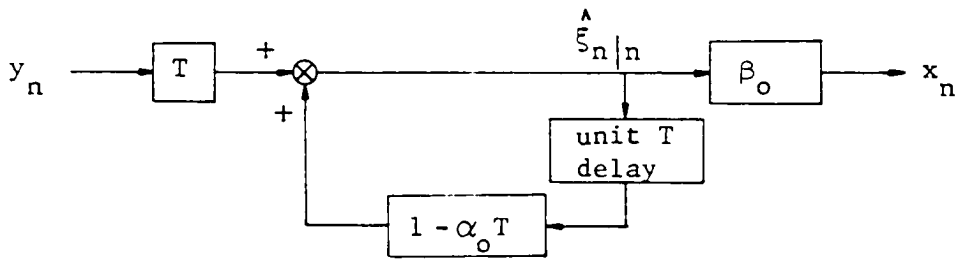
It is noted here that equation (6-31) is a first-order simplification of the equation (6-27) using a series expansion of the exponential term. Block diagrams of the resulted equations (6-29), (6-30) and (6-31) are illustrated in Figure 6.8.

The numbers of multiplications and additions for this simplified new filter are the same as those for the impulse invariant z-transform derived filter and the matched z-transform derived filter which need two multipliers, one adder and one memory. The bilinear z-transform derived filter needs two multipliers, two adders and one memory.

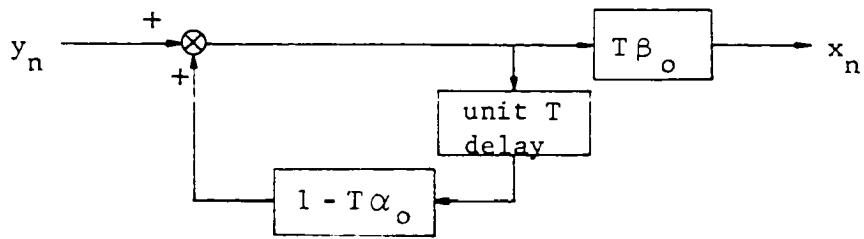
From equations (6-11) and (6-18), the w-transform of the discrete Kalman-Bucy derived filter becomes, with $w = j\sigma$,

$$W_N(j\sigma) = \frac{\beta_o}{\alpha_o} \frac{1 + j\sigma}{1 + \left(\frac{2}{\alpha_o T} - 1\right)j\sigma}. \quad (6-32)$$

where $\sigma = \tan \frac{\omega}{\omega_o} \pi$.



a) A block diagram for equations (6-29) and (6-30)



b) A simplified representation of a

Figure 6.8. Block diagrams of the first-order time-fixed discrete Kalman-Bucy derived filter

The frequency response of the continuous filter (6-26) is

$$W(j\omega) = \frac{\beta_o}{\alpha_o} \frac{1}{1 + \frac{1}{\alpha_o} j\omega}, \quad (6-33)$$

and frequency responses of z-transform derived filters are

$$W_I(j\sigma) = W_M(j\sigma) = \frac{\beta_o T}{1 - e^{-\alpha_o T}} \frac{1 + j\sigma}{1 + \frac{1 + e^{-\alpha_o T}}{1 - e^{-\alpha_o T}} j\sigma}, \quad (6-34)$$

and

$$W_B(j\sigma) = \frac{\beta_o}{\alpha_o} \frac{1}{1 + \frac{2}{\alpha_o T} j\sigma}, \quad (6-35)$$

where $\sigma = \tan \frac{\omega}{\omega_o} \pi$.

For the small T, the break frequencies of those sampled-data digital filter poles are

$$\omega_N = \frac{\omega_o}{\pi} \tan^{-1} \frac{1}{\frac{2}{\alpha_o T} - 1} \quad \text{in the new filter,}$$

$$\omega_I = \omega_M = \frac{\omega_o}{\pi} \tan^{-1} \frac{1}{\frac{1 + e^{-\alpha_o T}}{1 - e^{-\alpha_o T}}} \quad \text{in the impulse invariant and matched z-transform derived filters,}$$

(6-36)

$$\omega_B = \frac{\omega_o}{\pi} \tan^{-1} \frac{1}{\frac{2}{\alpha_o T}} \quad \text{in the bilinear z-transform derived filter.}$$

and they are very close to the continuous filter break frequency $\omega_c = \alpha_o$. The break frequencies which appeared in the new filter, the impulse invariant z-transform derived filter and the matched z-transform derived filter, become large as the sampling interval becomes small. This follows from

$$\omega_\tau = \frac{\pi}{2T}, \quad (6-37)$$

which is one quarter of the sampling frequency, and hence this break frequency exists outside of our interests.

3. Example 3

A general second-order time-fixed continuous filter given by the equation (6-38) and its block diagram shown in Figure 6.9 will now be considered.

$$W(s) = \frac{\beta_1 s + \beta_o}{s^2 + \alpha_1 s + \alpha_o} \quad (6-38)$$

where parameters α_o , α_1 , β_o and β_1 are assumed to be positive constants, and the denominator has complex conjugate poles. In order to be applicable to z-transform methods, this transfer function is also assumed to be factored as

$$W(s) = \beta_1 \frac{s + \gamma}{(s + \gamma)^2 + \theta^2} + \frac{\beta_o - \gamma\beta_1}{\theta} \frac{\theta}{(s + \gamma)^2 + \theta^2}. \quad (6-39)$$

Then the z-transform derived filters have the following sampled-data digital filter transfer functions:

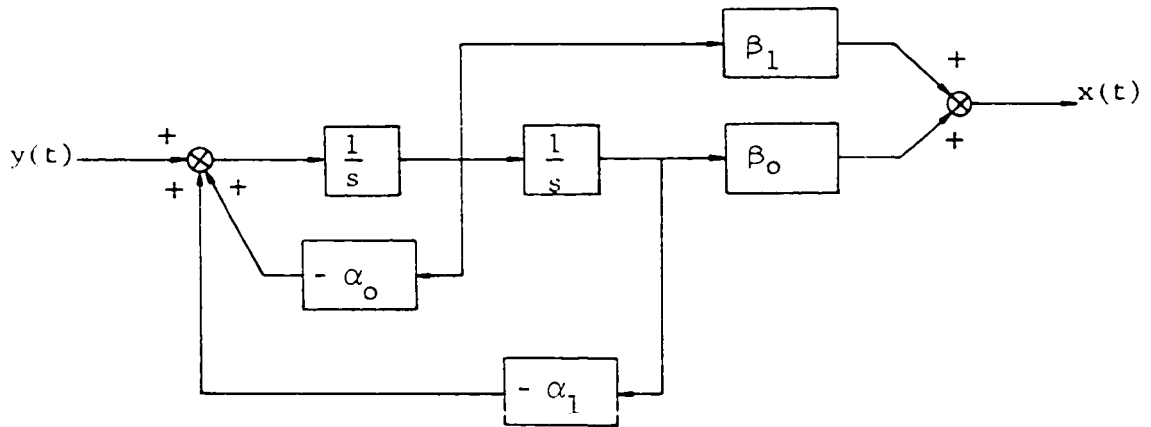


Figure 6.9. A general block diagram of the second-order time-fixed continuous filter

$$W_I(z) = \frac{\beta_1 T \{1 - e^{-\gamma T} (\cos \theta T) z^{-1}\} + \frac{\beta_0 - \gamma \beta_1}{\theta} T e^{-\gamma T} (\sin \theta T) z^{-1}}{1 - 2e^{-\gamma T} (\cos \theta T) z^{-1} + e^{-2\gamma T} z^{-2}}, \quad (6-40)$$

$$W_B(z) = \frac{\frac{\beta_1 T}{2} (1+z^{-1}) \left\{ \left(\frac{\gamma T}{2} + 1 \right) + \left(\frac{\gamma T}{2} - 1 \right) z^{-1} \right\} + \frac{\beta_0 - \gamma \beta_1}{4} T^2 (1+z^{-1})^2}{\left(\frac{\gamma^2 - \theta^2}{4} T^2 + \gamma T + 1 \right) + 2 \left(\frac{\gamma^2 + \theta^2}{4} T^2 - 1 \right) z^{-1} + \left(\frac{\gamma^2 + \theta^2}{4} T^2 - \gamma T + 1 \right) z^{-2}} \quad (6-41)$$

and

$$W_M(z) = \frac{\beta_1 T (1 - e^{-\gamma T} z^{-1}) + (\beta_0 - \gamma \beta_1) T^2}{1 - 2e^{-\gamma T} (\cos \theta T) z^{-1} + e^{-2\gamma T} z^{-2}}. \quad (6-42)$$

Now the discrete Kalman-Bucy derived filter is obtained from the equation (6-38) as follows:

$$1) \quad A = \begin{bmatrix} 0 & 1 \\ -\alpha_0 & -\alpha_1 \end{bmatrix}$$

$$\underline{M} = [\beta_0 \quad \beta_1]$$

$$2) \quad A\bar{P} + \bar{P}A' + \begin{bmatrix} 0 & 0 \\ 0 & 2 \end{bmatrix} = 0$$

$$\text{therefore } \bar{P} = \begin{bmatrix} \frac{1}{\alpha_0 \alpha_1} & 0 \\ 0 & \frac{1}{\alpha_1} \end{bmatrix}$$

$$\text{hence } \bar{P}^{-1} = \begin{bmatrix} \alpha_0 \alpha_1 & 0 \\ 0 & \alpha_1 \end{bmatrix}$$

$$3) F = A + \begin{bmatrix} 0 & 0 \\ 0 & 1 \end{bmatrix} \bar{P}^{-1} = \begin{bmatrix} 0 & 1 \\ -\alpha_0 & 0 \end{bmatrix}$$

$$\underline{H} = [0 \quad 1] \bar{P}^{-1} = [0 \quad \alpha_1]$$

$$4) F_n = I + TF = \begin{bmatrix} 1 & T \\ -\alpha_0 T & 1 \end{bmatrix}$$

$$\underline{H}_n = [0 \quad \alpha_1]$$

$$\underline{M}_n = [\beta_0 \quad \beta_1].$$

The discrete Kalman-Bucy derived filter equations are

$$\begin{bmatrix} \hat{\xi}_1 \\ \hat{\xi}_2 \end{bmatrix}_{n|n} = \begin{bmatrix} 1 & T \\ -\alpha_0 T(1 - \alpha_1 T) & 1 - \alpha_1 T \end{bmatrix} \begin{bmatrix} \hat{\xi}_1 \\ \hat{\xi}_2 \end{bmatrix}_{n-1|n-1} + \begin{bmatrix} 0 \\ T \end{bmatrix} y_n, \quad (6-43)$$

and

$$x_n = [\beta_0 \quad \beta_1] \begin{bmatrix} \hat{\xi}_1 \\ \hat{\xi}_2 \end{bmatrix}_{n|n}. \quad (6-44)$$

The sampled-data digital filter transfer function of the above equations is

$$W_N(z) = \frac{\beta_1 T + (\beta_0 T^2 - \beta_1 T)z^{-1}}{1 - (2 - \alpha_1 T)z^{-1} + (1 - \alpha_1 T)(1 + \alpha_0 T^2)z^{-2}} \quad (6-45)$$

The block diagrams for the new digital filter are shown in Figure 6.10.

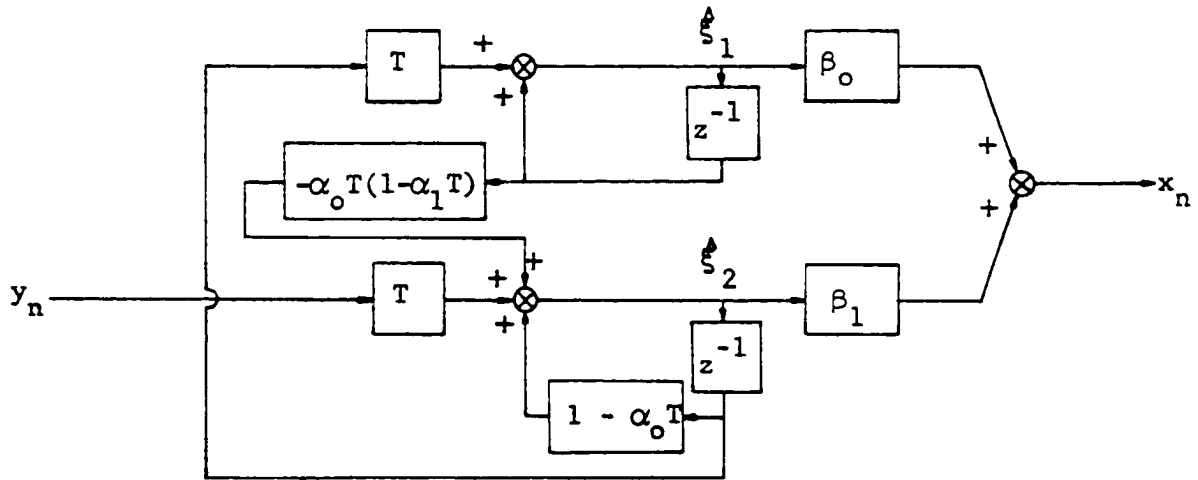
The number of additions and multiplications for this new filter realization is compared to those of equations (6-40), (6-41) and (6-42). The new filter realization needs three adders, four multipliers and two memories which are the same as in the impulse invariant z-transform derived filter and the matched z-transform derived filter, and which is one less adder than in the bilinear z-transform derived filter.

The frequency response of the new filter transfer function (6-45) is solved to be

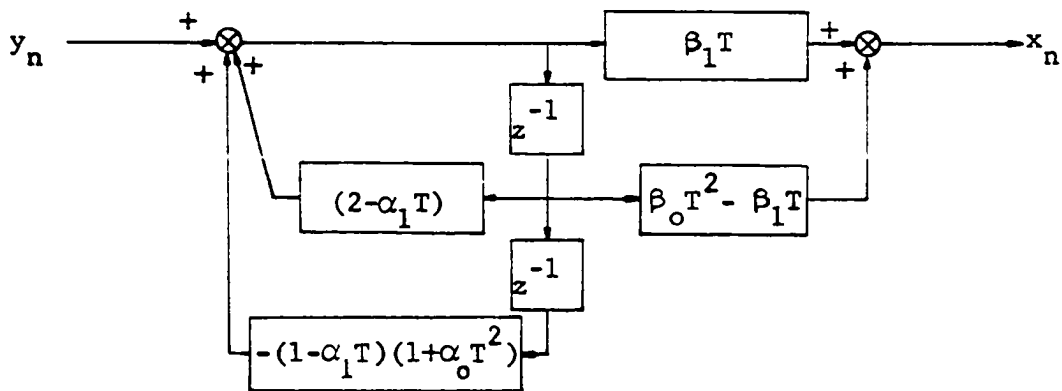
$$W_N(j\omega) = \frac{(2\beta_1 T - \beta_0 T^2)(j\omega)^2 + 2\beta_1 T j\omega + \beta_0 T^2}{\{4 - 2\alpha_1 T + \alpha_0 T^2(1 - \alpha_1 T)\}(j\omega)^2 + 2\{\alpha_1 T - \alpha_0 T^2(1 - \alpha_1 T)\}j\omega + \alpha_0 T^2(1 - \alpha_1 T)} \quad (6-46)$$

While frequency response functions of z-transform derived filters are

$$W_I(j\omega) = \frac{\{\beta_1 + \beta_1 e^{-\gamma T} \cos \theta T - \frac{\beta_0 - \gamma \beta_1}{\theta} e^{-\gamma T} \sin \theta T\} T (j\omega)^2 + 2\beta_1 T j\omega + \{\beta_1 - \beta_1 e^{-\gamma T} \cos \theta T + \frac{\beta_0 - \gamma \beta_1}{\theta} e^{-\gamma T} \sin \theta T\} T}{(1 + 2e^{-\gamma T} \cos \theta T + e^{-2\gamma T})(j\omega)^2 + 2(1 - e^{-2\gamma T})j\omega + (1 - 2e^{-\gamma T} \cos \theta T + e^{-2\gamma T})} \quad (6-47)$$



(a) A block diagram for equations (6-43) and (6-44)



(b) A simplified representation of (a)

Figure 6.10. Block diagrams of a second-order time-fixed discrete Kalman-Bucy derived filter

$$W_B(j\sigma) = \frac{2\beta_1 T j\sigma + \beta_0 T^2}{4(j\sigma)^2 + 4\gamma T j\sigma + (\gamma^2 + \theta^2) T^2}, \quad (6-48)$$

and

$$W_M(j\sigma) = \frac{\{\beta_1 T(1 - \gamma T + e^{-\gamma T}) + \beta_0 T^2\} (j\sigma)^2 + 2\{\beta_1 T(1 - \gamma T) + \beta_0 T^2\} j\sigma + \{\beta_1 T(1 - \gamma T - e^{-\gamma T}) + \beta_0 T^2\}}{(1 + 2e^{-\gamma T} \cos \theta T + e^{-2\gamma T}) (j\sigma)^2 + 2(1 - e^{-2\gamma T}) j\sigma + (1 - 2e^{-\gamma T} \cos \theta T + e^{-2\gamma T})} \quad (6-49)$$

where

$$\sigma = \tan \frac{\omega}{\omega_0} \pi.$$

Fidelity of each filter is compared using equations (6-46), (6-47), (6-48), (6-49) and the continuous filter frequency response function. Break frequencies in each filter response equation have about the same positions as those of the continuous filter, and, hence, they have the same responses as the continuous filter inside the quarter-sampling frequency.

4. Example 4

Now to be considered is a three-pole Butterworth low-pass filter with a cutoff frequency $\omega_c = 1$. The continuous filter transfer function is given as

$$W(s) = \frac{1}{s^3 + 2s^2 + 2s + 1}. \quad (6-50)$$

Its block diagram is shown in Figure 6.11.

The discrete Kalman-Bucy derived filter is obtained as follows:

$$1) \quad A = \begin{bmatrix} 0 & 1 & 0 \\ 0 & 0 & 0 \\ -1 & -2 & -2 \end{bmatrix}$$

$$\underline{M} = [1 \quad 0 \quad 0]$$

$$2) \quad \bar{A}\bar{P} + \bar{P}A' + \begin{bmatrix} 0 & 0 & 0 \\ 0 & 0 & 0 \\ 0 & 0 & 2 \end{bmatrix} = 0$$

$$\bar{P} = \begin{bmatrix} \frac{2}{3} & 0 & -\frac{1}{3} \\ 0 & \frac{1}{3} & 0 \\ -\frac{1}{3} & 0 & \frac{2}{3} \end{bmatrix} \quad \bar{P}^{-1} = \begin{bmatrix} 2 & 0 & -3 \\ 0 & 3 & 0 \\ -3 & 0 & 2 \end{bmatrix}$$

$$3) \quad F = A + \begin{bmatrix} 0 & 0 & 0 \\ 0 & 0 & 0 \\ 0 & 0 & 1 \end{bmatrix} \bar{P}^{-1} = \begin{bmatrix} 0 & 1 & 0 \\ 0 & 0 & 1 \\ 0 & -2 & 0 \end{bmatrix}$$

$$\underline{H} = [0 \quad 0 \quad 1] \bar{P}^{-1} = [1 \quad 0 \quad 2]$$

$$4) \quad F_n = I + TF = \begin{bmatrix} 1 & T & 0 \\ 0 & 1 & T \\ 0 & -2T & 1 \end{bmatrix}$$

$$\underline{H}_n = [1 \quad 0 \quad 2]$$

$$\underline{M}_n = [1 \quad 0 \quad 0].$$

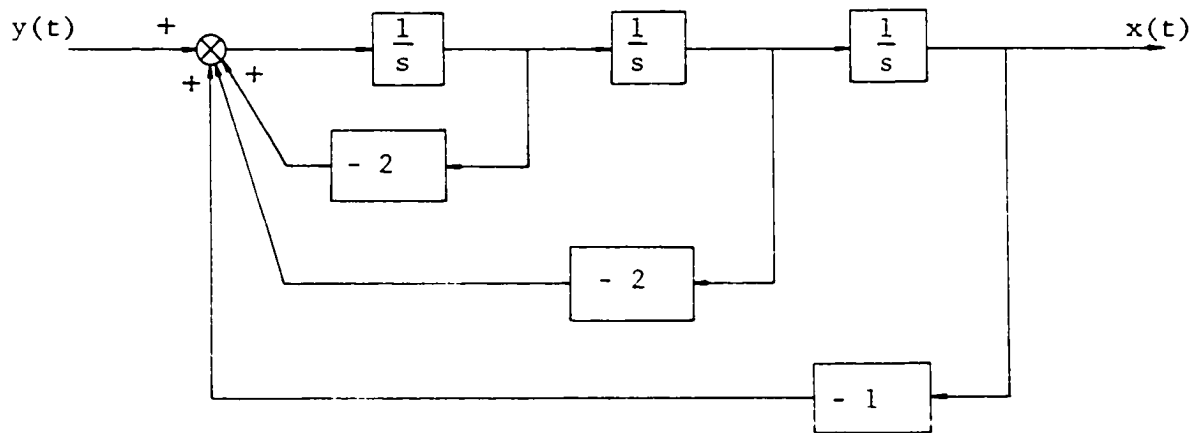


Figure 6.11. A block diagram of a three-pole Butterworth low-pass filter

The discrete Kalman-Bucy derived filter equations are

$$\begin{bmatrix} \hat{\xi}_1 \\ \hat{\xi}_2 \\ \hat{\xi}_3 \end{bmatrix}_{n|n} = \begin{bmatrix} 1 & T & 0 \\ 0 & 1 & T \\ -T & -2T+3T^2 & 1-2T \end{bmatrix} \begin{bmatrix} \hat{\xi}_1 \\ \hat{\xi}_2 \\ \hat{\xi}_3 \end{bmatrix}_{n-1|n-1} + \begin{bmatrix} 0 \\ 0 \\ 1 \end{bmatrix} y_n \quad (6-51)$$

and

$$x_n = [1 \quad 0 \quad 0] \begin{bmatrix} \hat{\xi}_1 \\ \hat{\xi}_2 \\ \hat{\xi}_3 \end{bmatrix}_{n|n}, \quad (6-52)$$

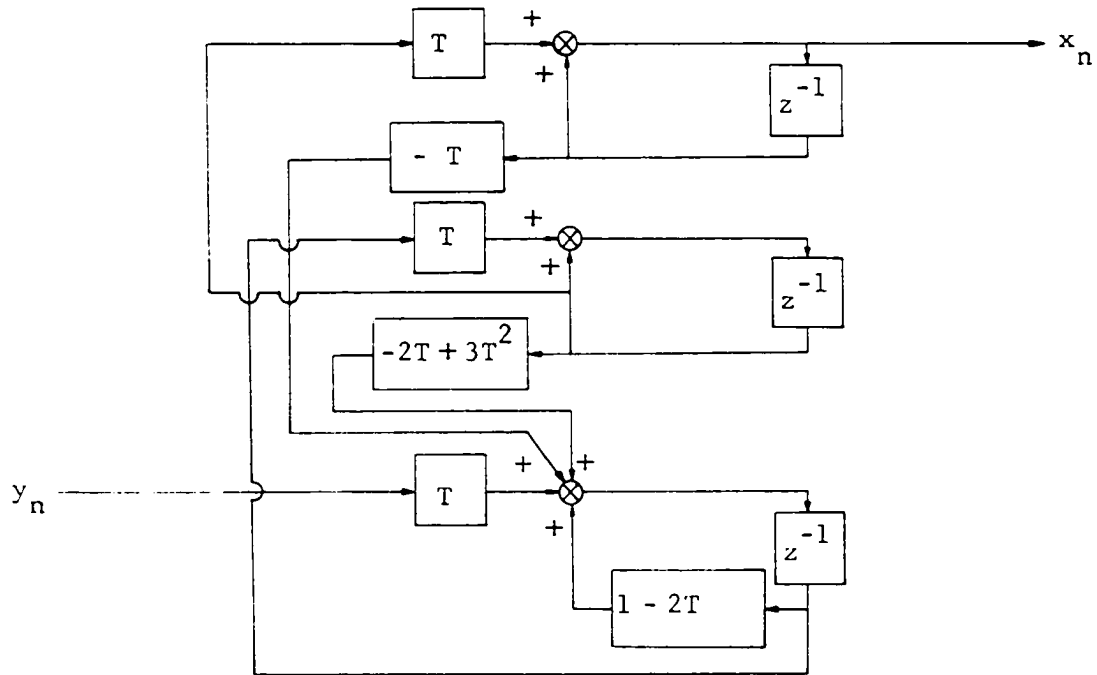
where the initial conditions are given with zeros such as

$$\begin{bmatrix} \hat{\xi}_1 \\ \hat{\xi}_2 \\ \hat{\xi}_3 \end{bmatrix}_{0|0} = \begin{bmatrix} 0 \\ 0 \\ 0 \end{bmatrix}.$$

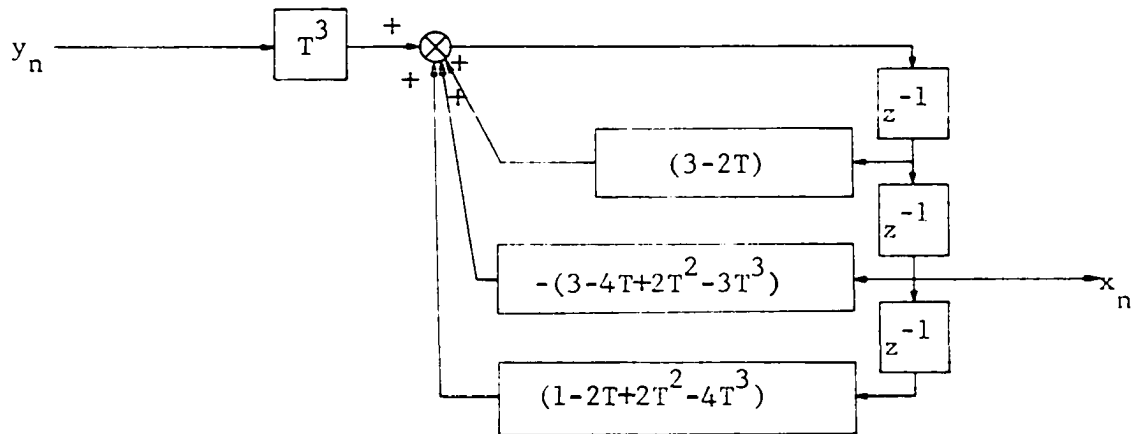
Equations (6-51) and (6-52) are combined in the sampled-data digital filter transfer function form as

$$W_N(z) = \frac{T^3 z^{-2}}{1 - (3 - 2T)z^{-1} + (3 - 4T + 2T^2 - 3T^3)z^{-2} - (1 - 2T + 2T^2 - 4T^3)z^{-3}} \quad (6-53)$$

where z^{-1} is a delay operator. The block diagrams of equations (6-51), (6-52) and (6-53) are shown in Figure 6.12. The frequency response function of this new filter is



a) A block diagram of equations (6-51) and (6-52)



b) A simplified block diagram of a

Figure 6.12. Block diagrams of three-pole Butterworth low-pass discrete Kalman-Bucy derived filter

$$W_N(j\sigma) = \frac{T^3(1-j\sigma)(1+j\sigma)^2}{T^3 + (4T^2 - 9T^3)j\sigma + (8T - 8T^2 + 15T^3)(j\sigma)^2 + (8 - 8T + 4T^2 - 7T^3)(j\sigma)^3} \quad (6-54)$$

and its approximation for the small T is

$$W_N(j\sigma) \approx \frac{(1-j\sigma)(1+j\sigma)^2}{(1 + \frac{2}{T}j\sigma)\{1 + \frac{2}{T}j\sigma + (\frac{2}{T}j\sigma)^2\}}, \quad (6-55)$$

where $\sigma = \tan \frac{\omega}{\omega_0} \pi$.

It should be noticed here that the denominator of this equation (6-55) is close in form to that of the given continuous filter transfer function.

The impulse invariant z-transform of the continuous filter equation (6-50) is obtained as

$$W_I(z) = \frac{T}{1 - e^{-T}z^{-1}} - \frac{T - Te^{-\frac{1}{2}T}(\cos\frac{\sqrt{3}}{2}T + \frac{1}{\sqrt{3}}\sin\frac{\sqrt{3}}{2}T)z^{-1}}{1 - 2e^{-\frac{1}{2}T}(\cos\frac{\sqrt{3}}{2}T)z^{-1} + e^{-T}z^{-2}}, \quad (6-56)$$

and its frequency response function is

$$W_I(j\omega) = \frac{T\{v_0 + v_1 j\omega + v_2 (j\omega)^2\}(1 + j\omega)}{\{(1 - e^{-T}) + (1 + e^{-T})j\omega\}\{\delta_0 + \delta_1 j\omega + \delta_2 (j\omega)^2\}}, \quad (6-57)$$

where

$$\delta_0 = 1 - 2e^{-\frac{1}{2}T} \cos \frac{\sqrt{3}}{2}T + e^{-T}$$

$$\delta_1 = 2 - 2e^{-T}$$

$$\delta_2 = 1 + 2e^{-\frac{1}{2}T} \cos \frac{\sqrt{3}}{2}T + e^{-T}$$

$$v_0 = 2e^{-T} - e^{-\frac{1}{2}T} \left(\cos \frac{\sqrt{3}}{2}T - \frac{1}{\sqrt{3}} \sin \frac{\sqrt{3}}{2}T \right) - e^{-\frac{3}{2}T} \left(\cos \frac{\sqrt{3}}{2}T + \frac{1}{\sqrt{3}} \sin \frac{\sqrt{3}}{2}T \right)$$

$$v_1 = -2\left\{ e^{-T} - e^{-\frac{3}{2}T} \left(\cos \frac{\sqrt{3}}{2}T + \frac{1}{\sqrt{3}} \sin \frac{\sqrt{3}}{2}T \right) \right\}$$

and

$$v_2 = (e^{-\frac{1}{2}T} - e^{-\frac{3}{2}T}) \cos \frac{\sqrt{3}}{2}T - \frac{1}{\sqrt{3}} (e^{-\frac{1}{2}T} + e^{-\frac{3}{2}T}) \sin \frac{\sqrt{3}}{2}T.$$

The bilinear z-transform derived filter of the continuous filter transfer function (6-50) is

$$W_B(z) = \frac{\frac{T^3}{8(1 + \frac{T}{2})(1 + \frac{T}{2} + \frac{T^2}{4})} (1 + z^{-1})^3}{\left(1 - \frac{1 - \frac{T}{2}}{1 + \frac{T}{2}} z^{-1}\right) \left\{ 1 - 2 \frac{1 - \frac{T^2}{4}}{(1 + \frac{T}{2} + \frac{T^2}{4})} z^{-1} + \frac{(1 - \frac{T}{2} + \frac{T^2}{4})}{(1 + \frac{T}{2} + \frac{T^2}{4})} z^{-2} \right\}} \quad (6-58)$$

and its frequency response function is

$$W_B(j\omega) = \frac{1}{\left(\frac{2}{T}j\omega + 1\right)\left\{\left(\frac{2}{T}j\omega\right)^2 + \frac{2}{T}j\omega + 1\right\}}. \quad (6-59)$$

The matched z-transform derived filter corresponding to the equation (6-50) is

$$W_M(z) = \frac{T^3}{(1 - e^{-T}z^{-1})(1 - 2e^{-\frac{1}{2}T}\cos\frac{\sqrt{3}}{2}Tz^{-1} + e^{-T}z^{-2})}, \quad (6-60)$$

and its corresponding frequency response function is

$$W_M(j\omega) = \frac{T^3(1 + j\omega)^3}{\left\{(1 - e^{-T}) + (1 + e^{-T})j\omega\right\}\left\{(1 - 2e^{-\frac{1}{2}T}\cos\frac{\sqrt{3}}{2}T + e^{-T}) + 2(1 - e^{-T})j\omega + (1 + 2e^{-\frac{1}{2}T}\cos\frac{\sqrt{3}}{2}T + e^{-T})(j\omega)^2\right\}}. \quad (6-61)$$

As hardware requirements, the new filter (6-53) needs four multipliers, three adders and three memories. The z-transform derived filters all require more elements. The impulse invariant z-transform derived filter (6-56) requires six multipliers, four adders and three memories. The bilinear z-transform derived filter (6-58) needs four multipliers, six adders and three memories. And the matched z-transform derived filter (6-60) requires four multipliers, three adders and three memories, which are same numbers as the new digital filter.

The fidelity of each filter is shown in Table 6.1, where the sampling interval is $T = 1/100$ seconds.

Table 6.1. Fidelity comparison (unit db)

frequency	continuous	DK-BD	impulse	bilinear	matched
5	-89.83	-89.67	-89.85	-90.06	-89.67
10	-107.89	-107.73	-107.90	-108.77	-107.73
20	-125.95	-124.12	-126.06	-129.74	-124.13

Equations (6-53), (6-56), (6-58) and (6-60) are used for calculating frequencies of amplitudes. They are compared to the original frequency response function of the given continuous filter. As can be seen from this table, the discrete Kalman-Bucy derived filter has as good fidelity as the others. The quarter-sampling frequency in this example is $f = 25$. The last frequency in the above table is close to this. A sample printout of the computer program used for the responses is shown in Appendix D.

VII. CONCLUSIONS

This dissertation develops a systematic design approach for digital implementation of any continuous filter whose transfer function is given with a ratio of two polynomials.

The new feature of this method, called the discrete Kalman-Bucy derived filter, begins with the equivalent Kalman-Bucy filter which is composed of the fixed-gain scalar-measurement Kalman-Bucy filter and the additional output equation. The state representation of the denominator of the continuous filter transfer function leads to a fixed-gain Kalman-Bucy filter and that of the numerator becomes an output equation. Both transient-state and steady-state behaviors of the equivalent Kalman-Bucy filter are identically conditioned to those of the given analog filter. The input signal of the equivalent Kalman-Bucy filter is supposed to be generated by an imagined stochastic process model. Noise statistics in the stochastic signal and measurement process models are assumed to be simple so that the matrix Riccati-type steady-state covariance equation is represented as a set of simple first-order linear differential equations.

The next step is the discretization of the equivalent Kalman-Bucy filter. The fixed-gain Kalman-Bucy filter is discretized to be a form of the discrete Kalman filter and the output equation is obtained simply by sampling at discrete time. Since the gain is also fixed after discretization and the sampling interval in discretization is small enough, the discretized equivalent Kalman-Bucy filter is simplified,

which is called a discrete Kalman-Bucy derived filter. It is also shown that the new filter produces the same degree of stability as its analog counterpart.

One advantage of this approach is that the procedure obtaining the discrete Kalman-Bucy derived filter is a simple routine matter. It is summarized with four steps:

- 1) State representation of the given continuous filter transfer function (4-1), which gives $A(t)$ and $\underline{M}(t)$ from equations (4-5) and (4-7), respectively.
- 2) Solution of the steady-state covariance equation and its inverse matrix, where the steady-state covariance equation is given by the equation (4-38) in the time-fixed parameter system, or by the equation (4-39) in the time-varying parameter system.
- 3) Continuous parameter determinations, where the continuous parameters $F(t)$ and $\underline{H}(t)$ are obtained from equations (4-19) and (4-20) in the time-fixed parameter system, or from equations (4-26) and (4-27) in the time-varying parameter system, respectively.
- 4) Discrete parameter determinations, where the discrete parameters F_n , \underline{H}_n and \underline{M}_n are obtained from equations (5-27), (5-12) and (5-15), respectively.

Then the discrete Kalman-Bucy derived filter equations are given by equations (5-39) and (5-40), or the simplified equation is obtained from the equation (6-11) which is the desired sampled-data digital filter transfer function.

The other advantage is that the new filter design method can apply to the time-varying parameter continuous filter as well as to the time-fixed parameter continuous filter. The only difference in the design procedure exists in the second step. However, the time-varying parameters in the numerator of the given filter transfer function do not affect the compositions of the new filter. When the denominator of the analog filter consists of time-varying parameters, the steady-state matrix covariance equation in the second step becomes a set of first-order differential equations. While, if the denominator of the given transfer function is time fixed, then the steady-state matrix covariance equation becomes a set of algebraic equations. Since the set of first-order differential equations is linear, the second step will also be simple. As an example of a time-varying parameter filter, a finite-time averaging filter is designed. The impulse response and the step response of this digital filter are compared to its continuous responses and they are shown to be invariant after being discretized.

Another advantage of the new approach is that the resulting discrete Kalman-Bucy derived filter is better in some cases than the z-transform derived filters. Comparisons are made on the hardware requirements of digital implementations and the fidelity of each sampled-data digital filter in terms of frequency response. Since the discrete Kalman-Bucy derived filter equations (5-39) and (5-40) can be combined, the number of multiplications and additions required for the new sampled-data digital filter is equal to or less than any other sampled-data digital

filters. The combined equation (6-11) of the discrete Kalman-Bucy derived filter is compared to other sampled-data digital filter transfer functions.

Three examples which are time-fixed are considered in this dissertation. A first-order real pole filter designed by the new approach has the same amplitude response as that of the continuous filter within the quarter-sampling frequency as shown in the equations (6-36) and (6-37). In this case, the number of hardware requirements is the same as that of the impulse invariant z-transform derived filter and that of the matched z-transform derived filter, and is smaller than that of the bilinear z-transform derived filter. A second-order filter which has complex conjugate poles is designed by the discrete Kalman-Bucy derived filter approach. The amplitude frequency response in this example has satisfactory results when compared to its original continuous filter response within the quarter-sampling frequency. The figure of hardware requirements in this new filter is the same as in the case of the first-order filter. That is, the number of hardware requirements is the same as that of the impulse invariant z-transform derived filter, and that of the matched z-transform derived filter. And a three-pole Butterworth low-pass filter is designed by the discrete Kalman-Bucy derived filter method. The frequency response function in this example shows that the new filter design approach can be applied to higher-order continuous filters, and the responses of discrete Kalman-Bucy derived filters are very close to the original continuous responses. The number of hardware requirements in this third-order filter are less than those of the impulse

invariant z-transform derived filter, or those bilinear z-transform derived filter, and the matched z-transform derived filter. By these examples, the results of the discrete Kalman-Bucy filter approach are demonstrated to be satisfactory.

VIII. LITERATURE CITED

1. Blackman, R. B., Linear Data Smoothing and Prediction in Theory and Practice, Reading, Mass., Addison-Wesley, 1965.
2. Bode, H. W. and Shannon, C. B., "A simplified derivation of linear least square smoothing and prediction theory," Proc. IRE, vol. 38, pp. 417-425, April 1950.
3. Boxer, R. and Thaler, S., "A simplified method of solving linear and nonlinear systems," Proc. IRE, vol. 44, pp. 89-101, January 1956.
4. Brown, R. G., "Kalman filter notes," unpublished class notes, Electrical Engineering Dept., Iowa State University, June 1962.
5. Brown, R. G. and Nilsson, J. W., Introduction to Linear System Analysis, New York, N. Y., John Wiley & Sons, 1962.
6. Burrs, C. S. and Parks, T. W., "Time domain design of recursive digital filters," IEEE Trans. Audio and Electroacoustics, vol. AU-18, pp. 137-141, June 1970.
7. Carney, R., "Design of a digital notch filter with tracking requirements," IEEE Trans. Space Electronics and Telemetry, vol. SET-9, pp. 109-114, December 1963.
8. Davenport, W. B., Jr. and Root, W. L., An Introduction to the Theory of Random Signal and Noise, New York, N. Y., McGraw-Hill, 1958.
9. Gold, B. and Rader, C. M., "Effects of quantization noise in digital filters," Proc. IEEE, vol. 55, pp. 688-689, May 1967.
10. Golden, R. M., "Digital filter synthesis by sampled-data transformation," IEEE Trans. Audio and Electroacoustics, vol. AU-16, pp. 321-329, September 1968.
11. Golden, R. M. and Kaiser, J. F., "Design of wide-band sampled-data filters," Bell Sys. Tech. J., vol. 43, pt. 2, pp. 1533-1546, July 1964.
12. Griffin, R. E. and Sage, A. D., "Large and small scale sensitivity analysis of optimum algorithms," IEEE Trans. Automatic Control, vol. AC-13, pp. 320-329, August 1968.

13. Jackson, L. B., Kaiser, J. F. and McDonald, H. S., "An approach to the implementation of digital filters," IEEE Trans. Audio and Electroacoustics, vol. AU-16, pp. 413-421, September 1968.
14. James, M., Nichols, N. B. and Phillips, R. S., Theory of Servomechanics, New York, N. Y., McGraw-Hill, 1947.
15. Johnson, C. W., Lindorff, D. P. and Nordling, C. G. A., "Extension of continuous-data system design techniques to sampled-data control systems," Trans. AIEE, vol. 74, pt. 2, pp. 252-263, September 1955.
16. Jury, E. I., Theory and Application of the z-transform Method, New York, N. Y., John Wiley and Sons, 1964.
17. Kaiser, J. F., "Design methods for sampled-data filters," Proc. First Allerton Conf. on Circuit and System Theory, pp. 221-236, Monticello, Ill., November 1963.
18. Kaiser, J. F., "Some practical consideration in the realization of linear digital filters," Proc. Third Allerton Conf. on Circuit and System Theory, pp. 621-633, Monticello, Ill., October 1965.
19. Kalman, R. E., "A new approach to linear filtering and prediction problems," Trans. ASME J. Basic Engrg. Ser. D, vol. 82, pp. 35-45, March 1960.
20. Kalman, R. E., "New methods in Wiener filter theory," Proc. of the First Symposium on Engineering Application of Random Function Theory and Probability, Edited by Bogdanoff, J. L. and Kozin, F., New York, N. Y., John Wiley and Sons, 1963.
21. Kalman, R. E. and Bucy, R., "New results in linear filtering and prediction theory," Trans. ASME J. Basic Engrg. Ser. D, vol. 3, pp. 95-108, March 1961.
22. Knowles, J. B. and Olcayto, E. M., "Coefficient accuracy and digital filter response," IEEE Trans. Circuit Theory, vol. CT-15, pp. 31-41, March 1968.
23. Lansing, J. L. and Battin, R. H., Random Processes in Automatic Control, New York, N. Y., McGraw-Hill, 1956.
24. Lindorff, D. P., Theory of Sampled-data Control Systems, New York, N. Y., John Wiley and Sons, 1965.
25. Nishmura, T., "Error bounds of continuous Kalman filters and the application to orbit determination problems," IEEE Trans. Automatic Control, vol. AC-12, pp. 268-275, June 1967.

26. Otens, R. K., "An elementary design procedure for digital filters," IEEE Trans. Audio and Electroacoustics, vol. AU-16, pp. 330-335, September 1968.
27. Parzen, E., Stochastic Processes, San Francisco, Cal., Holden Day, 1962.
28. Penrose, R., "On best approximate solutions of linear matrix equations," Proc. Camb. Phil. Soc. vol. 52, pp. 17-19, January 1956.
29. Price, C. F., "An analysis of the divergence problem in the Kalman filter," IEEE Trans. Automatic Control, vol. AC-13, pp. 696-702, December 1968.
30. Rader, C. M., "On digital filtering," IEEE Trans. Audio and Electroacoustics, vol. AU-16, pp. 303-314, September 1968.
31. Rader, C. M. and Gold, B., "Digital filter design techniques in the frequency domain," Proc. IEEE, vol. 55, pp. 149-171, February 1967.
32. Ragazzini, J. R. and Franklin, G. F., Sampled-data Control Systems, New York, N. Y., McGraw-Hill, 1958.
33. Schlee, F. H., Standish, C. J. and Toda, N. F., "Divergence in Kalman filter," AIAA Journal, vol. 5, pp. 1114-1120, June 1967.
34. Sorenson, H. W., "On the error behavior in linear minimum variance estimation problems," IEEE Trans. Automatic Control vol. AC-12, pp. 557-562, October 1967.
35. Truxal, J. G., Automatic Feedback Control System Synthesis, New York, N. Y., McGraw-Hill, 1955.
36. White, S. A., "Coefficient-word-length requirement for accurate pole location for digital filters," Second Asilomar Conference on Circuit and Systems, pp. 208-210, November 1, 1968.
37. White, S. A. and Mitsutomi, T., "The IC digital filter: a low-cost signal-processing tool," Control Engineering, vol. 17, pp. 58-68, June 1970.
38. Wiener, N., Extrapolation, Interpolation and Smoothing of Stationary Time Series, Cambridge, Mass., The M.I.T. Press, 1949.

IX. ACKNOWLEDGEMENTS

The author wishes to thank his major professor Dr. R. G. Brown, for his encouragement and suggestions concerning the research reported in this dissertation.

The author also wishes to acknowledge the Engineering Research Institute at Iowa State University for providing financial support through Project Themis under the Office of Naval Research Contract N00014-68A-0162.

To his wife, Hee Soon, no acknowledgement can possibly convey the appreciation for her sacrifices and understanding over the past years.

X. APPENDIX A

This appendix will derive the expression for the state transition matrix in general time-varying parameter system. Suppose the system is given by equation (4-28), and the homogeneous solution of this equation is of interest, the state transition matrix is obtained from this solution. That is, the homogeneous equation is

$$\dot{\underline{\xi}}(t) = F(t)\underline{\xi}(t) . \quad (A-1)$$

Then by integrating this equation, the integral equation

$$\underline{\xi}(t) = \underline{\xi}(\tau) + \int_{\tau}^t F(\lambda)\underline{\xi}(\lambda)d\lambda \quad (A-2)$$

is obtained, where $\underline{\xi}(\tau)$ is an initial condition at time $t = \tau$. This equation is called a vector Volterra equation.

Equation (A-2) can be solved by repeated substitution of the right side of the integral equation into the integral for $\underline{\xi}(\lambda)$. For example, the first iteration is

$$\underline{\xi}(t) = \underline{\xi}(\tau) + \int_{\tau}^t F(\lambda)\left\{\underline{\xi}(\tau) + \int_{\tau}^{\lambda} F(\sigma)\underline{\xi}(\sigma)d\sigma\right\}d\lambda . \quad (A-3)$$

The expression can be simplified somewhat by the introduction of the integral operator Γ , where is defined by

$$\Gamma() = \int_{\tau}^t ()d\lambda . \quad (A-4)$$

Using this operator notation, the equation (A-2) becomes

$$\underline{\xi}(t) = \underline{\xi}(\tau) + \Gamma[F(\lambda)]\underline{\xi}(\lambda). \quad (\text{A-5})$$

If the procedure shown in equation is continued, then $\underline{\xi}(t)$ is obtained as the Neumann series

$$\begin{aligned} \underline{\xi}(t) = \{ & I + \Gamma[F(\lambda)] + \Gamma[F(\lambda)\Gamma[F(\sigma)]] + \Gamma[F(\lambda)\Gamma[F(\sigma)\Gamma[F(\omega)]]] \\ & + \dots \} \underline{\xi}(\tau). \end{aligned} \quad (\text{A-6})$$

The first term in the parentheses is I, the unit matrix. The second term is the integral of the $F(\lambda)$ between limits τ and t . The third term is found by premultiplying $\Gamma[F(\sigma)]$ by $F(\lambda)$ and then integrating this product between limits τ and t . The other terms are found in like manner.

If elements of $F(\lambda)$ remain bounded between limits of integration, then this series is absolutely and uniformly convergent. This series defines a square matrix $\Lambda(F)$ which is called the matrizant as

$$\Lambda(F) = I + \Gamma(F) + \Gamma[F\Gamma(F)] + \Gamma[F\Gamma[F\Gamma(F)]] + \dots \quad (\text{A-7})$$

If both sides of the above equation are differentiated with respect to t , the fundamental property of the matrizant

$$\frac{d}{dt} \Lambda(F) = F\Lambda(F) \quad (\text{A-8})$$

is obtained. Therefore this $\Lambda(F)$ is indeed the solution to equation (A-1), and as such represents the desired state transition matrix for the time-varying system. Thus

$$\phi(t, \tau) = \Lambda(F), \quad (\text{A-9})$$

or

$$\underline{\xi}(t) = \Lambda(F)\underline{\xi}(\tau) = \phi(t, \tau)\underline{\xi}(\tau). \quad (\text{A-10})$$

For a small integral interval, the matrizant (A-7) is approximated with the first order term. This follows

$$\phi[(n+1)T, nT] \approx I + TF(nT), \quad (\text{A-11})$$

where T is the time interval from τ to t .

XI. APPENDIX B

Basic arithmetic units are described in this appendix. Three basic operations to be realized in the implementation of a digital filter are delay, addition (or subtraction), and multiplication. Serial delays (z^{-1}) are realized simply as single-input single-output shift registers. Serial full adders are used for additions and serial-parallel multipliers are employed for digital implementations.

A. Shift Register

A register is defined as a device which is capable of storing information. The register illustrated in Figure B.1 consists of n storage devices, each a flip-flop. In this register, the binary bit stored in each flip-flop will be shifted one place to the right each time a clock pulse is applied. A register that is constructed in this manner is called a shift register.

The sequence of signals representing the numbers to be read in is connected to the input lines at the left in Figure B.1. Each time a bit is to be read into the register, a clock pulse is applied. This causes all the bits in the register to be shifted right, and the new bit to be stored in the leftmost storage device. If the series of signals representing the number to be read in are contained on a single line, an inverter may be used to form the necessary wave shape for the other input line since there are two input lines to the register.

If a steady train of clock pulses is applied to the shift register input line, the input waveform will be reproduced at the outputs from

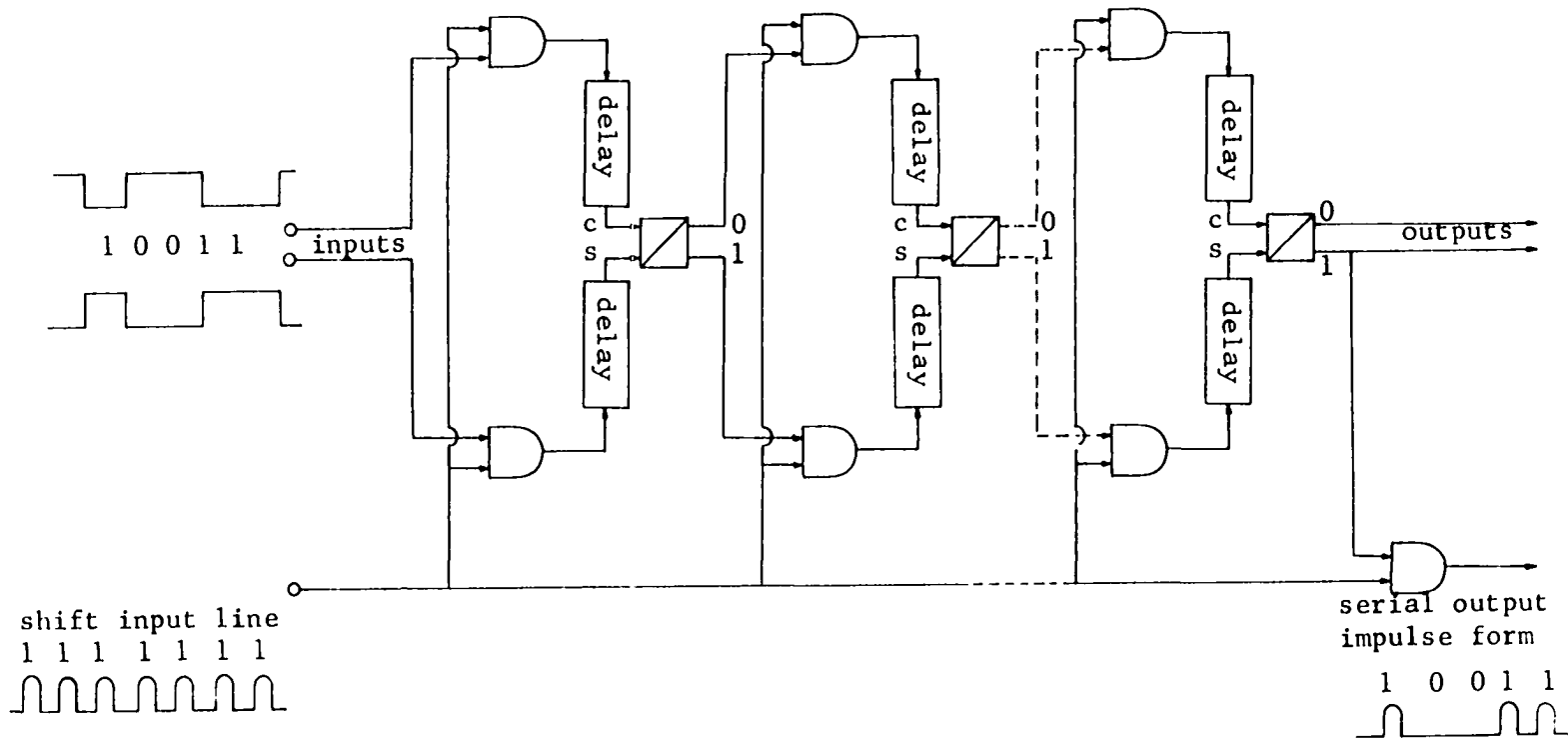


Figure B.1. Shift register

the register after a number of clock pulses equal to the number of storage devices (bits) in the register have been applied. If the clock pulses are spaced $2\ \mu\text{sec}$ apart, the length of the register is 12 bits, the input waveform will appear at the output from the shift register after a delay of $24\ \mu\text{sec}$. The outputs from the flip-flops will be in essentially d-c form; however, the d-c signals may be converted to pulses with the width and amplitude of the clock pulses by connecting AND gate to the 1 output of the last flip-flop and pulsing it with the clock pulses (refer to Figure B.1).

B. Serial Operation of Full Adder

The inputs to the serial adder consist of two numbers, the addend and augend, each represented by a series of signals. Figure B.2 illustrates a block diagram of a serial adder and set of input and output waveforms. The inputs to the full adder are, for instance, 0110, the addend, and 0011, the augend. Notice that the least significant bits of numbers to be added are the rightmost parts of the waveforms in Figure B.2. The most significant bits are leftmost bits. The signals representing the least significant bits of two numbers being added arrive at the adder first because the carry bit must be added in with the next least significant pair of augend-addend bits.

There is a delay line in the carry loop. This delay is equal to one bit time, or the delay which occurs between successive bits to be added. In this manner the carries from the least significant bits are propagated to the most significant bits, just as in the parallel adder.

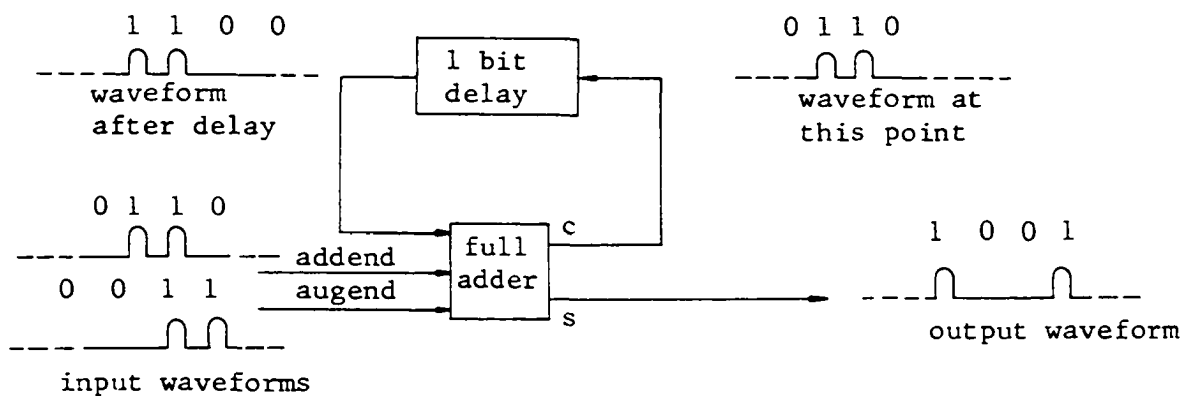


Figure B.2. Serial operation of full adder

For instance, in Figure B.2, a carry occurs when both the second and the third least significant bits of augend and addend are added. This carry is delayed and added in with the next pair of addend-augend bits.

C. 2's Complement System

Since the 2's complement system has the advantage of not requiring an end-around carry during addition, inputs of the serial full adder are used in this system. It is also possible to construct a complemeter for serial machines which will form the 2's complement of a serial number as it passes through the complemeter. Because it is easy to form 2's complements in serial machines, this system is most used.

The circuit in Figure B.3 complements every bit which passes after not including, the first 1. For instance, if binary 1010 is the number to be complemented, the complemeter will first receive a 0, then 1, then another 0, and finally a 1. The circuit will not complement to 0, nor will it complement the first 1 it receives, but it will complement every bit thereafter. The final complemented number will be 0110, the 2's complement of 1010.

The flip-flop must be reset before the circuit is used to form 2's complement of another number.

D. Serial-parallel Multiplier

To reduce the multiplication time in a serial arithmetic unit, the serial-parallel multiplier should be considered. The required binary multiplication is performed by successive shifting and addition.

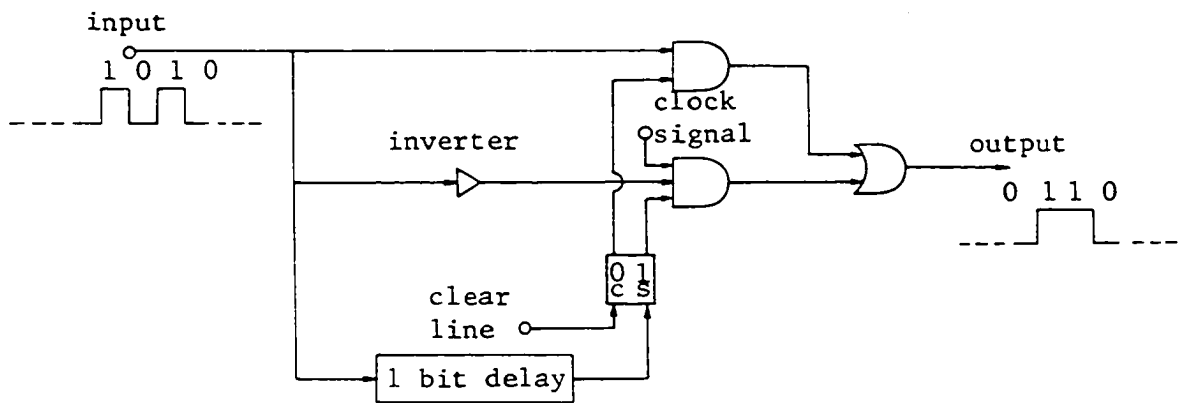


Figure B.3. Serial 2's complement circuit

The logic is controlled by the multiplier binary digits. When the multiplier and the multiplicand are represented with 2's complements, and when one inverter, AND gate, OR gate, and flip-flop are added, sign bits of two inputs need not to be picked up and stored specially at the beginning. Hence the serial-parallel multiplier is also supplied with 2's complement inputs.

In the multiplication process, if the specific binary bit is a 1, the multiplicand of an appropriate weight is added to the sum of the partial product, and if the multiplier bit is 0, no addition is performed. The multiplier does not care about the length of the input data word, but it does need a signal on one control line to signify the start of the data word and a second pulse to alert it to the sign bit (last bit) of the input. The bit rate of this data input can be between 10 kcps and 1.5 Mcps when the scaling coefficient is a word of up to eight data bits in length, plus sign.

The basic circuit for multiplying the two numbers is shown in Figure B.4. The multiplicand (a_1 through a_{N+1}) is stored in the set of $N+1$ flip-flops that from the memory section, a_{N+1} being the least significant bit, and a_1 being sign bit. The multiplier appears serially on the indicated input line with the least significant bit appearing first with input having 2's complement. This implementation requires N full adders.

When a 1 bit appears on the multiplier serial input line, the stored multiplicand is gated to the adders through AND gates except the a_1 section. The sums generated by the adders are delayed 1-bit time and

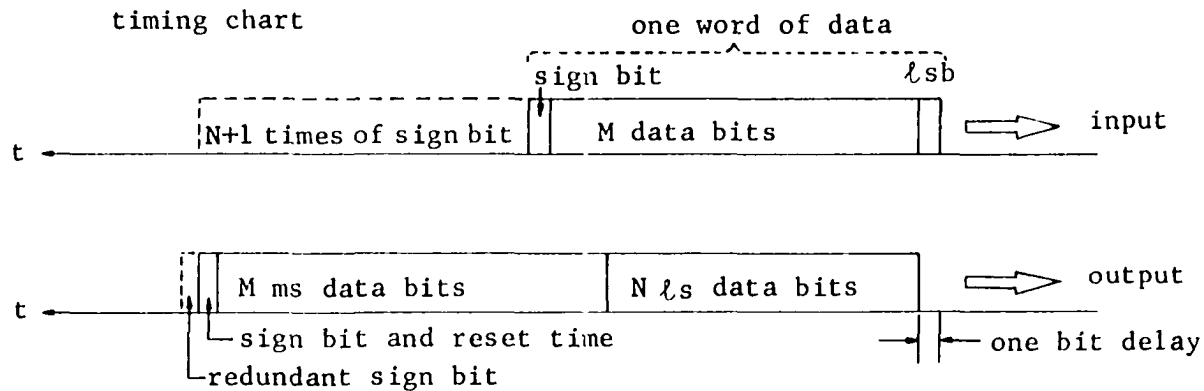
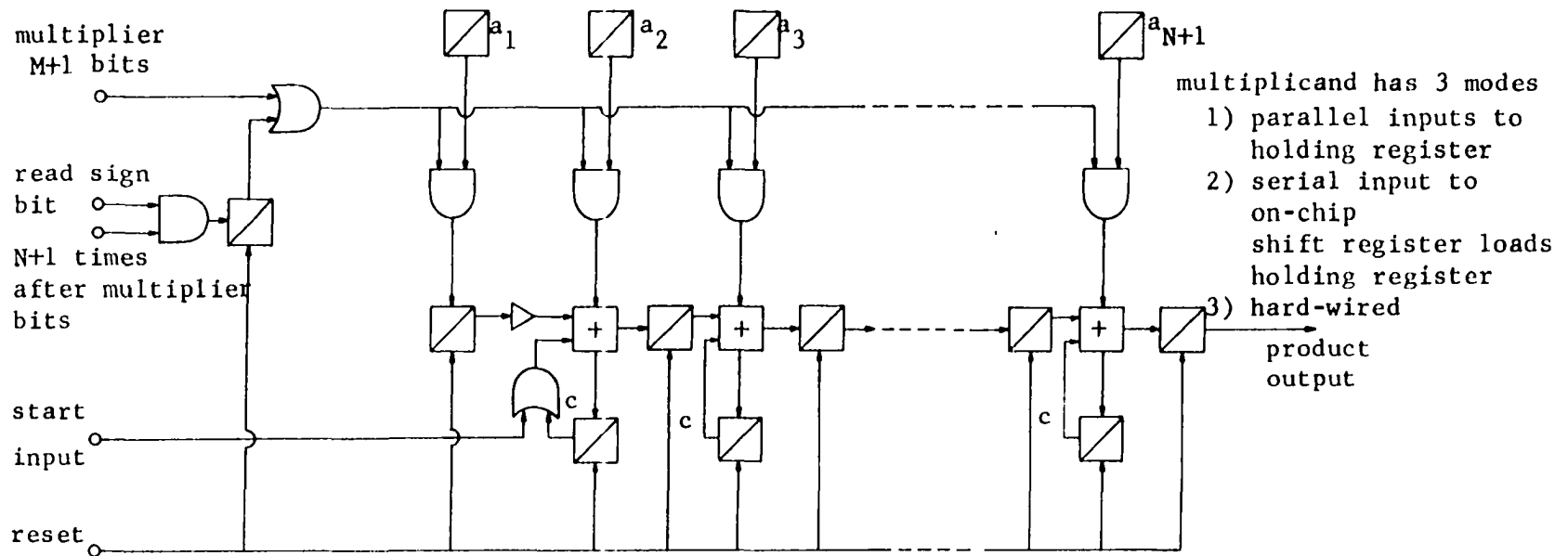


Figure B.4. Serial-parallel multiplier in 2's complement system and its timing chart

are used as inputs to the next adders. The carries from the adders are stored in the 1-bit delay elements, the outputs of which are fed back into the adders during the next clock time. When a 0-bit appears, all zeros are added to the 1-bit right-shifted partial sum. The most significant bit of the product will appear at the output gate during clock time $M+N+1$. This timing chart is also shown in Figure B.4. Here the serial multiplier input after $M+1$ data bits is $N+1$ times of the sign bit. They are added by the machine with one AND, OR, DELAY, and flip-flop units.

As the addition or shift is being performed, the product bits become available at the output gate, one bit at each bit time. The multiplication is completed in one word time, during which the least significant half of the product is delivered at the output gate when used the same length of multiplier and multiplicand words. The most significant half is in the shift registers being added with duplicating sign bits. It takes another one word time to shift its contents out of the multiplier. Therefore, the multiplication takes total of two word times without one word time for storing the multiplicand.

XII. APPENDIX C

This appendix derives the simple discrete Kalman-Bucy derived filter equation given by equation (6-10) from equations (6-7) and (6-8).

The matrix state equation (6-7) is noted as a set of first-order difference equations where the delay operator z^{-1} is used as $\xi_{n-1|n-1}^\Lambda = z^{-1} \xi_{n|n}^\Lambda$. They are

$$\xi_1^\Lambda = z^{-1} \xi_1^\Lambda + Tz^{-1} \xi_2^\Lambda$$

$$\xi_2^\Lambda = z^{-1} \xi_2^\Lambda + Tz^{-1} \xi_3^\Lambda$$

·
·
·

$$\xi_m^\Lambda = z^{-1} \xi_m^\Lambda + Tz^{-1} \xi_{m+1}^\Lambda \quad (\text{for } m = 1, 2, \dots, k-1) \quad (\text{C-1})$$

·
·
·

$$\xi_{k-1}^\Lambda = z^{-1} \xi_{k-1}^\Lambda + Tz^{-1} \xi_k^\Lambda$$

and

$$\begin{aligned} \xi_k^\Lambda &= \theta_1(nT)z^{-1} \xi_1^\Lambda + \theta_2(nT)z^{-1} \xi_2^\Lambda + \dots + \theta_m(nT)z^{-1} \xi_m^\Lambda + \dots \\ &\quad + \theta_{k-1}(nT)z^{-1} \xi_{k-1}^\Lambda + \theta_k(nT)z^{-1} \xi_k^\Lambda + Ty_n \end{aligned}$$

Note that the $n-1|n-1$ and $n|n$ subscripts have been omitted for convenience. Making arrangements such that the same state in each equation except the last equation is summed and arranged, the first $(k-1)$ equations become

$$\begin{aligned}
\hat{\xi}_1 &= \frac{Tz^{-1}}{1-z^{-1}} \hat{\xi}_2 \\
\hat{\xi}_2 &= \frac{Tz^{-1}}{1-z^{-1}} \hat{\xi}_3 \\
&\vdots \\
\hat{\xi}_m &= \frac{Tz^{-1}}{1-z^{-1}} \hat{\xi}_{m+1} \\
&\vdots \\
\hat{\xi}_{k-1} &= \frac{Tz^{-1}}{1-z^{-1}} \hat{\xi}_k.
\end{aligned} \tag{C-2}$$

As can be seen, the term $\frac{Tz^{-1}}{1-z^{-1}}$ is common to all equations and one

state is a multiplication of this common term with the next state.

Consequently, every state can be written in terms of one common state,

such as $\hat{\xi}_k$. These are shown as

$$\begin{aligned}
\hat{\xi}_1 &= \left(\frac{Tz^{-1}}{1-z^{-1}}\right)^{k-1} \hat{\xi}_k \\
\hat{\xi}_2 &= \left(\frac{Tz^{-1}}{1-z^{-1}}\right)^{k-2} \hat{\xi}_k \\
&\vdots \\
\hat{\xi}_m &= \left(\frac{Tz^{-1}}{1-z^{-1}}\right)^{k-m} \hat{\xi}_k \\
&\vdots \\
\hat{\xi}_{k-1} &= \left(\frac{Tz^{-1}}{1-z^{-1}}\right) \hat{\xi}_k.
\end{aligned} \tag{C-3}$$

Note that (k-1) states have simple relationships with the last state $\hat{\xi}_k$ in this procedure. Now substituting (C-3) into the last equation of (C-1), leads to

$$\begin{aligned} \hat{\xi}_k &= \theta_1(nT)z^{-1} \left(\frac{Tz^{-1}}{1-z^{-1}} \right)^{k-1} \hat{\xi}_k + \theta_2(nT)z^{-1} \left(\frac{Tz^{-1}}{1-z^{-1}} \right)^{k-2} \hat{\xi}_k + \dots \\ &+ \theta_m(nT)z^{-1} \left(\frac{Tz^{-1}}{1-z^{-1}} \right)^{k-m} \hat{\xi}_k + \dots + \theta_{k-1}(nT)z^{-1} \left(\frac{Tz^{-1}}{1-z^{-1}} \right) \hat{\xi}_k \\ &+ \theta_k(nT)z^{-1} \hat{\xi}_k + Ty_n. \end{aligned} \quad (C-4)$$

This is equivalent to

$$\begin{aligned} \{ 1 - \theta_1(nT)z^{-1} \left(\frac{Tz^{-1}}{1-z^{-1}} \right)^{k-1} - \theta_2(nT)z^{-1} \left(\frac{Tz^{-1}}{1-z^{-1}} \right)^{k-2} - \dots \\ - \theta_m(nT)z^{-1} \left(\frac{Tz^{-1}}{1-z^{-1}} \right)^{k-m} - \dots - \theta_{k-1}(nT)z^{-1} \left(\frac{Tz^{-1}}{1-z^{-1}} \right) \\ - \theta_k(nT)z^{-1} \} \hat{\xi}_k = Ty_n. \end{aligned} \quad (C-5)$$

Simplifying the above equation, a simple solution for $\hat{\xi}_k$ is obtained such that

$$\hat{\xi}_k = \frac{Ty_n}{1 - \left\{ \sum_{i=1}^k \theta_i(nT)z^{-1} \left(\frac{Tz^{-1}}{1-z^{-1}} \right)^{k-i} \right\}}. \quad (C-6)$$

Therefore, the rest of states are solved by substituting (C-6) into equations (C-3). These become

$$\begin{aligned}
\hat{y}_1 &= \frac{\left(\frac{Tz^{-1}}{1-z^{-1}}\right)^{k-1}}{1 - \left\{ \sum_{i=1}^k \theta_i (nT) z^{-1} \left(\frac{Tz^{-1}}{1-z^{-1}}\right)^{k-i} \right\}} T y_n \\
\hat{y}_2 &= \frac{\left(\frac{Tz^{-1}}{1-z^{-1}}\right)^{k-2}}{1 - \left\{ \sum_{i=1}^k \theta_i (nT) z^{-1} \left(\frac{Tz^{-1}}{1-z^{-1}}\right)^{k-i} \right\}} T y_n \\
&\vdots \\
\hat{y}_m &= \frac{\left(\frac{Tz^{-1}}{1-z^{-1}}\right)^{k-m}}{1 - \left\{ \sum_{i=1}^k \theta_i (nT) z^{-1} \left(\frac{Tz^{-1}}{1-z^{-1}}\right)^{k-i} \right\}} T y_n \quad (C-7) \\
&\vdots \\
\hat{y}_{k-1} &= \frac{\left(\frac{Tz^{-1}}{1-z^{-1}}\right)}{1 - \left\{ \sum_{i=1}^k \theta_i (nT) z^{-1} \left(\frac{Tz^{-1}}{1-z^{-1}}\right)^{k-i} \right\}} T y_n .
\end{aligned}$$

Now, from the output equation (6-8), the output x_n can be written in terms of input y_n as follows.

$$x_n = [\beta_0(nT) \beta_1(nT) \cdots \beta_m(nT) \cdots \beta_{k-2}(nT) \beta_{k-1}(nT)] \begin{bmatrix} \hat{\xi}_1 \\ \hat{\xi}_2 \\ \vdots \\ \hat{\xi}_m \\ \vdots \\ \hat{\xi}_{k-1} \\ \hat{\xi}_k \end{bmatrix} \Big|_{n|n}$$

$$= \sum_{i=1}^k \beta_{i-1}(nT) \hat{\xi}_i$$

$$= \frac{\sum_{i=1}^k \beta_{i-1}(nT) \left(\frac{Tz^{-1}}{1-z^{-1}}\right)^{k-i}}{1 - \left\{ \sum_{i=1}^k \theta_i(nT) z^{-1} \left(\frac{Tz^{-1}}{1-z^{-1}}\right)^{k-i} \right\}} T y_n \quad (C-8)$$

Here, the subscript $n|n$ is omitted for simplicity again. Then the input-output relationship is represented as $W_N(z)$ such that

$$W_N(z) = \frac{T \left\{ \sum_{i=1}^k \beta_{i-1}(nT) \left(\frac{Tz^{-1}}{1-z^{-1}}\right)^{k-i} \right\}}{1 - \left\{ \sum_{i=1}^k \theta_i(nT) \left(\frac{Tz^{-1}}{1-z^{-1}}\right)^{k-i} \right\} z^{-1}}, \quad (C-9)$$

where the z^{-1} term in the denominator has been taken out from the parenthesis.

The resulting equation (C-9) is going to be simplified for $(1 - z^{-1})$ term. Since the highest term of $(\frac{Tz^{-1}}{1 - z^{-1}})$ in the numerator of the above equation (C-9) is when $i=1$, that is the order of $(k-1)$, $(\frac{1}{1 - z^{-1}})^{k-1}$ is divided such that

$$W_N(z) = \frac{(\frac{1}{1 - z^{-1}})^{k-1} T \{ \sum_{i=1}^k \beta_{i-1} (nT) (Tz^{-1})^{k-i} (\frac{1}{1 - z^{-1}})^{-i+1} \}}{1 - (\frac{1}{1 - z^{-1}})^{k-1} \{ \sum_{i=1}^k \theta_i (nT) (Tz^{-1})^{k-i} (\frac{1}{1 - z^{-1}})^{-i+1} \} z^{-1}} \quad (C-10)$$

This reduces the final form of $W_N(z)$ as

$$W_N(z) = \frac{T \{ \sum_{i=1}^k \beta_{i-1} (nT) (Tz^{-1})^{k-i} (1 - z^{-1})^{i-1} \}}{(1 - z^{-1})^{k-1} - \{ \sum_{i=1}^k \theta_i (nT) (Tz^{-1})^{k-i} (1 - z^{-1})^{i-1} \} z^{-1}} \quad (C-11)$$

which is equation (6-10).

XIII. APPENDIX D

A sample printout of the computer program which was used to conduct the simulation studies in the third order low-pass filter example follows in this appendix. This program simulates sampled-data digital filters and calculates frequency responses of those filters. Continuous filter amplitude was found first and compared to the others. Three frequencies as shown in Table 6.1 are studied specifically. Initial conditions in this simulation are modified and set to converse steady states within short period of time.


```

C
C   FREQUENCY RESPONSES OF THE THIRD ORDER FILTER IN EXAMPLE 4
C   THIS PROGRAM COMPUTES FREQUENCY RESPONSES WITH A GIVEN SINUSOIDAL
C   INPUT Y(INPUT) AND A SAMPLING INTERVAL DELTAT
C   L INDICATES NUMBER OF FREQUENCY POINTS TO BE COMPUTED, M INDICATES
C   NUMBER OF SAMPLING POINTS WITHIN ONE INPUT PERIOD, AND N INDICATES
C   NUMBER OF DATA POINTS
C   INITIAL CONDITIONS DELAY(I) ARE CHOSEN SUCH THAT THE FILTER
C   CONVERGES ITS STEADY STATE IN SHORT PERIOD OF TIME
C
C   DOUBLE PRECISION A(3,3), AMPLIT(5,3), CONSNT(11), DELAY(3), DELTAT
1, F(3,3), FREQCY(3), GAIN, OMEGA, PI, STATE(3), TWOPI, X, XMAX
1, XMIN, XTOP, Y(20)
C   DOUBLE PRECISION DCOS, DEXP, DLOG10, DMAX1, DMIN1, DSIN, DSQRT
C   INTEGER INCREM(3), M(3)
C
C   READ CONSTANTS AND SINUSOIDAL INPUT
C
C   READ (5,10) L, N, DELTAT
10  FORMAT (2(10X,I5),10X,F12.5)
C   READ (5,11) (M(I),I=1,L)
11  FORMAT (3(10X,I5))
C   READ (5,12) (FREQCY(I),I=1,L)
12  FORMAT (3(10X,F12.5))
C   J=M(1)
C   READ (5,13) (Y(I),I=1,J)
13  FORMAT (10X,D23.16)
C   PI=+3.141592653585793
C   TWOPI=+2.0*PI
C   DO 14 K=1,L
C   INCREM(K)=M(1)/M(K)
14  CONTINUE
C
C   ENTER AMPLITUDE RESPONSES OF THE CONTINUOUS FILTER
C
C   WRITE (6,100)
100  FORMAT ('1'///6X,'AMPLITUDE RESPONSES OF THE CONTINUOUS FILTER'///

```

```

16X,'FREQUENCY',9X,'OMEGA',16X,'GAIN',19X,'AMPLITUDE (DB)')
DO 102 K=1,L
OMEGA=TWOPI*FREQCY(K)
CONSNT(1)=OMEGA*OMEGA
CONSNT(2)=+1.0-CONSNT(1)
CONSNT(3)=+1.0+CONSNT(1)
CONSNT(4)=CONSNT(1)+CONSNT(2)*CONSNT(2)
CONSNT(5)=DSQRT(CONSNT(3))
CONSNT(6)=DSQRT(CONSNT(4))
GAIN=+1.0/CONSNT(5)/CONSNT(6)
AMPLIT(1,K)=-20.0*DLOG10(CONSNT(5))-20.0*DLOG10(CONSNT(6))
WRITE (6,101) FREQCY(K), OMEGA, GAIN, AMPLIT(1,K)
101 FORMAT (1X,F12.5,5X,F12.5,2(5X,D23.16))
102 CONTINUE

```

C
C
C

ENTER THE DISCRETE KALMAN-BUCY DERIVED FILTER RESPONSES

```

A(1,1)=+1.0
A(1,2)=DELTAT
A(1,3)=0.0
A(2,1)=A(1,3)
A(2,2)=A(1,1)
A(2,3)=A(1,2)
A(3,1)=A(1,3)
A(3,2)=-2.0*DELTAT
A(3,3)=A(1,1)
DO 200 I=1,2
DO 200 J=1,3
F(I,J)=A(I,J)
200 CONTINUE
F(3,1)=-DELTAT*A(1,1)+(1.0-2.0*DELTAT)*A(3,1)
F(3,2)=-DELTAT*A(1,2)+(1.0-2.0*DELTAT)*A(3,2)
F(3,3)=-DELTAT*A(1,3)+(1.0-2.0*DELTAT)*A(3,3)
WRITE (6,201) ((A(I,J),J=1,3),I=1,3), ((F(I,J),J=1,3),I=1,3)
201 FORMAT ('1'///6X,'DISCRETE KALMAN-BUCY DERIVED FILTER'///6X,'STATE
1 TRANSITION MATRIX'///3(3(5X,D23.15))///6X,'COEFFICIENT MATRIX'///3
1(3(5X,D23.16))//)

```

```

DO 209 K=1,L
  READ (5,202) (DELAY(I),I=1,3)
202 FORMAT (5X,3D23.16)
  WRITE (6,203) FREQCY(K), (DELAY(I),I=1,3)
203 FORMAT ('1'///6X,'THE DISCRETE KALMAN-BUCY FILTER RESPONSE AT FREQ
  UENCY',F12.5///6X,'INITIAL CONDITIONS'///3(5X,D23.16)///4X,'N',2X,'
  IM',10X,'INPUT',18X,'OUTPUT',18X,'STATE 1',17X,'STATE 2',17X,'STATE
  1 3')
  INCREA=0
  INPUT=0
  XMAX=0.0
  XMIN=0.0
  XTOP=0.0
  DO 207 J=1,N
    INCREA=INCREA+1
    INPUT=INPUT+INCREM(K)
    IF (INCREA.LE.M(K)) GO TO 204
    INCREA=1
    INPUT=INCREM(K)
204 CONTINUE
  STATE(1)=F(1,1)*DELAY(1)+F(1,2)*DELAY(2)+F(1,3)*DELAY(3)
  STATE(2)=F(2,1)*DELAY(1)+F(2,2)*DELAY(2)+F(2,3)*DELAY(3)
  STATE(3)=F(3,1)*DELAY(1)+F(3,2)*DELAY(2)+F(3,3)*DELAY(3)
  1+DELTAT*Y(INPUT)
  X=STATE(1)
  DO 205 I=1,3
    DELAY(I)=STATE(I)
205 CONTINUE
  WRITE (6,206) J, INCREA, Y(INPUT), X, (STATE(I),I=1,3)
206 FORMAT (1X,I4,1X,I2,5(1X,D23.16))
  IF (J.LT.N-M(K)) GO TO 207
  XMAX=DMAX1(XMAX,X)
  XMIN=DMIN1(XMIN,X)
  XTOP=DMAX1(XTOP,XMAX,-XMIN)
207 CONTINUE
  AMPLIT(2,K)=+20.0*DLOG10(XTOP)
  WRITE (6,208) FREQCY(K), FREQCY(K), AMPLIT(2,K)

```

```
208 FORMAT (////6X,'AMPLITUDE RESPONSE OF THE DISCRETE KALMAN-BJCY DE  
'RIVED FILTER AT FREQUENCY',F12.5//6X,'FREQUENCY',3X,'AMPLITUDE (D  
18)')//1X,F12.5,5X,D23.16)
```

```
209 CONTINUE
```

```
C  
C  
C
```

```
ENTER THE IMPULSE INVARIANT Z-TRANSFORM DERIVED FILTER RESPONSES
```

```
CONSNT(1)=-DELTAT/2.0  
CONSNT(2)=+3.0  
CONSNT(3)=DELTAT*DSQRT(CONSNT(2))/2.0  
CONSNT(4)=-DELTAT  
CONSNT(5)=DEXP(CONSNT(1))  
CONSNT(6)=DCOS(CONSNT(3))  
CONSNT(7)=DSIN(CONSNT(3))/DSQRT(CONSNT(2))  
CONSNT(8)=DEXP(CONSNT(4))  
CONSNT(9)=+2.0*CONSNT(5)*CONSNT(6)  
CONSNT(10)=CONSNT(8)  
CONSNT(11)=CONSNT(5)*(CONSNT(6)+CONSNT(7))  
WRITE (6,300) (CONSNT(I),I=8,11), DELTAT
```

```
300 FORMAT ('1'//6X,'THE IMPULSE INVARIANT Z-TRANSFORM DERIVED FILTER  
1'//6X,'DENOMINATOR COEFFICIENTS'//3(5X,D23.16)//6X,'NUMERATOR CO  
EFFICIENT'//5X,D23.16//6X,'GAIN'//5X,D23.16)
```

```
DO 308 K=1,L
```

```
READ (5,301) (DELAY(I),I=1,3)
```

```
301 FORMAT (5X,3D23.16)
```

```
WRITE (6,302) FREQC(K), (DELAY(I),I=1,3)
```

```
302 FORMAT ('1'//6X,'THE IMPULSE INVARIANT Z-TRANSFORM DERIVED FILTER  
1 RESPONSE AT FREQUENCY',F12.5//6X,'INITIAL CONDITIONS'//3(5X,D23.  
116)//4X,'N',2X,'M',10X,'INPUT',18X,'OUTPUT',18X,'STATE 1',17X,'ST  
ATE 2',17X,'STATE 3')
```

```
INCREA=0
```

```
INPUT=0
```

```
XMAX=0.0
```

```
XMIN=0.0
```

```
XTOP=0.0
```

```
DO 306 J=1,N
```

```
INCREA=INCREA+1
```

```

INPUT=INPUT+INCREM(K)
IF (INCREA.LE.M(K)) GO TO 303
INCREA=1
INPUT=INCREM(K)
303 CONTINUE
STATE(1)=CONSNT(8)*DELAY(1)+DELTAT*Y(INPUT)
STATE(2)=CONSNT(9)*DELAY(2)-CONSNT(10)*DELAY(3)+DELTAT*Y(INPUT)
STATE(3)=DELAY(2)
X=STATE(1)-STATE(2)+CONSNT(11)*STATE(3)
DO 304 I=1,3
DELAY(I)=STATE(I)
304 CONTINUE
WRITE (6,305) J, INCREA, Y(INPUT), X, (STATE(I),I=1,3)
305 FORMAT (1X,I4,1X,I2,5(1X,D23.16))
IF (J.LT.N-M(K)) GO TO 306
XMAX=DMAX1(XMAX,X)
XMIN=DMIN1(XMIN,X)
XTOP=DMAX1(XTOP,XMAX,-XMIN)
306 CONTINUE
AMPLIT(3,K)=+20.0*DLOG10(XTOP)
WRITE (6,307) FREQCY(K), FREQCY(K), AMPLIT(3,K)
307 FORMAT (////6X,'AMPLITUDE RESPONSE OF THE IMPULSE INVARIANT Z-TRA
INSFORM DERIVED FILTER AT FREQUENCY',F12.5//6X,'FREQUENCY',3X,'AMP
LITUDE (DB)'//1X,F12.5,5X,D23.16)
308 CONTINUE
C
C   ENTER THE BILINEAR Z-TRANSFORM DERIVED FILTER RESPONSES
C
CONSNT(1)=+1.0+DELTAT/2.0+DELTAT*DELTAT/4.0
CCNSNT(2)=(+1.0-DELTAT/2.0)/(+1.0+DELTAT/2.0)
CCNSNT(3)=+2.0*(+1.0-DELTAT*DELTAT/4.0)/CONSNT(1)
CONSNT(4)=(+1.0-DELTAT/2.0+DELTAT*DELTAT/4.0)/CONSNT(1)
CCNSNT(5)=+2.0
CONSNT(6)=DELTAT*DELTAT*DELTAT/(+8.0*(+1.0+DELTAT/2.0)*CONSNT(1))
WRITE (6,400) (CONSNT(I),I=2,6)
400 FORMAT ('1'//6X,'THE BILINEAR Z-TRANSFORM DERIVED FILTER'//6X,'D
ENOMINATOR COEFFICIENTS'//3(5X,D23.16)//6X,'NUMERATOR COEFFICIENT

```

```

1'//5X,D23.16//6X,'GAIN'//5X,D23.16)
DO 408 K=1,L
READ (5,401) (DELAY(I),I=1,3)
401 FORMAT (5X,3D23.16)
WRITE (6,402) FREQCY(K), (DELAY(I),I=1,3)
402 FORMAT ('1'//6X,'THE BILINEAR Z-TRANSFORM DERIVED FILTER RESPONSE
1 AT FREQUENCY',F12.5//6X,'INITIAL CONDITIONS'//3(5X,D23.16)//4X,
1'N',2X,'M',10X,'INPUT',18X,'OUTPUT',18X,'STATE 1',17X,'STATE 2',17
1X,'STATE 3')
INCREA=0
INPUT=0
XMAX=0.0
XMIN=0.0
XTOP=0.0
DO 406 J=1,N
INCREA=INCREA+1
INPUT=INPUT+INCREM(K)
IF (INCREA.LE.M(K)) GO TO 403
INCREA=1
INPUT=INCREM(K)
403 CONTINUE
STATE(1)=CONSNT(2)*DELAY(1)+CONSNT(6)*Y(INPUT)
CONSNT(7)=STATE(1)+DELAY(1)
STATE(2)=CONSNT(3)*DELAY(2)-CONSNT(4)*DELAY(3)+CONSNT(7)
STATE(3)=DELAY(2)
X=STATE(2)+CONSNT(5)*DELAY(2)+DELAY(3)
DO 404 I=1,3
DELAY(I)=STATE(I)
404 CONTINUE
WRITE (6,405) J, INCREA, Y(INPUT), X, (STATE(I),I=1,3)
405 FORMAT (1X,I4,1X,I2,5(1X,D23.16))
IF (J.LT.N-M(K)) GO TO 406
XMAX=DMAX1(XMAX,X)
XMIN=DMIN1(XMIN,X)
XTOP=DMAX1(XTOP,XMAX,-XMIN)
406 CCNTINUE
AMPLIT(4,K)=+20.0*DLOG10(XTOP)

```

```

WRITE (6,407) FREQCY(K), FREQCY(K), AMPLIT(4,<)
407 FORMAT (////6X,'AMPLITUDE RESPONSE OF THE BILINEAR Z-TRANSFORM DE
RIVED FILTER AT FREQUENCY',F12.5//6X,'FREQUENCY',3X,'AMPLITUDE (D
18)'//1X,F12.5,5X,D23.16)
408 CONTINUE

```

C
C
C

```

ENTER THE MATCHED Z-TRANSFORM DERIVED FILTER RESPONSE

```

```

CONST(1)=-DELTAT
CONST(2)=-DELTAT/2.0
CONST(3)=+3.0
CONST(4)=DELTAT*DSQRT(CONST(3))/2.0
CONST(5)=DEXP(CONST(1))
CONST(6)=+2.0*DEXP(CONST(2))*DCOS(CONST(4))
CCNST(7)=CONST(5)
CONST(8)=DELTAT*DELTAT*DELTAT
WRITE (6,500) (CONST(I),I=5,8)
500 FORMAT ('1'//6X,'THE MATCHED Z-TRANSFORM DERIVED FILTER'//6X,'DE
INOMINATOR COEFFICIENTS'//3(5X,D23.16)//6X,'GAIN'//5X,D23.16)
DO 508 K=1,L
READ (5,501) (DELAY(I),I=1,3)
501 FORMAT (5X,3D23.16)
WRITE (6,502) FREQCY(K), (DELAY(I),I=1,3)
502 FCRMAT ('1'//6X,'THE MATCHED Z-TRANSFORM DERIVED FILTER RESPONSE
1AT FREQUENCY',F12.5//6X,'INITIAL CONDITIONS'//3(5X,D23.16)//4X,'
IN',2X,'M',10X,'INPUT',18X,'OUTPUT',18X,'STATE 1',17X,'STATE 2',17X
1,'STATE 3')
INCREA=0
INPUT=0
XMAX=0.0
XMIN=0.0
XTOP=0.0
DO 506 J=1,N
INCREA=INCREA+1
INPUT=INPUT+INCREM(K)
IF (INCREA.LE.M(K)) GO TO 503
INCREA=1

```

```

INPUT=INCREM(K)
503 CONTINUE
STATE(1)=CONSNT(5)*DELAY(1)+CONSNT(8)*Y(INPUT)
STATE(2)=CONSNT(6)*DELAY(2)-CONSNT(7)*DELAY(3)+STATE(1)
STATE(3)=DELAY(2)
X=STATE(2)
DO 504 I=1,3
DELAY(I)=STATE(I)
504 CONTINUE
WRITE (6,505) J, INCREA, Y(INPUT), X, (STATE(I),I=1,3)
505 FORMAT (1X,I4,1X,I2,5(1X,D23.16))
IF (J.LT.N-M(K)) GO TO 506
XMAX=DMAX1(XMAX,X)
XMIN=DMIN1(XMIN,X)
XTOP=DMAX1(XTOP,XMAX,-XMIN)
506 CONTINUE
AMPLIT(5,K)=+20.0*DLOG10(XTOP)
WRITE (6,507) FREQCY(K), FREQCY(K), AMPLIT(5,K)
507 FCRMAT (////6X,'AMPLITUDE RESPONSE OF THE MATCHED Z-TRANSFORM DER
IVED FILTER AT FREQUENCY',F12.5//6X,'FREQUENCY',3X,'AMPLITUDE (DB
1)'//1X,F12.5,5X,D23.16)
508 CONTINUE
C
C   AMPLITUDE RESPNCSES ARE SUMMARIZED
C
WRITE (6,600) (FREQCY(K),(AMPLIT(I,K),I=1,5),K=1,L)
600 FCRMAT ('1'//6X,'FIDELITY COMPARISONS'//6X,'FREQUENCY',4X,'CONT
INUOUS FILTER',7X,'D K-B D FILTER',10X,'I I FILTER',9X,'BILINEAR F
ILTER',8X,'MATCHED FILTER'//(3X,F12.5,5(2X,D20.13)/))
STOP
END

```



```

L          3, N          100, DELTAT          01
M(1)      20, M(2)      10, M(3)          5
FREQ.(1)   5          , FREQ.(2)      10          , FREQ.(3)      20
Y(1)      +0.3090169943749474D 00
Y(2)      +0.5877852522924731D 00
Y(3)      +0.8090169943749474D 00
Y(4)      +0.9510565162951536D 00
Y(5)      +0.1000000000000000D+01
Y(6)      +0.9510565162951536D 00
Y(7)      +0.8090169943749474D 00
Y(8)      +0.5877852522924731D 00
Y(9)      +0.3090169943749474D 00
Y(10)     0.0000000000000000D 00
Y(11)     -0.3090169943749474D 00
Y(12)     -0.5877852522924731D 00
Y(13)     -0.8090169943749474D 00
Y(14)     -0.9510565162951536D 00
Y(15)     -0.1000000000000000D+01
Y(16)     -0.9510565162951536D 00
Y(17)     -0.8090169943749474D 00
Y(18)     -0.5877852522924731D 00
Y(19)     -0.3090169943749474D 00
Y(20)     0.0000000000000000D 00
D05+0.3283447901643439D-04-0.6512192913414144D-04-0.3150293613478769D-01
D10+0.4107707677287839D-05-0.8139024668703403D-05-0.1538019454103424D-01
D20+0.5077395922280117D-06-0.1006390575512467D-05-0.6880893005032306D-02
I05-0.3153654000280100D-01-0.3481307487395726D-01-0.3244511696155941D-02
I10-0.1538438996672949D-01-0.1579339758972721D-01-0.4050011079434255D-03
I20-0.6881411654581103D-02-0.6931958484080774D-02-0.5005140789052404D-04
B05-0.3902843369577172D-06+0.7495434169987150D-05+0.8046483711150760D-05
B10-0.1903914162133389D-06+0.7962258644378170D-06+0.1006707975176546D-05
B20-0.8516175925281235D-07+0.3683776673633695D-07+0.1244831093026464D-06
M05-0.3153654000280100D-05+0.2838228402580538D-04+0.3218382593181051D-04
M10-0.1538438996672948D-05+0.2407209328445202D-05+0.4026614571823660D-05
M20-0.6881411654581482D-06-0.2002431112360320D-06+0.4979069945460436D-06

```