Variable sampling period adaptive control based on reinforcement learning

João M. Lemos[1], Francisco Parente[1], and Rita Cunha[2]

[1] INESC-ID, Instituto Superior Técnico, Universidade de Lisboa, Portugal,
jlml@inesc-id.pt,
franciscolopesparente@tecnico.ulisboa.pt,
[2] ISR, Instituto Superior Técnico, Universidade de Lisboa, Portugal,
rita@isr.tecnico.ulisboa.pt

Abstract. This article addresses the design of an adaptive control algorithm, based on reinforcement learning, for a class of bilinear systems with an accessible disturbance. The main contribution consists of using a variable sampling period, indexed to the control variable, to implement a warped time scale, in which the plant appears as a linear system to the controller. As a result, it is possible to perform aggressive manoeuvres, that consist of sudden reference changes defined by high amplitude step functions. Simulations show that the algorithm proposed is able to outperform the direct use of reinforcement learning adaptive control with a constant sampling period.

Keywords: Adaptive control, reinforcement learning, variable sampling, bilinear systems

1 Introduction

Although adaptive control based on linear quadratic reinforcement learning is able to control nonlinear plants so as to track a reference, aggressive manoeuvres, that amount to sudden reference changes of high amplitude, may result in poor performance or even lead to instability of the closed-loop. This drawback stems from the fact that, when using a linear quadratic formulation of reinforcement learning, the controller converges to a linearized controller that is only able to act locally and is unable to generate the adequate control actions to transfer the state of a nonlinear plant across different operating regimes. In order to do state changes of large amplitude, one must thus change the reference slowly enough so that the controller gains have enough time to adapt to the different intermediate operating regimes. Therefore, in order to perform fast state changes, one must embed in the control action some form of compensation of the plant nonlinearity.

For that sake, one possibility consists of using a feedback linearization like approach together with a linear control law applied to the resulting association. An example, closely related to the class of plants considered here is a distributed collector solar thermal field [2] (chapter 6). Another approach, that is limited to a class of bilinear plants, consists in making a change of the time variable, in a way described below. This approach was used with great success in adaptive predictive control of solar plants [5, 6, 4],

and demonstrated in actual plants, but the same idea can be explored for other adaptive control algorithms. This class of algorithms was name WARTIC-state and WARTIC-io algorithms, with WARTIC standing for "WARped TIme Controller".

The main contribution of this work consists of a reinforcement learning based adaptive control algorithm that uses a variable sampling period, indexed to the control variable, to implement a warped time scale, in which the plant appears as a linear system to the controller. The algorithm is named WARTIC-RL.

The article is organised as follows: after this introduction, the problem, including the definition of the class of plants considered, is formulated in section 2; the strategy for variable sampling period that yields an equivalent linear plant in discrete time is explained in section 3, and the reinforcement learning based adaptive controller that explores this strategy is described in section 4; section 5 shows simulation results that compare the performance yielded by this algorithm with constant sampling adaptive controller; finally, section 6 draws conclusions.

2 Problem formulation

Consider the first order bilinear plant described by

$$\frac{dx}{dt} = -axu + bd, \tag{1}$$

where a and b are constant, but unknown, parameters, $u \in \mathbb{R}$ is the manipulated variable, $x \in \mathbb{R}$ is the state, and $d \in \mathbb{R}$ is an accessible disturbance.

Let $r_x \in \mathbb{R}$ be the reference to track and $t \in \mathbb{R}$ continuous time. The controller to design acts, according to a zero-hold definition, in discrete time instants t_i $i = 0, 1, 2, \ldots$, and starting from state $x(t_k)$ at discrete time t_k so as to minimize the infinite horizon quadratic discounted cost

$$V(x(t_k)) = \sum_{i=k}^{\infty} \gamma^{i-k} \left[(x(t_i) - r_x(t_i))^2 + Ru(t_i)^2 \right], \tag{2}$$

with $R > 0$ a weight in the control action and $0 < \gamma < 1$ a discount factor, required by reinforcement learning.

Aspects related to tracking of constant references, that require the inclusion of integral action, are ignored here since this work is focused on improving the stability basin by using a change of the time variable.

3 Variable sampling period

Define the new time scale τ as the solution of the differential equation

$$\frac{d\tau}{dt} = u \tag{3}$$

and the virtual control variable v by

$$v := \frac{d}{u}. \tag{4}$$

If the value of $v(t)$ is computed using a control algorithm, the actual control variable to apply to the plant, u, is, of course, computed by

$$u(t) = \frac{d(t)}{v(t)}. \tag{5}$$

By using the above definition, and the chain rule of derivatives, the dynamics of the plant (1) is written, in the time scale τ, as

$$\frac{dx}{d\tau} = -ax + bv, \tag{6}$$

i. e., becomes linear.

On the other side, using the formula for the inverse derivative, and (3), the following expression is obtained

$$\frac{dt}{d\tau} = \frac{1}{u}. \tag{7}$$

Approximating this expression using finite differences, yields

$$t_{k+1} = t_k + \frac{h}{u(t_k)}, \tag{8}$$

where h is the constant sampling time considered in the time scale τ, in which the plant admits a linear model. This expression defines the next sampling time in the time scale t (the time "as we see it in our watch") such that the controller "sees" a linear plant.

4 Reinforcement learning based adaptive control

Figure 2 shows the block diagram that defines the architecture of the variable sampling adaptive controller (WARTIC-RL).

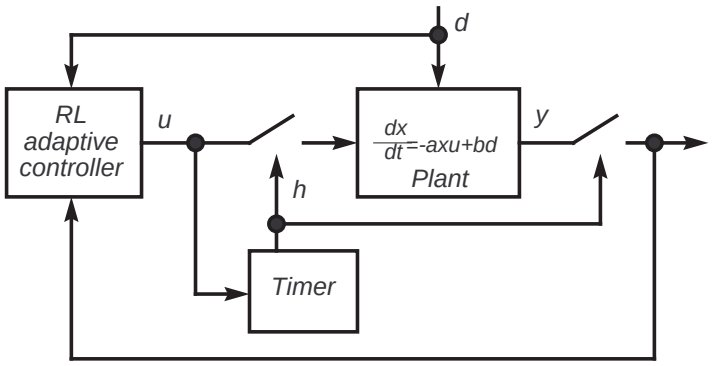

Fig. 1. Architecture of the variable sampling adaptive controller.

The timer is defined by (8), together with logic that ensures that the sampling period is always positive, and between *a priori* chosen minimum and maximum bounds, τ_{min} and τ_{max}.

The controller is an adaptive reinforcement learning controller based on Q-learning and policy iteration, as described in [3], but with the quality function Q approximation parameters estimated using directional forgetting recursive least squares (DF-RLS) [1].

4.1 Reinforcement learning adaptive control

In order to obtain a reinforcement learning adaptive controller, and following [3], define the instantaneous reward obtained when using a control policy u,

$$r(x_k,u_k) := (x_k - r_{x,k})^2 + Ru_k^2, \tag{9}$$

where the shorthand notation $x_k := x(t_k)$, $u_k := u(t_k)$, $r_{x,k} := r_x(t_k)$ has been used, and observe that the resulting cost verifies the Bellman equation

$$V(x_k) = r(x_k,u_k) + \gamma V(x_{k+1}). \tag{10}$$

Since V does not explicitly depends on u, obtaining the control law from this function requires the knowledge of the parameter b. Since the objective is to obtain a model free adaptive control algorithm, one resorts to the *quality function* [3], that depends both on the state and the control variables, in an explicitly manner, and is defined by

$$Q(x_k,u_k) := r(x_k,u_k) + \gamma V(x_k,u_k). \tag{11}$$

The quality function is such that, for the control policy u,

$$Q(x_k,u_k) = V(x_k) \tag{12}$$

and verifies the Bellman-like equation

$$Q(x_k,u_k) = r(x_k,u_k) + \gamma Q(x_{k+1},u(x_{k+1})), \tag{13}$$

where $u(x_{k+1})$ stands for the control action that results from applying the control policy u to the state x_{k+1}.

In the absence of constraints, the control is obtained from Q by solving

$$\frac{\partial}{\partial u} Q(x_k,u) = 0. \tag{14}$$

As shown in [3], for linear plants with quadratic criteria, the quality function is given by a linear combination of the entries of the vector $\phi(z_k) = z_k \otimes z_k$, where $\otimes$ denotes the Kronecker product and $z_k = [x_k\, u_k]^\top$. For a first order plant,

$$\phi_k = \begin{bmatrix} x_k^2 \\ u_k^2 \\ x_k u_k \end{bmatrix}. \tag{15}$$

The quality function is approximated by

$$Q(x,u) = W^\top \phi(x,u), \tag{16}$$

where the vector of weights $W = [w_1 \dots w_N]^\top$, with $N = \dim \phi_k$, is estimated by least squares. For this sake, observe that the Bellman like equation (13), together with (16), yield

$$W^\top (\phi(z_k) - \gamma\phi(z_{k+1})) = r(x_k,u_k), \tag{17}$$

that defines a linear regression model, in which $r(x_k,u_k)$ is expressed as a linear combination of the entries of the regressor vector $\phi(z_k) - \gamma\phi(z_{k+1})$.

Furthermore, from (14), ((16), and, for the special case of a first order plant, (15), the approximation of the optimal control is given by the feedback law

$$u_k = Fx_k + \eta_k, \tag{18}$$

with the controller gain given by

$$F = -\frac{w_3}{2w_2}. \tag{19}$$

The variable η denotes a dither noise signal injected in order to fulfill a persistency of excitation condition. An important controller tuning knob is the standard deviation of η, that must be as low as possible not to reduce tracking performance, but high enough so that W is identifiable.

The above reasoning justifies the following

Policy Iteration Algorithm
Initialize. Select a value of F that stabilizes the closed-loop.
Time loop
For $k = 1$ up to k_{end} perform the steps

- **Policy Evaluation Step**. Compute the least squares estimate of W using the regression model (17).
- **Policy improvement step**. Update the control using (18), (19).

end

□

4.2 Directional forgetting Recursive Least Squares

The estimates of the parameters W in the linear regression model (17) are obtained with the directional forgetting version of recursive least squares (DF-RLS) [1]. Directional forgetting is useful in situations where there are identifiability problems, as might be the case in reinforcement leaning adaptive control, where the dimension of the vector of parameters to estimate is higher than the number of the parameters of the plant model. Poor identifiability might cause parameter drift, leading to numerical problems and even drift of the controller gains themselves. For the estimation of the parameters in the general regression model $y(k) = \theta^\top \varphi(k)$, where θ is the vector of parameters to

estimate, $\varphi(k)$ is the regressor at time k and $y(k)$ is the independent data, the DF-RLS is implemented by recursively executing the following equations [1]

$$\beta(k) = 1 - \lambda + \frac{1-\lambda}{\varphi^{\top}(k)P(k\ 1)\varphi(k)} \tag{20}$$

$$\mathbb{K}(k) = \frac{P(k-1)\varphi(k)}{1+\varphi^{\top}P(k-1)\varphi(k)[1-\beta(k)]} \tag{21}$$

$$\hat{\theta}(k) = \hat{\theta}(k\ 1) + \mathbb{K}(k)[y(k) - \varphi^{\top}(k)\hat{\theta}(k-1)] \tag{22}$$

The expression for β ensures that $P(k) \succ 0$ if $P(0) \succ 0$.

In this case, $\theta = W$, $\varphi(k) = \phi(z_k) - \gamma\phi(z_{k+1})$, and $y(k) = r(x_k, u_k)$, and $0 < \lambda \leq 1$ is the forgetting factor in the direction of the incoming information.

5 Simulation results

Hereafter, three simulation examples are presented:

1. A sudden reference jump is applied, using the basic algorithm with constant sampling period, resulting in an unstable behaviour;
2. The reference slowly changes between the extreme values considered in the previous example. If the rate of change is slow enough, the algorithm will track the reference;
3. The conditions are as in the first example, but the sampling period is now variable according to the WARTIC-RL adaptive control algorithm. Opposite to the first example, the plant output tracks now the reference.

In all the examples there is an offset of the average value of the plant output, when tracking a constant reference. This offset was not of concern here and could be cancelled by the incorporation of integral effect. In all cases, a stabilizing controller is used in the initial samples.

5.1 Example 1: sudden reference jump with constant sampling

This example uses reinforcement learning based adaptive control with a constant sampling rate of $h = 0.01s$ in order to establish a baseline for the comparison of other situations, described in examples 2 and 3. The results for the plant output are shown in figure 2. The reference to track consists initially of a square wave of small amplitude, in the sense that the plant operates close to the same working point. At $t = 250s$, there is a very large change of the reference, and the closed-loop becomes unstable.

5.2 Example 2: slow change with constant sampling

In example 2 a constant sampling period is also used and the plant starts from a situation similar to the one in the beginning of example 1, and then, as shown in figure 3, the average value of the reference is gradually increased. The controller is able to stabilize

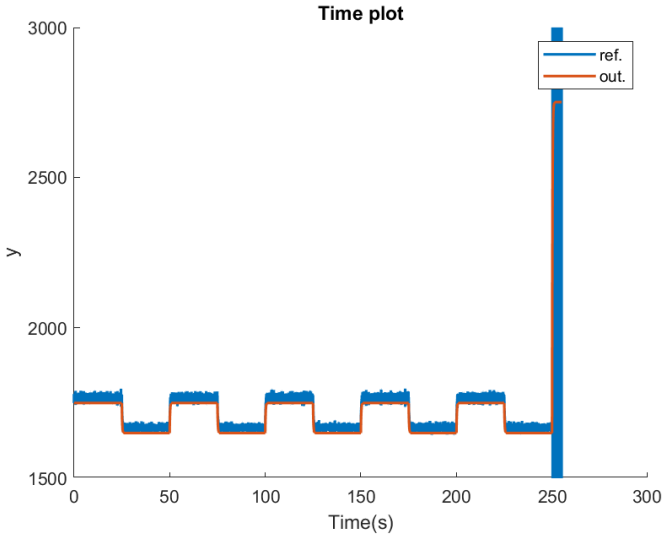

Fig. 2. Example 1. Plant output Constant sampling period with a sudden reference change.

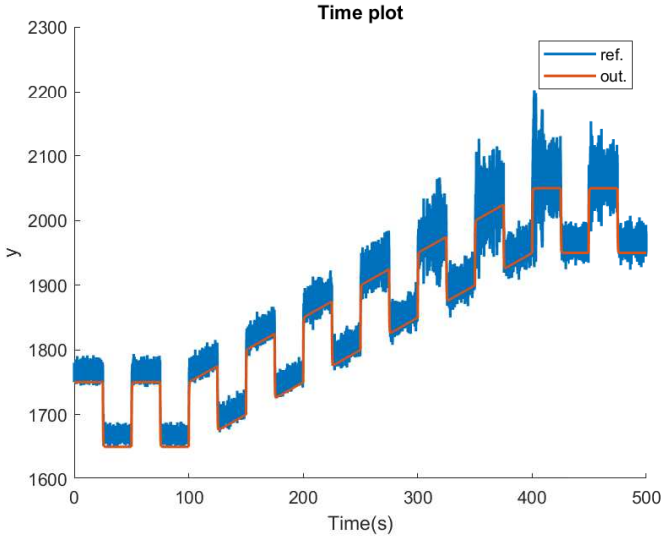

Fig. 3. Example 2. Plant output. Constant sampling period with a slow reference change.

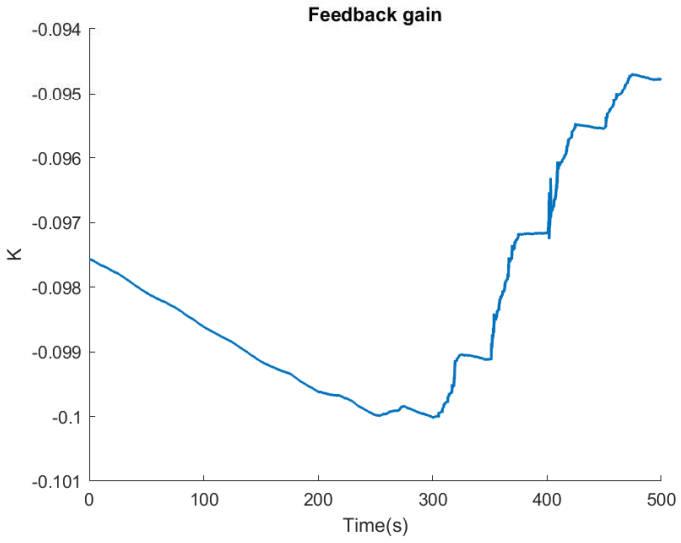

Fig. 4. Example 2. Controller gains. Constant sampling period with a slow reference change.

the closed-loop and gradually tracks the reference, although with some performence degradation (the variance of the output around the reference during periods in which the reference is constant increases). This behaviour is possible because, since the rate of change of the operating point is low, the gains, shown in figure 4, have enough time to adapt.

5.3 Example 3: sudden reference jump with variable sampling

The results of example 3 are shown in figure 5, where the WARTIC-RL algorithm is used. The conditions are the same as in example 1: after a period in which the reference has small changes around an operating point, there is an instantaneous, large change of the reference to another operating point. Opposite to what happened in example 1, in which the large amplitude reference change induced instability, the output perfectly tracks the reference. Since, due to the change in the sampling period, the controller ”sees” a constant linear plant model, the controller gain changes very little, as shown in figure 6.

6 Conclusions

By using a time varying sampling interval it is possible to change the time scale in continuous time such that, in discrete time, a class of bilinear plants admits a linear model. When using a reinforcement learning based adaptive controller, this fact, allows to perform aggressive manoeuvres in the plant, defined by very large step changes in the

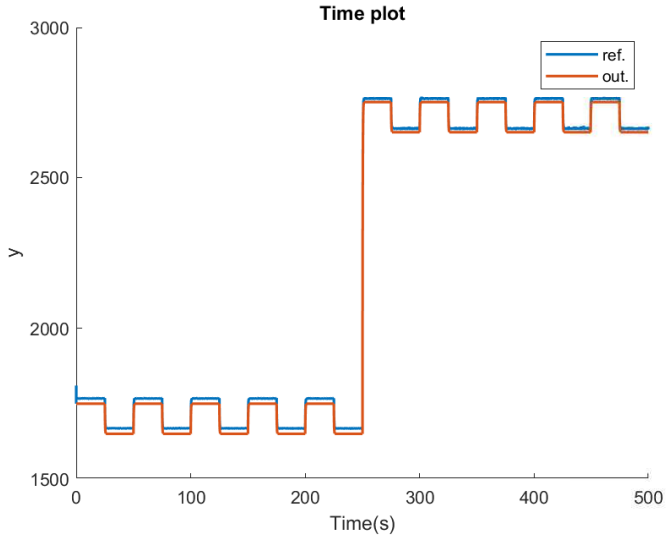

Fig. 5. Example 3. Plant output. Variable sampling period with a fast reference change.

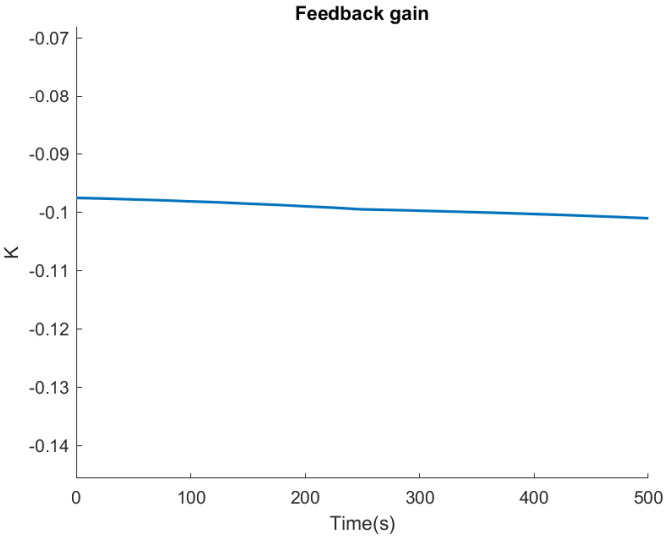

Fig. 6. Example 3. Controller gains. Variable sampling period with a fast reference change.

reference. For a constant sampling controller, the resulting behaviour would be unstable. The new algorithm proposed is named WARTIC-RL.

Although the class of plant is restricted, it comprises examples of significant practical engineering importance, such as temperature control in distributed collector solar thermal plants. Furthermore, it is remarked that, for the sake of explaining the core ideas in a simple way, only the first order case has been addressed, but the same method can be applied to higher order plant models.

7 Acknowledgements

This work was aupported by national funds through FCT, Fundação para a Ciência e a Tecnologia, under project UIDB/50021/2020.

References

1. R. Kulhavý (1987). Restricted exponential forgetting in real-time identification. *Automatica*, 23(5):589-600. https://doi.org/10.1016/0005-1098(87)90054-9
2. Lemos, J. M., Neves-Silva, R. and Igreja, J. M. *Adaptive control of solar energy collector systems*. Springer (2014). DOI:10.1007/978-3-319-06853-4
3. Lewis, F. L., Vrabie, D., and Vamvoudakis, G. (2012). Reinforcement learning and feedback control. *IEEE Control Systems Magazine*, 32(6): 76-105. DOI:10.1109/MCS.2012.2214134
4. Pin, G., Falchetta, M., and Fenu, G. (2008). Adaptive time-warped control of molten salt distributed collector solar fields. *Control Eng. Practice*, 16:813-823. DOI:10.1016/j.conengprac.2007.08.008
5. Silva, R. N., Lemos, J. M. and Rato, L. M. (2003) Variable sampling adaptive control of a distributed collector solar field. *IEEE Trans. Control Syst. Techn.* 11(5):765-772. DOI:10.1109/TCST.2003.816407
6. Silva, R. N., Rato, L. M., and Lemos, J. M. (2003) Time scaling internal predictive control of a solar plant. *Control Eng. Practice*, 11(12):1459-1467. DOI:10.1016/s0967-0661(03)00112-6