

Self-Triggering Based on Lyapunov with Adaptive Control Law for WNCS

E. Pecker Marcosig, F. E. Felicioni and A. Zanini

Abstract— This paper addresses the use of Lyapunov functions to self-triggering control implementation. In order to do this a condition, which comes from a sampling technique named Lyapunov sampling, is proposed. The results show the average sampling frequency reduction that can be assessed with this self-triggering method with respect to event and periodic sampling schemes. Using this to dynamically adjust sampling-time, a Lie-algebraic approach is considered to guaranteed closed-loop stability and performance. These results are particularly useful for wireless networked control systems.

Keywords— Wireless Networked Control Systems, Resource-Constrained Systems, Lyapunov Sampling, Discrete-Time Switched Systems.

I. INTRODUCCIÓN

ES cada vez más frecuente la utilización de redes de comunicaciones para transmitir información entre los distintos elementos de un sistema de control por realimentación. Es así como surgen los llamados Sistemas de Control sobre Redes (NCS) y en particular los Sistemas de Control sobre Redes Inalámbricas (WNCS). En lo atinente al control, la inclusión de una red en el lazo trae aparejado problemas tales como: pérdida y desincronización de paquetes de datos, retardos y variación del período de muestreo. Si la red es inalámbrica se pueden sumar los problemas de redundancia de señales cuando existe más de un sensor, seguridad en el acceso a la red, *visibilidad* entre nodos y restricciones energéticas. Estos efectos pueden degradar significativamente el desempeño del sistema de control e incluso inestabilizarlo. No hay que perder de vista que tanto los NCS como los WNCS también tienen requerimientos de tiempo real.

Con la implementación de NCS y WNCS el uso eficiente de los recursos (entre ellos: red, procesamiento y batería) cobra relevancia. El muestreo periódico lleva a una utilización de estos que no necesariamente es adecuada debido a que la ley de control se ejecuta de forma uniforme sin atender al estado del sistema. La ventaja de su utilización radica en que en general lleva a una simplificación de la implementación y del diseño puesto que esta teoría está consolidada. Por otro lado, se podría usar algún esquema de muestreo asincrónico que tome en consideración el estado de los recursos y de la planta para decidir cuándo tomar una muestra. El muestreo no periódico recibió mucha atención en la década del 60 cuando

se comenzó a trabajar con control digital quedando relegada hasta hace unos pocos años [2]. Lo que se busca con este tipo de muestreo es muestrear cada vez que sea necesario [9]. Entonces, una forma de enfrentar las problemáticas mencionadas anteriormente consiste en recurrir a esquemas de muestreo asincrónico como el muestreo por eventos y el automuestreo. El automuestreo es anticipativo prediciendo el próximo instante de muestreo mientras que el muestreo por eventos necesita monitorear una condición cuyo no cumplimiento produce un evento [1]. De esta manera el primero se presenta como una opción interesante para NCS y WNCS.

Al utilizar un periodo de muestreo variable surgen problemas teóricos y se hace necesario analizar el comportamiento del controlador de modo que el sistema no pierda performance ni estabilidad. En [5] se utiliza un controlador por realimentación de estados ajustado por el método LQ para un sistema que conmuta entre dos periodos de muestreo. En [10] se demuestra que algunos sistemas diseñados con LQ podrían inestabilizarse ante ciertas secuencias de conmutación entre los periodos de muestreo mencionados. En ese trabajo el problema se resuelve recurriendo a un diseño de controlador utilizando un sistema de Inecuaciones Matriciales Lineales (LMI's). En [7] se aplican técnicas de álgebra de Lie resoluble para reajustar el controlador ante cambios del periodo de muestreo asegurando estabilidad en el sentido de Lyapunov.

En este trabajo se presenta una técnica de automuestreo que incluye la actualización de los parámetros del regulador. En los apartados correspondientes se analiza tanto el desempeño del sistema en lazo cerrado como las condiciones a cumplir para garantizar la estabilidad.

El resto de este trabajo se estructura como sigue. En la Sección II se repasan algunos conceptos que serán utilizados a lo largo del artículo. La Sección III describe cómo adaptar los parámetros de un controlador por realimentación de estados ante cambios en el periodo de muestreo utilizando álgebra de Lie, de modo de garantizar performance y estabilidad del sistema conmutado. El concepto de auto-muestreo y las ventajas que supone para sistemas de control sobre redes se explica en la Sección IV. También se propone un algoritmo para el cálculo del periodo de muestreo en función del estado del sistema basados en el muestreo de Lyapunov y se aplica sobre una planta simulada. Finalmente en la Sección V se resumen los resultados obtenidos.

II. CONSIDERACIONES PREVIAS

A. Muestreo no periódico o asincrónico:

Dado un sistema de tiempo continuo expresado en variables de estado el sistema discreto equivalente se representa por

E. Pecker Marcosig, Universidad de Buenos Aires (UBA), Buenos Aires, Argentina, epecker@fi.uba.ar

F. E. Felicioni, Universidad de Buenos Aires (UBA), Buenos Aires, Argentina, flaviafelicioni@gmail.com

A. Zanini, Universidad de Buenos Aires (UBA), Buenos Aires, Argentina, azanini@fi.uba.ar

la Ec. (1). En este la entrada \mathbf{u}_k se mantiene constante dentro de cada periodo de muestreo (retención de orden cero) y éste a su vez podría ser variable.

$$\begin{cases} \mathbf{x}_{k+1} = \Phi_{i_k} \mathbf{x}_k + \Gamma_{i_k} \mathbf{u}_k \\ \mathbf{y}_k = C_c \mathbf{x}_k + D_c \mathbf{u}_k \end{cases} \quad (1)$$

El subíndice i_k en la Ec. (1) es la secuencia de conmutación y se define por la Ec. (2). Ésta define el juego de matrices Φ_i y Γ_i dentro de un conjunto \underline{P} que corresponde al instante k .

$$\begin{aligned} i_k: [0, +\infty) &\rightarrow \underline{P} \\ k &\mapsto [1, 2, \dots, p] \end{aligned} \quad (2)$$

En la Ec. (1), $\mathbf{x}_k \in \mathbb{R}^n$ representa al vector de estados, $\mathbf{u}_k \in \mathbb{R}^m$ representa al vector de entradas e $\mathbf{y}_k \in \mathbb{R}^p$ representa al vector de salidas. Por otro lado, las matrices Φ_{i_k} y Γ_{i_k} son funciones de h_{i_k} , donde $h_{i_k} = t_{k+1} - t_k$ con $k \in \mathbb{N}$ representa el tiempo entre instantes de muestreo. La función h_{i_k} toma valores definidos en el conjunto $S = \{h_1, h_2, \dots, h_p\}$. Esta asignación se observa en la Ec. (2). Ya que la cantidad de periodos de muestreo es finita entonces también lo será el número de posibles matrices Φ_{i_k} y Γ_{i_k} . El conjunto finito de matrices estará dado por la Ec. (3). Para esto se asumió que el instante en que se actualiza la salida \mathbf{y}_k coincide con la lectura de \mathbf{u}_k .

$$\begin{aligned} \Phi_{i_k} &\triangleq \Phi(h_{i_k}) = e^{A h_{i_k}} \\ \Gamma_{i_k} &\triangleq \Gamma(h_{i_k}) = \int_0^{h_{i_k}} e^{A\tau} d\tau B \\ i_k &= 1, 2, \dots, p \end{aligned} \quad (3)$$

Este sistema es un Sistema de Tiempo Discreto Conmutado (DTSS). En el caso en que el periodo de muestreo $h_{i_k} = t_{k+1} - t_k$ sea constante las matrices Φ y Γ estarán dadas por [3].

B. Control por Realimentación de Estados:

Cuando se utiliza un controlador por realimentación de estados se asume que se cuenta con información de todos ellos o con un observador de orden reducido o completo. La acción de control se genera entonces como $\mathbf{u}_k = \mathbf{K}\mathbf{x}_k$ con $\mathbf{K} \in \mathbb{R}^{m \times n}$ vector de ganancias de estado.

Con esto, la ecuación de estados en lazo cerrado queda como en la Ec. (4).

$$\mathbf{x}_{k+1} = \underbrace{(\Phi_{i_k} + \Gamma_{i_k} \mathbf{K}_i)}_{\Phi_{i_k}^{CL}} \mathbf{x}_k \quad (4)$$

Y la matriz de estados de lazo cerrado queda definida en la Ec. (5).

$$\begin{aligned} \Phi_{i_k}^{CL} &= \Phi_{i_k} + \Gamma_{i_k} \mathbf{K}_i \\ i &= 1, 2, \dots, p \end{aligned} \quad (5)$$

En la Ec. (6) se observa cómo se calcula la evolución a N pasos a partir de un estado inicial en un sistema de tiempo discreto a lazo cerrado.

$$\mathbf{x}_{k+N} = \prod_{j=k}^{k+N} \Phi_{i_j}^{CL} \mathbf{x}_k \quad (6)$$

La Ec. (6) se convierte en la Ec. (7) cuando se utiliza periodo de muestreo constante.

$$\mathbf{x}_{k+N} = \underbrace{(\Phi^{CL} \dots \Phi^{CL})}_{N \text{ veces}} \mathbf{x}_k = (\Phi^{CL})^N \mathbf{x}_k \quad (7)$$

C. Análisis de Estabilidad:

Para analizar la estabilidad de un sistema como el de la Ec. (1) se utilizará la teoría de Lyapunov. Esta teoría se puede aplicar tanto para sistemas no lineales, lineales y conmutados como los descritos en la sección anterior. Para esto se define la Función Cuadrática de Lyapunov (QLF) V_k como en la Ec. (8), con $P^T = P > 0$.

$$V_k \triangleq V(\mathbf{x}_k) = \mathbf{x}_k^T P \mathbf{x}_k \quad (8)$$

Para analizar la estabilidad hay que estudiar el comportamiento de ΔV_{k+1} . Esta variación se calcula como se muestra en la Ec. (9) (para el caso en que no hay conmutación).

$$\begin{aligned} \Delta V_{k+1} &= V_{k+1} - V_k \\ &= \mathbf{x}_{k+1}^T P \mathbf{x}_{k+1} - \mathbf{x}_k^T P \mathbf{x}_k \\ &= \mathbf{x}_k^T ((\Phi^{CL})^T P \Phi^{CL} - P) \mathbf{x}_k \end{aligned} \quad (9)$$

Para que el sistema sea estable se debe cumplir que $\Delta V_{k+1} < 0$, llevando a la Ec. (10).

$$(\Phi^{CL})^T P \Phi^{CL} - P < -Q \quad (10)$$

con $Q^T = Q > 0$.

Para el DTSS el análisis de estabilidad es diferente pues este análisis debe repetirse para cada periodo de muestreo (Ec. (11)). El conjunto de periodos de muestreo se denomina $S = \{h_1, h_2, \dots, h_p\}$ y las matrices de lazo cerrado correspondientes estarán contenidas en el conjunto $M = \{\Phi_i^{CL}, i \in [1 \dots p]\}$. Entonces se contará con un sistema de LMIs.

$$(\Phi_i^{CL})^T P \Phi_i^{CL} - P < -Q \quad (11)$$

Para que el sistema conmutado sea estable la matriz P debe ser la misma para todo $i \in \underline{P}$. La V_k hallada de esta manera se denomina Función Cuadrática de Lyapunov Común de todo el sistema (CQLF). La existencia de una CQLF es una condición suficiente para la estabilidad del sistema conmutado con secuencia de conmutación i_k arbitraria.

Partiendo de un punto inicial \mathbf{x}_0 en el espacio de estados se

define la siguiente recurrencia: $\mathbf{x}_{k+1} \in \{\Phi_{i_k}^{CL} \mathbf{x}_k, \Phi_{i_k}^{CL} \in M\}$. Con i_k se elige para cada instante k una matriz $\Phi_{i_k}^{CL}$ del conjunto M . El conjunto de todas las posibles secuencias de conmutación, i_k , define al conjunto completo de posibles trayectorias del sistema en lazo cerrado en el espacio de estados. Una trayectoria es un conjunto de vectores \mathbf{x}_k . Al conjunto de todas las secuencias posibles de vectores \mathbf{x}_k se lo denomina Inclusión Lineal Discreta (DLI). Se llama DLI(M) al conjunto de todas las posibles trayectorias del sistema.

Para que el sistema conmutado sea asintóticamente estable se debe cumplir que cualquier trayectoria \mathbf{x}_k descrita por DLI(M) cumpla: $\lim_{k \rightarrow \infty} \mathbf{x}_k = 0$. Por más que la cantidad de periodos de muestreo y por lo tanto de matrices de lazo cerrado $\Phi_{i_k}^{CL}$ sea finita ($i_k \in [1, 2, \dots, p]$) la cantidad de secuencias de conmutación es infinita y asegurar que todas ellas decaen a cero es imposible. En su lugar se utiliza el Radio Espectral Conjunto (JSR) del conjunto M , $\rho(M)$, para verificar esta condición.

Se verifica que la DLI(M) es asintóticamente estable si y sólo si $\rho(M) < 1$ [8]. No obstante, el cálculo del JSR no es sencillo. Sin embargo este se puede acotar superiormente utilizando la aproximación de la norma elipsoidal.

Se puede ver en [8] que para el caso general se cumple la Ec. (12).

$$(\Phi_i^{CL})^T P \Phi_i^{CL} \leq \mu^2 P \quad (12)$$

donde μ^2 es la aproximación de la norma elipsoidal. Esta última se puede reescribir como la Ec. (13).

$$V_{k+1} \leq \mu^2 V_k \quad (13)$$

De esta manera una condición suficiente de estabilidad del DLI(M) es que $\mu < 1$.

D. Muestreo de Lyapunov:

El muestreo de Lyapunov fue inicialmente introducido en [12]. Este esquema de muestreo se basa en la función de Lyapunov de forma de producir eventos sólo cuando se requiere desde el punto de vista de la estabilidad. Los eventos están relacionados con la variación de la función de Lyapunov entonces el próximo instante de muestreo se determina cuando se cumple la siguiente condición:

$$V_{k+1} = \eta V_k \quad (14)$$

donde $\eta \in \mathbb{R}^+$ es un parámetro que puede ser fijo como en [12] o variable como en [6], y está en el intervalo $0 < \eta < 1$ de modo que el muestreo lleve a estados de menor energía. A partir de esta expresión se puede intuir que valores de η cercanos a 1 se traducirán en periodos de muestreo pequeños y viceversa. El sistema será estable puesto que la energía decrece entre un evento y el siguiente.

A partir de la Ec. (14) se definen hiper-superficies de nivel de la función de Lyapunov en el espacio de estados y cada vez que la trayectoria se cruza con alguna de éstas se produce un evento. Siendo V_k una función cuadrática estas hiper-superfi-

cias de nivel serán hiper-elipses cuyo radio dependerá del valor de η . Para asegurar que siempre habrá muestreo, es decir, que la secuencia de muestras será infinita es necesario elegir su valor de forma adecuada. Esta correcta elección del η se da planteando la siguiente condición: $\eta^* < \eta < 1$. El cálculo de η^* se presenta en [12] aunque no es fácil de calcular por su elevado costo computacional y porque no siempre se puede asegurar su existencia para sistemas no lineales. En [6] se realizan algunas simplificaciones para permitir el cálculo en línea y así poder modificarlo de forma dinámica. Esta variación dinámica permitiría elegir un η que agrande el periodo de muestreo cuando el sistema se encuentra en equilibrio o cerca de él.

Cabe mencionar que la estabilidad del sistema controlado está garantizada incluso si el tiempo entre muestras es considerable. En este punto es importante recordar que el sistema discreto proviene de un sistema continuo muestreado y, por lo tanto, el máximo tiempo entre muestras está limitado por el Teorema de Shannon.

En [6] se analiza la respuesta ante perturbaciones y cambio en la referencia del sistema utilizando muestreo de Lyapunov. Se observa que el sistema es robusto frente a estos dado que V_k también sufre cambios. Para esto hubo que agregar una condición de seguridad que cuando detecta que $\Delta V_{k+1} > 0$ produzca un evento. Sin este agregado no se volvería a cumplir la condición de la Ec. (14).

III. DISEÑO DEL CONTROLADOR

Dada la planta en tiempo continuo junto con el conjunto $S = \{h_1, h_2, \dots, h_p\}$ de valores posibles para el periodo de muestreo, el problema consiste en encontrar un conjunto de ganancias del controlador \mathbf{K}_i con $i = 1, 2, \dots, p$, para el caso de realimentación de estados, tales que el sistema discreto conmutado equivalente de la Ec. (1) sea estable cuando se aplica el control $\mathbf{u}_k = \mathbf{K}_i \mathbf{x}_k$.

A. Diseño de controladores por Álgebra de Lie Resoluble:

El álgebra de Lie de matrices $g = \{M_i, i \in \underline{p}\}$ generada por la familia de matrices cuadradas $P = \{M_1, M_2, \dots, M_p\}$ con respecto al corchete de Lie $[M_\nu, M_\gamma] = M_\nu M_\gamma - M_\gamma M_\nu$, es el espacio generado por la aplicación iterativa de esta operación sobre las matrices de la familia [11].

Para el álgebra de Lie g , la secuencia $g^{(m)}$ se define como: $g^{(0)} = g$, $g^{(m+1)} = [g^{(m)}, g^{(m)}] \subset g^{(m)}$. Si $g^{(m)} = 0$ para algún m entonces se dice que g es *resoluble*. El parámetro m recibe el nombre de índice de resolubilidad.

Dada el álgebra de Lie g , la secuencia g^m se define como: $g^0 = g$, $g^{m+1} = [g, g^m] \subset g^m$. Si $g^m = 0$ para algún m entonces se dice que g es *nilpotente*.

La nilpotencia de un álgebra de Lie es una condición suficiente para su resolubilidad.

El álgebra de Lie $g = \{M_i, i \in \underline{p}\}$ es resoluble *sii* el álgebra de Lie $\tilde{g} = \{\tilde{M}_i, i \in \underline{p}\}$ es resoluble, donde $\tilde{M}_i = R^{-1} M_i R$, con R matriz de una transformación semejante que convierte las matrices M_i a una semejante triangular superior.

Siendo M_i las matrices de lazo cerrado del DTSS (Ec. (5)) la resolubilidad del álgebra definida con esta familia de matrices garantiza la existencia de una CQLF [11]. En la Sección II.C se vio que si una familia de sistemas que constituyen un sistema conmutado tiene una CQLF entonces se garantiza la estabilidad asintótica para cualquier secuencia de conmutación.

Si se hace que la secuencia $g^{(m)}$ generada por las matrices de lazo cerrado en P se anule para algún $m \leq n$ esto permite establecer ecuaciones de diseño de los parámetros de los controladores. Cuando se considera el índice de resolubilidad $m = 1$ que conduce a sistemas de segundo orden ($n = 2$) con una entrada existe una solución algebraica exacta para adaptar los parámetros del controlador a las matrices de la familia [8]. Esta solución está dada por las Ec. (15) y Ec. (16), donde a_{ij} son los elementos de $\Phi(h_r)$, b_i son los elementos de $\Gamma(h_r)$, c_i y d_i son los elementos de C y D respectivamente, y K_i las ganancias del controlador.

$$K_{i_1} = \frac{A}{C} \quad (15)$$

$$K_{i_2} = \frac{B}{C} \quad (16)$$

con A , B y C definidos en las siguientes ecuaciones.

$$A = K_2 c_{21} d_2 b_1 - K_2 c_{21} b_2 d_1 + c_{21} K_1 d_1 b_1 - d_1 c_{11} a_{21} + d_1 c_{22} b_2 K_1 - d_1 c_{11} b_2 K_1 + a_{12} c_{21} d_2 - d_2 c_{12} a_{21} - c_{12} b_2 K_1 d_2 + d_1 c_{22} a_{21} + c_{21} d_1 a_{11} - c_{21} d_1 a_{22}$$

$$B = -(d_1 a_{21} c_{12} + K_1 b_2 d_1 c_{12} - K_1 d_2 b_1 c_{12} - a_{12} d_2 c_{22} + b_2 K_2 d_2 c_{12} - b_1 K_2 d_2 c_{22} + b_1 K_2 d_2 c_{11} - d_2 a_{11} c_{12} - d_1 a_{12} c_{21} + a_{12} d_2 c_{11} + a_{22} d_2 c_{12} - d_1 b_1 K_2 c_{21})$$

$$C = -K_1 d_2 b_1 d_1 - b_1 K_2 d_2^2 - d_1 a_{11} d_2 + d_1^2 a_{21} + d_1 d_2 a_{22} + K_1 b_2 d_1^2 + b_2 K_2 d_2 d_1 - a_{12} d_2^2$$

Para analizar qué sucede al utilizar esta adaptación se utilizará el doble integrador de la Ec. (17) tomado de [4].

$$\begin{cases} \dot{x}(t) = \begin{bmatrix} 0 & -23.8 \\ 0 & 0 \end{bmatrix} x(t) + \begin{bmatrix} 0 \\ -23.8 \end{bmatrix} u(t) \\ y(t) = [1 \quad 0] x(t) \end{cases} \quad (17)$$

Se utiliza el controlador por realimentación de estados de la Ec. (18) diseñado con el método LQR de control óptimo [4]. Para éste se ha considerado $h = 20ms$.

$$K = [0.6739 \quad -1.3424] \quad (18)$$

En la Figura 1 se muestra la evolución de los autovalores de lazo cerrado del doble integrador al aumentar el periodo de muestreo desde $h_{min} = 10ms$ pasando por el valor nominal y llegando hasta $h_{max} = 90ms$. Se observa que para un valor de $h = 62.6ms$ los autovalores salen de la circunferencia unitaria haciendo inestable al sistema. En la Figura 2 se muestran las ventajas de la adaptación del controlador mediante las Ec. (15) y Ec. (16).

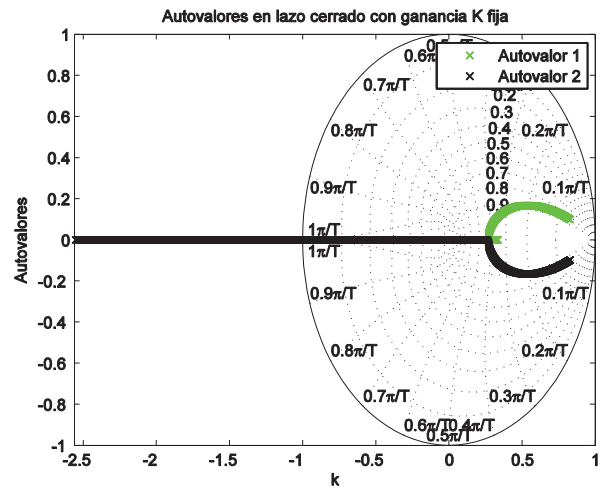


Figura 1. Autovalores del sistema de tiempo discreto a lazo cerrado al variar h entre 10ms y 90ms (sin adaptación de la ley de control).

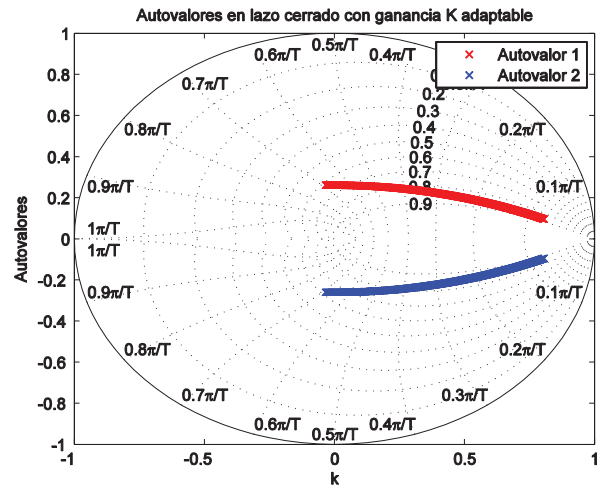


Figura 2. Autovalores del sistema de tiempo discreto a lazo cerrado al variar h entre 10ms y 90ms (con adaptación de la ley de control).

IV. AUTOMUESTREO

En la Figura 3 se muestra la idea básica del automuestreo. En este caso es el sensor quien determina el periodo de muestreo y los nodos controlador y actuador funcionan por eventos, siendo un evento la recepción de un paquete de datos. En la Figura 4 se muestra el intercambio de información entre los nodos sensor y controlador. Por otro lado, en el caso periódico el muestreo se controla por una señal de reloj que comparten todos los nodos y que nada tiene que ver con el estado del sistema. Se asumirá que tanto el procesamiento en cada nodo como el tiempo de transmisión de los paquetes de datos en la red son despreciables respecto de los periodos de muestreo considerados. Se asume también que todos los estados del sistema pueden ser medidos.

El objetivo del auto-muestreo es la utilización de los recursos del sistema de forma más eficiente. Particularmente, los WNCS llevan necesariamente a contemplar el uso de algoritmos que sean *conscientes de los recursos*.

Además del propio uso de los recursos en un lazo se puede tener más de un lazo de control compartiendo la misma red.

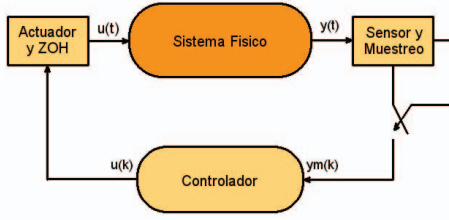


Figura 3. Esquema del Auto-Muestreo. El sensor es quien predice el próximo instante de muestreo. El controlador y el actuador funcionan por eventos.

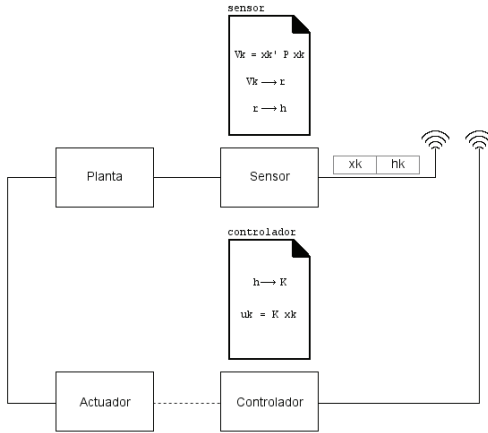


Figura 4. Información intercambiada entre los nodos.

La forma más simple de implementar un esquema de muestreo asincrónico consiste en elegir entre dos periodos de muestreo en función del error de seguimiento de una referencia. A partir de esto se puede definir un umbral de error tal que cuando la salida se encuentre lo suficientemente cerca de la referencia se utilice el periodo de muestreo más grande, y en caso contrario se conmute al periodo de muestreo más pequeño. En la Figura 5 se observa la conmutación entre periodos de muestreo para lo cual se precisa de tres parámetros: el umbral de error, h_{max} y h_{min} .

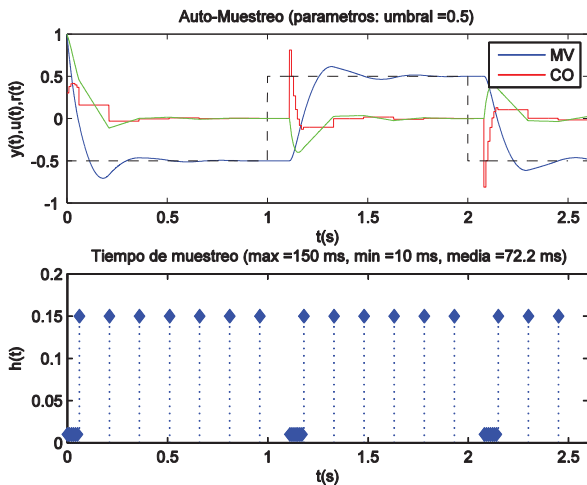


Figura 5. Selección entre dos periodos de muestreo en función de un umbral de error.

La principal desventaja de este método radica en que de querer elegir entre un conjunto mayor de periodos de muestreo es necesario agregar más umbrales complicando la implementación. A continuación se verá una forma automática de selección del periodo de muestreo que extiende esta idea al caso de automuestreo.

A. *Función de Costo:*

Utilizando la ley de adaptación del controlador por álgebra de Lie resoluble y para el caso de sistemas de segundo orden la relación entre la función de Lyapunov en el instante $k + 1$ con respecto al instante k se tiene según la Ec. (19). Esta expresión se demuestra en [8].

$$\begin{aligned}
 V_{k+1} &= \mathbf{x}_{k+1}^T P \mathbf{x}_{k+1} \\
 &= \mathbf{x}_k^T (\Phi^{CL})^T P \Phi^{CL} \mathbf{x}_k \\
 &= |\rho(\Phi^{CL}(h_i))|^2 \mathbf{x}_k^T P \mathbf{x}_k \\
 &= |\rho(\Phi^{CL}(h_i))|^2 V_k
 \end{aligned}
 \tag{19}$$

Esta forma es similar a la Ec. (14) donde η está dada por $|\rho(\Phi^{CL}(h_i))|^2$. Este valor de η se puede pensar como un porcentaje de decaimiento de V_k .

Utilizando la Ec. (19) se tendrá que para el caso periódico dada la energía del sistema en el instante t_k , para un tiempo de comparación de $T = Nh$ la energía decae según la Ec. (20). El tiempo T se divide entonces en N intervalos de ancho h , es decir, $T = h + h + \dots + h$ y en cada intervalo la energía decae $|\rho(\Phi^{CL}(h))|^2$.

$$V_{t_k+T} = V_{k+N} = |\rho(\Phi^{CL}(h))|^{2N} V_k
 \tag{20}$$

Para el caso no periódico este decaimiento estará dado por la Ec. (21) donde los h_j pertenecen al conjunto de la secuencia de muestreo y se pueden repetir o no aparecer alguno de ellos.

$$V_{t_k+T} = V_{k+N} = \prod_{i=1}^N |\rho(\Phi^{CL}(h_{i_k}))|^2 V_k
 \tag{21}$$

Entonces, de forma genérica se puede definir la función de costo dada por la Ec. (22) para comparar las respuestas obtenidas por el muestreo periódico y diversas estrategias de muestreo no periódico.

$$r_j = \prod_{i=1}^j |\rho(\Phi^{CL}(h_{i_k}))|^2
 \tag{22}$$

Esta función costo relaciona la energía inicial con la final en un intervalo de tiempo T . Dado un T fijo se podrían analizar todas las secuencias de periodos de muestreo posibles dentro de ese intervalo y los costos asociados a ellas. La de costo mayor será la que permita un mayor decrecimiento de la

energía del sistema. Sin embargo, esto conduciría en general a la elección del periodo de muestreo más pequeño en el conjunto llevando a un esquema periódico. Por esta razón, habría que encontrar un mecanismo que relacione a la función de Lyapunov con la función de costo. En la siguiente sección se abordará este tema.

La matriz P de la CQLF se puede calcular según [8]. Esta P asegura la estabilidad del DTSS ante cualquier secuencia de conmutación y su cálculo explícito se asegura si los autovalores de cada matriz en la familia (Ec. (5)) son pares complejos conjugados y estables. En la Ec. (23) se repite este cálculo de P y se ve que se utilizan dos matrices de la familia E_j y $E_k \in \mathbb{R}^{2 \times 2}$ que conmutan, es decir: $[E_j, E_k] = 0$. E_j es la matriz de lazo cerrado del sistema (Ec. (5)). Se denomina e_{ik}^j a los elementos de la matriz E_j . Se observa que sólo se utilizan los coeficientes de la matriz E_j puesto que las demás matrices de lazo cerrado consideradas conmutan con ésta.

$$P = \begin{bmatrix} -\frac{e_{21}^j}{e_{12}^j} & \frac{e_{11}^j - e_{22}^j}{2 e_{12}^j} \\ \frac{e_{11}^j - e_{22}^j}{2 e_{12}^j} & 1 \end{bmatrix} \quad (23)$$

B. Algoritmo para el cálculo del periodo de muestreo:

Se propone un conjunto S con los diferentes periodos de muestreo que se podrán utilizar para muestrear al sistema de tiempo continuo. El mayor de estos periodos de muestreo se define como el intervalo de tiempo T que funciona como un tiempo de comparación. Luego se analizan las diferentes formas en que se pueden combinar los otros periodos de muestreo en S para entrar exactamente dentro de este tiempo.

Para un conjunto de periodos de muestreo $S = \{h_1 = 10ms, h_2 = 30ms, h_3 = 50ms\}$ las diferentes formas en que se pueden ordenar para sumar $50ms$ son: $h_3 = h_1 + h_1 + h_1 + h_1 + h_1$, $h_3 = h_2 + h_1 + h_1$ y $h_3 = h_3$ (Tabla I). Cada una de estas elecciones tiene un peso asociado dado por la Ec. (22). Las secuencias $h_3 = h_1 + h_2 + h_1$ y $h_3 = h_1 + h_1 + h_2$ tienen el mismo peso que la $h_3 = h_2 + h_1 + h_1$ y por lo tanto no se consideran.

TABLA I. DIFERENTES COMBINACIONES DE LOS PERIODOS DE MUESTREO EN $S = \{10MS, 30MS, 50MS\}$. EL TIEMPO DE COMPARACIÓN T ES DE 50MS.

| h | Secuencias | Costo |
|-------|---|---|
| h_1 | $h_1 \quad h_1 \quad h_1 \quad h_1 \quad h_1$ | $ \rho(\Phi^{CL}(h_1)) ^{10}$ |
| h_2 | $h_2 \quad h_1 \quad h_1$ | $ \rho(\Phi^{CL}(h_2)) ^2 \rho(\Phi^{CL}(h_1)) $ |
| h_3 | h_3 | $ \rho(\Phi^{CL}(h_3)) ^2$ |

En la Figura 6 se muestra cómo a partir del valor de la función de Lyapunov en un instante de muestreo k se elige el valor de la función de costo más cercana que a su vez determina el valor del periodo de muestreo necesario. La selección de r a partir de V_k se realiza en el sensor (Figura 4). La elección de una recta para relacionar a la función de Lyapunov

con la función de costo se debe a que es simple y asocia correctamente los valores de una con los de la otra.

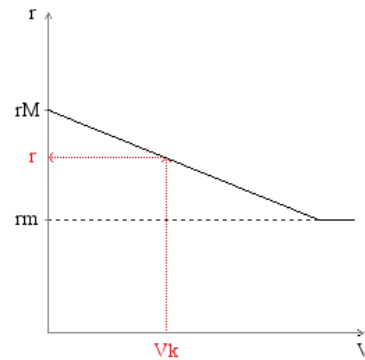


Figura 6. Método de selección de la función costo a partir del valor de la función de Lyapunov. Los valores máximo y mínimo de r son r_M y r_m respectivamente.

Una vez elegido un costo que está asociado a una secuencia de periodos de muestreo se elegirá como próximo periodo de muestreo al mayor dentro de la secuencia (primera columna de la Tabla I). De esta manera el sensor conoce de antemano el próximo instante de muestreo $k + 1$, pudiendo pasar a un estado de mínimo consumo y generando una interrupción que lo saque de este estado tan pronto como se cumpla el tiempo. En ese momento el sensor realiza las mediciones, calcula la función de Lyapunov, determina el costo y envía ambos datos al nodo controlador.

C. Ejemplo de Aplicación:

A continuación se aplicará el método de la Sección IV.B al ejemplo del doble integrador (Ec. (17)). Se define un conjunto $S = \{10ms, 30ms, 60ms, 90ms, 120ms, 150ms\}$ donde el tiempo de comparación es $h_{max} = 150ms$.

La CQLF calculada de acuerdo con la Ec. (23) se muestra en la Ec. (24).

$$P = \begin{bmatrix} 0.9905 & -0.8686 \\ -0.8686 & 1 \end{bmatrix} \quad (24)$$

La respuesta obtenida en este caso se observa en la Figura 7. Se observa que la media del periodo de muestreo está dada por $h_{med} = 139ms$ con valores máximo y mínimo dados respectivamente por $h_{min} = 60ms$ y $h_{max} = 150ms$. La respuesta para el caso de utilizar la conmutación entre dos periodos de muestreo $h_{max} = 150ms$ y $h_{min} = 10ms$ en función de si el error de seguimiento está por encima o por debajo de un umbral ($umbral = 0.5$) respectivamente se observa en la Figura 5. Usando el muestreo de Lyapunov propuesto en [12] se alcanza un periodo de muestreo promedio $h_{med} = 32.6ms$, con $h_{min} = 0.25ms$ y $h_{max} = 43.8ms$ (utilizando $\eta = \eta^* = 0.1719$). Las curvas obtenidas con este último método y con muestreo periódico no se reproducen aquí por una cuestión de espacio.

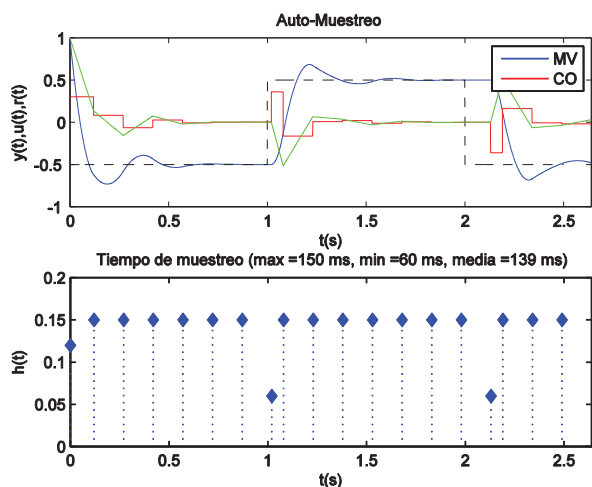


Figura 7. Automuestreo utilizando muestreo de Lyapunov y adaptación de la ley de control para un doble integrador.

En la Figura 8 se compara la evolución de la función de Lyapunov aplicando muestreo periódico, el muestreo de Lyapunov propuesto en [12], conmutación entre dos periodos de muestreo y automuestreo. Se observa que en los cuatro casos se logra el decaimiento de V , aunque en el caso del automuestreo se logra utilizando menor cantidad de muestras.

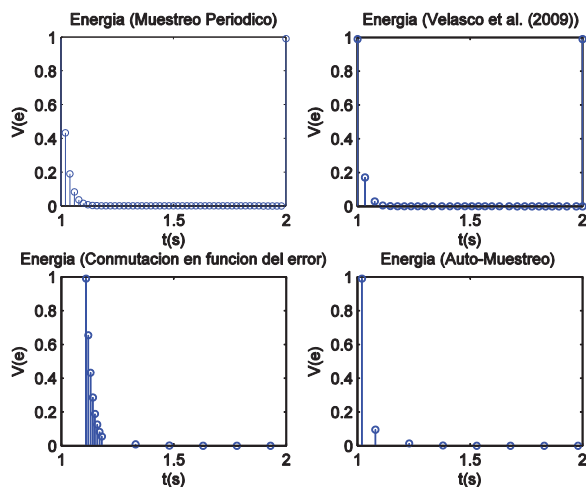


Figura 8. Evolución de la función de Lyapunov para: muestreo periódico, muestreo de Lyapunov ([12]), conmutación entre dos periodos de muestreo y automuestreo.

V. CONCLUSIONES Y TRABAJO FUTURO

El algoritmo de automuestreo basado en el muestreo de Lyapunov propuesto en este trabajo permite reducir la tasa de muestreo cuando es posible, logrando un mejor aprovechamiento de los recursos y permitiendo reducir retardos y pérdida de datos debido al menor tráfico en la red. A diferencia de otros trabajos con muestreo de Lyapunov ([12] y [6]) en los que se utiliza control por realimentación de estados con ganancia fija en este trabajo se propone la adaptación de la ley de control. Esta adaptación se realiza mediante álgebra de Lie ([8]) permitiendo asegurar que a pesar de la variación en el periodo de muestreo el sistema permanecerá estable y se podrá asegurar una dada performance. A su vez esto se consigue

independientemente de la secuencia de periodos de muestreo generada. Además los cálculos involucrados para sistemas de segundo orden son sencillos y no implican un gran costo computacional.

Esta propuesta forma parte de una cadena de trabajos cuyo fin es el análisis y diseño mediante simulación e implementación de WNCS.

Como trabajo futuro se investigarán otras funciones para vincular a la función de Lyapunov con la función de costo y con ellas las ventajas y desventajas asociadas.

AGRADECIMIENTOS

Este trabajo es parte de una tesis de doctorado en ingeniería, soportado por una beca Conicet y el subsidio UBACyT 20020100100333 otorgado por la Universidad de Buenos Aires.

REFERENCIAS

- [1] K.-E. Årzén, "A Simple Event-Based PID Controller", in Preprints 14th IFAC World Congress, vol.18, pp. 423-428, 1999.
- [2] K. J. Åström and B. Bernhardsson, "Comparison of Periodic and Event Based Sampling for First-Order Stochastic Systems", in Proc. 14th IFAC World Congress, vol. 11, pp. 301-306, 1999.
- [3] K. J. Åström and B. Wittenmark. Computer-Controlled Systems: Theory and Design, third edition. Prentice Hall. 1997.
- [4] A. Camacho, P. Martí, M. Velasco, C. Lozoya, R. Villa, J.M. Fuertes and E. Griful, "Self-Triggered Networked Control Systems: An Experimental Case Study", in IEEE International Conference on Industrial Technology (ICIT), pp. 123-128, 2010.
- [5] A. Cervin, J. Eker, B. Bernhardsson and K.-E. Årzén, "Feedback-Feedforward Scheduling of Control Tasks", Real-Time Systems, vol. 23, no. 1-2, pp- 25-53, 2002.
- [6] S. Durand, N. Marchand and J. Fermi Guerrero Castellanos, "Simple Lyapunov Sampling for Event-Driven Control" in Proc. 18th IFAC World Congress, vol. 18, pp. 8724-8730, 2011.
- [7] F. E. Felicioni, S. Junco and H. Haimovich, "Control de Sistemas Discretos LTI Conmutados en base a Métodos Lie-Algebraicos, con Aplicación al Control sobre Redes de Comunicación" in Proc. XXI Congreso Argentino de Control Automático (AADECA), 2008.
- [8] F. E. Felicioni, "Estabilidad y Performance de Sistemas Distribuidos de Control", Tesis Doctoral, Universidad Nacional de Rosario, Rosario, Argentina, 2010.
- [9] W. P. M. H. Heemels, K. H. Johansson and P. Tabuada, "An Introduction to Event-Triggered and Self-Triggered Control", in Proc. 51st IEEE Conf. Decision and Control, pp. 3270-3285, 2012.
- [10] M. Schinkel, W.-H. Chen and A. Rantzer, "Optimal Control for Systems with Varying Sampling Rate", in Proc. American Control Conference, vol. 4, pp. 2979-2984, 2002.
- [11] J. Theys, "Joint Spectral Radius: Theory and Approximations", Ph.D. thesis, Univ. de Louvain, France, 2005.
- [12] M. Velasco, P. Martí and E. Bini, "On Lyapunov Sampling for Event-Driven Controllers", in Proc. IEEE Conf. Decision Control, pp. 6238-6243, 2009.



Ezequiel Pecker Marcosig se graduó de Ingeniero Electrónico de la Universidad de Buenos Aires (UBA), Buenos Aires, Argentina, en 2011. Actualmente está realizando su doctorado en la misma universidad dentro del Instituto de Tecnologías del Hidrógeno y Energías Sostenibles (ITHES) y goza de una beca Conicet. Sus principales áreas de interés son: sistemas de control sobre redes, redes de

sensores inalámbricos y muestreo asincrónico.



Flavia Felicioni nació en Rosario, Argentina en 1976. Se graduó de Ingeniera Electrónica en la Universidad Nacional de Rosario (UNR) en mayo de 2003. Realizó su doctorado en co-tutela y en 2010 obtuvo los títulos de Doctora en Ingeniería de la Universidad Nacional de Rosario y Doctora en Informática del Institute National Polytechnique de

Lorraine (INPL), Nancy, Francia. Trabajó en docencia e investigación durante 11 años en la Universidad Nacional de Rosario y desde 2011 es Profesora en la Universidad de Buenos Aires (UBA) de Argentina. Desde 2010 se desempeña en el área de Instrumentación y Control de la Comisión Nacional de Energía Atómica (CNEA), Buenos Aires, Argentina. Sus áreas de interés son: Control de sistemas conmutados, Sistemas distribuidos de control y Sensores inalámbricos.



Anibal Zanini nació en Entre Ríos Argentina en 1954. Se graduó de Ingeniero Electricista en la Universidad Nacional de Rosario (UNR), Rosario, Santa Fe, Argentina, en 1977. Obtuvo el título de Doctor en Ingeniería en la Universidad Politécnica de Madrid (UPM), Madrid, España, en 1983. Ha trabajado durante más de 20 años en la industria en el

área de la Automática. Actualmente es Profesor en la Universidad de Buenos Aires (UBA), Argentina y miembro del Instituto de Tecnologías del Hidrógeno y Energías Sostenibles (ITHES). Sus áreas de interés son el Control de Procesos, Identificación de Sistemas y Controladores Adaptativos.