Dynamic Modelling and Control of Industrial Processes
with Particle Filtering Algorithms

Rubén Morales-Menéndez*, Nando de Freitas, David Poole and Jim Mutch**
Computer Science Department
University of British Columbia, Vancouver, BC, Canada V6T 1Z4

Federico Guedea-Elizalde***
Systems Design Engineering
University of Waterloo, Waterloo, ON, Canada N2L 3G1

Abstract
We use a probabilistic approach to estimate the operating conditions and guide an automatic control system for industrial processes. The jump Markov linear Gaussian (JMLG) model is adopted to describe process behavior as a dynamic mixture of linear models. Based on the JMLG model, we use Particle Filtering (PF) algorithms to make real-time estimates of the operating conditions of the process. The PF estimate is used to adapt an automatic feedback control system. We tested our approach against three standard control strategies using a real nonlinear process. The results indicate that implementation of a PF state estimator can lead to better control strategies.

Keywords: Particle Filtering, State Estimation, Jump Markov Linear Gaussian

1 Introduction
Early estimation of changes in operating conditions, disturbances, or faults in automatic control systems is valuable. State estimation allows us to take sooner so as to improve performance or to avoid worst-case conditions.

We want to estimate the states of industrial processes by measuring continuous variables over time. Online state estimation in dynamic systems is a NP combinatorial optimization problem (Doucet and Andrieu 2001). Particle Filtering (PF) methods have been successfully proposed as a practical solution (Doucet, Gordon and Krishnamurthy 2001, Morales-Menéndez, de Freitas and Poole 2002, Hutter and Dearden 2003).

This paper is organized as follows. Section 2 presents the fundamentals of the dynamic model and PF algorithms. Section 3 describes our nonlinear industrial process, the identification procedure, the state estimation, and the implemented control systems. Section 4 discusses the results, and section 5 concludes the paper.

2 Fundamentals
The behavior of nonlinear processes can often be represented by a set of linear Gaussian models; therefore, we will use this idea to model our industrial process. Based on this model, Particle Filtering algorithms will be used for state estimation.

* rmm@itesm.mx, ITESM, Center for Industrial Automation, México
** {nando,poole,mutch}@cs.ubc.ca
*** fguedea@pami.uwaterloo.ca
2.1 Dynamic model

Hidden Markov Models (HMMs) and State Space Models (SSMs) are well-known frameworks. Combining these structures, we can generate a hybrid model which has sufficient parameters to represent technical processes. This model is called Jump Markov Linear Gaussian (JMLG). Formally, we can represent the JMLG model as

\[ z_t \sim p(z_t | z_{t-1}) \]  
\[ x_{t+1} = A(z_{t+1})x_t + B(z_{t+1})\gamma_{t+1} + F(z_{t+1})u_{t+1} \]  
\[ y_t = C(z_t)x_t + D(z_t)v_t \]

where \( z_t \in \{1, \ldots, n_z\} \) denotes the possible discrete modes the system can be in, \( x_t \in \mathbb{R}^{n_x} \) denotes the continuous states, and \( y_t \in \mathbb{R}^{n_y} \) are the continuous observable variables. \( \gamma_t \in \mathbb{R}^{n_\gamma} \) and \( v_t \in \mathbb{R}^{n_v} \) are exogenous inputs for the process and measurement noise; they are i.i.d. Gaussian with covariance \( \mathcal{Q} \) and \( \mathcal{R} \). \( p(z_t | z_{t-1}) \) is a probabilistic transition function over the discrete modes. The initial states are \( z_0 \sim p(z_0) \) and \( z_0 \sim \mathcal{N}(\mu_0, \Sigma_0) \). The parameters \( (A(\cdot), B(\cdot), C(\cdot), D(\cdot), F(\cdot)) \) are matrices with \( D(\cdot)D(\cdot)' > 0 \) for any \( z_t \). The JMLG model has the properties of observability and controllability, which are important for this application.

2.2 Particle filtering methods

The goal in the Bayesian inference task is to compute the distribution \( p(z_{1:t} | y_{1:t}) \) given the observations and the JMLG model. This distribution is derivable from the full posterior distribution \( p(x_{1:t}, z_{1:t} | y_{1:t}, u_{1:t}) \) using standard marginalization. This cannot be done analytically, thus numerical approximation methods such as Particle Filtering must be used. PF (Doucet, de Freitas and Gordon 2001) is a powerful tool for Bayesian state estimation in nonlinear systems. One can approximate a posterior probability distribution over unknown state variables by a set of particles drawn from this distribution.

Particle Filtering (PF). PF uses a set of weighted samples to approximate the posterior probability distribution. Given \( N \) particles \( \{x_{0:t-1}^{(i)}, z_{0:t-1}^{(i)}\}_{i=1}^N \) at time \( t-1 \), approximately distributed according to \( p(x_{0:t-1}^{(i)}, z_{0:t-1}^{(i)} | y_{1:t-1}) \), PF allows us to sequentially compute \( N \) particles \( \{x_{0:t}^{(i)}, z_{0:t}^{(i)}\}_{i=1}^N \) approximately distributed according to \( p(x_{0:t}^{(i)}, z_{0:t}^{(i)} | y_{1:t}) \) at time \( t \).

Since we cannot sample from the posterior directly, the PF update is accomplished by introducing an appropriate importance proposal distribution \( q(x_{0:t}, z_{0:t}) \) from which we can obtain particles. The choice of the transition priors as proposal makes efficient sequential importance sampling possible. The basic algorithm consists of two steps, sequential importance sampling and selection.

Rao-Blackwellized Particle Filtering (RBPF). Greater computational efficiency can be achieved considering that the Kalman Filter part of the model is analytically solvable (Doucet, de Freitas, Murphy and Russell 2000). The Rao-Blackwell equation allows the factorization \( p(x_{0:t}, z_{0:t} | y_{1:t}) = p(x_{0:t} | y_{1:t}, z_{0:t})p(z_{0:t} | y_{1:t}) \). We can visualize the inference task in two steps. The density \( p(x_{0:t} | y_{1:t}, z_{0:t}) \) is Gaussian and can be computed analytically if we know the marginal posterior density \( p(z_{0:t} | y_{1:t}) \). The importance weights for \( z_t \) are given by the predictive density \( p(y_t | y_{1:t-1}, z_{1:t}) \).

\(^1\) For clarity, we omit the control signal \( u_t \) from the arguments of the probability distributions.
Look-ahead RBPF. While evaluating the importance weights for particles at time $t$, \textit{la-RBPF} (Morales-Menéndez et al. 2002) looks ahead one step to see which way the measurements are going at time $t + 1$. It then uses this information to compute better weights at time $t$. The optimal importance weights in \textit{look-ahead RBPF} are given by $\sum_{z_t=1}^{n_z} \mathbb{P}(z_t|z_{t-1}, y_{1:t-1}, z_{0:t-1}, z_t) \mathbb{P}(z_t|z_{t-1})$. This works because the weights do not depend on $z_t$. The basic sequential steps are Kalman prediction, Selection, Sequential Importance Sampling, and Kalman updating.

3 Application

3.1 Nonlinear level process

The level tank is a widely studied system; it can represent the interesting dynamic behaviors of many industrial processes, e.g. boiler drums. Fig. 1 shows our experimental process\(^2\). The instruments have standard communication with a Honeywell UDC 6300 Controller (not shown).

Manual bypass valves, $V_1$ and $V_2$, are used to physically implement different behaviors. Five discrete operational modes were defined.

![Level tank process and instrumentation diagram.](image)

3.2 Identification.

Parametric identification is carried out based on the Least Squares Estimation (LSE) algorithm. Experimental step changes were applied by altering the input signal. The transient step response follows a first order plus dead time (FOPDT) model. Using the FOPDT model, we obtain the deterministic discrete state space representation. Next we determine the process and measurement noise matrices and refine the deterministic matrices using the \textit{Expectation-Maximization} (EM) algorithm (Ghahramani and Hinton 1996)\(^3\). We repeated this procedure for each discrete mode. The set of equations and the engineered transition matrix $p(z_t|z_{t-1})$ complete the JMLG model. The left graph in

\(^2\) Standard ISA nomenclature has been used to identify the instrumentation.

\(^3\) We thank Zoubin Ghahramani for providing the EM algorithm code.
3.3 State estimation

We tested the three PF algorithms with the level-tank by physically implementing a sequence of discrete modes and recording the observations (Morales-Menéndez et al. 2002). At each time step, we use the Maximum A Posteriori (MAP) to identify the most probable discrete mode. Diagnosis error is defined as the percentage of time steps during which the discrete mode is not estimated properly. We performed 25 independent runs of 1,000 time-steps each. The right graph in Fig. 2 shows the results for different numbers of particles. **Look-ahead RBPF** gave the best performance.

3.4 Control system

We want to use PF for state estimation in real-world domains. The PF algorithm will estimate the most likely discrete mode. Then, given this information, the PID controller will select the appropriate tuning parameters for each behavior. First, for comparison purposes, we implemented three standard control strategies that cope with nonlinear processes without estimating discrete modes. Then we implemented a fourth strategy incorporating la-RBPF for state estimation. Each control system should keep the process variable (PV) close to the set point (SP).

**Feedback control system.** The PID controller was tuned for a single discrete mode with the most conservative tuning parameters. The left graphs in Fig. 3 show the feedback control system’s performance. The upper plot shows the discrete mode over time, the middle plot shows the SP and the PV, and the lower plot shows the output controller (OP).

**Feedforward control.** Feedback systems must wait until the error appears before taking the appropriate action. Feedforward control systems compensate for the error before it appears. They must measure an indicator of the possible problem, calculate the needed manipulation, and act before the error appears. We implemented this strategy by measuring the input flow (FT-101). See the results in the middle graphs of Fig. 3.

**Cascade control.** This technique is used to improve the dynamic response of a feedback control system having changing conditions mainly in the manipulated variable.
We implemented a master PID controller for the level-tank and a slave PID controller for the input flow. The right graphs in Fig. 3 show the performance.

**Feedback control system plus la-RBPF algorithm.** Finally, we augmented the feedback system with la-RBPF to first estimate the most likely discrete mode. This allows the controller to pick the appropriate set of tuning parameters for each behavior. Fig. 4 shows the control system and results.

**4 Results**

**Modelling and Estimation.** The left graphs of Fig. 2 illustrate the nonlinearity of our process and its successful representation by the JMLG model. The right graphs show the very low diagnosis error of PF algorithms as the number of particles grows. Look-ahead RBPF demonstrates the best performance with very low mean and variance.

**Control System Application.** PID is a robust controller but performs best with linear systems. The left graph in Fig. 3 shows the feedback system performance. In some discrete modes the PV approaches the SV quickly. However, in other discrete modes, the process has a long settling time. The dynamic behavior for discrete modes $z_t = 1$
and $z_t = 4$ is very different. The SSE\(^4\) is 2.98. The feedforward strategy gives much better performance (middle plots in Fig. 3) with SSE = 0.473; however, it comes at the cost of an additional sensor/transmitter. Note the poor performance for some discrete modes. Cascade control (right plots in Fig. 3) yields excellent results (SSE = 1.32) when the right tuning parameters are used; however, the slave control system becomes unstable if the dynamic behavior changes considerably. Also, cascade control demands an additional sensor/transmitter and an additional PID controller.

Our proposal combining \textit{la-RBPF} with feedback control (Fig. 4) gives excellent results (SSE = 1.11). While this is not as low as the feedforward strategy for this relatively simple process, it is less expensive and more stable. In more complex systems having more variables, the benefits of combining PF techniques with feedback control systems should become even more apparent. Feedforward and other standard control strategies will break down when many variables change at once, but our combined approach is immune to this effect. This will be explored in a future paper.

**Conclusions**

Particle Filtering algorithms can be implemented for state estimation in dynamic systems. The experiments showed that combining feedback control systems and PF techniques can result in better performance. This creates new possibilities for sophisticated control strategies. One opportunity for improvement would be a method for automatically updating the parameters of the JMLG model.

**Acknowledgments.** Morales-Menéndez was partially supported by the Government of Canada (ICCS) and UBC’s CS department. Poole and de Freitas are supported by NSERC.

**References**


\(^4\) SSE = $\frac{1}{N} \sum_{t=1}^{N} e^2$, where the error $e = SP - PV$. 