USING PATTERN RECOGNITION IN CONTROLLER
ADAPTATION AND PERFORMANCE EVALUATION

Ralph F. Hinde, Jr. and Douglas J. Cooper†
Department of Chemical Engineering
University of Connecticut, U-222
Storrs, CT 06269-3222

ABSTRACT
This work presents pattern recognition-based methods for
controller adaptation and performance evaluation. These
methods comprise a passive model-based adaptive control
algorithm that is simple to use, easy to understand, stable, and
fairly robust in a wide variety of applications. Controller
adaptation in this work uses excitation diagnostics to initiate
batch-wise regression of a process model to dynamic closed-loop
process data. The process model is then employed in model-
based controller tuning relations to update the controller's
character. Controller performance evaluation is used to
determine appropriate adjustments to the tuning relations such
that an accurate process model will produce desired controller
performance. These adaptive techniques are implemented using
vector quantizing neural networks as efficient pattern recognition
tools.

The adaptive algorithm is presented in a structure that
allows for the implementation of these advanced techniques
without requiring the replacement of an existing feedback
controller. This is demonstrated using a simulated nonlinear third
order process and an IMC tuned PI controller with Smith
Predictor.

INTRODUCTION
The manner in which a controller is performing may be
determined by observing the recent history of either the
controller error or the process output after a sustained set point
step change. Patterns in these histories reveal the degree of effort
the controller is exerting. If a controller is too aggressive, the set
point response pattern will be marked by an excessive overshoot
and slow damping. Similarly, a sluggish controller will produce a
response pattern that is overdamped with no overshoot.

A pattern-based approach to performance feedback for
controller adaptation was first proposed by Bristol [1]. This
unique approach determines appropriate updates to the
parameters of a PID controller based on measures of the
controller performance. The measures used are features of the
set point response pattern such as overshoot and damping ratio.
The difference between observed features and operator specified
desired features is used to tune a PID controller in a manner
reminiscent of Ziegler and Nichols [2]. Other researchers have
explored the use of feature-based pattern recognition in adaptive
process control including the development of a commercial
product, the Foxboro EXACT [3-6].

Whereas these methods have been proven to be
successful, they have two limitations. The first limitation is that
the algorithms are designed to update only PID controllers. An
algorithm that can adapt several different types of feedback
controllers would be more widely applicable. The second
limitation is that robust feature extraction requires a cumbersome
rule base.

Extracting features from the set point response pattern
involves locating the local maxima and minima within that
response pattern. This becomes very difficult as process
nonidealities arise. Nonidealities such as significant measurement
noise and nonstationarity mask the location of the maxima and
minima, corrupting the estimates of the features. The rule base
grows and becomes cumbersome as the effects of these
nonidealities are identified and accounted for [7,8]. A simpler and
more efficient method of pattern recognition would prove to be
beneficial.

Recently, artificial neural networks (ANNs) have been
found to be efficient tools for pattern recognition and
classification [9-11]. Seeing the potential for such tools,
researchers have employed ANNs in pattern-based adaptive
process control strategies. These include controller performance
feedback driven adaptive pole placement and a pattern based
approach to open-loop PID controller tuning [12,13].

The authors have explored the use of ANNs for
performance feedback model-based adaptive control. By adapting
a process model rather than a specific controller, the algorithm
is widely applicable to any one of a number of model-based
controllers. Through this work, ANNs have been found to be less
sensitive to nonidealities than feature-based pattern recognition
methods. This is due to the manner in which ANNs consider
the entire pattern and not just features of that pattern [8,14].

Unfortunately, performance feedback model-based
adaptation has been found to contain a drawback when
performed alone. Model accuracy is often sacrificed when a
process model is updated based solely on the controller
performance. For example, if the value of the IMC closed loop
time constant, $r_c$, should no longer provide desired controller
performance given an accurate process model, adaptation will
adjust one or more model parameters to account for the error in
$r_c$. The accuracy of other model-based algorithms, such as Smith
Prediction of the controller error, is then compromised. This may
lead to a reduction of controller performance rather than an
improvement.

What is required is an adaptive algorithm that keeps an
accurate model for wide applicability and insures that the model
produces desired performance. This work fulfills that requirement
by extending the use of ANN-based pattern recognition for
controller adaptation and performance evaluation. Two ANNs
are used in a unique approach to excitation diagnostics for
closed-loop passive process identification and model-based
controller design. A third ANN is used in performance feedback
diagnostics for adaptation of model-based controller tuning
relations. Both methods employ the ART2-A vector quantizing
neural network (VQN) architecture of Carpenter et al. for the
pattern recognition task [15]. These adaptive strategies are
presented in a framework that allows for their application to an
existing feedback controller without calling for the replacement
of that controller.

This new adaptive algorithm is shown in Figure 1 being
implemented with an existing feedback controller. When on-line,
the adaptive algorithm first uses a performance diagnostics VQN to perform pattern recognition on set point step change response patterns to determine if adaptation is necessary. If the observed performance is not desired, the excitation diagnostic VQNs monitor process data and determine when sufficient dynamics exist for regression of a process model [16]. Once model parameters have converged, a second model-based controller is designed. The second controller is then put on-line in a manner which overrides the effort of the existing controller without replacing it. Excitation diagnostics are then continually performed leading to further updates of the process model. Likewise, performance diagnostics continue to monitor controller performance and adjust the controller design algorithms so that accurate model parameters will produce desired controller performance.

A simulated third order nonlinear process is employed in a demonstration of this work. IMC tuned PI controllers with Smith Predictor are used as both the existing and second controllers.

### Adaptive PI Control

The velocity form of the PI algorithm computes the incremental control action, \( a(t) \), at sample number \( t \) as:

\[
\Delta u(t) = K_c [e(t) - e(t-1) + e(t)at/r_1] \tag{1}
\]

where \( K_c \) is the controller gain and \( r_1 \) is the reset time [17,18]. Here, a Smith predictor calculates the controller error, \( e(t) \), as the difference between the set point, \( e_{set} \), and the value of the process output, \( y(t) \), predicted one dead time into the future [19].

A first order plus dead time (FOPDT) model is used for the prediction. The difference form of the FOPDT model is:

\[
y'(t) = y'(t-1) + \exp(-at/r_p)ay'(t-1) \\
    + K_p (1 - \exp(-at/r_p))a(t-k) \tag{2}
\]

where \( y'(t) \) is the predicted process output, \( K_p \) is the steady state gain, \( r_p \) is the dominant time constant and \( k \) is one plus the integer number of sample times, \( \Delta t \), in one dead time, \( r_d \). The prediction is obtained by performing \( k \) iterations on Equation (2).

The PI tuning relations employed are derived from the internal model control (IMC) architecture [20]. Having compensated for the dead time, the process gain and time constant from the FOPDT model are combined with a first order filter to obtain the IMC tuning relations:

\[
K_c = \frac{r_p}{r_cK_p} \quad r_1 = r_p \tag{3}
\]

where \( r_c \) is the one IMC tuning knob, is the desired closed loop time constant. This algorithm is made adaptive by updating \( K_c \), \( \tau_p \), \( \tau_d \) and \( r_c \) in Equations (2) and (3).

To implement this model based controller, \( K_p \), \( r_p \) and \( \tau_d \) must be specified at start-up. These parameters are typically based on the regression of data collected from a single open loop step test made in the start-up operating regime. The sample time, \( \Delta t \), is set equal to 0.04 \( r_p \). The parameter \( r_c \) is set to a value that produces desired controller performance which in this work is a quick response with very little overshoot.

The closed loop time constant is defined in this work as:

\[
\tau_c = \gamma \tau_p \tag{4}
\]

where \( \gamma \) is initially equal to 0.50 and is later adjusted through performance feedback. The closed loop time constant is also allowed to change as adaptation changes the value of \( r_p \).

### Pattern Recognition

Pattern-based adaptation in this work involves the extension of previous pattern recognition methods to performance feedback diagnostics and process excitation diagnostics [8,16]. ART2-A VQNs are employed for the pattern recognition task [15]. What follows is a summary of VQN pattern recognition and the ART2-A architecture. A further discussion of the ART2-A VQN as applied to this work may be found in [16].

VQNs take incoming patterns and assign them to specific discrete classes. As shown in Figure 2, this is done by comparing the incoming pattern to a library of exemplar patterns where each exemplar pattern represents a specific pattern class. A matching score between each exemplar pattern and the incoming pattern is developed in the comparison. Matching scores give a measure of how closely the incoming pattern resembles their associated exemplar patterns. The exemplar pattern with the largest matching score is the one that is most similar to the incoming pattern.

For the ART2-A VQN, classification does not end with identification of the exemplar pattern with the largest matching score. If the incoming pattern is unlike any exemplar pattern in the library, the largest matching score will be low representing a weak match. Therefore, a vigilance test is used to make sure that the best match is a good match. The vigilance test simply checks to see if the largest matching score is above a minimum value called the vigilance parameter. If it is, then the vigilance test is passed and the incoming pattern is recognized as belonging to the class represented by that exemplar pattern.

Before the VQN can recognize and classify incoming patterns, the library of exemplar patterns must be developed through training. Training involves introducing the VQN to numerous patterns that cover the breadth and depth of the patterns the network needs to classify. The VQN forms its exemplar patterns by clustering similar training patterns together.

In training, the first pattern introduced to the network becomes the first exemplar pattern. The second training pattern is compared to the existing exemplar pattern. If the vigilance test is passed, the two patterns are clustered to form a new first exemplar pattern. If the vigilance test is failed, then the second training pattern becomes the second exemplar pattern. Training continues in this manner until no new exemplar patterns are formed or until room for new exemplar patterns is no longer available. Obviously, the value of the vigilance parameter is key to exemplar pattern formation.

### Performance Feedback Diagnostics

Performance feedback diagnostics serve two purposes in the adaptive algorithm. They first decide whether or not controller performance is desired. They then determine an appropriate action that will maintain or obtain desired performance. This is accomplished by performing VQN pattern recognition on the recent history of the controller error after sustained set point step changes.

The VQN first classifies the incoming set point response pattern as belonging to a class displaying a certain degree of controller performance. Each controller performance class has associated with it a corrective action that will restore or retain desired performance. In past work, performance-based corrective actions were made to the process model [8]. Again, when an adaptive or corrective action is made to the process model, model accuracy is often compromised in an effort to improve performance. Therefore, an alternative corrective action is necessary.
Model-based control algorithms such as IMC and DMC typically have one or more adjustable parameters that the practitioner may use to obtain desired controller performance given an accurate process model. In the IMC PI tuning relations, this parameter is the closed loop time constant. In DMC, the parameter is the input suppression factor [21]. This work employs an IMC tuned PI controller with Smith Predictor so corrective action is taken on the closed loop time constant, $r_c$.

The performance feedback diagnostic VQN is, therefore, trained on set point response patterns that have been developed by adjusting $\gamma$ used in the determination of $r_c$ in Equation (4) from its correct value. An IMC tuned PI controller with Smith Predictor is implemented on a simulated linear second order process. The value of $\gamma$ is then mismatched from that value that produces desired controller performance (0.5 in this work). A set point step change is made and the resulting response pattern and associated $\gamma$ mismatch are sent to the VQN for training.

During training, the VQN not only clusters similar training patterns, but it clusters their associated $\gamma$ mismatches as well. In this way, when an incoming response pattern is classified as being similar to a certain exemplar pattern, the $\gamma$ mismatch associated with that exemplar pattern is known. In order to restore desired controller performance, the correct adaptive action to take is to make the inverse mismatch to the present value of $\gamma$.

In this work, $\gamma$ mismatches of one third to three times 0.5 are used to train the network. In order to achieve a library of manageable size, the vigilance parameter is set to 0.9998 which results in 80 exemplar patterns. Training patterns consist of 50 samples where controller sample time is equal to 0.107$r_c$. When performance feedback diagnostics are on-line, response pattern collection begins whenever there is a set point change. As mentioned previously, the controller sample time for implementation is 0.04$r_c$. Response patterns must then be collected in a manner such that 50 samples cover the time span of five times the present estimate of the dominant process time constant. If the set point changes during pattern collection then collection is aborted until the next set point change. This avoids collecting patterns from non-sustained set point step changes that the network has not been trained to recognize.

**Process Excitation Diagnostics**

With performance feedback diagnostics insuring that an accurate process model provides desired performance, what remains to be determined is a way to maintain process model accuracy. The most reliable approach is to periodically regress a model to closed-loop process data. However, successful closed-loop identification requires that the process data be dynamic and that information contained in the process data not be masked by measurement noise [22].

Pattern-based approaches to determining a controller's behavior have been extended to determining process behavior. This has lead to the development of a process excitation diagnostic methodology [16]. Rather than classify long term response patterns interminently as in performance feedback diagnostics, excitation diagnostics employ two VQNs working in tandem to continually observe short term (one time constant's duration) patterns in both process variables. The objective is to find many short term dynamic trends in both process variables to signify that a global process dynamic event is occurring that may be used in process identification.

Each VQN is trained to recognize generic dynamic patterns in its respective process variable. The training patterns are generated by randomly adjusting the set point of the controller of choice (IMC tuned PI control with Smith Predictor in this work) implemented on a simulated second order process. Using a vigilance parameter of 0.9200 results in 400 exemplar patterns in the process input VQN and 100 exemplar patterns in the process output VQN. Two separate VQNs are required because process input history patterns reflect the nature of the controller whereas process output history patterns reflect the nature of the process. The need for more exemplar patterns in the process input VQN results from the fact that the process input is freely manipulated by the controller while the process output responds according to the slower dynamics of the process.

Unlike performance diagnostics, excitation diagnostics do not require the identification of the exemplar pattern that is most like the incoming history pattern. Instead, only the result of the vigilance test needs to be known. Through past work employing VQNs as pattern recognition tools the authors have noticed that the matching score not only provides a measure of resemblance, but may also be used to indirectly determine the signal-to-noise ratio of the data within the incoming history pattern [8,16]. The larger the noise component in the data, the lower the matching score will be. Therefore, if the vigilance test is passed, then the process variable history pattern is declared to be both dynamic and relatively free from noise. The results of the vigilance tests from both VQNs are sent to a decision maker.

The decision maker is a simple rule base system that receives dynamic classifications from the VQNs and determines both when the process model should be updated and which data should be used. The decision maker looks for relatively simultaneous multiple dynamic trend classifications from both VQNs. This is done by keeping a running sum of the results of the vigilance tests from each VQN. A passed test is assigned +1 and a failed test is assigned -1. The minimum sum is zero.

A process variable dynamic state is defined when that variable's running sum is incremented up to a trigger value. This trigger value is set equal to five which represents one dominant time constant's duration of dynamics since the VQNs are activated every five samples. A dynamic flag for that process variable is then set to one and its running sum is reset to zero.

When both process variable dynamic flags are equal to one within one dominant time constant's duration of either other, sufficient dynamics for process model update have been found and the modeling algorithm is activated. In this work, the modeling algorithm is a batchwise regression minimizing a sum of squared errors.

Data for the model regression are collected from the present back to the last steady state. A steady state is declared when either both process variable running sums are simultaneously equal to zero for a trigger value's duration or when one running sum is equal to zero for one estimated response time as long as the process input is not saturated. To help insure proper characterization and converged process model parameters, an additional model regression is performed just prior to the first steady state after a dynamic event.

Using these rules, a set point response pattern, for example, may be modeled three or four times during its duration. Each modeling instance will contain more information than the last until a new steady state is found.

Once the dynamic data have been modeled, the new model parameters must be tested for accuracy and convergence before they are implemented. Spurious results from the modeling algorithm may produce poor control. Such is often the case at the first modeling instance of a dynamic event where there is minimal information available in the data. Poor control may also result from model parameters produced by data that are dominated by the dynamics of an unmeasured process disturbance.

The first step in determining the validity of the data is to consider the new estimate of the process gain. Vogel and Edgar note that the value of the new estimate of $K_p$ provides an indication of the validity of the new model parameters [23]. If the
new estimate of $K_p$ does not fall within a reasonable range of the previous estimate, than the new model is assumed false or corrupted and is discarded. This is first determined by checking if the sign of the gain has changed. If the new estimate has the opposite sign of the old previously implemented estimate then the model is determined to be invalid. Next, if the new estimate of $K_p$ is approximately 20 times larger or smaller than the present estimate then the new estimate is considered invalid and the new model is rejected.

If the new model is not rejected, then it is valid and each parameter is checked for convergence. Convergence is determined separately as each new model parameter estimate is compared to its associated present valid estimate. Each model parameter whose estimate is determined to be converged is then implemented in the internal controller model.

In order to avoid over-adapting the overall algorithm, adjustments to $\gamma$ suggested by the performance diagnostics are not implemented if excitation diagnostics have lead to significant model parameter changes during the set point response event.

**ADAPTIVE CONTROL DEMONSTRATION**

A simulated onedimensional process is used to demonstrate the adaptive algorithm. The process consists of three FOPDT difference equations in series making the overall process character third order. The gains and time constants of the FOPDT difference equations are all initially equal to 1.00 and the overall true dead time is equal to 12.2t.

The process is made nonideal through the addition of measurement noise, process nonlinearity and model order mismatch. Measurement noise is simulated by adding random error with a standard deviation of 0.10 to the process output. Process nonlinearity is introduced by making the overall process gain a nonlinear function of the process input as:

$$K_p = (20.0)[u(t)/50.0] - 1.0$$

At the beginning of the demonstration, both the process input and output equal 50.0. The use of a FOPDT model as the controller's internal model in controlling this third order process introduces the nonideality of model order mismatch. Model order mismatch also exists in that the training patterns for the performance diagnostic VQN were made using a simulated second order process and not a third order process.

A batchwise regression of a FOPDT model to an open-loop step test made in the start-up operating regime produces model parameter estimates of $K_p = 1.00$, $\tau_p = 2.70$ and $\zeta = 2.42$. The existing controller is an IMC tuned PI controller with Smith Predictor that is designed using the model parameter estimates and $\gamma = 0.50$. The parameters of the existing controller are then fixed and not allowed to change throughout the demonstration. Again, the sample time, $\Delta t$, equals 0.04$t$.

Figure 3 shows the result of implementing the adaptive algorithm on this demonstration process. The adaptive algorithm is activated at the beginning of the demonstration and the performance feedback diagnostic VQN begins to look for set point step change response patterns to evaluate. The set point is stepped from 50.0 to 52.5 at sample 500 and the performance feedback diagnostic VQN begins collecting the response pattern. A complete response pattern is diagnosed at sample 625. As the existing controller is properly tuned for this operating regime, desired performance is found to exist and no adaptation is made.

At sample 1000, the set point is stepped from 52.5 to 60. The process input is increased during the response to this set point change. This results in the process gain increasing to approximately 1.6 which produces a very aggressive response marked by large overshoot and slow damping. The displayed performance is correctly diagnosed by the performance diagnostic VQN as being aggressive at sample 1125. This leads to the activation of the excitation diagnostic VQNs. At sample 1335, the first steady state is found allowing dynamic event identification to begin. Note that no model has been identified yet so the existing controller is still in command.

At sample 1500, the set point is stepped from 60.0 up to 62.5 and the performance diagnostic VQN begins collecting a new response pattern. At that time, the excitation diagnostic VQNs begin to identify the presence of significant process excitation within both process variables. This leads to a total of three modeling instances during the set point response event. The second modeling instance at sample 1590 produces converged estimates of all three model parameters. At that time, the second controller is designed as an IMC tuned PI controller with Smith Predictor and put on-line so as to override the effort of the existing controller. Initially, $\tau_c = 0.50$.

Since the second controller did not take command until halfway through this set point response event, the response pattern appears very aggressive. The performance diagnostic VQN accordingly recommends a large adjustment to $\gamma$. However, since the estimate of $K_p$ significantly changed during the set point response, the adjustment to $\gamma$ that is suggested is not implemented so as not to over-adapt the second controller.

The response to the fourth set point change made at sample 2000 is diagnosed as being only slightly sluggish. As a result, only a small adjustment to $\gamma$ is suggested. Four modeling instances are identified and acted upon by the excitation diagnostic VQNs during this response, but no significant changes are made in the model parameters. Since the small degree of sluggishness is not found to be accounted for by model parameter error, the suggested adjustment to $\gamma$ is made.

Desired performance identical to that displayed at the first set point change is found to exist at the last set point change made at sample 2500. Since no significant change is made in either the model parameters or the closed loop time constant, it is concluded that the adaptive algorithm has successfully updated both the process model and the tuning relations regaining desired performance.

**ACKNOWLEDGEMENT**

Acknowledgement is gratefully made to the National Science Foundation through Grant CTS-9008596, Connecticut Innovations Inc. through an Elias Howe grant, the University of Connecticut Precision Manufacturing Center, and Larry Megan.

**LITERATURE CITED**


