Proceeding of the 2004 American Control Conference
Boston, Massachusetts June 30 - July 2, 2004

A Controller Performance Monitor

Samuel O. Owusu, and R. Russell Rhinehart, Fellow, ISA

***Abstract*— This work proposes a simple, robust, efficient, and practicable method to automatically flag poor control performance. It uses only the run length of the actuating errors. Run length is defined as a State, and transitions between States are then modeled as a Markov Chain. Transition probabilities are then compared with the control limits established from a user-defined period of good control.**

I. INTRODUCTION

Controllers are tuned for desirable performance. However, over time, control performance degrades due to changing process factors, and what once was a good controller becomes a bad one, unable to control the process efficiently. Good control performance is necessary for process safety, product quality, and profitable manufacturing practice; and real time detection and diagnosis of faults have become an integral part of process design [8]. However, only about a third of industrial controllers provide an acceptable level of performance, despite the performance measures developed within the past 10 years [1, 6].

A typically large process operation consists of hundreds of control loops, often operating under varying conditions. Maintenance of these loops is generally the responsibility of either a lead operator, control engineer, or an instrument technician; but other responsibilities, coupled with the tediousness of consistently monitoring a large number of loops, often result in control problems being overlooked for long periods of time [3].

In any control scheme, a deviation from setpoint is a function of both controller performance and the plant disturbance spectrum, and it is a general requirement that any controller performance assessment technique should have at least the following basic attributes: 1) Be independent of disturbance or setpoint spectrums, 2) Be able to be automated, 3) Require minimum specification of process dynamics, and 4) Be sensitive to detuning or process model mismatch [3].

Manuscript received September 15, 2003. The authors appreciate the guidance and financial support of the industrial sponsors of the Measurement Control and Engineering Center. The original concept for this work was conceived by Dr. Ricki Ingalls and Dr. Sherry Avery of the Oklahoma State University School of Industrial Engineering and Management.

S. O. Owusu is a PhD candidate in the School of Chemical Engineering, Oklahoma State University, Stillwater, OK 74078-5021. samueoo@okstate.edu.

R. R. Rhinehart is Bartlett Chair and Head of the School of Chemical Engineering, Oklahoma State University, Stillwater, OK 74078-5021, and author for correspondence. rrr@okState.edu

A comprehensive approach for controller performance monitoring usually includes the following: 1) Determination of the capability of the control system, 2) Development of suitable statistics for monitoring the performance of existing system, and 3) Development of methods for diagnosing the underlying causes for changes in performance of the control system. Among the various methods for detecting changes in an industrial process, statistical methods generally predominate in using sampled data for decision analysis [11].

Minimum variance control (MVC) and its derivatives, popular benchmarks for control performance evaluation, are based on the work of Harris, *et al*. [2]. MVC requires that the process delay be known; however, the delay in chemical processes changes during routine operation, and online estimation is often not practicable. A control performance technique that does not require any process knowledge will be desirable.

An automated, goodness of control performance monitor developed by Rhinehart [9] uses the ratio of the expected variance of the deviation of the controlled variable from the setpoint to one half of the expected variance of the deviation of two consecutive process measurements. It then compares the current ratio-static value with some critical values to indicate performance changes. This technique however, just like other performance monitoring techniques compared a single index value to a trigger value to judge the performance. It did not consider the distribution of index values; and, therefore, was not accepted as a completely functional approach for performance assessment [7].

The Linear Quadratic Gaussian (LQG) benchmark was proposed as a more appropriate tool for assessing the performance of controllers [4]. However, calculation of LQG benchmark requires a complete knowledge of the process model, which is often a demanding requirement or simple not possible for on-line assessment.

Recently the chi-squared goodness-of-fit statistic to compare the distribution of a performance index (run length) within a window of data to a reference run length distribution in order to determine the performance of a controller was proposed. A statistically significant change in the distribution is indicative of a significant change in controller performance. The technique uses only routine plant data and is suited for online application [7]. Although it uses the generally robust chi-squared test, the theoretical foundation is not tractable.

The technique developed in this work also uses run length distribution data. However, it is based on binomially distributed variables, which provides a more satisfying basis for fundamental analysis.

II. Development of the Health Monitor

A. Definitions

A few definitions will help the reader.

Data: Actuating error or process-model mismatch.

Run length: The number of contiguous past data of like sign between consecutive zero crossings.

Zero Crossing: The switching of sign of the actuating errors from + to –, or – to +. In the special case where the error is equal to zero, it does not signify a zero crossing and the same sign as the prior State is maintained.

Transition: The change in State, e.g. a run of +2 to –1.

State: The run length with a sign representing whether run was comprised of – or + data. Further, if the number of States were limited to a maximum value of +3 or –3 for instance then a run length of 5 negative errors would be in State of –3.

Transition Probability: The probability of moving to another State at the next observation.

Count: The number of times a State has been occupied (Cumulative State value).

Window: A reference period of the past N samplings which provide data for statistical comparison at each sampling.

B. Measurement of Actuating Errors

When a controller is in operation, the actuating errors (Setpoint minus Controlled Variable) are generated sequentially as shown in Figure 1.

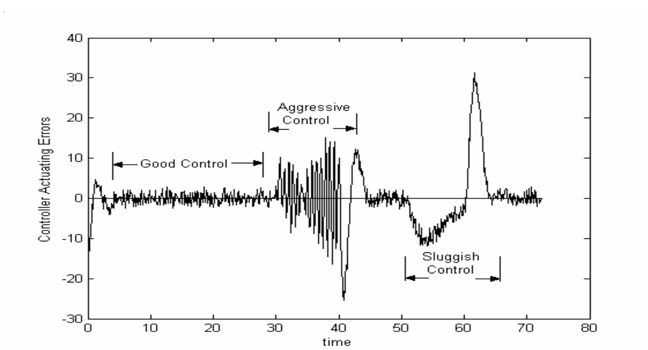

Fig.1. Controller Actuating Errors

Labeled is a period of good control when the controller was able to desirably manipulate the controlled variable in order to make the process stable and minimize deviations of the process output variable from the process setpoint. Figure 1 also illustrates a period when the controller was aggressive, resulting in increased oscillations in the process output. Lastly, a period of sluggish control is labeled. Not shown are examples of constraint encounters, sticktion, or continuous disturbances, all of which are flagged by this proposed method. The actuating errors as they occur are labeled showing their run length in Figure 2. Errors above the mean are labeled as positive and those below are labeled as negative. If the actuating error persists on one side of the

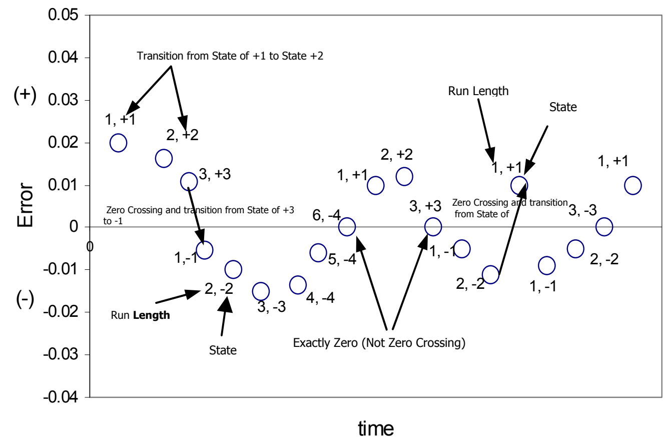

Fig. 2. Controller Run length, Zero Crossing, State and Transition State (O represents an actuating error value. The adjacent number is the State).

mean, the run length numbering continues on that side with the appropriate sign. However, anytime a zero crossing occurs, it signifies a sign change, and the state numbering also changes from positive to negative (or vice versa) and begins again with either +1 or –1 as appropriate. If the error has a value of zero, it is not a zero crossing, and the run length continues to increase, with the State bearing the same sign as the prior State. Shown here, for illustration, the maximum State number is 4. Runs of 5, 6, or more, remain in a state of ±4.

C. Model States as a Markov Chain

A Markov Chain is a probabilistic model describing the State transition of a system where the present State depends only on the immediately past State and not on the manner the system arrived at this particular State [10]. See Figure 3. Given that an actuating error run is in a State of +i, it only has 2 (binomial) transition options for the next observation. It can either move to a State of +i+1 or make a zero crossing to a State of -1. The directed paths in Figure 3 indicate the allowable state transitions. If it is in a State of -i, it also has 2 options of either moving to a State of –i-1 or making a zero crossing to a State of +1. This Markov Chain Model is shown with k States (k = 8). In general the run length can be any number, but for practical purposes, it is appropriate to limit States. Any run length higher than the extreme State (E) remains in the extreme State. Intuitively, the authors believe that the number of States should reveal about 80% of the State transitions. Based on this, the authors defined twelve (12) total States with the States of ±6 as the extreme States for this work. This implies that all run lengths of positive errors of 6 and above are labeled as being in State +6.

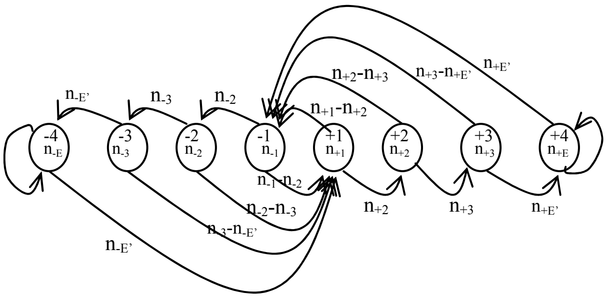

Fig. 3. A Schematic of Markov Chain with 8 States

For each State, the number of times that the actuating error is in a State +i or –i is denoted by n_{+i} or n_{-i}. These counters are shown within the circles representing State nodes. Since the run can not stay in an interior state, the number of times that runs enter an interior State is the same as the number of times that a run existed in that State. Therefore, the number of transitions from a State of +i-1 to a State of +i will, also, be given by n_{+i}. The number of transitions from an interior State of +i to a State of -1 will then be given by $n_{+i} - n_{+i+1}$. The analysis is equivalent for runs that exist in negative States. If P(+i) denotes the probability of transition from an interior State of +i to a State of -1, then:

$$p(+i) = \frac{n_{(+i)} - n_{(+i+1)}}{n_{(+i)}} \qquad (1)$$

When the run length enters the penultimate State (+E-1, or –E+1) or the extreme State (+E or -E), the determination of the transition probability is different from Equation (1). In the Penultimate State (i.e. +3 to +4 or -3 to -4 in Figure 3), the transition probability is given by:

$$p(+3) = \frac{n_{+3} - n_{(+E')}}{n_{(+3)}} \qquad (2)$$

while in the extreme State, the transition probability from the State of +E, to the State of -1 is given by:

$$p(+E) = \frac{n_{(+E')}}{n_{(+E)}} \qquad (3)$$

where, $n_{(+E)}$ denotes the total number of times the run length existed in the extreme State (either entered or reentered), and $n_{(+E')}$ denotes only the total number of times that the run length left the penultimate State (i.e.+E-1) and entered the extreme State. The same analogy is true for *p(-i)* if all the "+" signs in (1), (2) and (3) above are replaced with "-" signs.

D. Structure of Proposed Health Monitor

Figure 4 shows a flow chart of the entire structure of the proposed health monitor. The steps performed are described below. For each stage, at each sampling time, measure the process controlled variable (CV) and the setpoint (SP), and calculate the error (SP-CV).

Stage 1 - Initialization: Collect data for the entire length of a good control period as defined by the user. After data has been collected, calculate the transition probabilities associated with each State. If the actuating error run is in a State of +1, then it can only make a transition to a State of +2 with probability denoted by 1-P(+1) or make a zero crossing to a State of -1 with probability denoted by P(+1). Similarly, if the actuating error is in a State of +2, then it can only make a transition to a State of +3 with probability denoted by 1-P(+2) or make a zero crossing to a State of -1 with probability denoted by P(+2). A probability 1-P(+E) denotes that the actuating error persists in the extreme positive State, while 1-P(-E) denotes that the run persists in the negative extreme State. Once transition probabilities are calculated, determine the control limits for the data collected using binomial statistics. Store the control limits and use them as checks for future transition probabilities.

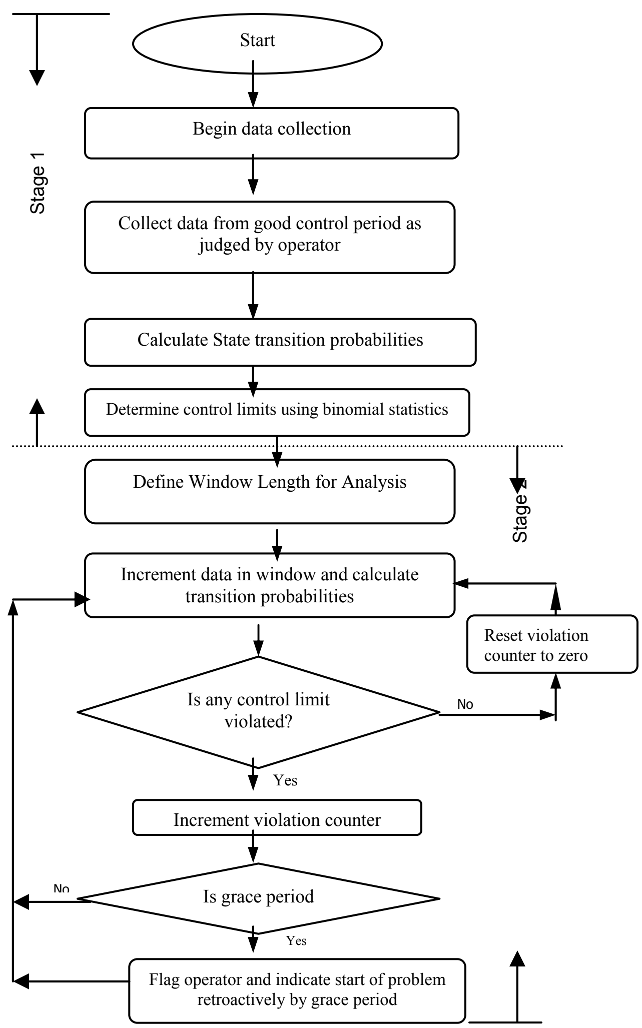

Fig. 4. Process Flow Chart for Performance Monitor

Stage 2 – Monitor: In the moving window of data, compare each transition probability with the control limits. If any transition probability lies outside the control limits, it indicates data in the new window has a significantly different behavior from the "good" reference. This could be the result of a setpoint change or a disturbance. The controller needs time to adjust to any such disturbance. Hence, a grace period equal to the closed loop settling time (CLST) plus the window length is allowed. If after the grace period is exceeded, the controller has still not been able to adjust to the disturbance, then the monitor should raise a flag and provide a retroactive time stamp based on the grace period. If the controller is able to adjust to the disturbance within the grace period, the violation counter is reset to zero.

E. Binomial Statistics to Check for Violations in Transition Probabilities.

The binomial distribution describes the chances that an outcome will occur X times in N trials and is used to determine control limits, which are used as operating limits to check the transition probabilities for violations. It is applicable under conditions where: 1) All the trials are identical, 2) Each has only two possible outcomes, 3) The probabilities of the two outcomes remain constant, and 4) The trials are independent [5]. For X outcomes in N trials, the binomial distribution makes use of the binomial formula:

$$P(X/N) = \binom{N}{X} p^{X} (1-p)^{N-X}$$

$$P(X/N) = \frac{N!}{X!(N-X)!} p(event)^{X} p(not\ event)^{N-X} \quad (4)$$

Equation (4) however, defines the probability for one State transition given a level of significance α. However, there are a total of 'k' states within a window, and each State transition is associated with a level of significance denoted by α_i (where i, denotes State in this work). Since the monitor flags if any one of the k transitions is in any of the extreme regions of the binomial density distribution as shown in Figure 5, the composite level of significance for the entire health monitor (i.e. any one of k events) will be different from an individual event. Let the composite (total) level of significance of the monitor be denoted by α_T. Then α_T = sum of the extreme tails in Figure 5. So P(Data is in extreme region) = α_T and P(Data not in extreme region) = $1-\alpha_T$ = P(no violation). Let $P(T_i)$ = probability of the i^{th} transition, then $P(T_i)$ = probability that T_i is not extreme, where the null hypothesis, H_0: (All State transitions are equal to the reference period). Then, $P(H_0:)$ = P(T_{-6} is not extreme and T_{-5} is not extreme and …T_{-1} is not extreme and T_1 is not extreme…and T_6 is not extreme). Mathematically, $P(H_0:)$ = P(T_{-6} is not extreme)*…* P(T_{-1} is not extreme)* P(T_1 is not extreme)*…* P(T_6 is not extreme) = $1-\alpha_T$. Hence, $1-\alpha_T$ = [1-P(T_{-6} is extreme)]*…* [1-P(T_{-1} is extreme)]* [1-P(T_{+1} is extreme)]*…* [1-P(T_{+6} is extreme)]. Or,

$$P(H_0:) = [1-\alpha_{-6}]*...*[1-\alpha_{-1}]*[1-\alpha_{+1}]*...*[1-\alpha_{+6}].$$

Commonly, α_i is chosen to be identical for each transition. Then,

$$1-\alpha_T = (1-\alpha_i)*(1-\alpha_i)*(1-\alpha_i)*...$$
$$= (1-\alpha_i)^k$$

Hence,

$$\alpha_i = 1-\sqrt[k]{(1-\alpha_T)} \quad (5)$$

where k is the total number of States. Equation (5) establishes the critical alpha, level of significance, for each individual State transition beyond which the monitor should flag. Since this is a two-tailed test, it implies that for each transition, the lower limit below which the controller should flag is $\alpha_i/2$ and the upper limit is $1-\alpha_i/2$. This establishes the level of significance for calculating the control on individual Stare transition probabilities.

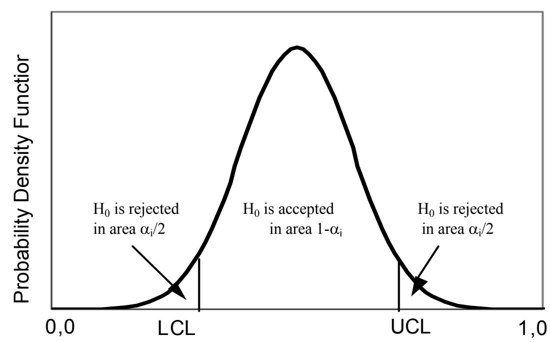

Fig. 5. Binomial Probability Density Function

F. Calculating the Control Limits

Let X = number of State transitions. Then, given H_0: and N = total number of State transitions, from Equation (4),

$$P(X_i|N_i) = \frac{N_i!}{X_i!(N_i-X_i)!} P(T_i)^{X_i} (1-P(T_i))^{N-X_i}$$

where i denotes a particular State and $0 \le X \le N$.

G. Lower Control Limit (LCL)

To find the lower control limits for which $\alpha_i/2 = \sum_{X=0}^{LCL} P(X|N)$, the cumulative sum of $P(X_i|N_i)$ is determined until two cumulative values denoted by c_o and c_N bracket the lower control value $\alpha_i/2$. Once that is established, the lower control Limit is given by interpolation as:

$$LCL = \frac{X}{N} = \frac{X_O + \left[\frac{(\alpha_i/2 - C_O)}{(C_N - C_O)}\right]}{N} \quad (6)$$

H. Upper Control Limit (UCL)

Similarly, in order to find the Upper Control Limits for which $1-\alpha_i/2 = \sum_{X=0}^{UCL} P(X|N)$, the cumulative sum of $P(X_i|N_i)$ is determined until two cumulative (c_o and c_N) values bracket the upper control value $1-\alpha_i/2$. Once that is established, the upper control limit is given by interpolation as:

$$UCL = \frac{X}{N} = \frac{X_O + \left[\frac{((1-\alpha_i/2)-C_O)}{(C_N - C_O)}\right]}{N} \qquad (7)$$

It is possible to determine the control limits above by using the binomial approximation to a normal distribution. However, the analysis used in the work makes no such approximation. The analysis is more fundamentally exact and would not lead to control limits that might violate the 0 and 1 probability limits when there happen to be few data in a State.

H. Grace Period

In order for the monitor not to flag for any short-lived disturbances, its essential to provide a grace period during which time the controller is allowed to adjust to any internal or external disturbance or setpoint change that is introduced. If a disturbance is introduced, a controller must be allowed to adjust to the desired setpoint within a reasonable time frame equivalent to the closed loop settling time (CLST). So, during monitoring, a period at least equivalent to the CLST must be provided. However, usually after the CLST is exceeded, the monitor must be allowed to collect enough data to remove the upset from the analysis window to determine if the controller has adjusted or not. Consequently, a period equivalent to one window length must be provided for the health monitor to collect data. So, a grace period equivalent to the CLST, in units of samplings, plus the window length also in units of samplings (CLST + window) must be provided before the monitor flags. The advantage of this is that once the monitor flags, it is quite certain that the controller could be having some problems and might require immediate attention. The grace period eliminates calling operator attention to brief, but recoverable events.

J. Window

In general, by setting a desired significance level, the probability of making Type-I error is fixed. However, the probability of making Type-II errors β, has to be accounted for as well. In general both Type-I and Type-II errors depend on the number of samples collected per transition, additionally β depends on α. A logical way to reduce Type-II errors [5] will be to choose a large enough window size. In this work a window size of 2000 data points was used. However, the larger the window size the greater the time to detect poor control. The authors acknowledge that this work is still in progress and future endeavors include studying the effects of number of States, window size, sampling rate, and grace period in minimizing Type-I and Type-II errors, and speed of detection and recovery.

III. Performance Monitor Demonstration

Several simulations were conducted to demonstrate the application of the health monitor. In this work, results of one case study using an overall level of significance (α_T = 1%) is shown. First, a standard PID controller was tuned using Cohen and Coon tuning parameters to control a second-order plus deadtime (SOPDT) process. A schematic diagram of the system is shown in Figure 6 where, d = Disturbance; SP = Setpoint; C = Control Variable; MV=Manipulated Variable;

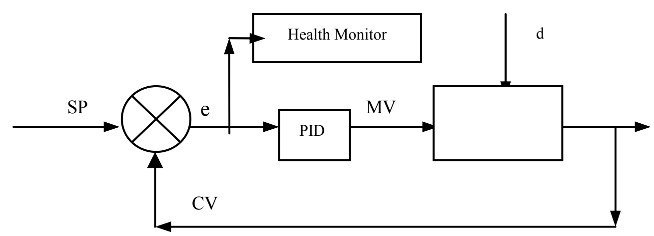

Fig. 6. Schematic of a Closed-Loop Process Control Unit

The simulated data included randomly generated noise using MatLAB. The performance output with the monitor running in tandem with the controller is shown in Figure 7. The total number of samples analyzed is a little over 40,000. With a time step of 0.1 second, this translates to a total duration of approximately 4000 seconds (≅ 1.1 hour) of monitoring.

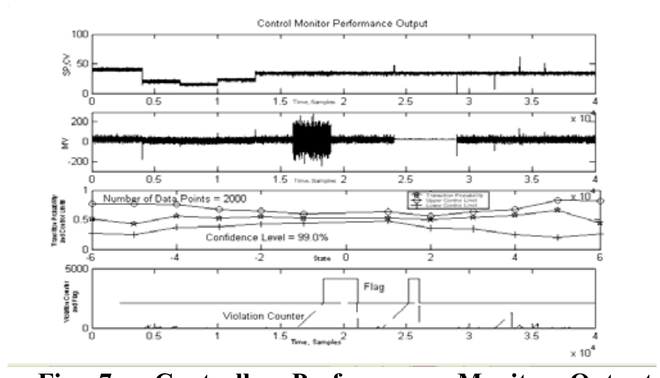

Fig. 7. Controller Performance Monitor Output (Sampling Period = 0.1s, Window length = 2000 samples, Startup Period = 250 Samples, Grace Period 2050 Samples, Violation Counter Trigger = Length of Grace Period +1, Overall Level of Significance (α_T) = 1%).

During this period various events were introduced into the control loop in order to determine the ability of the health monitor to detect poor controller performance. After the startup period of 250 samplings, the control limits were determined after the next 2000 samples were collected. At sampling 4000, a setpoint changed was made. The controller could not immediately place the controlled variable at setpoint, so the run length distribution violated a control limit and the violation counter started counting. However, the controller recovered to normal behavior within the grace period and the violation counter was reset to zero.

At sampling 7000 another setpoint change was made. Again, The controller could not immediately place the controlled variable at setpoint, so the run length distribution violated a control limit and the violation counter started counting. But, again, the upset was short lived and the monitor reset the counter to zero. Further, the smaller setpoint changes that were introduced at samplings 10,000 and 13,000 did not significantly upset the controlled behavior and the analysis window was not affected. Other times when the counter started counting might be due the fact that there is a 1% chance (α_T = 1%) of H_0 being rejected when in fact it should not (Type-I error). But, in all such instances the flag was not raised because good control was recovered within the grace period, and the monitor reset the counter to zero.

At sampling 16,000, the controller was made too aggressive by increasing the controller gain K_c by a factor of 5. The monitor detected this and started the violation counter. After the grace period was reached and the controller had still not recovered, a flag was raised indicating something was seriously wrong either with the controller or within the control loop. Between sampling 19,000 and 23,000, the controller was restored to normal performance by resetting the controller gain to its original value, the monitor immediately detected that; and, within about one window length of data from sampling 19,000, the monitor stopped flagging.

Between sampling 23,000 and 26,000, the controller was made sluggish by changing K_c by a factor of 0.1. Again, the monitor detected that the controller was not performing well, so it started the violation counter. By the time the grace period was reached the controller had not recovered, so the flag was raised. The controller was restored to normal mode at sampling 26,000. When the monitor detected a return to good control the flagging was stopped.

Further disturbances introduced in the control loop included an external disturbance at sampling 30,000, 33,000 and 36,000 where the manipulated variable was altered by a factor of -3, -1/3, and 4 respectively. In all these instances, the controller recovered within the grace period and no flag was raised.

IV. Perspective

This monitoring technique is capable of identifying changes in the control behavior in real time. While it does not indicate exactly what and where a problem could be, the moment a flag is raised is an indication that something significant persists in the control loop, which requires attention. Although, the scheme has been developed for a single controlled variable (CV), it appears applicable to all multivariable systems (apply to each CV) and to any model (apply to process-model output residuals).

V. Conclusion

A monitoring technique is hereby proposed to detect and flag poor control performance for a single variable control loop. The technique uses routine plant operating data. It neither requires *a-priori* knowledge of the process (such as model or process deadtimes and delays) nor the controller. It only requires the process setpoint and a representative data of the controlled variable during a period of good control.

References

[1] Desborough, L., and Miller R., "Increasing Customer Value of Industrial Control P Performance Monitoring - Honeywell's Experience", CPC VI, 2001.

[2] Harris, T. J., Seppala, C. T., and Jofriet, P. J., "Recent Developments in Controller Performance Monitoring and Assessment Techniques" AIChE Symposium Series 326 (1998).

[3] Hugo, J, A., "Process Controller Performance Monitoring and Assessment", Control Arts Inc., 2000.

[4] Kadali, R., and Huang, B., "Controller Performance Analysis with LQG Benchmark Obtained Under Closed Loop Conditions", ISA Transactions, 41, 512-532, 2002.

[5] Mann, P. S. "Introductory statistics", 2nd, John Wiley & Sons, 1995.

[6] Merritt, R., "Some Process Plants are Out of Control" Control for the Process Industries, 16(2), 30, 2003

[7] Li, Q., Whiteley, J. R., and Rhinehart, R. R., "An Automated Performance Monitor for Process Controllers", Control Engineering Practice, In Press, 2004.

[8] Ralston, P., Depuy, G., and Graham, J. H., "Computer-Based Monitoring and fault diagnosis: A Chemical Process case study", ISA Transactions, 40, 85-98, 2001.

[9] Rhinehart, R. R., "A Watch Dog for Controller Performance Monitoring", Proceedings of the 1995 American Control Conference. Seattle, WA. 1995.

[10] Tamir, A., "Applications of Markov Chains in Chemical Engineering", Elsevier Science B.V., NY 1998.

[11] Tatara, E., and Cinar, A., "An Intelligent System for Multivariate Statistical Process Monitoring and Diagnosis", ISA Transactions, 41, 225-270, 2002.