Event-based PID control in a flexible manufacturing process

Octavian DUCA  
Doctoral school of Fundamental Sciences and Engineering  
University “Dunarea de Jos” of Galati  
Galati, Romania  
octavian.duca@yahoo.com

Eugen MINCA  
Faculty of Electrical Engineering, Electronics and Information Technology  
Valahia University of Targoviste  
Targoviste, Romania  
eugenia.minca@gmail.com

Daniela Cernea  
Department of Automation and Electrical Engineering  
University “Dunarea de Jos” of Galati  
Galati, Romania  
daniela.cerneaga@ugal.ro

Adrian FILIPESCU  
Department of Automation and Electrical Engineering  
University “Dunarea de Jos” of Galati  
Galati, Romania  
Adrian.Filipescu@ugal.ro

Marius PAUN  
Institute of Multidisciplinary Research for Science and Technology  
Valahia University of Targoviste  
Targoviste, Romania  
paun.marius2009@yahoo.com

Abstract— In the flexible manufacturing environment some processes that until now did not need a close loop control need to be controlled so that the precision of the task permits the continuing of the manufacturing. As the processes are optimized and the manufacturing consists of multiple parallel processes, there is a need of assuring that a product is at a certain location at a certain time. In the control of a task, usually, the output of the system is measured in a timely manner at a certain sampling time. This is usually not possible in the manufacturing processes as a product position is given based on events, by sensors placed at certain locations. Certain event-based PID controllers were proposed and implemented in other control areas but in this controllers the event triggering is given by the error passing a certain level. This kind of controller cannot be implemented given the necessity of high frequency measurement of product position. In this paper we propose an PID event-base controller that uses as triggering event the activation of one of the manufacturing process sensors by the product. As the sensors are placed only in key positions the number of activations is limited. Based on these sensors the PID controller determines the necessary control signals between the key locations.

Keywords—Event-Based control, PID control, Flexible manufacturing

I. INTRODUCTION

Even if is the most researched type of control, time triggered control of manufacturing processes cannot always be used in some instances, based on the type of process. In the event-based processes, most of the time, an open loop pre-defined command is sent to the actuator on an event-based rule. This kind of control cannot cope with the variability of a flexible manufacturing process optimal operating [1]. In the optimization process and line balancing the processing and transportation durations are considered known and constant across the process duration. In the flexible manufacturing processes these durations aren’t always constant. To maintain some constant durations between some events (e.g. entering and exiting a workstation) the transportation speeds can be modified to control the arriving of a product on time [2].

In the literature most of the control of the positioning of a product on a manufacturing process is made in a time-triggered manner based on a sampling period. The control law is computed and updated every time this sampling period is reached [3]. This field is widely investigated even in the cases of measurement losses that introduces some asynchronous behavior. However fewer studies cover cases of event-based sampling, with a sampling time event-triggered, for example the arriving of a product to a certain point. Most event-based control research focus on the passing over a certain error threshold as an event-trigger and updating the control values based on the event [4].

This event-based types of control have as the main objective the reduction of the control values updates and as a result the reduction of the computational power and data transmission, especially in low power devices [5]. Through the monitoring of measurement variation at the sensor, the number of signals sent to the actuator decreases so that the command values modifies only when needed [6]. But for the monitoring of the measured values at the sensors and the error threshold that triggers the event an intelligent sensor or the allocation of controller computing power is needed for certain processes [7].

In the manufacturing environment the events that need to trigger the control laws are usually binary sensors that signal the arriving of the measured process at a certain point. This kind of signals that indicates a discrete event of a continuous process cannot be used to verify the error as a triggering event [8]. From this point of view the triggering event that activates the modification of the command value needs to be the presence of process at the level of a certain binary sensor (e.g. the presence of a part at a certain point in a work-station).

For this kind of event-based control a certain event-triggered algorithm for the activation of the controller is necessary [9]. The activation of the controller will be in some regards similar with the literature error level event triggering, as the command values are modified only when the controller is active. But from the perspective of the number of activations of the controller the differences will be significant, as in the manufacturing process the product location it verifies in a small number of points.

This paper is organized as follows: in Section 2 the time-triggered PID control will be presented as a reference base and detail the event-based PID control with the error variation as the event-trigger. The Section 3 will present the flexible manufacturing process considered for controlling. In Section 4 will present the algorithm of the event-triggering control proposed Section 5 will analyze the algorithm in certain cases and Section 6 will present the conclusions.
II. PID CONTROL

To better compare the proposed work with the existent controllers in this section the time-based PID controller and the event-based PID controller first proposed in [10] are recalled.

A. Time-Based PID Controller

In the PID control theory the S-domain controller is given by the equation:

\[ U(s) = K_p E(s) + \frac{1}{T_i s} E(s) + T_d s E(s) \]  

(1)

As the name suggests the controller can be divided in its three parts: proportional, integral and derivative, \( U_p, U_i, U_d \), the parameters of which are modified to improve the controller performances. The setpoint weighting is applied, for a more flexible structure, to \( U_p \) and \( U_d \). The three components of the PID can be represented as:

\[
\begin{align*}
U_p(s) &= K_p \cdot E(s) \\
U_i(s) &= \frac{K_i}{s} \cdot E(s) \\
U_d(s) &= K_d \cdot s \cdot E(s)
\end{align*}
\]

To avoid possible problems with the high frequency noise in the measurement a low-pass filter can be added on the derivative term, changing the derivative form to

\[
U_d(s) = \frac{N \cdot K_d}{K_d \cdot s + N} \cdot E(s)
\]

(3)

with \( N \) the filter gain.

Beside the filtering of the measurement noise in the output of the controller a saturation parameter can be set to better represent a real process, as the command to an actuator can be limited:

\[
\begin{align*}
U_{\text{comand}}(s) &= U_{\text{int}}, \quad U(s) < U_{\text{int}} \\
U_{\text{comand}}(s) &= U(s), \quad U_{\text{int}} \leq U(s) \leq U_{\text{up}} \\
U_{\text{comand}}(s) &= U_{\text{up}}, \quad U(s) > U_{\text{up}}
\end{align*}
\]

(4)

Combined to this saturation of the command value an anti-windup method must be used in most of the cases to eliminate possible control delays based on the integrator factor. Most of the time the clamping method is used, disengaging the integrator element for the duration of the command value being outside the saturation limits.

Based on the frequency domain form of the PID and discrete-time version of the controller can be obtained by mapping the s-plane to the z-plane, resulting in the time domain, in

\[
u(t_k) = u_p(t_k) + u_i(t_k) + u_d(t_k)
\]

(5)

with \( t_k = k \cdot \tilde{h} \) representing an instant moment in the sampling, with \( \tilde{h} \) the constant sampling period. Similar with the continuous time controller the proportional part is similar for the discrete time

\[
u_p(z) = K_p E(z)
\]

(6)

with the discrete time form of

\[
u_p(t_k) = K_p e(t_k)
\]

(7)

For the integral and derivative parts of the controller several solutions can be applied. A solution is represented by the backward difference approximation method based on the first-order Taylor series expansion. This expansion is based on \( s = \frac{1-z^{-1}}{\tilde{h}} \) and gives as a result

\[
\begin{align*}
U_i(z) &= \frac{K_i \cdot \tilde{h}}{1-z^{-1}} E(z) \\
U_d(z) &= \frac{N \cdot K_d}{(N \cdot \tilde{h} + K_d) - K_d \cdot z^{-1}} \cdot e(t_k)
\end{align*}
\]

(8)

Or for the time domain

\[
\begin{align*}
u_i(t_k) &= u_i(t_k - 1) + K_i \cdot \tilde{h} \cdot e(t_k) \\
u_d(t_k) &= \frac{K_d}{K_d + N \cdot \tilde{h}} \cdot u_i(t_k - 1) + \frac{N K_d}{K_d + N \cdot \tilde{h}} \cdot (e(t_k) - e(t_k-1))
\end{align*}
\]

(9)

These approximations will be more accurate as the \( \tilde{h} \) sampling period is smaller

B. Event-Based PID Controller

The first event-based PID controller proposal was made for the first time by Arzen [10] and improved in other publications. This controller consists of two parts: an event detection and the activation of the controller based on the event.

The event detector with time triggering is used is used in detecting the crossing of a certain level. This event detector runs with a sampling period \( \tilde{h} \) similar with the time-based PID. The event detector sends a request to the second part containing the controller when an event is detected. In the cases presented in the literature the event-triggering is represented by the absolute error value crossing a certain level \( \tilde{e} \) set by the user. This results in the condition:

\[
|e(t_k)| \geq \tilde{e},
\]

(10)

with \( t_k = \sum_i h_i \) as in the event-based control the sampling period, based on the number of event-triggers happened, varies resulting in \( h_i = t_k - t_{k-1} \) and can be determined only when a new event takes place.

When the event takes place the PID event-based controller will calculate and update the control signal based on the request received from the event detector. Between the events detected the control signal is kept constant as the length of the sampling time \( h_i \) is defined by two successive events. For both the derivative and integrative part the backward difference method approximation method is applied resulting:

\[
\begin{align*}
u_i(t_k) &= u_i(t_k - 1) + K_i \cdot h_i \cdot e(t_k) \\
u_d(t_k) &= \frac{K_d}{K_d + N \cdot h_i} \cdot u_i(t_k - 1) + \frac{N K_d}{K_d + N \cdot h_i} \cdot (e(t_k) - e(t_k-1))
\end{align*}
\]

(11)
with the variable $\bar{h}$ as a replacement for the constant period $h$ from equation (9).

Given the event-triggering condition (10), where an event is recorded only when the error passes a certain level, the sampling interval can become very large as the output signal can have small variations over a long period of time. To compensate for the perturbations a large sampling time can introduce, in [5] where proposed improvements over the algorithm. These compensations apply to the integral and derivative parts. A “forgetting” factor of the sampling interval is presented that after a long period of steady output value the value of the parameter $h$ is reduced so that it doesn’t affect the control signal in a very significant manner. This approach is similar to the anti-windup mechanism present in the control theories that compensate the error introduced by the saturation.

In the integral term an exponential function is used so that the sampling interval impact is decreased as the time of the steady-state increases, resulting in

$$h_{\text{exp}}(h_{k}) = h_{i}e^{\alpha(h - h_{i})}$$

with $\alpha$ used to increase/decrease the exponential sampling interval in equation (12). This kind of forgetting factor determine that the $h_{\text{exp}}(h_{k})$ gets closer to $h_{i}$ when this sampling value is small, and it gradually disappears as this value gets bigger. In a similar manner the exponential term is also used in the derivative term resulting in

$$h'_{\text{exp}}(h_{k}) = \bar{h} + (h_{i} - \bar{h})e^{\alpha(h - h_{i})}$$

with $\alpha$ used to increase/decrease the exponential factor in equation (13). This algorithm modification allows $h'_{\text{exp}}(h_{k})$ to have a value closer to $h_{i}$ when this value is small and more closer to the last sampling value period $\bar{h}$ when the value is large.

Using these modifications, an PID with exponential forgetting factor is obtained, of which the integrative and derivative values are:

$$
\begin{align*}
    u_{i}(t_{k}) &= u_{i}(t_{k-1}) + K_{i} \cdot h'_{\text{exp}}(h_{k}) \cdot e(t_{k}) \\
    u_{d}(t_{k}) &= K_{d} \cdot e(t_{k}) + K_{d} \cdot N \cdot h'_{\text{exp}}(h_{k}) \cdot u_{d}(t_{k-1}) + \\
    &+ \frac{N \cdot K_{d}}{K_{d} + N \cdot h'_{\text{exp}}(h_{k})} \cdot (e(t_{k}) - e(t_{k-1}))
\end{align*}
$$

III. FLEXIBLE MANUFACTURING PRODUCTION

The processes in which the event-based control is used the most are represented by the manufacturing processes, and especially the flexible manufacturing processes. In the classical manufacturing process we mostly have a time or event dependency, in the flexible manufacturing process is encountered a combination of time and event dependencies. These dependencies on time and events are needed primarily for the optimization of the process. In the optimization of the process the operations and transport durations need to have a constant duration, so that a product can arrive at a predetermined time at a workstation. To obtain the constant durations needed a control algorithm must be implemented.

A. Flexible manufacturing system hardware

The flexible manufacturing system considered (Fig.1) is composed of 7 workstations each controlled by a PLC (Fig.2), making the individual control easier. Each workstation is responsible for the assembly of a part of the product in a predetermined order. To better use the existent workstations a SCARA transportation system was implemented so that the product path through the workstations can differ based on the product configuration. Beside the flexible manufacturing in-line the manufacturing system is equipped with a in-cell manufacturing where the entire product is assembled in a single station (Station 3) equipped with a robotic arm. This results in a two parallel manufacturing processes that intersect in the quality control (QC) area.

For an optimal production the products on the two parallel flows must arrive on a predetermined manner at the intersection point. To ensure the following of the arriving of products at certain point at certain times a control algorithm is necessary.
B. Flexible manufacturing system control

In ensuring that the necessary assembly and transport durations are obtained, a better control method, different from the open loop sequential algorithms control, must be implemented. For this the implementation of a PID was considered. Given that the PID control is a used and very researched type of control ensures an easy adaptation.

In the manufacturing process the most encountered process is the process of transportation, predominantly on conveyors. On these conveyors a product is not monitored in a continuous manner. The position of the product is known on an event-base. The product will be detected by sensors placed in certain interest zones. In most cases the sensors are placed at entry and leaving of the workstation and at the processing/assembly area. These sensors can detect two points on the tray of the product given the possibility do detect the product in more positions.

In controlling the transportation in a workstation, it can be separated in two transportation control situations, the transport from the entry in the workstation to the assembly area and the transport from the assembly area to the leaving of the workstation. In the first control situation the speed of the conveyor can vary based on the delays introduced by the previous workstation.

In the second transportation control situation the speed of the conveyor varies based on the variability of the assembly duration. Given that the conveyors work at an optimal speed for energy consumption and maintenance the transportation speed can be increased to a certain point. If the speed increase cannot compensate the assembly delay, a certain delay will be transmitted to the next workstation.

As the position of the product can be measured only in some points the computing of the PID controller will be made only when the product is present at one of the positions. This is made possible by the fact that the sensors are monitored at a high frequency sampling time. As one of the sensors is activated the position of the product is updated and the algorithm that calculates the control value using the PID algorithm is activated updating the control value.

Given the fact that all the actuators and motors have a limited operating zone, the control value given by the PID is passed through a saturation process. In the saturation process the control value is compared with certain upper and lower limits and if the PID control value is over these limits the control value will take the limit value.

IV. EVENT-BASED PID CONVEYOR CONTROL

To ensure the conveyor control an event-based PID controller was implemented, starting from the literature work. Similar to the literature the activation of the PID controller and the calculation of the necessary conveyor speed will be based on an event-trigger. But in comparison with the examples in the literature the event trigger will not be based on the variation of the error outside a predefined limit, will be based on the arriving and activation of one of the three sensors by the product.

Given the number of sensors a small number of event-triggers will be recorded compared with the error limit event-triggers presented in literature. Also, as shown in literature, the sampling time can be verified and used as a trigger if it crosses a maximum pre-defined level, but is not always a necessity.

The resulting controller algorithm is:
**ALGORITHM 1: ALGORITHM FOR EVENT BASED PID CONTROL**

**Input:**
- `y_ref` //reference position
- `y` //actual position based on the last sensor activation

//Event trigger sensor active `s_act`

```plaintext
s_act = 1, if sensor1 or sensor2 or sensor3 == active
0, otherwise
```

**Output:**
- `u` //motor speed

1. //Calculate time between updates:
   - `h_pres = h_pres + h_nom`
   ```plaintext
   if (s_act == 1)
      e = y_ref - y
      u_p = K_p * (y_ref - y)
      u_d = T_d * (y_new - 2 * y + y_old) / T_d
      u_i = u_i + K_i * (y_e - y_old)
      u = u_p + u_i + u_d
   end
   ```

2. //saturation implementation:
   ```plaintext
   if (u < u_inf) then //command under a limit
      u_com = u
      u_i = u_i_old
   else if (u > u_sup) then //command over a limit
      u_com = u_sup
      u_i = u_i_old
   else
      u_com = u
      //anti-windup
      u_i = u_i + K_i * (y_e - y_old)
   end
   ```

3. //update values
   ```plaintext
   u_i_old = u_i
   y_old = y
   h_pres = 0 //reset of working time
   ```

**V. CASE STUDY**

Being a simple transportation process dependent only on the speed of the motor as a control signal, the system will be defined by the transfer function

\[ H(s) = \frac{1}{s} \]  \hspace{1cm} (15)

Based on this transfer function, a number of simulations and measurements were made to compare the evolution and performances of the two types of controllers. Based on the results, it can be observed that the evolution of the event-based PID is close to the continuous modeling of the system for a small transitory duration, given the control signal saturation (Fig.5).

Given the control signal saturation, having the parameters \( K_p = 0.912, \ T_I = 0.008, \ T_D = 0.009, \ N = 104.227 \) for both continuous and event-based PID, the control signal is similar for a large part of the simulation period (Fig.6.a), also in Figure 6.b can be seen the event-triggering signal representing the moments at which the product is present at one of the sensors and the PID controller is activated and the control signal updated. The tested parameters are determining an overshoot value of 0.74% that giving the control signal saturation gives an error of 0.74% in the continuous format and almost zero in the event-based control. The setpoint in the tests is represented at one of the desired positions of the product on the conveyor.

![Figure 5: Event-based and continuous PID in the case of different transitory time requirements](image)

In the case of PID parameters that determine a slower response of the controller, the differences between the event-based and continuous control are more relevant. For parameters \( K_p = 0.316, \ T_I = 0.003, \ T_D = 0.0028, \ N = 36.139 \), computed in Matlab, for both the event-based and continuous PID, as the response of the controller is slower, the differences between the event-based and continuous controlled outputs are grater than in the fast response controller as it can be seen in Figures 5 and 6. For this controller, the parameters’ overshoot values are similar at 0.74%.

![Figure 6: a) PID command values for event-based and continuous PID in the two cases, b) Activation of the PID controller based on the sensors detection](image)

The main disadvantage for the proposed event-based binary sensor triggering compared to the literature error level triggering controller is given by the discrete reference value...
range. If in the error level triggering controller, the reference can take any value on a known range and ensure a level of error, in this case to ensure a reasonable error level the reference can take only discrete values given by the used positions on the conveyor. If the value of the reference is outside these discrete values, the error will be given by the distance between the reference and the next sensor position. But as in the manufacturing process the targeted reference is the same as the sensor position this method can be used to update the control signal.

Given the saturation of the control signal that determine that the conveyor can go in only one direction some degree of stability is ensured. If the reference is placed between the positions of two sensors, or close to one of the sensors, the controller will position the part at the next sensor from the reference and stop the control signal. If the system permitted the conveyor to go back and forth the system will become unstable as the controller will try to position the product on the reference moving the product between the two sensors in an infinite loop.

VI. CONCLUSIONS
Event-based control of a process represents a particular case of control as in certain moments the controlled parameter cannot be measured. Is in most cases demanded by the process structure as the parameter value is needed only at certain values or the addition of new sensors is not possible.

In the case of the manufacturing process the proposed PID event-based control adds certain benefits. Based on the PID parameters certain behaviors of the controller can be obtained. Given the saturation of the control signal in the manufacturing processes, for a fast transitory duration, the proposed controller behaves similar to the continuous time PID, but as the transitory duration increases the differences are more visible. The stability of the controller is given primarily by the special saturation of the manufacturing process, as the conveyors usually travel in one direction. Based on this saturation the product stops ones it arrives to a certain sensor. Also, a shortcoming of the controller is given by the fact that the reference of the controller must take values from a discrete interval given by the key position in the manufacturing process.

Further research would imply more testing of the simulated result on the physical device. Also, the possibility of integration of a digital twin in the event-based control by measuring the necessary parameters in the digital twin and compare certain points with the physical system and adjust the control based on the obtained errors.

REFERENCES
[1] E. Kucera, M. Niznanska and S. Kozak, "Advanced techniques for modelling of as/rs systems in automotive," in 16th International Carpathian Control Conference (ICCC), Szilvasvarad, Hungary, 2015.
[2] D. Krenczyk, R. Davidrajuh and B. Skolad, "An Activity-Oriented Petri Net Simulation Approach for," in International Joint Conference SOCO’17-CISIS’17-ICEUTE’17, León, Spain., 2017.
[3] A. Visioli, "Research Trends for PID Controllers," Acta Polytechnica, vol. 52, no. 5, 2012.
[4] S. Durand, "Event-based stabilization of linear system with communication delays in the measurements," in Proceedings of the American Control Conference, 2013.
[5] S. Durand and J. Guerrero-Castellanos, "Event-based digital PID control," in Proceedings of the 1st IEEE International Conference on Event-Based Control, Communication, and Signal Processing, 2015.
[6] S. Durand, N. Marchand and J. Guerrero-Castellanos, "Event-based stabilization of nonlinear time-delay systems," in Proceedings of the 19th World Congress of IFAC, 2014.
[7] S. Durand, B. Boisseau and J. Guerrero-Castellanos, "Event-Based PID Control: Application to a Mini Quadrotor Helicopter," Journal of Control Engineering and Applied Informatics, vol. 20, no. 1, pp. 36-47, 2018.
[8] M. Dev, S. Jain, H. Kumar, B. Tripathi and S. Khan, "Various Tuning and Optimization Techniques Employed in PID Controller: A Review," in International Conference in Mechanical and Energy Technology., 2020.
[9] R. Borase, D. Maghade, S. Sondkar and S. Pawar, "A review of PID control, tuning methods and applications," Int. J. Dyn. Control, vol. 9, pp. 818-827, 2021.
[10] Arzen and K.E., "A simple event-based PID controller," in 14th World Congress of IFAC., 1999.