KYBERNETIKA — VOLUME *30* (1994), NUMBER 3, PAGES 233 – 244

A COMPARISON OF ALGORITHMS TO FILTER NOISY OBSERVATIONS OF A LINEAR DIFFERENTIAL SYSTEM DRIVEN BY BROWNIAN MOTION AND A SIMPLE MARKOV SWITCHING PROCESS

P. J. Browne

The problem under consideration is the filtering of Gaussian noise observations of a linear differential system driven by both Brownian motion and a Markov process switching in continuous time at a constant rate λ with state space $(-1, +1)$ usually referred to as a random telegraph process.

The algorithms compared are:

(1) The Interacting Multiple Model (IMM) algorithm.

(2) Differential equations driven by the innovation process for the mean and variance of the state and switching level derived from a representation for the posterior density of the joint process, which is in turn obtained from the fundamental filtering theorem for semi-martingales with Gaussian observation noise.

(3) A filter obtained by replacing the Markov switching process by a Gaussian process with equivalent second order properties. This gives rise to a Kalman–Bucy filter.

1. INTRODUCTION

The problem under consideration is to filter noisy observations of a linear differential system driven by Brownian motion and a simple Markov switching process. A. H. Jazwinski [4, Chapter 5], shows that for a wide class of loss functions optimal filtering implies computing the posterior distribution of the state, and the optimal estimate is the posterior mean. Consequently in this paper attention is restricted to the goal of finding the posterior distribution of the state, given a realisation of the observation process to the time in question.

The state $X_t \in R$ is generated by the stochastic differential equation:

$$\mathrm{d}X_t = M_t \, \mathrm{d}t + \rho \, \mathrm{d}W_t,$$

where $W_t \sim N(0, t)$; $\{W_t\}$ is standard Brownian motion independent of $\{M_t\}$. The process $\{M_t\}$ is a stationary Markov process with states $\{-1, +1)\}$ and switching at rate λ. The observation process $\{Y_t\}$ is defined by the stochastic differential equation:

$$\mathrm{d}Y_t = X_t \, \mathrm{d}t + \mathrm{d}\, Z_t.$$

Here $\{Z_t\}$ is Brownian motion independent of $\{W_t\}$ and $\{M_t\}$ with $Z_t \sim N(0, \sigma^2 t)$.

The problem is to construct the posterior distribution of the joint process (M_t, X_t) from the observed path y_s, $0 \le x \le t$, and prior on (M_0, X_0). It is assumed that $P(M_0 = -1) = P(M_0 = +1) = 1/2$ and $X_0 \sim N(0,1) = \frac{1}{2}$ and $X_0 \sim N(0,1)$. The generator of the continuous time Markov chain M_t is:

$$G = \begin{bmatrix} -\lambda & \lambda \\ \lambda & -\lambda \end{bmatrix}.$$

The associated transition matrix is:

$$R_t = \begin{bmatrix} \frac{1}{2} + \frac{1}{2}e^{-2\lambda t} & \frac{1}{2} - \frac{1}{2}e^{-2\lambda t} \\ \frac{1}{2} - \frac{1}{2}e^{-2\lambda t} & \frac{1}{2} + \frac{1}{2}e^{-2\lambda t} \end{bmatrix}.$$

Let P_t be a Poisson counting process; then the Markov chain may be represented as:

$$\mathrm{d}\, M_t = -2M_t\, \mathrm{d}P_t,$$

where $\{P_t\}$ is a Poisson counting process with rate λ. So the problem is described by a pair of stochastic differential equations:

$$\mathrm{d}X_t = M_t\, \mathrm{d}t + \rho\, \mathrm{d}W_t \tag{1}$$

$$\mathrm{d}M_t = -2M_t\, \mathrm{d}P_t, \tag{2}$$

with $\{W_t\}$ standard Brownian motion and $\{P_t\}$ a Poisson counting process with rate λ independent of $\{W_t\}$.

The observation equation is unchanged:

$$\mathrm{d}Y_t = X_t\, \mathrm{d}t + \mathrm{d}Z_t \tag{3}$$

$\{Z_t\}$ is Brownian motion independent of both $\{W_t\}$ and $\{P_t\}$. $Z_t \sim N(0, \sigma^2 t)$. Equation (1) is formally equivalent to the integral expression:

$$X_t = x_0 + \int_0^t M_t\, \mathrm{d}t + \rho \int_0^t \mathrm{d}W_t, \tag{4}$$

where the first integral is a M. S. Riemann integral and the second integral is an Itô integral. In a like manner equations (2) and (3) can be represented as:

$$M_t \;=\; m_0 - 2\int_0^t M_t\, \mathrm{d}P_t, \tag{5}$$

$$Y_t \;=\; y_0 + \int_0^t X_d\, \mathrm{d}t + \int_0^t \mathrm{d}Z_t. \tag{6}$$

There is no explicit solution, so some kind of approximation needs to be undertaken. Three different ways of attacking this problem are considered.

1. Interacting Multiple Model (IMM) algorithm

This is a discrete filtering approach originated by H. A. P. Blom from a method of reducing mixtures of Gaussian distributions to single Gaussian components.

2. Continuous Filtering Approach
This approach is the most mathematically sophisticated undertaken in this paper. It is an application of the fundamental filtering theorem for noisy observations of semimartingales with white Gaussian observation noise.

3. Second order equivalent filter
In this approach the Markov switching process is replaced by a Gaussian process with the same mean and auto covariance structure. As will be seen the simplified filter becomes a Kalman–Bucy filter.

Sections 2, 3 and 4 discuss these algorithms. A more detailed description is contained in a Technical report available from the author. Section 5 compares the three algorithms by Monte Carlo simulation. Section 6 gives the conclusions.

2. ALGORITHM 1

The IMM algorithm is a development of the generalised pseudo Bayes (GPB) method originated in 1971 by Jaffer & Gupta. Jaffer & Gupta's method is an attempt to control the problem of the exponentially increasing number of hypotheses that beset the full Bayesian approach in discrete time. Consider the problem here for example, and consider the chain history in terms of all possible Markov states. At time $t = 0$ there are two possibilities for the state; $M_0 = +1$ or -1. At time $t = \Delta t$ the possible histories are:

$$\{-1,-1\}\,\{-1,+1\},\ \{+1,-1\},\ \{+1,+1\}$$

a total of four possibilities. At time $t = 2\Delta t$ there are eight possibilities and so on. GPB(k) employs a bank of N^k Kalman filters matched to the N^k most recent possible paths, where N is the number of states in the Markov chain.

The IMM algorithm was first mentioned in the literature by H. A. P. Blom in 1984, but we refer to Blom [1] (1986) which explains both the IMM method and the GPB approach very clearly and discusses their relative merits.

The idea of the IMM algorithm is to alter the sequence of steps of the GPB(2) method into the following sequence: Markov switching; merge, time extrapolation, measurement update; MMSE estimate. Blom claims that the IMM method offers near GPB(2) performance at GPB(1) computational cost. The principal reason for the cost reduction is the reduction in the number of Kalman filters from four to two when N is 2. Space reasons preclude further consideration of the IMM algorithm except to say that a simplification of it has been devised to reduce the amount of computation involved in the merge operation. This simplification is termed algorithm Als in the simulation Section 5.

3. ALGORITHM 2

This approach is based on a continuous time model and the fundamental filtering theorem for noisy observations on semimartingales with white Gaussian observation noise. This theorem is presented as Theorem 18.4 on p.280 of Elliott [3], and can be referred to in Kwakernaak [5] and Blom [2].

In 1975 H. Kwakernaak employed this theorem to obtain a representation for the evolution of the posterior probability density function of a stochastic differential equation driven by a Poisson process, a problem similar in nature to the one considered here. From the representation he was able to derive differential equations for the evolution of the posterior mean, and all central moments. Each evolution equation for an individual posterior central moment involves the next higher central moment, and so one is faced with an infinite set of simultaneous differential equations. He dealt with this problem in two ways. The first method he adopted was to employ a Ritz–Galerkin method for approximately solving the conditional Fokker–Planck equation, while for the second method he truncated the stochastic differential equations satisfied by the cumulants of the posterior density function. He found that best results were achieved by considering the first seven central moments, and that his sophisticated approach offered at best only a small improvement over the Kalman filter. He also noted that if the time step chosen for the discrete approximation to the differential evolution expression is too large then instability results.

Blom in [2] employed the results in Kwakernaak's seminal paper to derive an expression for the evolution of the joint posterior probability density function of the state and switching state of a vector valued process which is a generalisation of the pair of equations presented as eqns (1) and (2). He, like Kwakernaak, derived a set of coupled differential equations describing the evolution of the posterior central moments, each equation involving the next higher central moment. However, he attacked this problem in a different way from Kwakernaak. He factorised the evolution equation for the posterior joint probability density function into an expression for the marginal distribution of the Markov switching process, and an expression for the posterior evolution of the state conditional on the Markov switching level. He argued that by this approach he could ignore third and higher order central moments. It is his approach that I have adopted in this paper.

Space considerations preclude a full presentation of the working associated with the following results. Suppose f is a complete right continuous filtration, and both the joint Markov process (X_t, M_t) defined by eqns (1) and (2), and the observation process Y_t from eqn (3) are adapted to f_t. Let $y_t = \sigma\{y_s;\, s \le t\}$ be an increasing sequence of sigma algebras generated by the observation process. Note that $y_t \subseteq f_t$. The notation $\hat{x}_t$ is defined as being the mean of the conditional probability density function of X_t given a realisation of y_t. Similarly $\hat{P}_{X_t,M_t}(x,m)$ denotes the conditional joint pdf of (X_t, M_t) given a realisation of y_t. Employment of the fundamental filtering algorithm results in the following evolution equation for $\hat{p}_{X_t,M_t}(x,m)$

$$\mathrm{d}\,\hat{p}_{X_t,M_t}(x,m) = L\,\hat{p}_{X_t,M_t}(x,m)\,\mathrm{d}t + \hat{p}_{X_t,M_t}(x,m)\,[x-\hat{X}_t]\,\frac{\mathrm{d}\nu_t}{\mathrm{d}\sigma^2}$$

where $\mathrm{d}\,\nu_t = d\,y_t - \hat{X}_t\,\mathrm{d}t$ is the innovation processs, L is the forward diffusion operator for the joint process and

$$\begin{aligned} L\,\hat{p}_{X_t,M_t}(x,m) \quad &= \quad -m\,\frac{\partial p_{X_t,M_t}(x,m)}{\partial x} + \frac{\rho^2}{2}\,\frac{\partial^2}{\partial x^2}\,\hat{p}_{X_t,M_t}(x,m) \\ &\qquad -\lambda\,\hat{p}_{X_t,M_t}(x,m) + \lambda\,\hat{p}_{X_t,M_t}(x,-m). \end{aligned}$$

This equation can be factorised into a differential equation using the marginal posterior probability distribution of the Markov switching level, and the posterior pdf of the state given the Markov switching level. These results are:

$$\begin{aligned}\mathrm{d}\,\hat{p}_{X_t,M_t}(x,m) &= \left[\hat{p}_{M_t}(m)\,L^m\,\hat{p}_{X_t|M_t}(x|m) + M\,\hat{p}_{X_t,M_t}(x,m)\right]\,\mathrm{d}t \\ &\quad + \hat{p}_{X_t,M_t}\,[x - \hat{X}_t]\,\frac{\mathrm{d}\nu_t}{\mathrm{d}\sigma^2}\end{aligned}$$

where:

$$\begin{aligned}L^m\,\hat{p}_{X_t|M_t}(x|m) &= -m\frac{\partial}{\partial x}\hat{p}_{X_t|M_t}(x|m) + \frac{\rho^2}{2}\frac{\partial^2}{\partial x^2}\hat{p}_{X_t|M_t}(x|m) \\ M\,\hat{p}_{X_t,M_t}(x,m) &= -\lambda\,\hat{p}_{X_t,M_t}(x,m) + \lambda\,\hat{p}_{X_t,M_t}(x,-m).\end{aligned}$$

Use of the Itô differential rule enables the following five equations to be obtained.

Let $\hat{q}_t = \frac{P(M_t=+1)}{P(M_t=-1)}$. So $\ln\hat{q}_t$ is the log likelihood ratio for Markov switching level $+1$ vs. Markov switching level -1. Then:

$$\mathrm{d}\ln\hat{q}_t = \left(\frac{\lambda}{\hat{q}_t} - \lambda\hat{q}_t\right)\,\mathrm{d}t + \left(\hat{X}_t(+1) - \hat{X}_t(-1)\right)\,\frac{\mathrm{d}y_t}{\sigma^2} + \left(\hat{X}_t^2(-1) - \hat{X}_t^2(+1)\right)\,\frac{\mathrm{d}t}{2\sigma^2}$$

where y_t is the observation made at time t from equation (3).

A pair of equations for the evolution of the posterior state mean given the Markov switching level are:

$$\mathrm{d}\hat{X}_t(+1) = \mathrm{d}t + \frac{\lambda}{\hat{q}_t}\left(\hat{X}_t(-1) - \hat{X}_t(+1)\right)\,\mathrm{d}t + \frac{\hat{Q}_t(+1)\,\mathrm{d}\nu_t(+1)}{\sigma^2}$$

where $\mathrm{d}\nu_t(+1) = \mathrm{d}y_t - \hat{X}_t(+1)\,\mathrm{d}t$

$$\mathrm{d}\hat{X}_t(-1) = -\mathrm{d}t + \lambda\,\hat{q}_t\left(\hat{X}_t(+1) - \hat{X}_t(-1)\right)\,\mathrm{d}t + \frac{\hat{Q}_t(-1)\,\mathrm{d}\nu_t(-1)}{\sigma^2}$$

where $\mathrm{d}\nu_t(-1) = \mathrm{d}y_t - \hat{X}_t(-1)\,\mathrm{d}t$.

If third and higher order central moments are neglected, this simplification enables a pair of equations for the evolution of the posterior state variance given the Markov switching level to be calculated:

$$\begin{aligned}\mathrm{d}\hat{Q}_t(+1) &= \rho^2\,\mathrm{d}t + \frac{\lambda}{\hat{q}_t}\left(\hat{Q}_t(-1) - \hat{Q}_t(+1) + (\hat{X}_t(-1) - \hat{X}_t(+1))^2\right)\mathrm{d}t - \frac{\hat{Q}_t^2(+1)}{\sigma^2}\,\mathrm{d}t \\ \mathrm{d}\hat{Q}_t(-1) &= \rho^2\,\mathrm{d}t + \lambda\,\hat{q}_t\left(\hat{Q}_t(+1) - \hat{Q}_t(-1) + (\hat{X}_t(+1) - \hat{X}_t(-1))^2\right)\mathrm{d}t - \frac{\hat{Q}_t^2(-1)}{\sigma^2}\,\mathrm{d}t.\end{aligned}$$

The above systems of five differential equations represents another approximate solution to the optimal filtering problem.

4. ALGORITHM 3

Let the time interval step be Δ. The state equation (1) can be rewritten as a first order approximation:

$$X_{t+\Delta} = X_t + M_t\,\Delta + W_{t+\Delta} - W_t$$

where $W_{t+\Delta} - W_t \sim N(0, \rho^2, \Delta)$.

It is possible to replace the discrete Markov chain M_t by a Gaussian process with an equivalent covariance structure. The autocovariance function for the Markov chain M_t, $\operatorname{cov}(M_t, M_\tau) = e^{-2|t-\tau|\lambda}$.

References to p. 123 of Jazwinski [4] shows that the linear stochastic differential equation:

$$\mathrm{d}M_t = -2\lambda M_t\,\mathrm{d}t + 2\sqrt{\lambda}\,\mathrm{d}\beta_t$$

with $\{\beta_t\}$ standard Brownian motion generates a stationary exponentially correlated Gaussian process with zero mean, and the desired autocovariance function.

This stochastic differential equation can be modelled by the first order approximation:

$$M_{t+\Delta} = M_t(1 - 2\lambda\Delta) + 2\sqrt{\lambda}(\beta_{t+\Delta} - \beta_t)$$

where $\beta_{t+\Delta} - \beta_t \sim N(0, \Delta)$.

The observation equation, equation (3) can be rewritten as a first order approximation:

$$B_t = X_t + \eta_t, \qquad \{\eta_t\} \sim \text{ iid } N\left(0, \frac{\sigma^2}{\Delta}\right).$$

The state and observation equations can be represented by the system of three equations:

$$\begin{bmatrix} X_{t+\Delta} \\ M_{t+\Delta} \\ B_{t+\Delta} \end{bmatrix} = \begin{bmatrix} 1 & \Delta & 0 \\ 0 & (1-2\lambda\Delta) & 0 \\ 1 & \Delta & 0 \end{bmatrix} \begin{bmatrix} X_t \\ M_t \\ B_t \end{bmatrix} + \begin{bmatrix} W_{t+\Delta} - W_t \\ 2\sqrt{\lambda}(\beta_{t+\Delta} - \beta_t) \\ W_{t+\Delta} - W_t + \eta_{t+\Delta}. \end{bmatrix}$$

Under the above assumption the posterior distribution of (X_t, M_t) is joint normal. It is possible to compute the mean vector and variance covariance matrix of the posterior distribution of (X_t, M_t) which characterises the distribution completely. It is also possible to check that the evolution equations for the mean vector and variance covariance matrix match that for the discrete Kalman Bucy filter.

5. COMPARISON OF ALGORITHMS BY SIMULATION

The algorithms discussed above were investigated by a wide range of simulations. The four figures presented here are typical.

		Symbol used in plots
A1	Algorithm 1	Δ
Als	Simplified version of algorithm 1	∇
A2	Algorithm 2	$\times$
A3	Algorithm 3	$\square$

Comparison of the various algorithms is achieved by Monte Carlo simulation. The simulator is based on a discrete first order approximation to the system equations (1 ,2, 3). A congruential random number generator is used in association with the Box–Muller algorithm to generate variates from a normal distribution. A random number seed is chosen for the first of a sequence of simulator runs and the operation of the random number generator ensures that each run of a sequence represents a suitably realistic random realisation of the simulation. Of course when one is comparing the four algorithms it is important to use the same sequence of random number seeds for each sequence of runs for the different algorithms, so that one is comparing likewith like.

Turning now to the initial conditions for the simulator, it is assumed that at the outset each filter algorithm knows the value of the state perfectly at time step $t = 0$, but is subsequently supplied with noisy observations of the state. The initial value of the simulator state is drawn from a $N(0, 1)$ distribution. Each run of the simulator lasts for 300 time steps for the four figures presented here. In order to discover the effectiveness of the algorithms A1 and A2 at tracking changes in the Markov switching level it was decided to introduce fixed jumps of the Markov process. The initial state of the Markov switching process for the simulator is set to $+1$, and there are fixed jumps of this process at $t = 50$, 150, 250. This corresponds roughly to a simulator value of λ of 10, since this would give three expected jumps in 300 time steps with a time interval of .001. The time step interval value is set to .001 throughout. For all four runs the simulator data is the same, namely ρ simulator $= .1$, σ simulator $= .01$.

Considering the initial conditions for A1 and A2, the variance of the state given the Markov switching state is set initially to zero, while the mean is set to the simulator state. The weights for the Markov switching process are set to $1/2$.

For A3, the variance of the state is set to zero, and the mean is set to the simulator state. The covariance between the state and the Gaussian process that is equivalent to the Markov switching process is set to zero; the variance of this process is set to 1, and the mean is drawn from a $N(0, 1)$ distribution.

In order to detect track loss for each run the difference between the actual position of the simulated target and the merged posterior mean (note that A3 finds this mean directly, but for A1 and A2 a merge operation is required) obtained by each algorithm is used. The difference is scaled by the posterior merged MMSE estimate standard deviation $\sqrt{\hat{Q}_t}$, and tested at a nominal significance level of .1 . Track loss is considered to have occurred when five consecutive estimates fail the test. Note that even though track loss is considered to have occurred the filters still continue to track, and thus a contribution is still made to the Rms plot discussed in the next section.

5.1. RMS Plot

This plot is presented as Fig. 1 and it portrays the Rms errors between the simulated target and the target position estimate for all three algorithms over 100 runs of 300 time steps each. Note that the filter values of ρ and σ are matched to the simulator values, i. e. ρ filter = .1, σ filter = .01. This is also roughly true for the λ parameter since the filter λ is set to 10.

Track loss statistics are calculated. A1 has lost 2 tracks out of 100, while Als has lost 10 tracks. A2 and A3 perform better with losses of just 1 track and 0 tracks respectively.

Having dealt with the preliminaries it is possible to turn to an examination of the plot. Note that the Rms error builds up from an initial zero value for all the algorithms because each filter knows the position of the simulated target exactly at the outset. The effect of the simulator jumps at steps 50, 150 and 250 can be seen. The worst performer is Als. A3 has poorer Rms performance initially but after approximately step 50 it settles down to perform well, and after step 50 there is little to choose between A1, A2 and A3. Examination of the computer loading for the various algorithms (not included here) gives a ranking of A3, A2, Als, Al with A3 using only 2/3 of the compute time required for Al.

Fig. 1.

5.2. Examination of Mismatch

The remaining three plots, Figs 2, 3, 4, form a group and are designed to show how well the various algorithms perform in a mismatch situation. That is to say where the value of a filter parameter ρ, σ, λ is not matched to the same simulator parameter. As before the data is printed on each plot, and the plots are:

1.	Average steps to track loss.	Fig. 2
2.	RMS errors at step 270.	Fig. 3
3.	Percentace trask losses	Fig. 4

These three plots explore simulator/filter mismatch in a single parameter; in this case the ρ parameter. The σ value for both simulator and filter is set to .01. The λ value for the filter algorithms is set at 10 to match the three fixed jumps of the simulator, as for Fig. 1. However the ρ value for the filter algorithms is set to vary from .01 to .2 in steps of .01, this range of values being evenly disposed about the simulator ρ value of .1. The abscissa of each plot represents the filter value for ρ with the simulator value of ρ being in the centre of the x axis. Each symbol on each plot represents the net result of 100 runs of 300 steps each with the simulator value of ρ determined by the intercept on the abscissa.

Considering the plots in detail, and examining Fig. 2, it can be seen that Als has initially a poor performance but that this improves rapidly as the filter ρ increases, while A1, A2 and A3 all have good performance. The best performer is A2.

Fig. 2.

Turning to Fig. 3, step 270 was chosen for computing the Rms errors because it occurs soon after the final simulator switching process jump at step 250. Als has the worst performance initially but this steadily improves as the filter ρ increases. Note that the best performance for Als occurs when the filter value for ρ is greater than the simulator value for ρ. The other three algorithms have similar performance for Rms error with A3 being the best performer. Note that the Rms error is not affected by filter/simulator ρ mismatch.

Fig. 3.

Finally Fig. 4 confirms the picture that has emerged from study of Figs 2 & 3. Als has initially a very poor performance that improves rapidly with increasing simulator ρ, while the other three algorithms enjoy similar good performance that is unaffected by ρ mismatch, with A2 being the best performer.

Fig. 4.

6. CONCLUSIONS

Examination of very many runs like Figs 1 – 4 with varying data has enabled the following conclusions to be made:

1. If the time step interval is too large A2 often becomes unstable. Kwakernaak [5] noted the same problem. A time step interval has been chosen here of .001, and for the data dealt with in this study it seems to be a reasonable choice.
2. A1 and A2 have an in built advantage over A3 for they compute the marginal distribution of the switching level, and the conditional distribution of the state given the switching level rather than the joint normal distribution of the state and Gaussian process in the case of A3. Extensive simulation has shown that A1 and A2 are most successful at predicting the Markov switching level with small values of σ for both simulator and filter.

3. Considering the situation where the simulator and algorithm have matched parameters, the discrimination between the various algorithms is least when $\rho \gg \sigma$, and at high levels of λ for both simulator and filter. There is little to choose between A2 and A3, and A1 is perhaps the worst performer. Als, the IMM simplification, performs best at low levels of λ.

   When a mismatch exists between the simulator parameters and the algorithm parameters, there is little to choose between the algorithms. The algorithms are most sensitive to mismatch in the parameters ρ and σ, and least in parameter λ. Again there is little to choose between A2 and A3, and A1 is perhaps the worst performer.

4. In terms of computing power A3 has the lightest load and A1 the heaviest load, and A2 is between as expected. Als, the IMM simplification, offers significant saving in compute time for the IMM algorithm.

(Received March 3, 1994.)

REFERENCES

[1] H. A. P. Blom: Overlooked potential of systems with Markovian coefficients. In: Proceedings of 25th Conference on Decison and Control, Athens, Greece, December 1986 pp. 1758–1764.

[2] H. A. P. Blom: Detection Filter Representations for Markov Jump Diffusions. NLR, TR, 82019 U, 1982.

[3] R. J. Elliott: Stochastic Calculus and Applications. Springer-Verlag, Berlin 1982.

[4] A. H. Jazwinski: Stochastic Processes and Filtering Theory. Academic Press, New York 1970.

[5] H. Kwakernaak: Filtering for systems excited by Poisson white noise. In: Control Theory, Numerical Methods and Computer System Modeling (A. Bensousan and J. L. Lions, eds., Lecture Notes in Economics and Mathematical Systems 107), Springer-Verlag, Berlin 1975, pp. 468–492.

P. J. Browne, Department of Mathematical Statistics, University of Sussex, Falmer, Brighton BN1 9QH. Great Britain.