Model-free control for unknown delayed systems

Maxime Doublet, Cédric Join, Frédéric Hamelin

▶ To cite this version:

Maxime Doublet, Cédric Join, Frédéric Hamelin. Model-free control for unknown delayed systems. 3rd Conference on Control and Fault-Tolerant Systems, SysTol 2016, Sep 2016, Barcelone, Spain. hal-01369776

HAL Id: hal-01369776

https://hal.archives-ouvertes.fr/hal-01369776

Submitted on 22 Sep 2016

Model-free control for unknown delayed systems

Doublet M.[1], Join C.[1,2,3], Hamelin F.[1]

[1] Université de Lorraine, CRAN, UMR 7039 Vandœuvre-lès-Nancy 54500, France
[2] ALIEN (ALgèbre pour Identification et Estimation Numériques), 24-30 rue Lionnois, BP 60120, 54003 Nancy, France
[3] Projet NON-A, INRIA Lille-Nord-Europe, France

Abstract—The use of model-free control (MFC) spreads now more and more in industry. Nevertheless, control unknown delayed systems with this method remains an open problem. In this contribution, we present the use of model-free control in this context and we propose a solution to improve the effectiveness of this approach using a parameter estimation.

Key words : *Model-free control, parameter estimation, delayed systems, unknown delay*

I. Introduction

A lot of dynamic systems present a delay more or less important. We can find this kind of systems in many domains as physics, chemistry, aeronautics,...[7], [5]. In presence of delays, different modifications arise (different schemes of models, problems of stablity in control,...) [4]. Indeed, sometimes systems become unstable or have a loss of performance as over/under shoots or output swings and the consequences could be disastrous.

There exist many ways to tackle and to solve the problem of delayed systems in the literature, only two are quoted here. The first one is the Smith predictor [6] [13], which is based on a classic control adjusted with respect to the delay. This method works only if the delay is relatively well-known.
On the other hand, we have the model predictive control [11] and [10], which is based on a cost function that we have to minimize to find the optimal control. This cost function depends on different parameters as the reference variable, the controlled variable,...This method requires a good knowledge of the model. In this paper, we will use another method.

Around ten years ago, a new approach, usually known as model-free control, has been proposed by M. Fliess and C. Join [2] and [8]. This method is based on an ultra-local model estimated at each time. We will show how this method can be applied on delayed systems for which the delay is unknown.

Usually, model-free control is based on the scheme presented in fig 1. We have to estimate F to make the input/output behaviour of the system be equivalent to an integrator cascade of order v. With a delayed system, this equivalent model does not remain valid. Intuitively, in this context, an
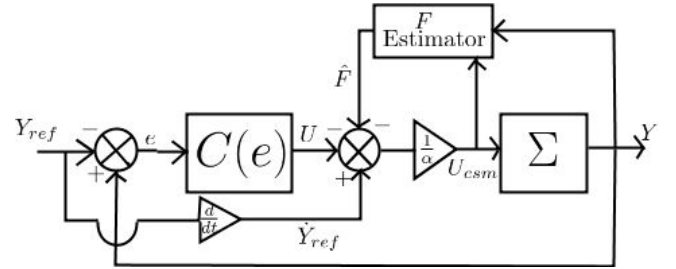

Fig. 1. MFC principle

equivalent model should be :

$$\frac{Y(s)}{U(s)} = \frac{e^{-\tau s}}{s^v} \tag{1}$$

If the delay τ is unknown, we cannot use this model (1). So, in this contribution we will present a new solution to solve this problem considering only stable delayed systems.

This paper is organized as follows, section 2 presents the model-free control. Section 3 explains our contribution to improve the effectiveness of classic MFC. The last section gives a concrete application and a conclusion is made in the last section.

II. Model-free control : usual approach

With model-free control method, we want the input/output behaviour of the considered system be ruled by equation (2) without knowing the exact model of the system :

$$y^{(v)}(t) = F(t) + \alpha u_{csm}(t) \tag{2}$$

where :

- $y(t)$ and $u_{csm}(t)$ are respectively the controlled output and input.
- v is the time derivative order, usually no more than 2 because it is sufficient to describe the behaviour of the system.
- $F(t)$ groups all the unknown signals (disturbance, noise,...) and the imperfections of the model. It is estimated at each sample time.
- α was until now considered as a constant [8]. It can be viewed as a scaling between $y^{(v)}(t)$ and $u_{csm}(t)$.

Assuming the estimation of $F(t)$, the control law is designed as :

$$u_{csm}(t) = \frac{1}{\alpha}\left(-F(t) - C(e(t)) + y_{ref}^{(v)}(t)\right) \qquad (3)$$

with $e(t) = y(t) - y_{ref}(t)$ and $C(e(t))$ is a functionnal of the error. It is chosen such as $y^{(v)} + C(e)$ is stable.

In the following, we approach $F(t)$ by a constant function (noted $\hat{F}$) in a short interval $[t - T, t]$ where T is the estimation horizon. By sliding this interval, it becomes $[0, T]$ with $\sigma \in [0, T]$ the time variable. It exists several ways to estimate $F(t)$. In this paper, we propose the continuous estimation of $F(t)$ using algebraic technics [3]. To do this, we take equation (2) with $v = 1$, because it is suffcient in the Laplace domain :

$$sY(s) - y(0) = \frac{\hat{F}}{s} + \alpha U_{csm}(s) \qquad (4)$$

Then, we derive this equation with respect to s to eliminate the initial condition $y(0)$ which is a constant term :

$$\frac{\hat{F}}{s^2} = -Y(s) - s\frac{dY(s)}{ds} + \alpha\frac{dU_{csm}(s)}{ds} \qquad (5)$$

Now, by applying the inverse Laplace transform on the interval $[0, T]$ we get :

$$\sigma\hat{F} = -y(\sigma) + (\frac{d}{d\sigma}\sigma y(\sigma)) - \alpha\sigma u_{csm}(\sigma) \qquad (6)$$

And finally, by evaluating (6) in T, we have :

$$T\hat{F} = -y(T) + (\frac{d}{d\sigma}\sigma y(\sigma))(T) - \alpha T u_{csm}(T) \qquad (7)$$

Remark : We use equation (7) to estimate F because the first derivative of $y(t)$ is generally not reachable (and in a more general context the vth derivative).

III. α ONLINE ADAPTATION AND SIMULATIONS

With the classic approach, the control is designed as given in equation (3). But now, with α variable, we will use this control law :

$$u_{csm}(t) = \frac{-\hat{F}(t)}{\hat{\alpha}(t)} - K_p e(t) + \frac{\dot{y}_{ref}(t)}{\hat{\alpha}(t)} \qquad (8)$$

We consider in the following $(v) = 1$. Furthermore, in the following, $C(e)$ will just be a proportionnal corrector K_p as in equation (8).
We do not divide K_p by α to make the correction applied to the error uniform during all the simulation. Furthermore, we can see that if α is constant, with a simple variable change, we come back in the classic approach with :

$$\tilde{K}_p = \frac{K_p}{\alpha} \qquad (9)$$

In the following, we will present how the adaptation of α is made.

A. α online adaptation

In the most proposed works, α is constant. Since a few years, the problem of its online estimation is discussed [1]. So the most important contribution of this paper is to propose an estimation algorithm of α.
As said before, α is a scaling between $y^{(v)}(t)$, with $v = 1$, and $u_{csm}(t)$. If the magnitude of the vth derivative of the output is going too high, the initial chosen value for α can become irrelevant. In this paper, we show that for important delays we have a loss of effectiveness with α constant. It is why we propose this new approach for α online estimation.

When α is variable, it leads that equations (5) and (7) become respectively :

$$\frac{\hat{F}}{s^2} = -Y(s) - s\frac{dY(s)}{ds} + \alpha(s) * \frac{dU_{csm}(s)}{ds} \qquad (10)$$

And :

$$T\hat{F} = -y(T) + (\frac{d}{d\sigma}\sigma y(\sigma))(T) - \alpha(T)T u_{csm}(T) \qquad (11)$$

Where $*$ is the convolution product.
Assuming that $\alpha(t)$ is constant for a short time lapse, the estimation of $F(t)$ remains valid.

The idea here, is to consider the error between $y(t)$ and $y_{ref}(t)$. From (3) it comes :

$$e(t) = \frac{1}{K_p}\left(\frac{-\hat{F}(t) + \dot{y}_{ref}(t)}{\alpha(t)} - u_{csm}(t)\right) \qquad (12)$$

Control objective is to track $y_{ref}(t)$ (in ideal way $e(t) = 0$) and then :

$$u_{csm}(t) = \frac{-\hat{F}(t) + \dot{y}_{ref}(t)}{\alpha(t)} \qquad (13)$$

So it comes the estimation of α :

$$\boxed{\hat{\alpha}(t) = \frac{-\hat{F}(t) + \dot{y}_{ref}(t)}{u_{csm}(t)}} \qquad (14)$$

And, the new control law becomes (8).

Now, we show some simulations explaining from where comes the difference between classic model-free control and our proposal.

Delayed systems are considered :

$$\frac{Y(s)}{U(s)} = G(s)e^{-\tau s} \qquad (15)$$

with $G(s)$ a stable transfer function. $U(s)e^{-\tau s}$ is equivalent to $u(t - \tau)$ with respect to the inverse Laplace transform. As explained in the introduction, we do not know the delay τ. We will explain the influence of the variable parameter. We set :

$$\Delta u(t) = u(t) - u(t - \tau) \qquad (16)$$

and with respect to the Laplace transform :

$$\Delta U(s) = U(s) - U(s)e^{-\tau s} \tag{17}$$

So, it comes :

$$Y(s) = G(s)U(s)e^{-\tau s} = G(s)(U(s) - \Delta U(s)) \tag{18}$$

We have to focus on the value of the $\Delta u(t)$ term because if this term is neglictable we come back in a classical situation without delay. The simulations presented here will show this value for both cases with α constant and α variable using a particular system which is a stable second order :

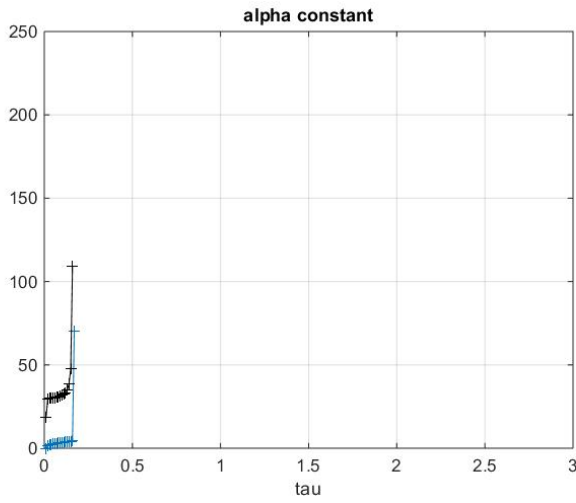$$\frac{Y(s)}{U(s)} = \frac{e^{-\tau s}}{s^2 + 1.4s + 1} \tag{19}$$

Fig. 2. In clear line $|u(t) - u(t-\tau)|$ and in black $||y(t) - y_{ref}(t)||$ in function of the delay τ in [s] for α constant.
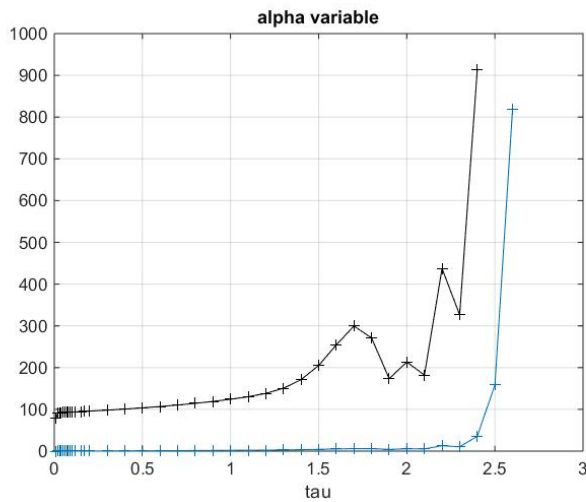

Fig. 3. In clear line $|u(t) - u(t-\tau)|$ and in black $||y(t) - y_{ref}(t)||$ in function of the delay τ in [s] for α variable.

On figures 2 and 3, we represent the value of $\Delta u(t)$ term and the quadratic error with respect to delay τ. On figure 2 we can see that with α constant, the system becomes quickly unstable (black line), that is why there is nothing more after a delay of around twenty samples (0.2s). But, before this limit the value of Δu is small enough so it is why the system remains stable (see [9] for more explanations).

On figure 3, we see that $\Delta u(t)$, the blue line, stays small a very long time thanks to the α adaptation. The system remains stable as long as Δu is unimportant. It means that we have more or less the required model given in equation (21). After a limit, around 200 samples (2s), the system becomes unstable.

So, when the estimation algorithm is used, we have shown that term $\Delta u(t)$ can be neglected and we come back in a well-known situation :

$$Y(s) \approx G(s)U(s) \tag{20}$$

B. Implementation issue

In this short subsection, we explain how the implementation for our simulations was made. We want to implement numerically the control. To do that, we have to discretize the equations given previously.

In a first time, the reference trajectory is obtained by filtering a step signal with a second order filter. Indeed, it is done to make the signal smoother and mainly, to have no problem with the derivative of the reference which appears in (3) or (12).

Next, the more important thing is to discretize the F estimation, given in equation (6), so it gives :

$$\hat{F}(k) = \frac{-y(k) + A(k) - kT_e\alpha(k-1)u_{csm}(k-1)}{kT_e} \tag{21}$$

with T_e the sampling period and where :

$$A(k) = ky(k) - (k-1)y(k-1) \tag{22}$$

$\hat{F}(k)$ is also filtered with a second order to be robust in presence of noise and then comes the adaptation of α :

$$\hat{\alpha}(k) = \frac{-\hat{F}_f(k) + \dot{y}_{ref}(k)}{u_{csm}(k-1)} \tag{23}$$

where $\hat{F}_f(k)$ is the filtered estimation of $\hat{F}(k)$.
And finally we get the control law :

$$u_{csm}(k) = \frac{-\hat{F}_f(k)}{\hat{\alpha}(k)} - K_pe(k) + \frac{\dot{y}_{ref}(k)}{\hat{\alpha}(k)} \tag{24}$$

with :

$$\dot{y}_{ref}(k) = \frac{y_{ref}(k) - y_{ref}(k-1)}{T_e} \tag{25}$$

In the next section, we show some simulations using this implementation.

IV. SIMULATIONS AND RESULTS

Here, we apply the algorithm explained in the previous section on an unknown delayed system and the efficiency of this approach is shown.

As said before, we consider here the following particular system, corresponding to (13):

$$\frac{Y(s)}{U(s)} = \frac{e^{-\tau s}}{s^2 + 1.4s + 1} \tag{26}$$

With :
- $Y(s)$ and $U(s)$ the output and the input.
- τ an unknown delay.

This is a stable second order delayed system. We control the system with the model-free control and we will present a comparison between classic model-free control and model-free control with α adaptation.
We do not need to consider a more complex system because control an unknown delayed system with good performances remains an open problem.

Each simulation has been done with the same parameters and the x-axis is graduated in time. Every time, "optimal" parameters were chosen, in particular α for the classic MFC. On figure 4, we can see the outputs obtained for
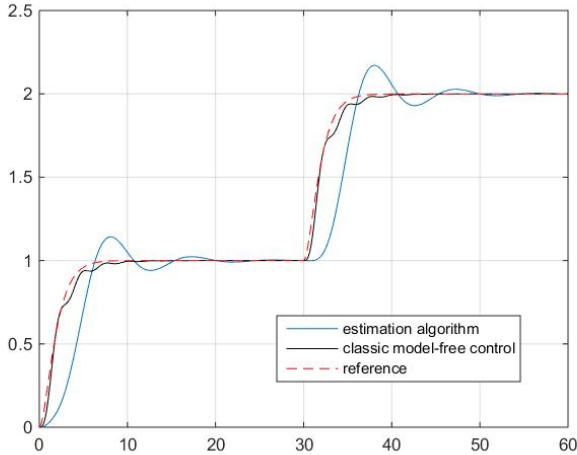

Fig. 4. Comparison between classic MFC and our proposal for a delay of 1 sample

a delay of one sample (0.01s) in both cases. We see that results are different because there is an over threshold with our proposed method but both results remain correct. For a delay of ten samples (0.1s) (figure 5), result remains equivalent for the proposed algorithm ; whereas result obtained with usual model-free control is degraded (some swings appear). This phenomenon increases with the delay and the system finally diverges as we can see in figure 6.
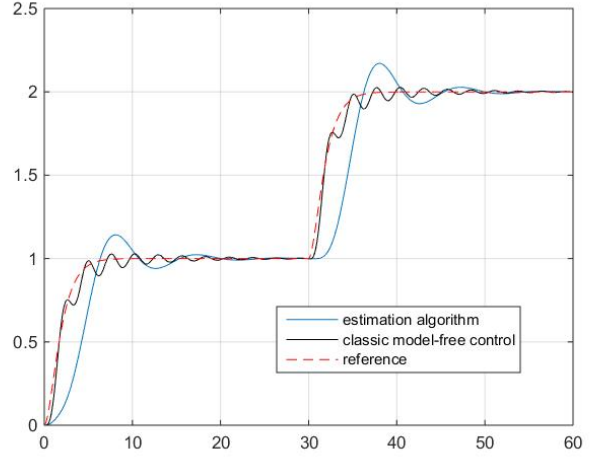

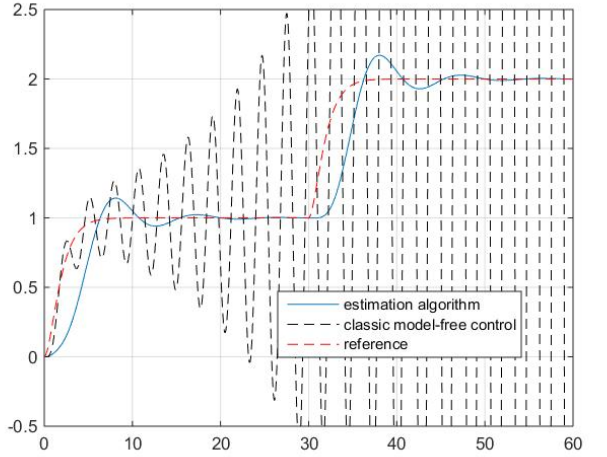Fig. 5. Comparison between classic MFC and our proposal for a delay of 10 samples

Fig. 6. Comparison between classic MFC and our proposal for a delay of 20 samples

With this three figures, we can see that when α is variable we clearly extend the range of accepted delay in front of the classic method without changing any parameters in the simulation for all cases.

Remark : *It is important to note that we do not care about the initial condition of α for the algorithm. Indeed, α is automatically adapted and it does not affect the output as shown in figure 7. On the contrary, if the chosen value for α in the classic method is not as well as needed (chosing this value by testing), some problems appear (figure 8). On figure 7 we can see that α becomes exactly identical in both cases with $\alpha(0) = 100$ and $\alpha(0) = 1$. It is clearly an advantage in front of the classic method.*
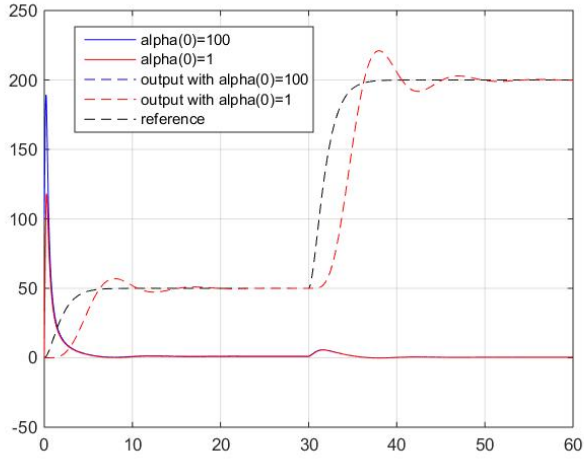
Fig. 7. Comparison of output with different initial conditions of α for a delay of 10 samples.
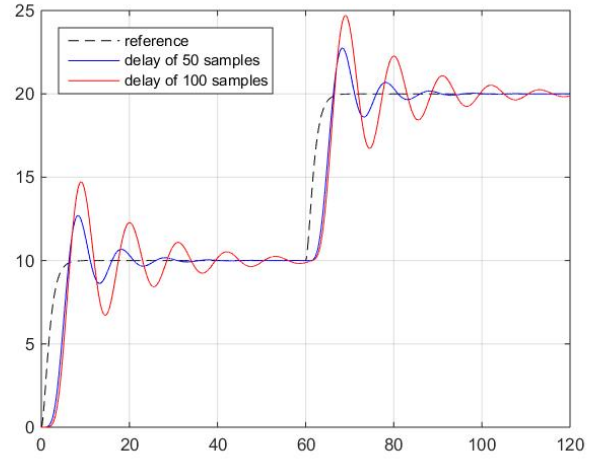

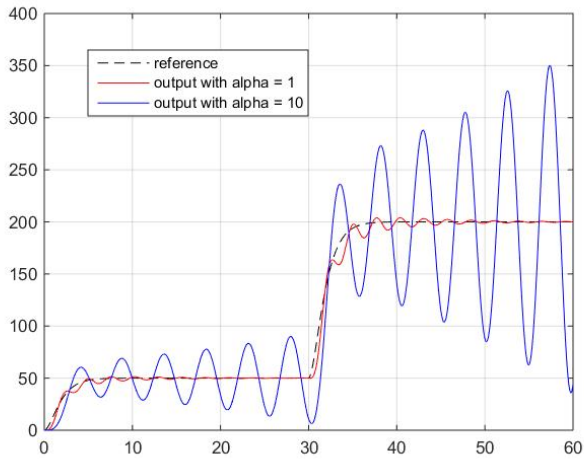Fig. 9. Comparison of output with different delays.

Fig. 8. Comparison of output with different α constant for a delay of 10 samples.

Now, some additive simulations to show the accepted range of delay for our proposed method.

As seen before, with the classic method of model-free control, a delay of around 10 samples is acceptable but no more in our particular case. After this limit, the system becomes unstable and the output diverges (figure 6). With the proposed algorithm we see that a limit far more high is acceptable, a bit less than 200 samples. On figure 9, we can see the different outputs for different delays (50 and 100 samples) with our method. In front of the classic MFC, a bigger delay is accepted, so we can say that our proposal improve the effectiveness of the classic MFC.

Here, a comparison with a classic PI controller is made as in [12] but using a MSE calculation. The controller has this form : $C(e) = (P_k + \frac{P_i}{s})(E(s))$. We consider that the delay is in : $\tau \in [0; \tau_{max}]$, with $\tau_{max} = 10[s]$ (100 samples), but we do not know its exact value. With a MSE calculation, we find the best tunning for P_k and P_i in the worst case, i.e. when the delay $\tau = \tau_{max}$. We find the following parameters : $P_k = 0.1$ and $P_i = 0.08$. With this tunning for the controller, we know that the system remains stable for any delay in the defined interval.
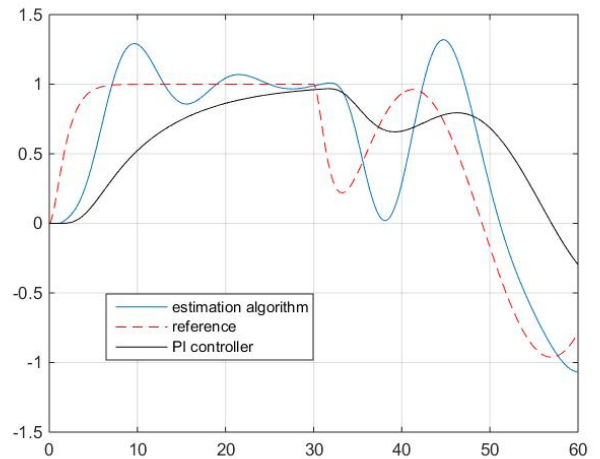

Fig. 10. Comparison between our proposed method and the "best" PI controller for a delay of 100 samples.

On figure 10, we can see the output of our system with a delay of 100 samples with the tunning of the controller

which garantees the smallest MSE. With the PI method we find $MSE_{PI} = 0.287$. On the opposite, with our proposed MFC and for the same delay, we find $MSE_{MFC} = 0.139$. With a MSE which is two times higher with the PI controller, we can say that our proposed method presents clearly an advantage against classic methods like PI controller when the delay is unknown.

To conclude this section, we show two simulations (figure 11 and 12) to prove the robustness of our method in presence of an output additive noise.
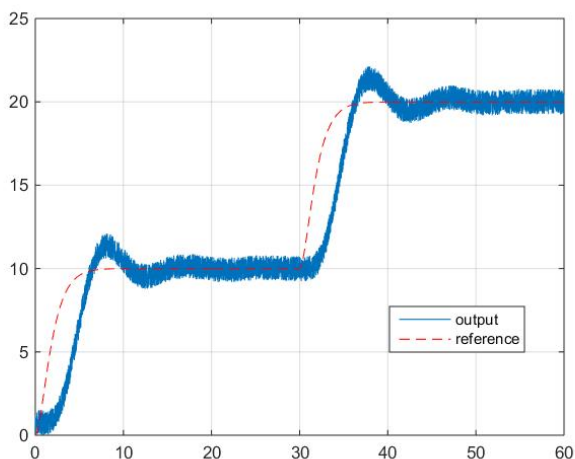

Fig. 11.   Results for a delay of 10 samples in the presence of noise.
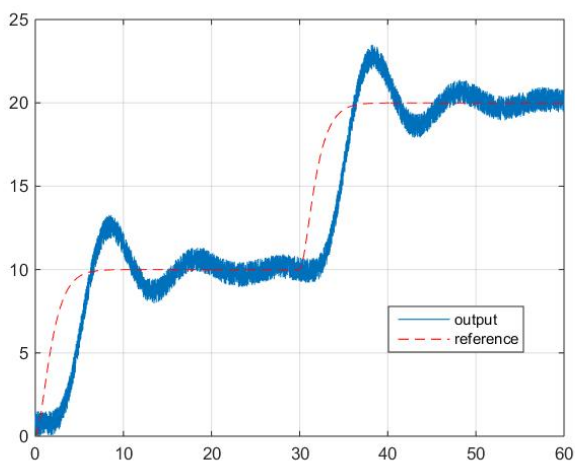

Fig. 12.   Results for a delay of 50 samples in the presence of noise.

V. CONCLUSION

In this contribution, unknown delayed systems were considered. We present a solution to control this kind of system using model-free control method. Furthermore, an improvement of this method was explained. Indeed, a method to adapt online the parameter α was presented. An explanation of the difference between usual model-free control and our proposal was done thanks to some simulations. With different simulations and different delays, the efficiency of this approach has been shown when the delay is unknown.
More, a comparison with a classic PI controller has been done. Thanks to a MSE calculation, we have shown that our proposal presents better results than the classic PI controller.

For the future, a more mathematical proof will be presented and this method will be applied on systems described by partial derivative equations because a link between unknown delayed systems and distributed systems can be made.

REFERENCES

[1] Francisco Javier Carrillo and Frédéric Rotella. Some contributions to estimation for model-free control. In *17th IFAC Symposium on System Identification SYSID 2015 Ű Beijing, China*, 2015.
[2] C.Join, J.Masse, and M.Fliess. A model-free control for an engine throttle: a preliminary study. *Journal Européen des Systèmes Automatisés, Hermès, 42 (2-3), 337-354*, 2008.
[3] H.Thabet, M.Ayadi, and F.Rotella. An algebraic control approach based on the estimation of an ultra-local broida model. In *4th International Conference on Systems and Control, Sousse, Tunisia.*, April 28-30, 2015.
[4] J.P.Richard. Time-delay systems : an overview of some recent advances and open problems. *Automatica, 39, 1667-1694*, 2003.
[5] K.Gu and S.I.Niculescu. Survey on recent results in the stability and control of time-delay systems. *Journal of dynamic systems measurement and control, 125(2), 158-165*, 2003.
[6] K.J.Astrom, C.C. Hang, and B.C. Lim. A new smith predictor for controlling a process with an integrator and long dead-time. *IEEE transactions on automatic control, 39(2), 343-345*, 1994.
[7] L.Denis-Vidal and C.Jauberthie. Identifiability of a nonlinear delayed-differential aerospace model. *IEEE Transactions on automatic control, 56(1), 154-158*, 2006.
[8] M.Fliess and C.Join. Model-free control. *International Journal of Control, 86(12), 2228-2252*, 2013.
[9] M.Fliess and C.Join. Stability margins and model-free control: A first look. In *13th European Control Conference, ECC'14, 454-459*, 2014.
[10] M.Nikolaou. Model predictive controllers : A critical synthesis of theory and industrial needs. *Advances in Chemical Engineering, 26, 131-204*, 2001.
[11] M.R.Garcia, C.Vilas, L.O.Santos, and A.A.Alonso. A robust multi-model predictive controller for distributed parameter systems. *Journal of Process Control, 22, 60-71*, 2012.
[12] P.A.Gédouin, E.Delaleau, J.M.Bourgeot, C.Join, S.Arab-Chirani, and S.Calloch. Experimental comparison of classical pid and model-free control: Position control of a shape memory alloy active spring. *Control Engineering Practice, 19, 433-441.*, 2011.
[13] S.Majhi and D.P.Atherton. Obtaining controller parameters for a new smith predictor using autotuning. *Automatica, 36, 1651-1658*, 2000.