Online Prediction under Model Uncertainty Via Dynamic Model Averaging: Application to a Cold Rolling Mill [1]

Adrian E. Raftery
University of Washington
Seattle, USA

Miroslav Kárný
Academy of Sciences
Czech Republic

Josef Andrýsek
Academy of Sciences
Czech Republic

Pavel Ettler
COMPUREG Plzeň, s.r.o.
Plzeň, Czech Republic

Technical Report no. 525
Department of Statistics
University of Washington
Seattle, USA

December 14, 2007

[1]Adrian E. Raftery is Blumstein-Jordan Professor of Statistics and Sociology, Box 354320, University of Washington, Seattle, WA 98195-4320; email: raftery@u.washington.edu. Miroslav Kárný is Head and Josef Andrýsek is Researcher, Department of Adaptive Systems, Institute of Information Theory and Automation (ÚTIA), Czech Academy of Sciences, Prague, Czech Republic; email: school/andrysek@utia.cas.cz. Pavel Ettler is with COMPUREG Plzeň, s.r.o., 30634 Plzeň, Czech Republic; email: ettler@compureg.cz. Raftery's research was supported by the DoD Multidisciplinary Research Initiative (MURI) administered by the Office of Naval Research under grant N00014-01-10745. The research of Kárný and Andrýsek was supported by grant number 1ET 100 750 401 from the Academy of Sciences of the Czech Republic, "Bayesian adaptive distributed decision making," and by grant number 1M0572 from the Czech Ministry of Education, Youth and Sports (MSMT), "Research Center DAR." This research was carried out while Raftery was visiting ÚTIA.

Abstract

We consider the problem of online prediction when it is uncertain what the best prediction model to use is. We develop a method called Dynamic Model Averaging (DMA) in which a state space model for the parameters of each model is combined with a Markov chain model for the correct model. This allows the "correct" model to vary over time. The state space and Markov chain models are both specified in terms of forgetting, leading to a highly parsimonious representation. The method is applied to the problem of predicting the output strip thickness for a cold rolling mill, where the output is measured with a time delay. We found that when only a small number of physically motivated models were considered and one was clearly best, the method quickly converged to the best model, and the cost of model uncertainty was small; indeed DMA performed slightly better than the best physical model. When model uncertainty and the number of models considered were large, our method ensured that the penalty for model uncertainty was small. At the beginning of the process, when control is most difficult, we found that DMA over a large model space led to better predictions than the single best performing physically motivated model.

Contents

1 Introduction **1**

2 The Rolling Mill Problem **3**

3 Dynamic Model Averaging (DMA) **5**
3.1 The One-Model Case 5
3.2 The Multi-Model Case 8
3.3 Connection to Static Bayesian Model Averaging 11

4 Results **12**

5 Discussion **18**

List of Tables

1 List of Models Used 13
2 Sample Statistics of Prediction Errors 15

List of Figures

1 Schematic drawing of a reversing cold rolling mill 4
2 Measurement Time Delay 5
3 Rolling mill data 6
4 Posterior Model Probabilities of Three Models 15
5 Posterior Model Probabilities for all 17 Models 16
6 Comparison of prediction errors for Model 3 and DMA 17

1 Introduction

We consider the problem of online prediction when it is uncertain what the best prediction model to use is. Online prediction is often done for control purposes and typically uses a physically-based model of the system. Often, however, there are several possible models and it is not clear which is the best one to use.

To address this problem, we develop a new method called Dynamic Model Averaging (DMA) that incorporates model uncertainty in a dynamic way. This combines a state space model for the parameters of each of the candidate models of the system with a Markov chain model for the best model. Both the state space and Markov chain models are estimated recursively, allowing which model is best to change over time. Both the state space and Markov chain models are specified using versions of forgetting, which allows a highly parsimonious representation. The predictive distribution of future system outputs is a mixture distribution with one component for each physical model considered, and so the best prediction is a weighted average of the best predictions from the different models.

The physical theory underlying the prediction or control problem is often somewhat weak, and may be limited essentially to knowing what the inputs are that could potentially influence the output of a system. In that case we consider a model space consisting of all possible combinations of inputs that are not excluded by physical considerations.

The DMA methodology combines various existing ideas, notably Bayesian model averaging, hidden Markov models, and forgetting in state space modeling. Bayesian model averaging (BMA) (Leamer 1978; Raftery 1988; Hoeting et al. 1999; Clyde and George 2004) is an established methodology for statistical inference from static datasets in the presence of model uncertainty, and has been particularly well developed for linear regression when there is uncertainty about which variables to include (Raftery, Madigan, and Hoeting 1997; Fernández, Ley, and Steel 2001; Eicher, Papageorgiou, and Raftery 2007). BMA usually addresses the problem of uncertainty about variable selection in regression by averaging over all possible combinations of regressors that are not excluded by physical considerations. BMA is restricted to static problems, however. An extension to dynamic updating problems was proposed by Raftery et al. (2005) in the context of probabilistic weather forecasting, using a sliding window estimation period consisting of a specified previous number of days. DMA is a recursive updating method rather than a windowing one.

The idea of the hidden Markov model is that there is an underlying unobserved discrete-valued process whose value affects the system state and which evolves according to a Markov chain. The idea seems first to have been proposed independently by Ackerson and Fu (1970) for the case where the noise in a Kalman filter is a Gaussian mixture, and by Harrison and Stevens (1971) for modeling time series that can have outliers and jumps in level and trend. The latter has been extended to the dynamic linear model (Harrison and Stevens 1976; West

and Harrison 1989) and the multiprocess Kalman filter (Smith and West 1983). The basic idea has since been widely used, often under different names, in different disciplines including speech recognition (Rabiner 1989) and genomics (Eddy 1998; Eddy 2004). In economics, the Markov switching model of Hamilton (1989) is widely used for time series in which the autoregressive parameters switch between different regimes. Markov switching models are also widely used for tracking moving or manoeuvering objects, particularly in aerospace engineering (Li and Jilkov 2005). An important unifying framework is the conditional dynamic linear model (CDLM) of Chen and Liu (2000), which includes several earlier proposals as special cases,

To specify DMA, we postulate the existence of a hidden Markov chain on the model space. This differs from many other hidden Markov applications because the definition of the state itself, and not just its value, depends on the current value of the chain. As a result, our method is not a special case of the CDLM, although it is related to it.

One of the difficulties with hidden Markov models is the need to make inference about the full sequence of hidden values of the chain, and the resulting computational explosion. Various approximations have been proposed, including finite-memory approximations (Ackerson and Fu 1970; West and Harrison 1989), the interacting multiple model (IMM) algorithm of Blom and Bar-Shalom (1988), which is popular in tracking applications (Mazor et al. 1998), and the ensemble Kalman filter (Evensen 1994), also known as the particle filter or sequential importance sampling. The fact that in our case the state vector has a different definition for each candidate model allows us to use a very simple approximation in which each model is updated individually at each time point. We show that this is equivalent to an age-weighted version of BMA, which makes it intuitively appealing in its own right.

In many previous hidden Markov applications there is considerable information about the evolution of the state. In the rolling mill application that motivated our work, there is considerable physical knowledge about the rolling mill itself, but little physical knowledge about how the regression model and its parameters (which define the system state in our setup) are likely to evolve. All that can be assumed is that the parameters are likely to evolve gradually in time, and that the model is likely to change infrequently. As a result, rather than specify the state space model fully, which is demanding and for which adequate information is not available, we specify the evolution of the parameters and the model by exponential forgetting (Fagin 1964; Jazwinsky 1970; Kulhavý and Zarrop 1993).

We illustrate the methodology by applying it to the prediction of the outgoing strip thickness of a cold rolling mill, ultimately for the purpose of either human or automatic control. The system input and three adjustable control parameters are measured, and the system output is measured with a time delay. While there is considerable understanding of the physics of the rolling mill, three plausible models have been proposed, of which two are physically motivated. We first implemented DMA for these three models, one of which turned

out to be clearly best. DMA converged rapidly to the best model, and the performance of DMA was almost the same as that of the best model — DMA allowed us to avoid paying a penalty for being uncertain about model structure.

We then extended the set of models to allow for the possibility of each of the inputs having an effect or not, yielding a much bigger model space with 17 models. In this case we found that DMA performed better than the best physical model in the initial, most unstable period when control is most difficult, and comparably over the rest of the period, when prediction errors were generally stable. Thus, even in this case with a larger model space, DMA allowed us to avoid paying a penalty for model uncertainty, and indeed allowed us to use the model uncertainty to increase the stability of the predictions in the early unstable period.

The rest of the paper is organized as follows. In Section 2 we describe the rolling mill problem that motivated our research. In Section 3 we describe the dynamic model averaging methodology, and in Section 4 we show the results of single-model and DMA prediction for the rolling mill. In Section 5 we discuss limitations, possible extensions and alternatives to the proposed methodology.

2 The Rolling Mill Problem

The problem that motivated our methodology is that of predicting the output of a cold rolling mill. A cold rolling mill is a machine used for reducing the thickness of metal strips; a schematic drawing is shown in Figure 1. Metal is passed through a gap and subjected to the rolling force. The strip thickness is measured by diamond-tipped contact meters on both sides of the rolling mill, providing measurements of the input and output thickness. A target thickness is defined, and this needs to be achieved with high accuracy depending on the nominal thickness. In our case a typical tolerance was $\pm$ 10 microns.

The output that we want to predict is the output strip thickness which is, under normal conditions, securely controlled using automatic gauge control (Ettler and Jirovský 1991). This control method can be improved manually by actions of the operators, potentially supported by a decision-support system (Quinn et al. 2003; Ettler, Kárný, and Guy 2005).

Reliable thickness prediction can improve control, particularly under adverse conditions that the automatic gauge control system is not designed to handle. These can arise at the beginning of a pass of the material through the rolling mill, when the thickness meters often do not operate for a period because there is a danger that they may be damaged. Adverse conditions can also arise for welded strips and for strip parts with uneven surfaces.

Important variables that can be adjusted to achieve the target are the size of the rolling gap governed by the roll positioning system, the rolling force, the input and output rolling speeds and – possibly – strip tensions. Rolling forces can be of the order of 10^6 N, and rolling

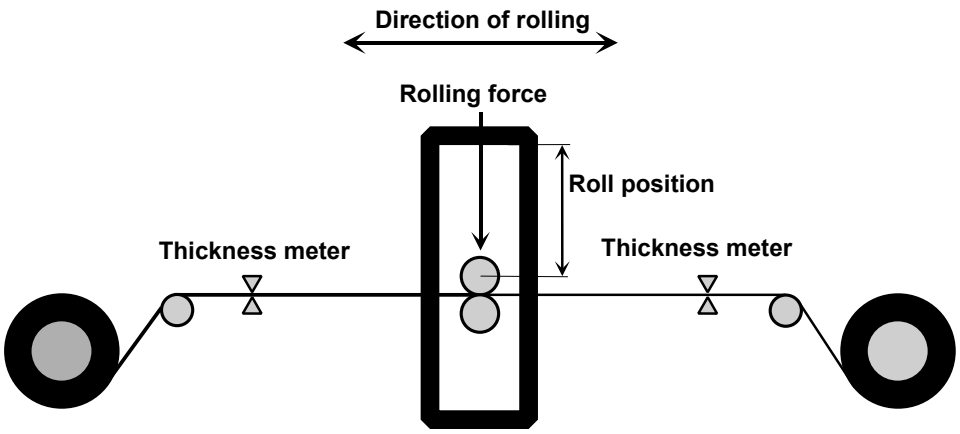


Figure 1: Schematic drawing of a cold rolling mill

speeds of the order of 0.1–8 m/s. The rolling gap cannot be reliably measured, but the roll position is available instead, measured against the mill frame. This position differs from the gap mainly because of frame elongation during rolling.

Modern rolling mill control systems allow every sample processed to be archived. Data collection is triggered by the strip movement. In our case, data samples are recorded every 4 cm, so that the sampling period varies according to the strip speed. For instance, if the speed is 1 m/s, the sampling period is 40 milliseconds. This imposes constraints on the complexities of online data processing, which must be completed well within the sampling period. These constraints are a key element of the problem, and imply that the methods used must be computationally efficient.

Our goal is to predict the output thickness for the piece of the strip just leaving the rolling gap. We wish to do this online, either to provide guidance to human operators controlling the system in real time in situations that preclude the use of automatic gauge control, or as an additional input to the control system. This is essentially a regression problem. However, an important complicating aspect of the data collection system is the time delay problem, illustrated in Figure 2.

Current values of the regressors are available for the piece of the strip in the rolling gap. However, the system output — the output strip thickness — is measured only with a delay d, which in our case was $d = 24$ samples. The task thus consists of predicting the output thickness, given the input thickness and other measured data at the gap. But data for estimating the regression relationship are available only with a delay d.

The data that we analyze come from a strip of length about 750 m that yielded 19,058 samples. The roll position was controlled by automatic gauge control with feedback. The data from the first 1,000 samples are shown in Figure 3. Figure 3(a) shows the deviations

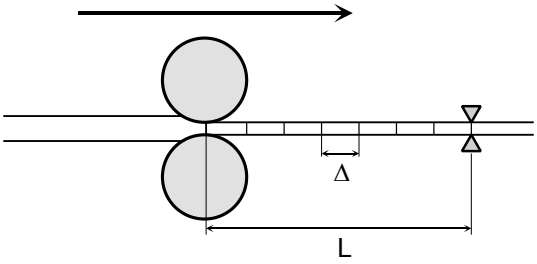


Figure 2: Measurement Time Delay. A measurement is made every time the material moves by a length Δ, equal to one sample. The output thickness currently at the gap is not measured until the material has moved a distance L beyond the gap. The measurement time delay is $d \approx L/\Delta$ samples, where d is a natural number.

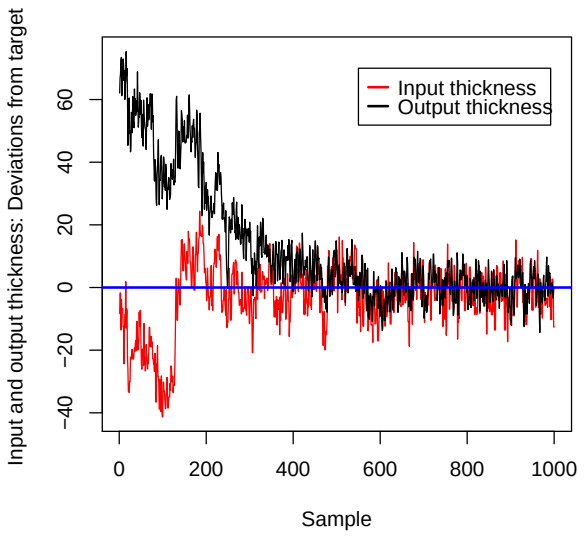
of the input and output thicknesses from their nominal values, which are 1398 and 1180 microns, respectively. (The input was about 200 microns thicker than the desired output, allowing a margin for the rolling mill to do its work of thinning the material.) Initially the output was too thick, and gradually converged to a relatively stable situation after about 600 samples. Figure 3(b) shows the roll position, Figure 3(c) the ratio of the output to the input rolling speeds, and Figure 3(d) the rolling force. The gradual adjustment of the controlled variable by the feedback controller can be seen. The two outliers in the rolling speed ratio at samples 42 and 43 provide a challenge to the system.

3 Dynamic Model Averaging (DMA)

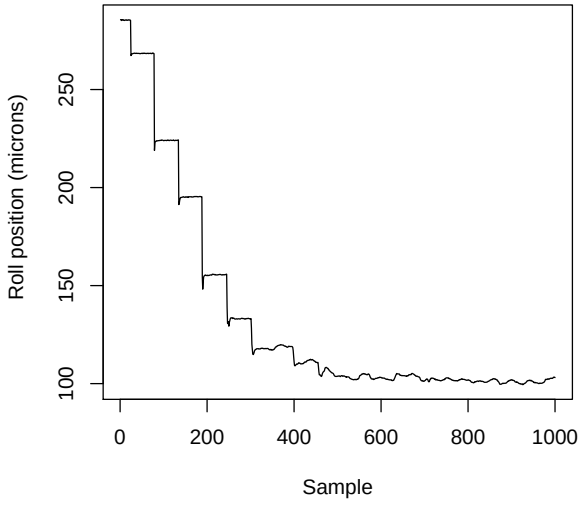
We first consider the situation where a single, physically-based regression-type model is available for prediction. We will review a standard state-space model with forgetting for adaptive estimation of the regression parameters and prediction with time delay. This is essentially standard Kalman filtering, but we review it here to fix ideas and notation. We will then address the situation where there is uncertainty about the model structure and several plausible models are available. For concreteness, we will use the rolling mill terminology.

3.1 The One-Model Case

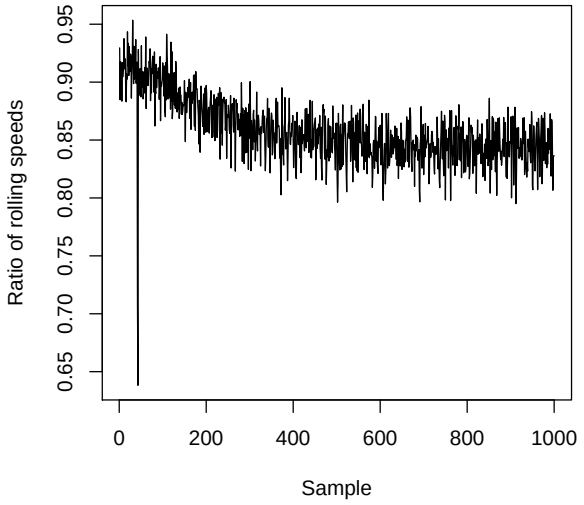
Let y_t be the deviation of the output thickness of sample t from its target value, and let $x_t = (x_{tj} : j = 1, \ldots, \nu)$ be the corresponding vector of inputs. Typically $x_{t1} = 1$, corresponding to the regression intercept, and x_t will also include the input thickness and other inputs and possibly also a past value of the output; the latter would lead to an ARX model (Ljung 1987).



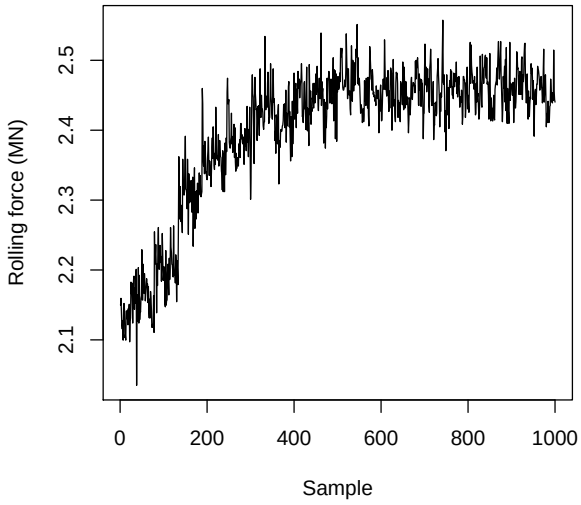
(a) Input and output thickness deviations from target



(b) Roll position



(c) Ratio of rolling speeds



(d) Rolling force

Figure 3: Data from the rolling mill: first 1,000 samples. (a) Input and output thickness deviations from their nominal values (in microns); (b) Roll position (in microns); (c) Ratio of output to input rolling speeds; (d) Rolling force at the gap (in MN).

The *observation equation* is then

$$y_t = x_t^T \theta_t + \varepsilon_t, \tag{1}$$

where a superscript T denotes matrix transpose, θ_t is a vector of regression parameters and the innovations ε_t are distributed as $\varepsilon_t \overset{\text{iid}}{\sim} N(0, V)$, where $N(\mu, \sigma^2)$ denotes the normal distribution with mean μ and variance σ^2. The regression parameters θ_t are allowed to evolve according to the *state equation*

$$\theta_t = \theta_{t-1} + \delta_t, \tag{2}$$

where the state innovations δ_t are distributed as $\delta_t \overset{\text{ind}}{\sim} N(0, W_t)$.

Inference is done recursively using Kalman filter updating. Suppose that $\theta_{t-1}|Y^{t-1} \sim N(\hat{\theta}_{t-1}, \Sigma_{t-1})$, where $Y^{t-1} = \{y_1, \ldots, y_{t-1}\}$. Then

$$\theta_t|Y^{t-1} \sim N(\hat{\theta}_{t-1}, R_t), \tag{3}$$

where

$$R_t = \Sigma_{t-1} + W_t. \tag{4}$$

Equation (3) is called the *prediction equation*.

Specifying the $\nu \times \nu$ matrix W_t completely is demanding, and often little information for doing so is available. Instead we follow Fagin (1964) and Jazwinsky (1970) and specify it using a form of *forgetting*. This consists of replacing (4) by

$$R_t = \lambda^{-1}\Sigma_{t-1}, \tag{5}$$

where λ is called the *forgetting factor* and is typically slightly below 1. The resulting model is a properly defined state space model, with $W_t = a\Sigma_{t-1}$, where $a = (\lambda^{-1} - 1)$. As pointed out, for example, by Hannan, McDougall, and Poskitt (1989), estimation in this model is essentially age-weighted estimation where data i time points old has weight λ^i, and the effective amount of data used for estimation, or the *effective window size*, is $h = 1/(1 - \lambda)$. This suggests that forgetting may be roughly comparable to windowing (Jazwinsky 1970), in which the last h available samples are used for estimation and are equally weighted. A windowing method like this was used in the multimodel context by Raftery et al. (2005).

Inference is completed by the *updating equation*,

$$\theta_t|Y^t \sim N(\hat{\theta}_t, \Sigma_t). \tag{6}$$

In (6),

$$\hat{\theta}_t = \hat{\theta}_{t-1} + R_t x_t^T (V + x_t^T R_t x_t)^T e_t, \tag{7}$$

where $e_t = y_t - x_t^T \hat{\theta}_{t-1}$ is the one-step-ahead prediction error, and

$$\Sigma_t = R_t - R_t x_t (V + x_t^T R_t x_t)^{-1} x_t^T R_t. \tag{8}$$

The inference process is repeated recursively as the measurements on a new sample become available. It is initialized by specifying $\hat{\theta}_0$ and Σ_0; we will discuss their values for the rolling mill problem in Section 4.

The information available for predicting y_t is d samples old because of the time-delay problem, and so we use

$$\hat{y}_t = x_t^T \hat{\theta}_{t-d-1}. \tag{9}$$

The observations innovations variance V needs to be specified by the user. Here we estimate it by a recursive method of moments estimator, using the fact that the one-step-ahead predictive distribution of y_t is given by

$$y_t | Y^{t-1} \sim N(x_t^T \hat{\theta}_{t-1}, V + x_t^T R_t x_t). \tag{10}$$

It follows that

$$V_t^* = \frac{1}{t} \sum_{r=1}^{t} [(y_t - x_t \hat{\theta}_{t-1})^2 - x_t^T R_t x_t]$$

is a consistent estimator of V in the sense that $V_t^* \longrightarrow V$ as $t \longrightarrow \infty$ in probability under the model defined by (1) and (2). It is not guaranteed, however, that $V_t^* > 0$. This leads us to define the recursive moment estimator

$$\hat{V}_t = \begin{cases} A_t & \text{if } A_t > 0; \\ \hat{V}_{t-1} & \text{otherwise,} \end{cases}$$

where

$$A_t = \left(\frac{t-1}{t} \right) \hat{V}_{t-1} + \frac{1}{t} (e_t^2 - x_t^T R_t x_t).$$

3.2 The Multi-Model Case

We now consider the case where multiple models, $M_1, \ldots, M_K$ are considered and there is uncertainty about which one is best. We assume that each model can be expressed in the form of equations (1) and (2), where the state vector for each model is different. They can be of different dimensions and need not overlap. The quantities specific to model M_k are denoted by a superscript (k). We let $L_t = k$ if the process is governed by model M_k at time t. We rewrite (1) and (2) for the multimodel case as follows:

$$y_t | L_t = k \sim N(x_t^{(k)T} \theta_t^{(k)}, V^{(k)}), \tag{11}$$

$$\theta_t^{(k)} | L_t = k \sim N(\theta_{t-1}^{(k)}, W_t^{(k)}). \tag{12}$$

We assume that the model governing the system changes infrequently, and that its evolution is determined by a $K \times K$ transition matrix $Q = (q_{k\ell})$, where $q_{k\ell} = P[L_t = \ell | L_{t-1} = k]$. The transition matrix Q has to be specified by the user, which can be onerous when the

number of models is large. Instead we again avoid this problem by specifying the transition matrix implicitly using forgetting.

Estimation of $\theta_t^{(k)}$ in general involves a mixture of K^t terms, one for each possible value of $L^t = \{L_1, \ldots, L_t\}$, as described by Ackerson and Fu (1970) and Chen and Liu (2000), for example. This number of terms rapidly becomes enormous, making direct evaluation infeasible, and many different approximations have been described in the literature. Here we are interested in prediction of the system output, y_t, given Y^{t-1}, and this depends on $\theta_t^{(k)}$ only *conditionally* on $L_t = k$. This leads to a simple approximation that works well in this case, consisting of simply updating $\theta_t^{(k)}$ conditionally on $L_t = k$ for each sample.

In this multimodel setup, the underlying state consists of the pair, (θ_t, L_t), where $\theta_t = (\theta_t^{(1)}, \ldots, \theta_t^{(K)})$. The quantity $\theta_t^{(k)}$ is defined only when $L_t = k$, and so the probability distribution of (θ_t, L_t) can be written

$$p(\theta_t, L_t) = \sum_{k=1}^{K} p\left(\theta_t^{(k)} | L_t = k\right)\ p(L_t = k). \tag{13}$$

The distribution in (13) is what we will update as new data become available.

Estimation thus proceeds analogously to the one-model case, consisting of a prediction step and an updating step. Suppose that we know the conditional distribution of the state at time $(t-1)$ given the data up to that time, namely

$$p(\theta_{t-1}, L_{t-1} | Y^{t-1}) = \sum_{k=1}^{K} p\left(\theta_{t-1}^{(k)} | L_{t-1} = k, Y^{t-1}\right)\ p(L_{t-1} = k | Y^{t-1}), \tag{14}$$

where the conditional distribution of $\theta_{t-1}^{(k)}$ is approximated by a normal distribution, so that

$$\theta_{t-1}^{(k)} | L_{t-1} = k, Y^{t-1} \sim N(\hat{\theta}_{t-1}^{(k)}, \Sigma_{t-1}^{(k)}). \tag{15}$$

The prediction step then involves two parts: prediction of the model indicator, L_t, via the model prediction equation, and conditional prediction of the parameter, $\theta_t^{(k)}$, given that $L_t = k$, via the parameter prediction equation. We first consider prediction of the model indicator, L_t. Let $\pi_{t-1|t-1,\ell} = P[L_{t-1} = \ell | Y^{t-1}]$. Then the *model prediction equation* is

$$\begin{aligned} \pi_{t|t-1,k} &\equiv P[L_t = k | Y^{t-1}] \\ &= \sum_{\ell=1}^{K} \pi_{t-1|t-1,\ell}\ q_{k\ell}. \end{aligned} \tag{16}$$

To avoid having to explicitly specify the transition matrix, with its K^2 elements, we replace (16) by

$$\pi_{t|t-1,k} = \frac{\pi_{t-1|t-1,k}^{\alpha}}{\sum_{\ell=1}^{K} \pi_{t-1|t-1,\ell}^{\alpha}}, \tag{17}$$

where α is a forgetting factor, which will typically be slightly less than 1. This increases the uncertainty by flattening the distribution of L_t. This is an instance of the multiparameter power steady model introduced independently by Peterka (1981) and Smith (1981), generalizing the one-dimensional steady model of Smith (1979). Although the resulting model does not specify the transition matrix of the Markov chain explicitly, Smith and Miller (1986) argued that this is not a defect of the model, since the data provide information about $\pi_{t-1|t-1,k}$ and $\pi_{t|t-1,k}$, but no additional information about Q.

With forgetting, the *parameter prediction equation* is

$$\theta_t^{(k)}|L_t = k, Y^{t-1} \sim N(\hat{\theta}_{t-1}^{(k}, R_t^{(k)}), \tag{18}$$

where $R_t^{(k)} = \lambda^{-1}\Sigma_{t-1}^{(k)}$.

We now consider the updating step, which again has two parts, model updating and parameter updating. The *model updating equation* is

$$\pi_{t|t,k} = \omega_{tk} / \sum_{\ell=1}^{K} \omega_{t\ell}, \tag{19}$$

where

$$\omega_{t\ell} = \pi_{t|t-1,\ell}\ f_\ell(y_t|Y^{t-1}). \tag{20}$$

In (20), $f_\ell(y_t|Y^{t-1})$ is the density of a $N(x_t^{(\ell)T}\hat{\theta}_{t-1}^{(\ell)}, V^{(\ell)} + x_t^{(\ell)T}R_t^{(\ell)}x_t^{(\ell)})$ distribution, evaluated at y_t.

The *parameter updating equation* is

$$\theta^{(k)}|L_t = k, Y^t \sim N(\hat{\theta}_t^{(k)}, \Sigma_t^{(k)}), \tag{21}$$

where $\hat{\theta}_t^{(k)}$ is given by (7) and $\Sigma_t^{(k)}$ is given by (8), in each case with the superscript (k) added to all quantities. This process is then iterated as each new sample becomes available. It is initialized by setting $\pi_{0|0,\ell} = 1/K$ for $\ell = 1, \ldots, K$, and assigning values to $\theta_0^{(k)}$ and $\Sigma_0^{(k)}$.

The model-averaged one-step-ahead prediction of the system output, y_t, is then

$$\begin{aligned} \hat{y}_t^{\mathrm{DMA}} &= \sum_{k=1}^{K} \pi_{t|t-1,k}\ \hat{y}_t^{(k)} \\ &= \sum_{k=1}^{K} \pi_{t|t-1,k}\ x_t^{(k)T}\ \hat{\theta}_{t-1}^{(k)}. \end{aligned} \tag{22}$$

Thus the multimodel prediction of y_t is a weighted average of the model-specific predictions $\hat{y}_t^{(k)}$, where the weights are equal to the *posterior predictive model probabilities* for sample t, $\pi_{t|t-1,k}$. For the rolling mill problem with delayed prediction, we take the predicted value of y_t to be

$$\hat{y}_t^{\mathrm{DMA}} = \sum_{k=1}^{K} \pi_{t-d|t-d-1,k}\ \hat{y}_t^{(k)}$$

$$= \sum_{k=1}^{K} \pi_{t-d|t-d-1,k}\, x_t^{(k)T}\, \hat{\theta}_{t-d-1}^{(k)}. \tag{23}$$

One could also project the model probabilities into the future and allow for model transitions during the delay period by replacing $\pi_{t-d|t-d-1,k}$ in (23) by $\pi_{t-d|t-d-1,k}^{\alpha^d}$, but we do not do this here.

We call the method *dynamic model averaging* (DMA), by analogy with Bayesian model averaging for the static linear regression model (Raftery, Madigan, and Hoeting 1997).

3.3 Connection to Static Bayesian Model Averaging

Standard Bayesian model averaging (BMA) addresses the static situation where the correct model M_k and its parameter $\theta^{(k)}$ are taken to be fixed but unknown. In that situation, the BMA predictive distribution of y_{T+d} given Y^T is

$$p(y_{T+d}|Y^T) = \sum_{k=1}^{K} p(y_{T+d}|Y^T, M_k) p(M_k|Y^T).$$

where $p(M_k|Y^T)$ is the posterior model probability of M_k. If, as here, all models have equal prior probabilities, this is given by

$$p(M_k|Y^T) = \frac{p(Y^T|M_k)}{\sum_{\ell=1}^{K} p(Y^T|M_\ell)},$$

where $p(Y^T|M_k) = \int p(Y^T|\theta^{(k)}, M_k) p(\theta^{(k)}|M_k) d\theta^{(k)}$ is the integrated likelihood, obtained by integrating the product of the likelihood, $p(Y^T|\theta^{(k)}, M_k)$, and the prior, $p(\theta^{(k)}|M_k)$, over the parameter space. See Hoeting et al. (1999) and Clyde and George (2004) for reviews of BMA.

Dawid (1984) pointed out that the integrated likelihood can also be written as follows:

$$p(Y^T|M_k) = \prod_{t=1}^{T} p(y_t|Y^{t-1}, M_k), \tag{24}$$

with Y^0 defined as the null set. The posterior model probabilities in BMA can be expressed using *Bayes factors* for pairwise comparisons. The Bayes factor for M_k against M_ℓ is defined as the ratio of integrated likelihoods, $B_{k\ell} = p(Y^T|M_k)/p(Y^T|M_\ell)$ (Kass and Raftery 1995). It follows from (24) that the log Bayes factor can be decomposed as

$$\log B_{k\ell} = \sum_{t=1}^{T} \log B_{k\ell,t}, \tag{25}$$

where $B_{k\ell,t} = p(y_t|Y^{t-1}, M_k)/p(y_t|Y^{t-1}, M_\ell)$ is the *sample-specific Bayes factor* for sample t.

In the dynamic setup considered in the rest of the paper, it follows from (17), (19) and (20) that

$$\log\left(\frac{\pi_{T|T,k}}{\pi_{T|T,\ell}}\right) = \sum_{t=1}^{T} \alpha^{T-t} \log B_{k\ell,t}, \tag{26}$$

where $B_{k\ell,t}$ is defined as in (25). Thus our setup leads to the ratio of posterior model probabilities at time T being equal to an exponentially age-weighted sum of sample-specific Bayes factors, which is intuitively appealing. When $\alpha = \lambda = 1$ there is no forgetting and we recover the solution for the static situation, as (25) and (26) are then equivalent.

4 Results

For the rolling mill problem, the four candidate predictor variables of the deviation of the output thickness from its target value for sample t, y_t, are:
u_t: the deviation of the input thickness from its nominal value for sample t,
v_t: the roll position (in microns),
w_t: the ratio of the output rolling speed to the input rolling speed, and
z_t: the rolling force applied to sample t.

In previous work, three models have been used, the first two of which are physically motivated (Ettler, Kárný, and Nedoma 2007). The third model was based on exploratory empirical work. All three have the regression form (1).

The first of these models is based on the gauge meter principle which has been used in this area for several decades (e.g. Grimble 2006). It works with the stretching of the mill housing during rolling. The output thickness satisfies the equation

$$Y_t = v_t + f(z_t),$$

where $f(z)$ is the nonlinear stretch function, depending on the rolling force z. If the stretch function is approximated by a linear function of the rolling force, the gauge meter principle leads to predicting output thickness as a linear function of the roll position and the rolling force.

The second physically-motivated model is based on the mass flow principle, which is commonly used for rolling mill control (e.g. Maxwell 1973). This follows from the continuity of material flow through the mill. It says that the ratio of input thickness to output thickness is equal to the ratio of output rolling speed to input rolling speed. This implies that output thickness can be predicted using the product of the input thickness and the ratio of the speeds. The ratio of the speeds is also included as a regressor in its own right to allow for the fact that a constant has been subtracted from the input thickness and to give some additional flexibility.

Table 1: List of Prediction Models Used. The first three models are the physically motivated ones. The remaining 14 are the empirical models considered. See text for the definitions of the variables.

#	Variables				
	u_t	v_t	w_t	z_t	$(u_t w_t)$
1	-	✓	-	✓	-
2	-	-	✓	-	✓
3	✓	✓	✓	-	-
4	-	-	-	-	-
5	-	-	-	✓	-
6	-	-	✓	-	-
7	-	-	✓	✓	-
8	-	✓	-	-	-
9	-	✓	✓	-	-
10	-	✓	✓	✓	-
11	✓	-	-	-	-
12	✓	-	-	✓	-
13	✓	-	✓	-	-
14	✓	-	✓	✓	-
15	✓	✓	-	-	-
16	✓	✓	-	✓	-
17	✓	✓	✓	✓	-

The predictors to which the three models correspond are as follows:

$$\begin{aligned} M_1: \quad x_t^{(1)} &= (1, v_t, z_t), \\ M_2: \quad x_t^{(2)} &= (1, w_t, u_t w_t), \\ M_3: \quad x_t^{(3)} &= (1, u_t, v_t, w_t). \end{aligned}$$

We considered prediction using each of these models individually, and DMA based on the three models.

The theory underlying the physically-based models does not exclude the possibility of removing any one of the four predictors from the model. To explore what happens when the model space is larger, we therefore considered all possible combinations of the four predictors; this yielded $2^4 = 16$ models. We considered DMA with 17 models: these 16 models, together with model M_2 which also includes the interaction term $(u_t w_t)$. The full list of models is shown in Table 1.

To initialize the process, $\hat{\theta}_0^{(k)}$ and $\Sigma_0^{(k)}$ need to be specified for each model M_k. In general this should be done using external information, including the machine specifications and data

on the rolling mill from other strips that have been run through; such data will generally accumulate rapidly. We did not have such data readily available, and to approximate this we used the data themselves to specify a prior distribution that was spread out relative to the precision available in the data. We specified $\hat{\theta}_0^{(k)} = 0$ for each k, and $\Sigma_0^{(k)} = \mathrm{diag}(s_1^{2(k)}, \ldots, s_{\mu_k}^{2(k)})$. We used $s_j^{2(k)} = \mathrm{Var}(y_t)/\mathrm{Var}(x_{t,j}^{(k)})$ for $j = 2, \ldots, \nu_k$. The rationale for this is that, in linear regression, a regression coefficient for an input variable X is likely to be less than the standard deviation of the system output divided by the standard deviation of X. This is always the case when there is just one input variable, by the Cauchy-Schwarz inequality, and empirical evidence suggests that it holds more generally; see, e.g., Figure 1 in Raftery, Madigan, and Hoeting (1997).

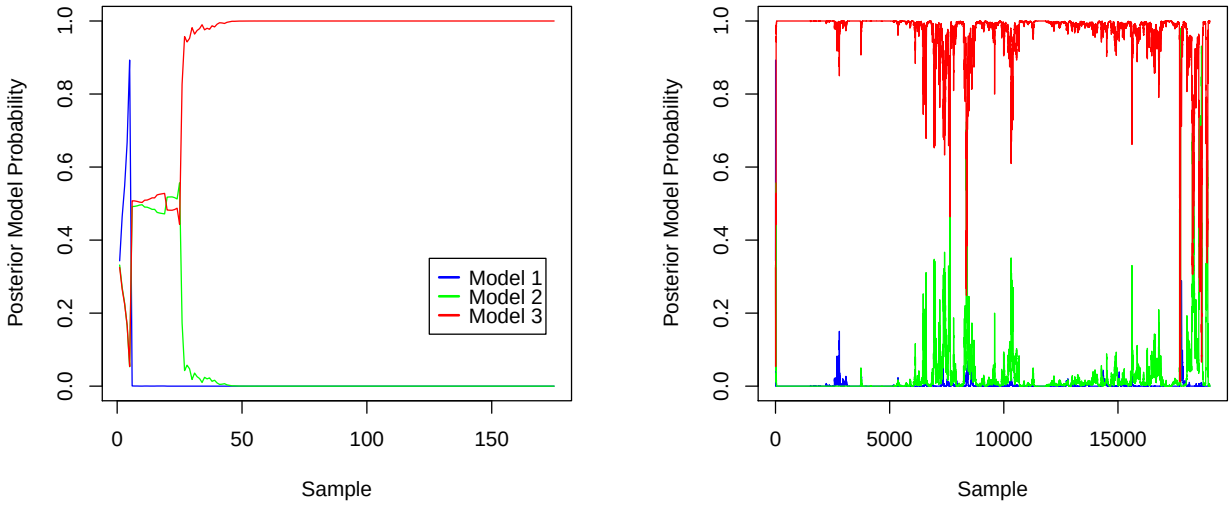
The prior variance of the intercept term, $s_1^{2(k)}$, is trickier. We fitted a static linear regression model to the full dataset and used $s_1^{2(k)} = \hat{\beta}_0^2 + \mathrm{Var}(y_t)$, where $\hat{\beta}_0$ is the estimated intercept. The resulting initial distributions proved to be amply spread out relative to the estimates, and the results were not sensitive to reasonable modifications to them.

We also needed to specify the forgetting factors, λ for the parameters and α for the models. In experiments we found that there was no gain from using different forgetting factors, and so we used $\lambda = \alpha$, leaving just one forgetting factor to be specified. We used $\lambda = \alpha = 0.99$ for all our experiments. We found that our results were relatively insensitive to changes in this value.

The evolution of posterior model probabilities for the three-model case is shown in Figure 4. This is a case where one of the models (M_3) is clearly superior to the other two (M_1 and M_2). DMA quickly picks this up, and from Figure 4(a) we see that the posterior probability of M_3 rapidly grows close to 1 once enough data have been collected to allow a comparison. However, this is not an absorbing state, and the other models occasionally become important, as can be seen from the plot for all 19,058 samples in Figure 4(b). Thus DMA is similar but not identical to prediction based solely on M_3 in this case.

The relative performance of the different methods is shown in Table 2. For all the better methods, the prediction errors had stabilized by around sample 200. A key issue for controlling the rolling mill is how quickly prediction stabilizes after the initial transient situation. Thus we report performance results for the initial period, samples 26–200 (the first 25 samples cannot be predicted using the inputs because of the time delay of 24 samples), and the remaining samples 201–19058. We report the mean squared value of the prediction error, MSE, the maximum absolute prediction error, MaxAE, and the number of samples for which the prediction error was greater than the desired tolerance of 10 microns.

Table 2 shows that M_3 was much better than M_1 or M_2 for both periods, particularly the initial period. For the initial period, DMA with 3 models was slightly better than M_3 on all three criteria we report, while for the stable period DMA was essentially the same as M_3. Thus DMA allows us to avoid paying a penalty for our uncertainty about model structure



(a) Samples 26–200

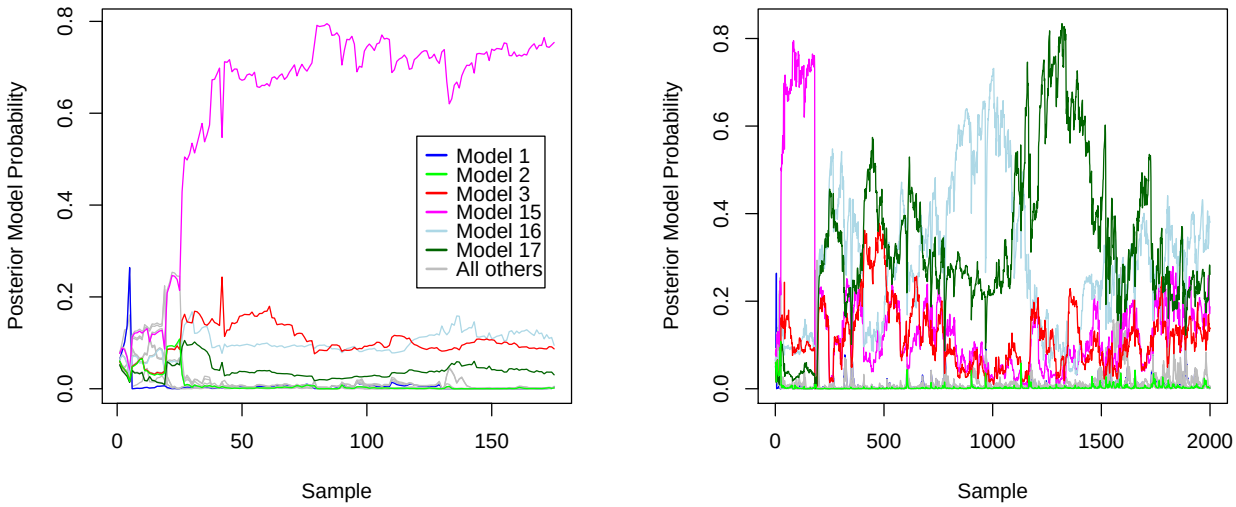
(b) All samples, 26–19058

Figure 4: Posterior Model Probabilities for the Three Initially Considered Models. The correspondence between models and colors is the same in both plots.

Table 2: Sample Statistics of Prediction Errors

Method	Samples 26–200			Samples 201–19058		
	MSE	MaxAE	#AE>10	MSE	MaxAE	#AE>10
Observed	2179.8	68.8	175	30.6	43.1	1183
Model 1	243.6	38.3	86	26.2	31.1	989
Model 2	345.5	41.7	118	26.8	41.4	914
Model 3	77.5	27.3	46	20.7	31.1	523
DMA – 3 models	76.1	26.3	45	20.7	31.1	520
DMA – 17 models	68.9	22.0	42	20.6	31.1	519

NOTE: MaxAE is the maximum absolute error. "#AE>10" is the number of errors above 10 microns in absolute value. The first line of the table refers to the deviations of the observed system output from the target value. Models 1, 2 and 3 are the three initially considered models described in the text.



(a) Samples 1–175

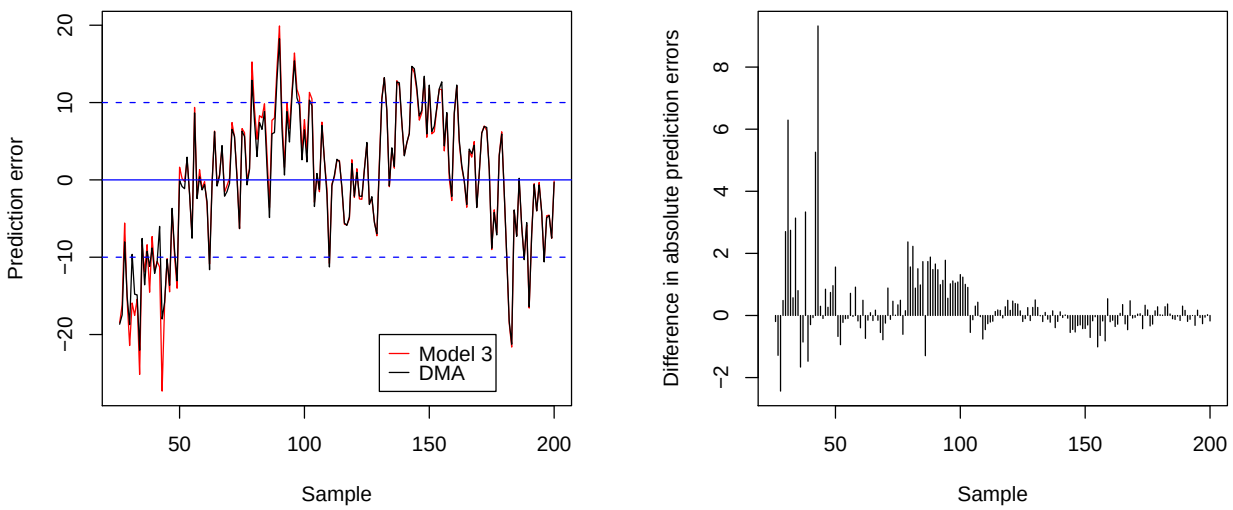
(b) Samples 1–2000

Figure 5: Posterior Model Probabilities for all 17 Models. The legends are the same in both plots.

in this case. Indeed, it even yields slightly more stable predictions in the initial unstable period, perhaps because it allows some weight to be given to the simpler models, M_1 and M_2 at the very beginning, before enough data has accumulated to estimate the more complex M_3 accurately.

We now consider what happens when the space of candidate models is much larger, and 17 models are considered. Figure 5(a) shows the evolution of the posterior model probabilities in the initial unstable period. Only four models (M_3, M_{15}, M_{16} and M_{17}) have more than negligible weight past the first 25 samples or so, and so, even with a large model space, DMA yields a relatively parsimonious solution. In the initial unstable period, the relatively simple model M_{15} had high weight, and in the later stable period, the more complex models M_{16} and M_{17} had more weight, as can be seen from Figure 5(b).

Table 2 shows, strikingly, that DMA with 17 models achieved significantly better performance in the initial unstable period than either M_3 or DMA with 3 models, on all three criteria. This may be because it allows weight to be put on simple, parsimonious models in the early period before stable data has accumulated to estimate more complex models reliably. In the later stable period, DMA with 17 models did slightly better than both DMA with 3 models, and than M_3 on its own. Thus DMA yielded clear gains in the initial unstable period and smaller ones in the later stable period. It allowed us to avoid paying a price for model uncertainty, even when the model space was larger. Overall, including all possible combinations of regressors led to better performance.



(a) Prediction errors for Model 3 and DMA (b) Difference in absolute prediction errors

Figure 6: Comparison of prediction errors for the best of the three initially considered models and for DMA: (a) Prediction errors for Model 3, and for DMA based on all 17 models. (b) Absolute prediction errors for Model 3 minus absolute prediction errors for DMA. Initial period, samples 26–200.

In order to investigate why DMA with 17 models did so well in the initial unstable period, Figure 6(a) shows the prediction errors for M_3 and DMA with 17 models. Figure 6(b) shows the absolute prediction error for DMA minus the absolute prediction error for M_3 (so that positive values correspond to DMA doing better). Up to sample 50, there are large positive values, and in samples 50–100 there are consistent nonnegligible positive values. In samples 100–200 the differences are much smaller. This provides support for our conjecture: in essence, DMA does better in the initial unstable period because it is more adaptive than a single model, even a good one such as M_3, and being adaptive is more important during the initial unstable period than later.

It is important for a real-time application such as this that methods run fast. Our experiments were run in the interpreted statistical language R on a 2005 Apple Powerbook G4 laptop. Computer time scaled roughly linearly with the number of samples times the number of models. DMA took about 2 milliseconds per model per sample. It seems reasonable to expect that running it on newer hardware would speed it up by a factor of at least 4, and that implementing the method in a more efficient, compiled language would speed it up by a factor of 10 or more, suggesting speeds of about 0.05 milliseconds per model per sample with good hardware and software in 2008. If one requires that computations take no more than 20 milliseconds per sample, this suggests that with an efficient implementation, DMA could be used for the rolling mill with up to about 400 models. It is thus well within the

range of practical application. Note that these time constraints preclude the use of methods that are much more computer-intensive than the ones described here.

5 Discussion

We have introduced a new method for real-time prediction of system outputs from inputs in the presence of model uncertainty, called dynamic model averaging (DMA). It combines a state space model for the parameters of the regression models used for regression with a Markov chain describing how the model governing the system switches. The combined model is estimated recursively. In experiments with data from a cold rolling mill with measurement time delay, we found that DMA led to improved performance relative to the best model considered in previous work in the initial unstable period, and that it allowed us to avoid paying a penalty for not knowing the correct model even when model uncertainty was considerable. Including all possible combinations of predictors gave better results than restricting ourselves to a small number of physically motivated models.

The procedure is largely automatic: the only user-specified inputs required are the forgetting factor and the prior mean and variance, which can be based on machine specifications and previous data. It would be possible to estimate the forgetting factor from external data by choosing it so as to optimize performance. It would also be possible to estimate it online so as to optimize predictive performance over past samples, but this would be more expensive computationally than the current method and would effectively preclude its use in the present application. We found that the method's performance was relatively insensitive to reasonable changes in the forgetting factor.

We have applied our method to model spaces of 3 and 17 models. For much bigger model spaces, however, it may not be feasible to run all the models in parallel. Such large model spaces do arise in regression problems, for example with moderate to large numbers of candidate regressors where all possible combinations are considered. It would be possible to modify the method for this kind of situation. One way to do this would be via an "Occam's window" approach (Madigan and Raftery 1994), in which only the current best model and other models whose posterior model probability is not too much less than that of the best model are "active," and are updated. When the posterior probability of a model relative to the best model falls below a threshold, it is removed from the active group. Inactive models are periodically assessed, and if their predictive performance is good enough, they are brought into the active group. Methods along these lines have been proposed in other contexts under the name of *model set adaptation* (Li 2005; Li, Zhao, and Li 2005).

We have used exponential forgetting for the parameters of each model (Fagin 1964; Jazwinsky 1970), as specified by (5). If this is used as the basis for an automatic control procedure, however, there is a risk that Σ_t may become degenerate because the control

process itself can induce high correlations between system inputs and output, and hence high posterior correlations between model parameters, which could lead to singularity or near-singularity of Σ_t. To avoid this, Kulhavý and Zarrop (1993) proposed "Bayesian forgetting," in which a prior distribution is added to the recursion at each iteration. If the prior distribution is Gaussian, this would amount to adding the prior covariance matrix to R_t in (5), thus essentially regularizing the updating process. We applied this in the present context and it made no difference for our data. However, it could be worthwhile for a linearly controlled system.

Various alternatives to the present approach have been proposed. Raftery et al. (2005) proposed a windowed version of Bayesian model averaging, in which the predictive distribution is a mixture with one component per candidate model, and is estimated based on a sliding window of past observations. Ettler, Kárný, and Nedoma (2007) proposed several methods including the predictors-as-regressors (PR) approach, which consists of recursively estimating each candidate model as above, and then running another recursive estimation with the predictors from each model as regressors. Practical aspects of the approach were elaborated in Ettler and Andrýsek (2007).

The DMA model given by (11) and (12) that underlies our work is related to the conditional dynamic linear model (CDLM) (Ackerson and Fu 1970; Harrison and Stevens 1971; Chen and Liu 2000) that has dominated work on adaptive hybrid estimation of systems that evolve according to a Kalman filter model conditionally on an unobserved discrete process. However, it is not a special case of the CDLM, because the form of the state vector $\theta_t^{(k)}$, and not just its value, depends on the model M_k. The CDLM would be given by (11) and (12) with $\theta_t^{(k)}$ replaced by θ_t, where θ_t is the same for all models M_k. It could be argued that DMA could be recast as a CDLM by specifying x_t to be the union of all regressors considered, and θ_t to be the set of regression coefficients for these regressors. In equations (11) and (12), the ν_k-vector $x_t^{(k)}$ would then be replaced by a ν-vector with zeros for the regressors that are not present in M_k.

The difficulty with this is that the state equation (12), now in the form $\theta_t|L_t = k \sim N(\theta_{t-1}, W_t^{(k)})$, would no longer be realistic. The reason is that when the model changes, for example from a model with two correlated regressors to one with just one of the two regressors, then there is likely to be a big jump in θ_t, not a gradual change. To be realistic, the state equation would thus have to involve both L_t and L_{t-1}, which would be unwieldy when the number of models is not small. Our formulation avoids this difficulty.

References

Ackerson, G. A. and K. S. Fu (1970). On state estimation in switching environments. *IEEE Transactions on Automatic Control 15*, 10–17.

Blom, H. A. P. and Y. Bar-Shalom (1988). The interacting multiple model algorithm for systems with Markovian switching coefficients. *IEEE Transactions on Automatic Control 33*, 780–783.

Chen, R. and J. S. Liu (2000). Mixture Kalman filters. *Journal of the Royal Statistical Society, Series B 62*, 493–508.

Clyde, M. and E. I. George (2004). Model uncertainty. *Statistical Science 19*, 81–94.

Dawid, A. P. (1984). Present position and potential developments: Some personal views: Statistical theory: The prequential approach. *Journal of the Royal Statistical Society, Series A 147*, 278–292.

Eddy, S. R. (1998). Profile hidden Markov models. *Bioinformatics 14*, 755–763.

Eddy, S. R. (2004). What is a hidden Markov model? *Nature Biotechnology 22*, 1315–1316.

Eicher, T., C. Papageorgiou, and A. E. Raftery (2007). Determining growth determinants: Default priors and predictive performance in Bayesian model averaging. Working Paper 76, Center for Statistics and the Social Sciences, University of Washington, Seattle.

Ettler, P. and J. Andrýsek (2007). Mixing models to improve gauge prediction for cold rolling mills. In *Preprints of the 12th IFAC Symposium on Automation in Mining, Mineral and Metal Processing*. Québec City: Université Laval.

Ettler, P. and F. Jirovský (1991). Digital controllers for škoda rolling mills. In M. K. K. Warwick and A. Halouskov (Eds.), *Lecture Notes in Control and Information Sciences, vol 158: Advanced Methods in Adaptive Control for Industrial Application*, pp. 31–35. Berlin: Springer-Verlag.

Ettler, P., M. Kárný, and T. V. Guy (2005). Bayes for rolling mills: From parameter estimation to decision support. In *Proceedings of the 16th IFAC World Congress*. Amsterdam: Elsevier.

Ettler, P., M. Kárný, and P. Nedoma (2007). Model mixing for long-term extrapolation. In *Proceedings of the 6th EUROSIM Congress on Modelling and Simulation*. Ljubljana: EUROSIM.

Evensen, G. (1994). Sequential data assimilation with nonlinear quasi-geostrophic model using Monte Carlo methods to forecast error statistics. *Journal of Geophysical Research 99*, 143–162.

Fagin, S. L. (1964). Recursive linear regression theory, optimal filter theory, and error analyses of optimal systems. *IEEE International Convention Record Part i*, 216–240.

Fernández, C., E. Ley, and M. F. J. Steel (2001). Benchmark priors for Bayesian model averaging. *Journal of Econometrics 100*, 381–427.

Grimble, M. J. (2006). *Robust Industrial Control Systems: Optimal Design Approach for Polynomial Systems*. John Wiley & Sons.

Hamilton, J. D. (1989). A new approach to the economic analysis of nonstationary time-series and the business cycle. *Econometrika*, 357–384.

Hannan, E. J., A. J. McDougall, and D. S. Poskitt (1989). Recursive estimation of autoregressions. *Journal of the Royal Statistical Society, Series B 51*, 217–233.

Harrison, P. J. and C. F. Stevens (1971). Bayesian approach to short-term forecasting. *Operational Research Quarterly 22*, 341–362.

Harrison, P. J. and C. F. Stevens (1976). Bayesian forecasting. *Journal of the Royal Statistical Society, Series B 38*, 205–247.

Hoeting, J. A., D. Madigan, A. E. Raftery, and C. T. Volinsky (1999). Bayesian model averaging: A tutorial (with discussion). *Statistical Science 14*, 382–417.

Jazwinsky, A. W. (1970). *Stochastic Processes and Filtering Theory*. New York: Academic Press.

Kass, R. E. and A. E. Raftery (1995). Bayes factors. *Journal of the American Statistical Association 90*, 773–795.

Kulhavý, R. and M. B. Zarrop (1993). On a general concept of forgetting. *International Journal of Control 58*, 905–924.

Leamer, E. E. (1978). *Specification Searches: Ad Hoc Inference With Nonexperimental Data*. New York: Wiley.

Li, X. R. (2005). Multiple-model estimation with variable structure - Part II: Model-set adaptation. *IEEE Transactions on Automatic Control 45*, 2047–2060.

Li, X. R. and V. P. Jilkov (2005). Survey of maneuvering target tracking. Part V: Multiple-model methods. *IEEE Transactions on Aerospace and Electronic Systems 41*, 1255–1321.

Li, X. R., Z. L. Zhao, and X. B. Li (2005). General model-set design methods for multiple-model approach. *IEEE Transactions on Automatic Control 50*, 1260–1276.

Ljung, L. (1987). *System identification: Theory for the User*. Englewood Cliffs, New Jersey: Prentice-Hall.

Madigan, D. and A. E. Raftery (1994). Model selection and accounting for model uncertainty in graphical models using Occam's window. *Journal of the American Statistical Association 89*, 1535–1546.

Maxwell, H. S. (1973). *Patent number 3762194: Constant Speed Driven Continuous Rolling Mill*. Assignee: General Electric Company.

Mazor, E., A. Averbuch, Y. Bar-Shalom, and J. Dayan (1998). Interacting multiple model methods in target tracking: A survey. *IEEE Transactions on Aerospace and Electronic Systems 34*, 103–123.

Peterka, V. (1981). Bayesian system identification. In P. Eykhoff (Ed.), *Trends and Progress in System Identification*, pp. 239–304. Oxford: Pergamon Press.

Quinn, A., P. Ettler, L. Jirsa, I. Nagy, and P. Nedoma (2003). Probabilistic advisory systems for data-intensive applications. *International Journal of Adaptive Control and Signal Processing 17*, 133–148.

Rabiner, L. R. (1989). A tutorial on hidden Markov models and selected applications in speech recognition. *Proceedings of the IEEE 77*, 257–286.

Raftery, A. E. (1988). Approximate Bayes factors for generalized linear models. Technical Report 121, Department of Statistics, University of Washington, Seattle.

Raftery, A. E., T. Gneiting, F. Balabdaoui, and M. Polakowski (2005). Using Bayesian model averaging to calibrate forecast ensembles. *Monthly Weather Review 133*, 1155–1174.

Raftery, A. E., D. Madigan, and J. A. Hoeting (1997). Model selection and accounting for model uncertainty in linear regression models. *Journal of the American Statistical Association 92*, 179–191.

Smith, A. F. M. and M. West (1983). Monitoring renal transplants: An application of the multiprocess Kalman filter. *Biometrics 39*, 867–878.

Smith, J. Q. (1979). Generalization of the Bayesian steady forecasting model. *Journal of the Royal Statistical Society, Series B 41*, 375–387.

Smith, J. Q. (1981). The multiparameter steady model. *Journal of the Royal Statistical Society, Series B 43*, 256–260.

Smith, R. L. and J. E. Miller (1986). A non-Gaussian state space model and application to prediction of records. *Journal of the Royal Statistical Society, Series B 48*, 79–88.

West, M. and P. J. Harrison (1989). *Bayesian forecasting and dynamic models*. New York: Springer-Verlag.