BAYESIAN PARAMETER ESTIMATION OF JUMP-LANGEVIN SYSTEMS FOR TREND FOLLOWING IN FINANCE

James Murphy, Simon Godsill

Department of Engineering, University of Cambridge
Trumpington Street, Cambridge, United Kingdom CB2 1PZ
Emails:{jm362, sjg30}@cam.ac.uk

ABSTRACT

In this paper we present a Bayesian method for parameter estimation in linear Jump-Langevin systems, i.e. systems driven by a linear, mean-reverting jump-diffusion trend process. Such models have been applied successfully to trend following in finance, in order to develop momentum-based trading strategies. Parameter estimation is based around a reversible-jump MCMC method for jump-time inference. Parameter estimation is demonstrated on both synthetic and financial time series, and estimated parameters are compared with *ad hoc* parameter estimates used in earlier work.

Index Terms— Jump-diffusion, parameter estimation, trend following, Bayesian, finance

1. INTRODUCTION

Momentum effects have been found to exist in numerous financial markets, including those for commodities [1], foreign exchange [2] and equities [3]. This is apparently in defiance of the *Efficient Market Hypothesis* of [4], which states that prices should not be predictable from analysing their past history. However, momentum effects appear to persist [5] and continue to be exploited, in spite of reports of their declining strength [6] or illusory nature [7]. Proposed explanations of these effects mostly divide into two groups: herding effects, whereby investors all buy similar assets (e.g. ones that have recently performed well) [8] [9], and delayed market reaction to news or changes in fundamentals (i.e. over multiple trading periods), meaning that the effect of incorporating new information in asset prices is not instantaneous [10] [5].

In order to exploit momentum effects for trading, it is necessary to identify trends that are present in asset prices. Many strategies exist for this, including ones as simple as buying shares that performed well in previous periods, which have been shown to be effective under certain circumstances [11]. However, for momentum trading, it is equally important to identify points at which trends change [12] since at those points a momentum strategy following a trend can make significant losses if it is unable to identify the change quickly and alter its position accordingly.

In [13], a model-based tracking algorithm was proposed to infer price momentum, using a particle filter for on-line trend inference. In order to cope with rapidly changing trends, the trend process was modelled as a jump-diffusion process allowing sudden trend changes to be modelled. Details of the model are given in section 2. There it was demonstrated that a trading system based on filtering using this model could detect momentum effects and be used to trade profitably, even in the presence of transaction costs. However, a limitation of the methodology presented was that the five system parameters were chosen *ad hoc*, attempting to maximize portfolio Sharpe ratio. This paper shows how Bayesian parameter estimation for this model (and Jump-Langevin type models in general) can be performed in a principled manner using a reversible jump Markov chain Monte-Carlo (MCMC) scheme to infer jump times in the trend process, and a Metropolis-within-Gibbs sampling scheme to infer parameter values. A related approach for parameter estimation in 1d continuous-time ARMA processes (without latent trend) is given in [14].

2. MODEL

The model considered here is that used in [13], a two-component model consisting of a ‘value’ x and ‘trend’ $\dot{x}$ component. The trend component is modelled as a mean-reverting random process, with mean reversion rate γ, subject to Gaussian noise of constant volatility σ^2 and random Gaussian jumps of volatility σ_J^2. Mean reversion within this model reflects a view that trends will fade over time. The governing stochastic differential equation (SDE) for the state dynamics is given by

$$\begin{bmatrix} dx_t \\ d\dot{x}_t \end{bmatrix} = \begin{bmatrix} 0 & 1 \\ 0 & -\gamma \end{bmatrix} \begin{bmatrix} x_t \\ \dot{x}_t \end{bmatrix} dt + \begin{bmatrix} 0 \\ \sigma \end{bmatrix} dW_t + \begin{bmatrix} 0 \\ \sigma_J \end{bmatrix} dJ_t, \quad (1)$$

where dW_t is the instantaneous change of a standard Brownian motion and dJ_t is the instantaneous change of a pure jump

This work was supported by the Engineering and Physical Sciences Research Council (EPSRC) grant number EP/K020153/1

process defined as

$$J_t = \sum_{k \in \{k | \tau_k < t\}} S_k,$$

where $S_k \sim \mathcal{N}\left(0, \sigma_J^2\right)$, so that $dJ_t = S_k$ at τ_k, i.e. the k^{th} jump occurs at time τ_k and its size is distributed as a zero-mean Gaussian random variable. Jump times τ_k are modelled as following a Poisson arrival process with rate λ, so that

$$p(\tau_k \mid \tau_{1:k-1}) = \text{Exp}\left(\tau_k\,;\lambda\right), \tag{2}$$

where Exp$(x\,;\lambda)$ is the exponential distribution density at x.

The i^{th} price observation, y_i, observed at time t_i, is assumed to be a noisy observation of the value process x_{t_i}, perturbed by Gaussian noise of fixed variance σ_{obs}^2:

$$y_i = x_{t_i} + v_{t_i}, \qquad v_{t_i} \sim \mathcal{N}(0, \sigma_{obs}^2). \tag{3}$$

Without jumps, the system in equation (1) is a Langevin system, with a zero-reverting trend process, sometimes known as the Singer model [15], which has a closed form solution [15]. For $s \leq t$,

$$p(X_t \mid X_s) = \mathcal{N}\left(X_t; F(s,t)X_s, R(s,t)\right),$$

with

$$\begin{aligned} F(s,t) &= \begin{bmatrix} 1 & \frac{1}{\gamma}(1 - e^{\gamma(s-t)}) \\ 0 & e^{\gamma(s-t)} \end{bmatrix}, \\ R(s,t) &= \sigma^2 \begin{bmatrix} q_{11}(s,t) & q_{12}(s,t) \\ q_{12}(s,t) & q_{22}(s,t) \end{bmatrix}, \end{aligned}$$

where

$$\begin{aligned} q_{11}(s,t) &= \frac{1}{2\gamma^3}(4e^{\gamma(s-t)} - 3 - e^{2\gamma(s-t)} - 2\gamma(s-t)), \\ q_{12}(s,t) &= \frac{1}{2\gamma^2}(e^{2\gamma(s-t)} + 1 - 2e^{\gamma(s-t)}), \\ q_{22}(s,t) &= \frac{1}{2\gamma}(1 - e^{2\gamma(s-t)}). \end{aligned}$$

When jumps are added to the system the state distribution can be calculated *conditional on the jump times*. For N jumps at times $\tau_1, ..., \tau_N$ such that $s < \tau_1 < ... < \tau_N < t$, this gives

$$p(X_t \mid X_s, \tau_{1:N}) = \mathcal{N}\left(X_t; F(s,t)X_s, S_N\right),$$

where the covariance S_N is given by the recursion

$$\begin{aligned} S_N &= R(\tau_N, t) + e^{A(t-\tau_N)} S_{N-1} e^{A'(t-\tau_N)}, \\ S_n &= R(\tau_{n-1}, \tau_n) + e^{A(\tau_n - \tau_{n-1})} S_{n-1} e^{A'(\tau_n - \tau_{n-1})} + \Sigma_J, \end{aligned}$$

for $n = 1, ..., N-1$. This requires that (nominally) $\tau_0 = s$. $A = \begin{bmatrix} 0 & 1 \\ 0 & -\gamma \end{bmatrix}$ is the system matrix and $\Sigma_J = \begin{bmatrix} 0 & 0 \\ 0 & \sigma_J^2 \end{bmatrix}$ is the jump covariance matrix.

Since $p(X_t \mid X_s, \mathcal{T})$ is linear Gaussian if jump times are known, a Kalman filter can be constructed to find the distribution of the system state at observation times, given a series of linear Gaussian observations, such as those in equation (3); details can be found in [13]. This can also evaluate the observation likelihood $p(y_{1:t} \mid \mathcal{T}, \theta)$, conditional on the jump times $\mathcal{T}$ and set of system parameters $\theta = \{\gamma, \sigma, \sigma_J, \sigma_{\text{obs}}, \lambda\}$.

3. SAMPLING JUMPS: REVERSIBLE JUMP MCMC

If the jump times $\mathcal{T}$ are unknown, as is usually the case, they can be estimated from M observations by sampling from the jump distribution $p(\mathcal{T} \mid y_{1:M}, \theta)$, using reversible jump MCMC [16], [17]. The state of the Markov chain is the entire set of jump times $\mathcal{T}$ and therefore proposals must be such that a series of accepted proposals is able to transform any set of jump times into any other. Furthermore, each of the proposals must be reversible, so that if a proposal mechanism exists that could propose a jump sequence $\mathcal{T}'$ given a current sequence $\mathcal{T}$, there is a proposal mechanism that could propose $\mathcal{T}$ when starting from $\mathcal{T}'$ (with non-zero density).

To this end three simple proposal types are allowed: a *move* proposal, in which one jump time is altered locally; a *birth* proposal, in which a new jump is created; and a *death* proposal, in which an existing jump is removed. These, along with their reversals, are shown in Fig. 1 and allow any starting sequence of jump times to be transformed to any other through a series of moves, births and deaths.

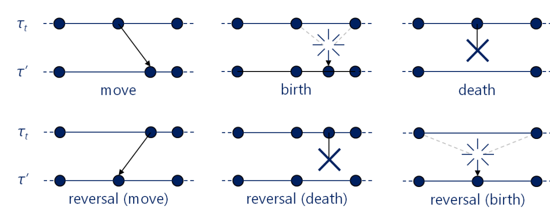

Fig. 1. The three basic types of proposal for state sequence updates, along with their reversals: move, birth and death

Birth Birth proposals involve generating a new jump time $\tau_* \sim \mathcal{U}(t_0, t_{\text{max}})$, where t_0 and t_{max} are the start and end times of the interval for which state inference is taking place. This jump time is added into the set of jump times so that $\mathcal{T}' = \mathcal{T} \cup \tau_*$. This proposal is accepted with probability $\min(1, \alpha_{\text{birth}})$ where

$$\alpha_{\text{birth}} = \frac{p(y_{1:M} \mid \mathcal{T}', \theta) p(\mathcal{T}' \mid \theta)(t_{\text{max}} - t_0)}{p(y_{1:M} \mid \mathcal{T}, \theta) p(\mathcal{T} \mid \theta)(N+1)},$$

and where N is the number of jumps in $\mathcal{T}$.

Death Death proposals are the reverse of birth proposals and can be created by choosing, uniform randomly, a jump time in the set $\mathcal{T}$ to remove. If this jump time is $\tau_\times$ then $\mathcal{T}' = \mathcal{T} \backslash \tau_\times$. This proposal is accepted with probability $\min(1, \alpha_{\text{death}})$

$$\alpha_{\text{death}} = \frac{p(y_{1:M} \mid \mathcal{T}', \theta) p(\mathcal{T}' \mid \theta) N}{p(y_{1:M} \mid \mathcal{T}, \theta) p(\mathcal{T} \mid \theta)(t_{\text{max}} - t_0)},$$

Move Move proposals do not change the state dimension and so are standard MCMC proposals. The strategy for creating move proposals used here is to uniform randomly choose a jump i to move from the existing jump sequence. If the time of this jump in $\mathcal{T}$ is τ, a proposal is created by moving this jump to a new position τ'. Thus $\mathcal{T}' = (\mathcal{T} \setminus \tau) \cup \tau'$. A suitable proposal for the new position is to add a Gaussian random variable to the current jump time. This gives $q_\tau(\tau' \mid \tau) = \mathcal{N}\left(\tau'; \tau, \sigma^2_{\text{move}}\right)$, where σ_{move} determines the scale of the proposed moves. The proposal $\mathcal{T}'$ is accepted with probability $\min(1, \alpha_{\text{move}})$, where

$$\alpha_{\text{move}} = \frac{p(y_{1:M} \mid \mathcal{T}', \theta) p(\mathcal{T}' \mid \theta)}{p(y_{1:M} \mid \mathcal{T}, \theta) p(\mathcal{T} \mid \theta)}.$$

The proposal is symmetric and so cancels in the above ratio.

Further mathematical details of this RJ-MCMC scheme for jump time inference can be found in [18], section 4.2.

In the above, the likelihood $p(y_{1:M} \mid \mathcal{T}', \theta)$ is calculated using the Prediction Error Decomposition of the Kalman filter as noted in section 2, detailed in [13]. The jump time prior $p(\mathcal{T} \mid \theta)$ can be calculated from the jump model as

$$p(\mathcal{T} \mid \theta) = (1 - \text{Exp}(t_{\text{max}} - \tau_N; \lambda)) \prod_{i=1}^{N} \text{Exp}(\tau_i - \tau_{i-1}; \lambda),$$

where $\tau_{1:N}$ are an ordered sequence of the N jump times in $\mathcal{T}$ with $\tau_1 < ... < \tau_N$ and (nominally) $\tau_0 = t_0$. This assumes that all jump times are in the range $[t_0, t_{\text{max}}]$, however if any jump time is outside this range $p(\mathcal{T} \mid \theta) = 0$.

4. SAMPLING PARAMETERS: GIBBS SAMPLER

Conditional on a sample of jump times, the system parameters can be sampled using a Metropolis-within-Gibbs scheme. In this, a parameter $\theta_i \in \{\gamma, \sigma, \sigma_J, \sigma_{\text{obs}}, \lambda\}$ is sampled from its full conditional distribution using the decomposition

$$p(\theta_i \mid \theta_{-i}, \mathcal{T}, y_{1:M}) \propto p(y_{1:M} \mid \theta, \mathcal{T}) p(\mathcal{T} \mid \theta) p(\theta_i \mid \theta_{-i}), \quad (4)$$

where $\theta_{-i} = \theta \setminus \theta_i$. The conditional likelihood of the observations $p(y_{1:M} \mid \theta, \mathcal{T})$ and the conditional likelihood of the jump sample $p(\mathcal{T} \mid \theta)$ can be evaluated as shown in the preceding sections. The distribution in equation (4) is not, in general, easy to sample. Sampling can be performed for each parameter θ_i using a Metropolis-Hastings step, with proposal density $q(\theta_i^* \mid \theta_i')$, where θ_i' is the current sample of parameter θ_i and θ_i^* is a proposal for θ_i. The acceptance probability for the proposal is given as $\min(1, \alpha_{\theta_i})$ with

$$\alpha_{\theta_i} = \frac{p(y_{1:M} \mid \theta_{-i}, \theta_i^*, \mathcal{T}) p(\mathcal{T} \mid \theta_{-i}, \theta_i^*) p(\theta_i^* \mid \theta_{-i})}{p(y_{1:M} \mid \theta_{-i}, \theta_i', \mathcal{T}) p(\mathcal{T} \mid \theta_{-i}, \theta_i') p(\theta_i' \mid \theta_{-i})} \frac{q(\theta_i' \mid \theta_i^*)}{q(\theta_i^* \mid \theta_i')}.$$

The prior $p(\theta_i \mid \theta_{-i})$ may depend on the other parameters, but does not need to, and should be chosen to reflect any existing beliefs about the distribution of the given parameter. In the absence of strong beliefs, vague priors can be chosen. A (symmetrical) Gaussian or Gaussian mixture random walk proposal can be used with variance chosen to match the scale over which a particular parameter is expected to vary.

4.1. Jump Rate

The jump rate λ can be sampled efficiently if an appropriate conjugate prior is chosen, since the inter-jump time is modelled as exponential with rate λ. In this case a Gamma ($\mathcal{G}$) prior on λ is conjugate and leads to the posterior distribution

$$p(\lambda \mid \mathcal{T}, y, \theta_{-\lambda}) = \mathcal{G}(\lambda; \alpha_\lambda + N, \beta_\lambda + T),$$

where N is the number of jumps in the current jump time sample $\mathcal{T}$ and T is the total observed time of the process (i.e. $t_{\text{max}} - t_0$). This distribution can easily be sampled, leading to an efficient Gibbs sampler for the jump rates. For the avoidance of confusion, the Gamma distribution here is defined as

$$\mathcal{G}(x;\ \alpha, \beta) = \frac{\beta^\alpha}{\Gamma(\alpha)} x^{\alpha-1} \exp(-\beta x),$$

whereas some definitions (including that of the gampdf function in Matlab) use $1/\beta$ as the second parameter.

The prior parameters can be interpreted (in a sense) as effectively 'adding' $\alpha_\lambda - 1$ additional jumps to the jump sequence and 'adding' β_λ extra time units to the observation period when compared to the likelihood distribution for λ, which is given by $L(\lambda) = \mathcal{G}(N + 1, T)$.

5. RESULTS

Figures 2 and 3 shows the result of applying the algorithm described above to synthetic data in order to estimate system parameters and jump times. The synthetic data used here consisted of 1000 observations generated from the model in equations (1), (2) and (3). Priors for all parameters were Gamma distributions so that $p(\theta_i \mid \theta_{-i}) = \mathcal{G}(\theta_i; \alpha_i, \beta_i)$, with the parameters $\alpha_i = \{1, 1.3, 2, 2, 1\}, \beta_i = \{50, 1, 0.05, 0.1, 10\}$ for $\theta_i \in \{\gamma, \sigma, \sigma_J, \sigma_{\text{obs}}, \lambda\}$. Since the priors (other than those for the jump-rates) are evaluated directly, it is straightforward to incorporate any other prior distribution for each parameter.

The results in figure 2 show good parameter estimation for this synthetic data, with the true parameters lying within

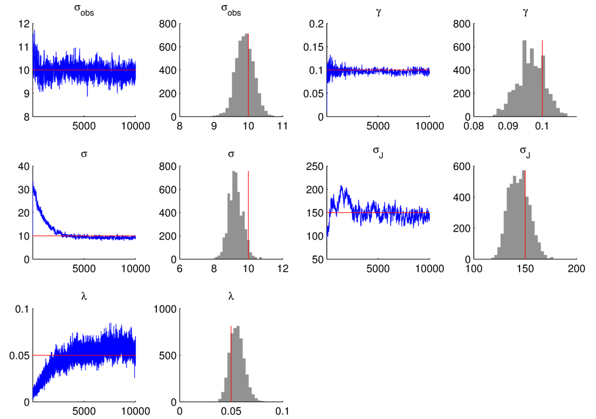

Fig. 2. Parameter estimation for synthetic data. Red lines show true parameter values. Left chart of each pair shows MCMC sequence for 10000 samples. Right chart of each pair shows histogram of final 5000 samples (5000 sample burn-in)

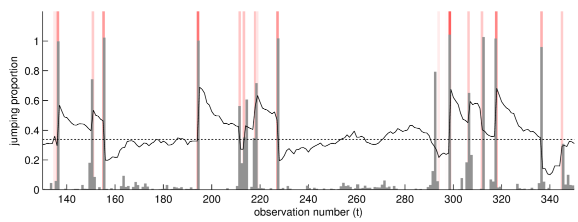

Fig. 3. Jump detection in trend process for synthetic data. Red bars show true jump positions, with colour intensity indicating jump intensity. Grey bars show proportion of samples in which a jump is present at each time (over 5000 samples, after 5000 sample burn-in). Black line is trend process (no scale shown), with dotted line indicating zero

the distributions of parameter samples for all parameters. As shown in figure 3, jump positions are also well estimated, with all jumps being identified and low false detection levels. These results are typical of those for synthetic data using this parameter estimation approach.

5.1. Financial Data

The algorithm was applied to 1500 daily USD/GBP exchange rates between June 2007 and March 2013 (multiplied by 1000 to have a similar scale to equity indices), giving the parameter estimation results shown in figure 4. As the data is daily (and comes from a major currency pair), it is possible to compare the parameter estimates derived here to those used in [13], which were estimated *ad hoc* by choosing parameters that produced good Sharpe Ratios in backtesting, rather than directly from the data; see table 1. These show that the estimates in [13], are somewhat different from those obtained using the model-based estimation procedure here. The parameters used

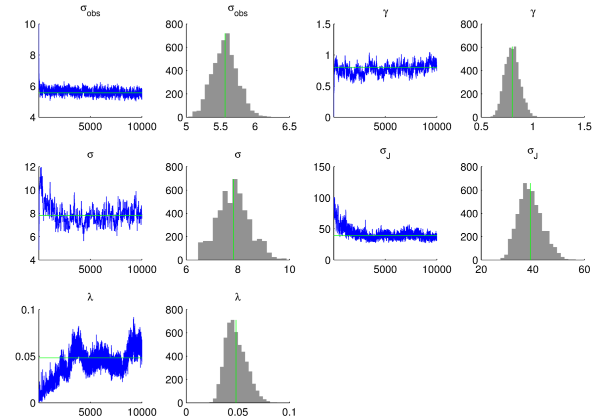

Fig. 4. Parameter estimation (10000 samples) for USD/GBP daily exhange rate from June 2007 to March 2013; green lines show mean parameter value over 5000 post-burn in samples

	σ_{obs}	γ	σ	σ_J	λ
USD/GBP	5.6 (0.18)	0.8 (0.07)	7.8 (0.6)	39 (4.6)	0.05 (0.01)
From [13]	26	0.2	7.0	120	0.2
S&P500	7.1 (0.46)	0.73 (0.14)	8.4 (1.1)	45 (7.7)	0.05 (0.02)
From [13]	20	0.2	4.1	70	0.2

Table 1. Estimated parameter means (standard deviations) compared with those used in [13]

in [13] overestimate jump scale σ_J and rate λ as well as observation noise scale σ_{obs}, but underestimate process noise σ and trend mean reversion rate γ. The high estimated rate of mean reversion indicates that trends of the sort detectable with this model are short-lived. Similar results were found when estimation was conducted using daily S&P500 index prices from October 2010 to March 2013 [18], as shown in table 1.

Matlab code for the system detailed in this paper can be found at **www-sigproc.eng.cam.ac.uk/Main/JM362**.

6. CONCLUSION

This paper has presented a batch method of parameter estimation for Jump-Langevin systems of the type used to model trends in financial data in earlier work [13], based on reversible jump MCMC for jump time detection. Tests on synthetic data show the algorithm is effective at correctly estimating system parameters and detecting the times of jumps in the trend process. Testing on financial data found differences between the *ad hoc* parameter values used in earlier work, and those estimated from the data, particularly for observation noise scale σ_{obs} and mean reversion rate γ. The effect of this mis-specification on the results in [13] is currently unknown. The results for USD/GBP exchange rates and S&P500 data appear to indicate that the types of trends described by this model are short-lived in the periods tested, as indicated by the high mean-reversion rate γ.

7. REFERENCES

[1] J. Miffre and G. Rallis, "Momentum strategies in commodity futures markets," *J. of Banking & Finance*, vol. 31, pp. 1863–1886, 2007.

[2] J. Okunev and D. White, "Do momentum-based strategies still work in foreign currency markets?," *Journal of Financial and Quantitative Analysis*, vol. 38, no. 02, pp. 425–447, 2003.

[3] K. Chan, A. Hameed, and W. Tong, "Profitability of momentum strategies in the international equity markets," *Journal of Financial and Quantitative Analysis*, vol. 35, no. 2, pp. 153–172, 2000.

[4] E.F. Fama, "Efficient capital markets: A review of theory and empirical work," *The Journal of Finance*, vol. 25, no. 2, pp. 383–417, 1970.

[5] C.S. Asness, T.J. Moskowitz, and L.H. Pedersen, "Value and momentum everywhere," *The Journal of Finance*, vol. 68, no. 3, pp. 929–985, 2013.

[6] S. Hwang and A. Rubesam, "The disappearance of momentum," *The European Journal of Finance*, 2013.

[7] D.A. Lesmond, M.J. Schill, and C. Zhou, "The illusory nature of momentum profits," *Journal of Financial Economics*, vol. 71, no. 2, pp. 349–380, 2004.

[8] R.J. Shiller, *Irrational exuberance*, Princeton University Press, 2005.

[9] D. Vayanos and P. Woolley, "An institutional theory of momentum and reversal," *NBER Working Papers*, 2008.

[10] H. Hong, T. Lim, and J.C. Stein, "Bad news travels slowly: Size, analyst coverage, and the profitability of momentum strategies," *Journal of Finance*, pp. 265–295, 2000.

[11] N. Jegadeesh and S. Titman, "Returns to buying winners and selling losers: Implications for stock market efficiency," *Journal of Finance*, pp. 65–91, 1993.

[12] L. Sun and C. Stivers, "Cross-sectional return dispersion and time variation in value and momentum premium," *Journal of Financial and Quantitative Analysis*, vol. 45, pp. 987–1014, 2010.

[13] H.L. Christensen, J. Murphy, and S.J. Godsill, "Forecasting high-frequency futures returns using online Langevin dynamics," *IEEE Journal of Selected Topics in Signal Processing*, vol. 6, no. 4, pp. 366–380, 2012.

[14] G. Yang and S.J. Godsill, "Bayesian inference for continuous-time ARMA models driven by jump diffusions," in *Statistical Signal Processing, 2007. SSP '07. IEEE/SP 14th Workshop on*, Aug 2007, pp. 99–103.

[15] Robert A Singer, "Estimating optimal tracking filter performance for manned maneuvering targets," *Aerospace and Electronic Systems, IEEE Transactions on*, , no. 4, pp. 473–483, 1970.

[16] P.J. Green, "Reversible jump Markov chain Monte Carlo computation and Bayesian model determination," *Biometrika*, vol. 82, no. 4, pp. 711, 1995.

[17] Peter J Green and David I Hastie, "Reversible jump MCMC," *Genetics*, vol. 155, no. 3, pp. 1391–1403, 2009.

[18] James Murphy, *Hidden States, Hidden Structures: Bayesian Learning in Time Series Models*, Ph.D. thesis, Department of Engineering, Cambridge University, 2013.