Experiment Design, Bayesian Estimation and Model Selection of an Autonomous Underwater Vehicle

Roger Stuckey[1] and Tristan Perez[2]

***Abstract*— This article investigates the design of experiments for parameter extraction and the identifiability of model parameters of an AUV. In addition, the application of Bayesian model selection techniques to determine the significant damping components is studied. The damping model is structured as a statistical linear model from which coefficients are selected based on a maximal evidence (in the Bayes sense) for a given set of data.**

I. INTRODUCTION

Often in vehicle dynamics modelling, the prior knowledge leads us to conjecture a particular model structure that may capture the essential dynamics that are of importance for the application under consideration. This then leads to two questions:

- How do we design an optimal experiment?
- What is the best result that can be obtained from such a experiment?

This document has a section dealing with each, following the section describing the vehicle model. Section III deals with the experiment design problem. If we adopt a particular model structure, then we seek to design an experiment u_t for $t = 1, \ldots, N$. which is in some way optimal. One way to prescribe such a design is to consider the information yielded by an experiment and attempt to maximise it. This can be captured by a reduction in entropy about the parameter set estimated by the experiment.

For the vehicle problem we can look at particular experiments and use the differential entropy maximisation. One experiment commonly used is the self-oscillating test, which is a generalisation of the zig-zag test used in ship manoeuvring. In his case, the optimisation is with respect to the maximum control input magnitude and the desired amplitude of oscillations. For simple systems, this can also be approximated by a doublet input, based on the control input amplitude and period.

Section IV explores the model structure selection problem. Even if we design the optimal experiment for the model structure that we seek to use, it may happen that given the constraints and the model structure selected, a sub-model could provide a better alternative. For example it is well known that free-flying experiments may not be as informative as captive model experiments.

[1]Roger Stuckey is with the Autonomous Systems Group, Maritime Division, Defence Science and Technology Organisation, Eveleigh, NSW 2015, Australia `roger.stuckey@dsto.defence.gov.au`

[2]Tristan Perez is with the Department of Electrical Engineering and Computer Science, Science and Engineering Faculty, Queensland University of Technology, Gardens Point, Brisbane, QLD 4000, Australia `tristan.perez@qut.edu.au`

This means that we can consider a set of model of varying complexity: $M_1, M_2, \ldots, M_m$. Where $M_i \subseteq M_m$. Then what we can do is based on our optimal experiment select the model with highest probability $P(M_i|D)$:

$$P(M_i|D, I) = \frac{P(D|M_i, I)P(M_i|I)}{P(D|I)}$$

where these can be computed using odds relative to the model M_m, so we don't have to compute the global likelihood $P(D|I)$.

This can then could be iterated with the experiment design if the result is that the optimal model is not M_m, namely, $M^\star \subset M_m$.

II. VEHICLE MODEL

The longitudinal vehicle model is based on a linearised 3 degree-of-freedom open-loop model of an AUV [1]. The REMUS 100 AUV model uses hydrodynamic coefficients from [2] and subsequently [3]. For the following investigation, a simple surge (x-axis) sub-model was considered. Following the notation of [4], the dynamical system can be expressed in state-space as:

$$\dot{u}(t) = f(u(t), n(t)) + \omega_x \tag{1}$$

$$y(t) = g(x(t)) + \omega_y \tag{2}$$

where u is the forward velocity and n is the propellor angular velocity.

Expanding force coefficient terms, one form of the sub-model can be given by:

$$(m - X_{\dot{u}}\dot{u} = X_u u + X_{u|u|}u|u| + (1-\tau)T_{n|n|}n|n| \tag{3}$$

For the extended Kalman Filter, the Jacobian matrices are:

$$\nabla_u = X_u + 2X_{u|u|}\sqrt{u^2} \tag{4}$$

and:

$$\nabla_X = \begin{bmatrix} u & 0 & 0 \\ u|u| & 0 & 0 \\ (1-\tau)n|n| & 0 & 0 \end{bmatrix}^T \tag{5}$$

where $X = \begin{bmatrix} X_u & X_{u|u|} & T_{n|n|} \end{bmatrix}$.

Several simplifications include neutral buoyancy ($W = B$), zero axial c.g. offset ($x_G = 0$) and the omission of propellor inflow for the thrust force.

III. EXPERIMENT DESIGN

For the purpose of system identification, an optimal experiment is one that will generate the greatest amount of information from the system of interest, thereby maximising the confidence which can be attributed to a particular model. Shannon [5] introduced the concept of entropy, which represents the average uncertainty in a random system and is equivalent to its information content:

$$H = -E_\theta[p(\theta)\log p(\theta)] \tag{6}$$

$$= \int_\Theta p(\theta)\log p(\theta) d\theta \tag{7}$$

Lindley [6] later used Bayes Theorem to propose the following criterion as a measure of the amount of information produced by an experiment:

$$H(p(\theta), I, D) = H(D) - H(I), \tag{8}$$

where $H(I)$ and $H(D)$ represent the entropy of the prior and the posterior of the system respectively, parameterised by $\Theta = \{\theta_0, \theta_1, \ldots, \theta_N\}$:

$$H(I) = \int_\Theta p(\theta|I)\log p(\theta|I)\, d\theta \tag{9}$$

$$H(D) = \int_\Theta p(\theta|I,D)\log p(\theta|I,D)\, d\theta \tag{10}$$

I is the background information and D is the data resulting from conducting an experiment. $H(p(\theta), I, D)$ therefore captures the reduction in entropy about the parameter set estimated by the experiment. We wish to discover a sequence of inputs which will maximise this criteria over some variable space $\beta \in \mathcal{B}$ thereby producing the most informative experiment. That is:

$$\beta^\star = \underset{\beta\in\mathcal{B}}{\operatorname{argmax}}\, H(p(\theta), I, D) \tag{11}$$

where β represents constraints on the input design, for example power or amplitude.

For a multivariate normal distribution of $\theta_1, \theta_2, ..., \theta_n$ with mean μ and covariance σ, the probability density function is:

$$f(\Theta) = \frac{1}{(2\pi)^{n/2}|\sigma|^{1/2}} e^{-\frac{1}{2}(\theta-\mu)^T\sigma^{-1}(\theta-\mu)} \tag{12}$$

And the joint differential entropy is given by [7]:

$$H(\Theta) = \frac{1}{2}E\left[\sum_{i,j}\tilde{\theta}_i(\sigma^{-1})_{ij}\tilde{\theta}_j\right] + \frac{1}{2}\ln((2\pi)^n|\sigma|)$$
$$= \frac{1}{2}\ln((2\pi e)^n|\sigma|) \tag{13}$$

where $\tilde{\theta} = \theta - \mu$.

For the state-space system under consideration, this can be written in terms of the error covariance for either prior or posterior parameter sets, $P(\Theta)$:

$$H(\Theta) = \frac{1}{2}\ln((2\pi e)^n|P(\Theta)|) \tag{14}$$

There are several techniques that could potentially be used for estimating the posterior. The sampling approach, used in the following section, is one method which can produce excellent estimates. However, it is quite CPU-intensive, typically requiring many thousands of samples to obtain representative probability distribution histograms. If the final entropy is to be maximised through some variable sweep or optimisation function, this becomes onerous. A better approach is to use a single-pass algorithm, such as a Kalman Filter, to simultaneously predict the system response and parameter estimates, along with their covariance estimates.

For the surge model, Square-Root form of the Unscented Kalman Filter (SRUKF) [8] was employed. The SRUKF has a number of advantages over the Extended Kalman Filter (EKF). These include numerical stability and guaranteed positive semi-definiteness of the state covariances. As the name suggests, the SRUKF propagates the square-root of the covariance thereby avoiding expensive recalculation at every step. Unlike the standard UKF, which typically uses Cholesky factorisation to generate the sigma points, the SRUKF generates the sigma points directly and requires only QR decomposition and an efficient update to the Cholesky factor to update the covariance square-root.

The state vector for the system comprised vehicle state and coefficient variables:

$$x = \begin{bmatrix} u & X \end{bmatrix} \tag{15}$$

And the observation vector consisted of the state velocity and acceleration:

$$y = \begin{bmatrix} u & \dot{u} \end{bmatrix} \tag{16}$$

Kalman Filters require initial covariance matrix estimates for the posterior error P, process noise Q and measurement noise R. These can be difficult to ascertain - particularly the former two. Excessively increasing the initial estimate for P causes the coefficient traces to jump around erratically, resulting in worse parameter estimates with overly conservative (large) standard errors. Increasing Q *artificially* improves the residual state response at the expense of the parameters, many of which can end up diverging over the time period. R can typically be estimated, using a high-pass filter for example, in the case of real data. Otherwise, increasing R by too much decreases the influence of the model thereby impeding convergence of the coefficient estimates.

In tuning the covariance estimates initially, the following attributes were sought:

- Close and/or improving residual state variables without following the high frequency noise signal *too* closely.
- Asymptotic behaviour in each posterior error covariance.
- Converging coefficient estimates, not dominated by erratic behaviour.

For the surge sub-model, the process and measurement noise covariance matrices were:

$$Q = diag(\begin{bmatrix} 10^{-6} & 10^{-7} & 10^{-7} & 10^{-10} \end{bmatrix})$$
$$R = diag(\begin{bmatrix} 10^{-2} & 10^{-2} \end{bmatrix})$$

Coeff	True	Estimates			
		EKF	UKF	SRUKF	INIT
X_u	-1.21	-1.21 (0.37)	-1.27 (0.38)	-1.28 (0.38)	-0.91 (100.00)
$X_{u\|u\|}$	-2.93	-2.93 (0.18)	-2.91 (0.18)	-2.91 (0.18)	-2.20 (100.00)
$T_{n\|n\|}$	6.28	6.28 (0.10)	6.28 (0.10)	6.28 (0.01)	4.71 (316.00)

TABLE I
SURGE-FORCE COEFFICIENT KALMAN FILTER ESTIMATES

Coeff	True	Estimates		
		SRUKF	SRUKF-C	INIT
X_u	-1.21	-1.27 (0.38)	-1.24 (0.37)	-0.91 (100.00)
$X_{u\|u\|}$	-2.93	-2.91 (0.18)	-2.93 (0.18)	-2.20 (100.00)
$T_{n\|n\|}$	6.28	6.28 (0.10)	6.28 (0.10)	4.71 (316.00)

TABLE II
SURGE-FORCE COEFFICIENT KALMAN FILTER ESTIMATES

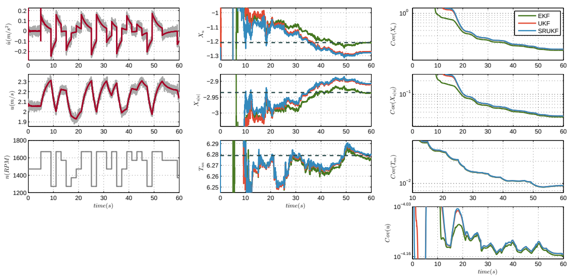

Fig. 1. EKF/UKF/SRUKF Model Response and Coefficient Estimates for Surge Manoueuvre: Random Step Input

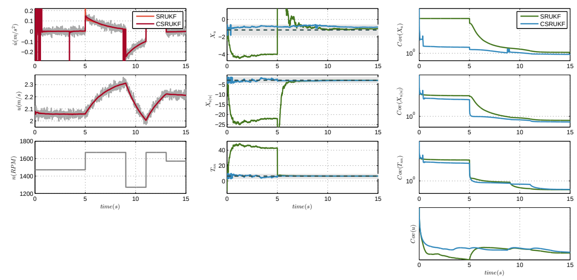

Fig. 2. SRUKF/SRUKF-C Model Response and Coefficient Estimates for Surge Manoueuvre: Random Step Input

And the initial estimate for the posterior error covariance was:

$$P = diag(\begin{bmatrix} 1 & 10^5 & 10^5 & 10 \end{bmatrix})$$

Prior to its use in the entropy maximisation procedure, the SRUKF was compared with the EKF and Unscented Kalman Filter (UKF) in terms of performance and accuracy. Table I and the corresponding Figure 1 illustrate the results for a sequence of random step inputs to the rotor (thrust) control. The estimates are listed with their standard deviation in braces and initial and true values. The initial estimates were set to 75% of their true values, intending to reflect some level of confidence in the model. In the figure, the system response is shown alongside the parameter and covariance estimates. The sample time for the simulation was 0.01s. Gaussian measurement noise was added to each observation variable with a standard deviation of approximately 5% of the uncorrupted signal amplitude.

The difference in the final estimates is very small for all three filters. The EKF produces marginally better results, converges more rapidly and is faster due to the inclusion of the Jacobian in the algorithm. However, the SRUKF maintains numerical stability as described above.

To aid the estimation further, the unscented transformation can also be constrained by clipping infeasible sigma points back onto the boundary of the feasible region [9]. This is only feasible with the SRUKF, since the state covariance estimate would otherwise tend to become singular during computation. Table II and Figure 2 illustrate the results for the constrained filter (SRUKF-C) using the same control input sequence over 15s. The upper and lower bounds were set at $\pm 50\%$ of the value for each coefficient.

For the coefficients, convergence is slightly more rapid and the final estimates closer using the constrained version. Most of the clipping occurs in the first 5s or so as can be seen in the coefficient traces. This has an adverse impact on the system state - namely $\dot{u}$ - which exhibits erratic behaviour during the first second. For this reason, the unconstrained SRUKF was chosen for the experiment design phase.

With a final estimate of the posterior error covariance, P, it was possible to calculate the reduction in entropy for a particular manoeuvre. For the surge sub-model examined, a simple repeated doublet-type input was employed; β was defined as a function of the thrust (propellor) control input amplitude, $n_{max}(RPM)$ and period, $\Delta T(s)$. To gain an appreciation of the effect of both variables over some space, a variable sweep was conducted for a 20×20 grid spanning the following range:

$$100.0 < (n_{max} - n_0) < 200.0$$
$$1.0 < \Delta T < 20.0$$

where the n_0 represents the trim value.

Figure 3 shows a surface plot of the reduction in entropy with thrust control input amplitude and period.

From the diagram, the increase in the entropy reduction with increasing n_{max} is pronounced. The variation in entropy with ΔT is not so clear, but appears to reach a maximum reduction around 10s. Beyond that, there is a diminishing increase; the doublet period would most likely be limited by other factors anyway, such as altitude or depth. In other words, for the repeated doublet control input the greatest reduction in entropy, or gain of information occurs for large amplitudes and periods greater than 10s. The ridges along the n_{max} axis correspond to discrete jumps in the number of whole doublets executed during the manoeuvre, since the period ΔT increases while the total simulation time remains fixed.

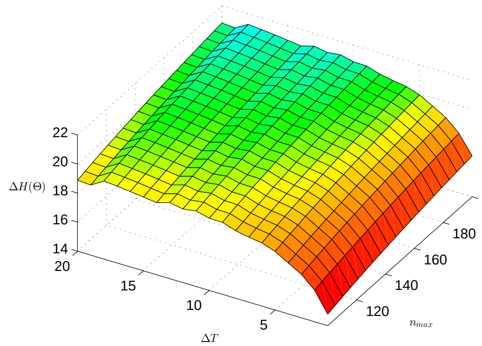

Fig. 3. Entropy Reduction with Control Input Parameters

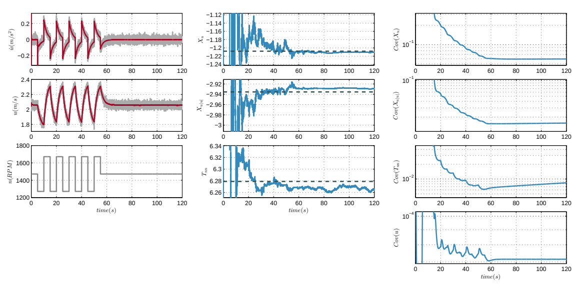

Fig. 4. SRUKF Model Response and Coefficient Estimates for Surge Manoueuvre

Figure 4 displays the system response, coefficient estimates and covariance estimates for the repeated doublet-type input with 10s period.

Note the long post-input time simulated to allow the system to stabilise, minimising the discrete jumps in entropy described above.

IV. ESTIMATION & MODEL SELECTION

Having established an optimal experiment for the maximisation of information, the model selection and estimation could take place. Since, initially at least, the estimation routine was not required to be embedded within an outer optimisation loop, a more accurate, longer-running technique was able to be employed. For this phase, a Markov Chain Monte Carlo (MCMC) global search using the PyMC software package [10] was chosen. By formulating a hierarchical stochastic model comprising prior distributions for the model and data, the posterior distributions could be obtained through sampling. Moreover, the previous assumption of normally-distributed priors was not necessary.

Figure 5 depicts the stochastic surge sub-model, with observation `obs_u` constructed with a 'mean' and precision. The 'mean' is represented by a deterministic function, `mean_u`, computed with the vehicle simulation model which is dependent on a set of stochastic coefficients `Xu`, `Xu|u|` and `Tn|n|`. The precision is modelled by another deterministic function `prec_u` and is dependent on a stochastic standard deviation `std_u`.

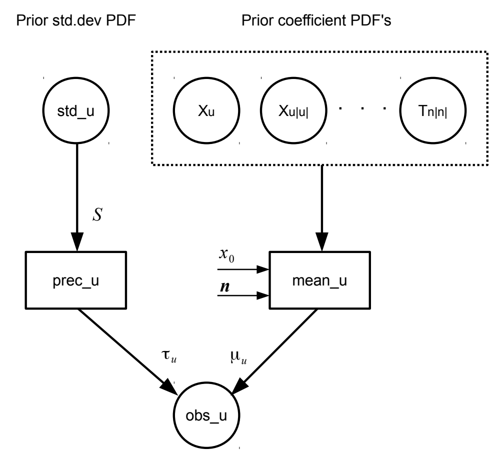

Fig. 5. Stochastic Surge Sub-Model Block Diagram

The circular blocks represent stochastic variables and the rectangular blocks deterministic functions.

The precision is given by:

$$\tau_u = \frac{1}{{\sigma_u}^2} \tag{17}$$

where σ_u is the standard deviation of the observation (forward velocity), modelled by a Uniform distribution with upper limit σ_{u_U}:

$$\sigma_u \sim U(0, \sigma_{u_U}) \tag{18}$$

The prior model coefficients are also modelled by Uniform distributions with limits $P_{u_L} < P_u < P_{u_U}$:

$$P_u \sim U(P_{u_L}, P_{u_U}) \tag{19}$$

where

$$P_u = \left[\begin{array}{cccc} X_u & X_{u|u|} & \cdots & T_{n|n|} \end{array} \right] \tag{20}$$

Finally, the observations have a Normal distribution as:

$$u \sim N(\mu_u(P_u), 1/\tau_u) \tag{21}$$

where $\mu_u(P_u)$ is a deterministic function with input arguments P_u.

In order to approximate the posterior distributions, PyMC provides a sampler and several step methods to choose from. For the vehicle model investigated, the Adaptive Metropolis step method, which is reported [11] to work more effectively with highly correlated variables. The Adaptative Metropolis step method works much like the Metropolis-Hastings step method, with the exception that its variables are block-updated using a multivariate jump distribution whose co-variance is tuned during sampling. From initial tests on the vehicle model, the Adaptive Metropolis step method yielded similar results, but is vastly quicker and reported to work more effectively on models with highly correlated variables.

Figure 6 shows the algorithm performance and output statistics for a two-stage MCMC run. The left column plots each of the stochastic variable (coefficient) traces over the entire run in green, overlaid with a convergence diagnostic over the first half. In order to alleviate scaling issues, the coefficients were all normalised by their true values. The first stage of 50K samples represents the 'burn-in' period and is discarded prior to calculation of the statistical results under the assumption that the Markov chain does not start near its converged state. It is possible to improve the starting values by seeding the Monte Carlo sampler with maximum *a-posteriori* estimates, however a burn-in stage is still recommended to improve mixing. The second stage of 50K samples represents the post 'burn-in' period, used for estimation of the posterior distributions. The convergence diagnostic is a time series approach [12], based on scores comparing the mean and variance of segments from the beginning and end of a single chain:

$$z = \frac{\bar{\theta}_a - \bar{\theta}_b}{\sqrt{Var(\theta_a) + Var(\theta_b)}}$$

where a is the early interval and b the late interval.

The z-scores are drawn as brown circles, overlaid on the traces in the first column, with the right-hand vertical axes units as standard deviations. If the chain has converged, the majority of points should fall within 2 standard deviations of zero, illustrated by the horizontal bands in each subplot.

The middle column plots the autocorrelation, for each of the stochastic variables. The raw autocorrelation is filled in light red, while the detrended version is overlaid as vertical lines in dark red.

The right column plots the posterior histograms for each of the stochastic variables in blue, with Normal approximations calculated using the mean and standard deviation of the traces drawn with dashed lines. Also shown are vertical lines representing the true values in red, mean values (estimates) in blue and vertical bands for the 95% Highest Probability Density (HPD) interval in grey.

Table III summarises the results from the same run.

For the surge model, the upper limit on the standard deviation for the forward velocity, σ_{u_U}, was set to 200% of that for the measurement noise added following simulation, as detailed in the previous section. For the prior distributions, the initial estimates (mean), $\mu_u(P_u)$, were set to 50% of

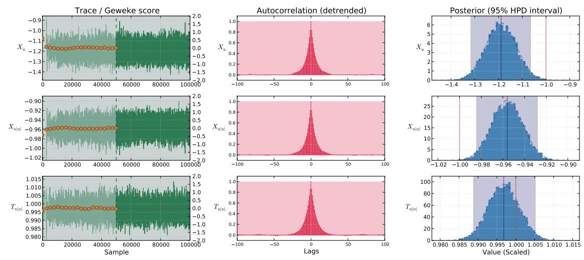

Fig. 6. Adaptive MCMC Run: 50K/50K samples

Coeff	True	Estimates	
		MCMC-A	INIT
X_u	-1.21	-1.44 (0.08)	-0.60
$X_{u\|u\|}$	-2.93	-2.81 (0.04)	-1.47
$T_{n\|n\|}$	6.28	6.28 (0.10)	3.14

TABLE III
SURGE-FORCE COEFFICIENT ESTIMATES

their true values and given wide bounds of $\pm 200\%$ of their absolute true values, intending to reflect some less confidence in the model.

The posterior estimates are within 3 standard deviations of their true values and exhibit near-normal distributions.

If the correct model is known *a-priori*, estimation via sampling is straightforward. Coefficients with a high degree of certainty, such as those obtained empirically, can be assigned uniform priors with relatively narrow bounds. Those with less certainty can be assigned wider bounds or uninformative priors. However, the model employed for analysis is often only an approximation to the physical system. Higher order terms may be included or omitted arbitrarily without any real basis, at the risk of under- or over-fitting. For example, [13] proposes a complex hydrodynamic damping model for ships, but for many purposes, this model is impractical and extremely difficult to estimate from recorded vehicle data. In model selection, we seek a parsimonious model comprising a subset of parameters which is well supported by the data.

Several criterion exist to assess parsimony within the Bayes formulation. In the past, Bayes factors [14] determined by Monte Carlo integration were used predominantly. More recently, several alternatives have become popular including the Deviance Information Criterion (DIC) and Bayesian Predictive Information Criterion (BPIC) [15]. The Deviance Information Criterion (DIC) is based on trade-off between the fit of the data and the corresponding complexity of the model.

Using the posterior mean deviance as a measure of fit:

$$\bar{D} = E_{\theta|y}[D] \tag{22}$$

The deviance is defined as:

$$D(\theta) = -2logp(y|\theta) \tag{23}$$

where $p(y|\theta)$ is the likelihood.

The complexity is measured by an estimate of the effective number of parameters:

$$p_D = E_{\theta|y}[D] - D(E_{\theta|y}[\theta]) \tag{24}$$
$$= \bar{D} - D(\bar{\theta}) \tag{25}$$

And the DIC is defined:

$$DIC = D(\theta) + 2p_D \tag{26}$$
$$= \bar{D} + p_D \tag{27}$$

The Bayes factor is simply a ratio of marginal likelihoods for competing models. Bayes factor may be calculated using likelihoods that have been integrated with respect to the unknown parameters.

$$BF_{i,j} = \frac{P(D|M_i, I)P(M_i|I)}{P(D|M_j, I)P(M_j|I)} \tag{28}$$

Using the same stochastic model framework developed above, the Bayes Factor was determined by calculating the ratio of average regularized likelihoods after sampling from the joint prior distributions to yield the log-likelihood for each model. Listing 1 outlines the Python code for computing the Bayes factor for n models.

```
# Set all variables to random values drawn from
# joint 'prior', meaning contributions of data
# to the joint distribution are not considered.
for m in models:
  loglikes[m] = zeros(iter)
  for i in xrange(iter):
    m.draw_from_prior()
    # Calculate log-likelihood
    loglikes[m][i] = m.obs_u.logp
    # Find max log-likelihood
    if (loglikes[m][i] > max_ll):
      max_ll = loglikes[m][i]
for m in models:
  # Regularize, exponentiate and average
  likes[m] = mean(exp(loglikes[m] - max_ll))
  # Multiply in the priors
  likepriors[m] = likes[m]*priors[m]
# Apply normalizing constant
sumlp = sum(likepriors.values())
for m in models:
  bf[m] = likepriors[m]/sumlp
```

Listing 1. Python Bayes Factor algorithm

To test the algorithm, we add an additional term to the simulation model, $X_{\Delta u^3}$, where Δu represents the change in forward velocity from trim. The coefficient was given an arbitrary value of 0.1 to impose a very small contribution to the total surge force. The Bayes factor was computed for two models: one with and one without the term, M-0 and M-1 respectively. Table IV summarises the models and results compiled from 100K samples.

The asterisk indicates which model was used to simulate the observations. Model M-1 is shown to be favourable, with the larger fraction of 0.5358, suggesting that the additional term $X_{\Delta u^3}$ is not supported by the data.

Model	BF	Coeff			
		X_u	$X_{u\|u\|}$	$T_{n\|n\|}$	$X_{\Delta u^3}$
M-0*	0.4642	✓	✓	✓	✓
M-1	0.5358	✓	✓	✓	

TABLE IV
BAYES FACTOR FOR SURGE-FORCE MODELS: $X_{\Delta u^3} = 0.1$

Model	BF	Coeff			
		X_u	$X_{u\|u\|}$	$T_{n\|n\|}$	$X_{\Delta u^3}$
M-0*	0.5501	✓	✓	✓	✓
M-1	0.4499	✓	✓	✓	

TABLE V
BAYES FACTOR FOR SURGE-FORCE MODELS: $X_{\Delta u^3} = 1.0$

Coeff	True	Estimates		
		Model M-0	Model M-1	INIT
X_u	-1.21	-0.97 (0.37)	-1.26 (0.07)	-0.60
$X_{u\|u\|}$	-2.93	-3.05 (0.19)	-2.90 (0.04)	-1.47
$T_{n\|n\|}$	6.28	6.28 (0.10)	6.28 (0.10)	3.14
$X_{\Delta u^3}$	0.10	7.78 (9.65)	–	0.05
Criterion		Model M-0	Model M-1	
DIC		-13939.8	-13940.6	
BPIC		-13934.6	-13935.9	
p_D		5.2	4.8	

TABLE VI
SURGE-FORCE COEFFICIENT ESTIMATES

Increasing the magnitude of the coefficient to 1.0 results in the following Bayes factor, shown in Table V.

In this case, favour has sided with the full model, M-0, with a fraction of 0.5501. Clearly, the influence of additional terms on the total surge force has a significant bearing on their importance as held by the data.

To gain more insight into the posterior estimates, MCMC runs were conducted for both models. Because of the uncertainty in the variance of the additional term, it was given an uninformative prior distribution. The observation data was again generated using the full model M-0 with $X_{\Delta u^3} = 0.1$. Table VI and Figures 7 and 8 outline the results for 50K samples with a 50K sample burn-in phase.

Looking at the first figure, although the posterior distribution for the additional term has been well sampled - as illustrated by the near normal density - the variance is very large, with a standard error nearly two orders of magnitude larger than the true value of the coefficient. In the second figure, the posterior estimates seem unaffected by the loss of one term. In fact, they are closer to their true values with a lower variance as verified in the table.

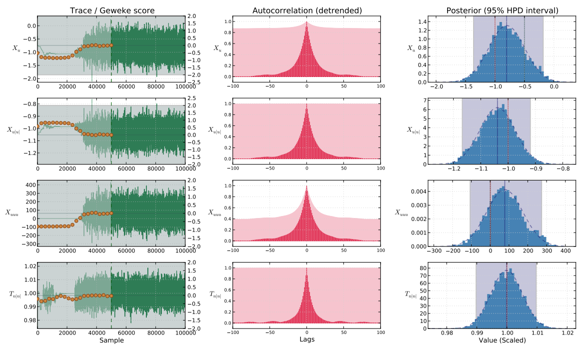

Fig. 7. Adaptive MCMC Run: 50K/50K samples, Model M-0

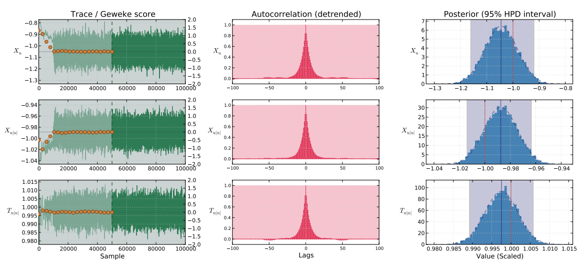

Fig. 8. Adaptive MCMC Run: 50K/50K samples, Model M-1

With lower values, both DIC and BPIC criterion also favour the reduced model, although their insensitivity to the change (less than 1%) makes them less useful as decision tools.

V. CONCLUSIONS

This document demonstrates a methodology for dynamic system identification based on two stages: experiment design to optimise the information produced and model selection for determination of the model structure. Moreover, it is feasible that the two stages could be repeated in an iterative manner in order to improve the accuracy and robustness of the estimated model. For a simple AUV surge sub-model, a sequence of control inputs that maximised the reduction in entropy were determined. Those same control inputs were then employed by a model in a hierarchical Bayesian estimation, to extract the coefficient distributions and select a parsimonious set.

APPENDIX

ACKNOWLEDGMENT

REFERENCES

[1] T. I. Fossen, *Handbook of Marine Craft Hydrodynamics and Motion Control*. Wiley, 2011.

[2] T. Prestero, "Verification of a six-degree of freedom simulation model for the remus autonomous underwater vehicle," Ph.D. dissertation, Massachusetts Institute of Technology and Woods Hole Oceanographic Institution, 2001.

[3] D. E. Sgarioto, "Control system design and development for the remus autonomous underwater vehicle," Defence Technology Agency, Tech. Rep. 240, May 2007.

[4] J. Refsnes, "Nonlinear model-based control of slender body auvs," Ph.D. dissertation, Norwegian University of Science and Technology, December 2007.

[5] C. E. Shannon, "A mathematical theory of communication," *Bell System Technical Journal*, vol. 27, 1948.

[6] D. Lindley, "On the measurement of information provided by an experiment," *Annals of Mathematical Statistics*, vol. 27, pp. 986–1005, 1956.

[7] T. M. Cover and J. A. Thomas, *Elements of Information Theory (Wiley Series in Telecommunications and Signal Processing)*. Wiley-Interscience, 1991.

[8] R. Van der Merwe and E. Wan, "The square-root unscented kalman filter for state and parameter-estimation," in *Acoustics, Speech, and Signal Processing, 2001. Proceedings. (ICASSP '01). 2001 IEEE International Conference on*, vol. 6, 2001, pp. 3461–3464 vol.6.

[9] R. Kandepu, L. Imsland, and B. A. Foss, "Constrained state estimation using the unscented kalman filter," in *Control and Automation, 2008 16th Mediterranean Conference on*, 2008, pp. 1453–1458.

[10] A. Patil, D. Huard, and C. Fonnesbeck, "PyMC: Bayesian stochastic modelling in python," *Journal of Statistical Software*, vol. 35, no. 4, pp. 1–81, July 2010. [Online]. Available: http://www.jstatsoft.org/v35/i04

[11] H. Haario, E. Saksman, and J. Tamminen, "An adaptive metropolis algorithm," *Bernoulli*, vol. 7, no. 2, pp. 223–242, April 2001. [Online]. Available: http://www.jstor.org/stable/3318737

[12] J. Geweke, "Evaluating the accuracy of sampling-based approaches to calculating posterior moments," in *Bayesian Statistics*, 1992, pp. 169–193.

[13] A. Ross, T. Perez, and T. Fossen, "A novel manoeuvring model based on low-aspect-ratio lift theory and lagrangian mechanics." in *IFAC Conference on Control Applications in Marine Systems (CAMS)*, Bol, Croatia, September 2007.

[14] R. E. Kass and A. E. Raftery, "Bayes factors," *Journal of the american statistical association*, vol. 90, no. 430, pp. 773–795, 1995.

[15] T. Ando, "Predictive bayesian model selection," *American Journal of Mathematical and Management Sciences*, vol. 31, no. 1-2, pp. 13–38, 2011. [Online]. Available: http://www.tandfonline.com/doi/abs/10.1080/01966324.2011.10737798