

Control en red basado en eventos: de lo centralizado a lo distribuido

María Guinaldo*, José Sánchez, Sebastián Dormido

Departamento de Informática y Automática, UNED, Juan del Rosal 16, 28040 Madrid, España

Resumen

Los sistemas de control en red (SCR) son aquellos en los que los diferentes elementos de un lazo de control (sensores, actuadores y controladores) se encuentran espacialmente distribuidos y la transmisión de información entre ellos tiene lugar a través de un canal de comunicación o red. La reducción de la cantidad de información transmitida juega un papel importante en el desempeño de estos sistemas, y reglas de comunicación no convencionales como el control basado en eventos, se han demostrado efectivas. En este artículo se revisan algunas de estas estrategias, centrándose en primer lugar en los SCR centralizados, para posteriormente estudiar esquemas de control distribuido, aplicados a sistemas de gran escala. Finalmente, algunos de los resultados teóricos se aplican al control de formaciones en un sistema de experimentación real.

Palabras Clave:

Control en red, control basado en eventos, control distribuido, sistemas de gran escala, sistemas multi-agente, control de formaciones, robots móviles.

1. Introducción

A pesar de que el mundo real es analógico, la mayoría de las aplicaciones de control se implementan en plataformas digitales que necesitan que la información en el lazo de control se intercambie de manera discreta entre los sensores, los actuadores y los controladores. Tradicionalmente, los instantes en los que este intercambio se produce son equidistantes, es decir, dados por el periodo de muestreo. La frecuencia de muestreo tiene que garantizar la estabilidad del sistema en todos los escenarios posibles, lo que a veces provoca una elección de la frecuencia conservadora. Además, todas las tareas se ejecutan de manera periódica independientemente de las necesidades de la planta.

Recientemente, ha habido un interés creciente en la idea de tener en cuenta el estado de la planta a la hora de decidir cuándo se ejecuta el control o se muestrea el sistema. En los sistemas de control basados en eventos, dichas tareas tienen lugar cuando se viola una cierta condición sobre el estado de la planta. Por tanto, hay una adaptación a las necesidades del proceso.

No hay una terminología uniforme a la hora de referirse a este concepto. Se pueden encontrar en la literatura distintas acepciones como control basado en eventos, control disparado por eventos, control *send-on-delta*, control por cruce de ni-

vel, control autodisparado o *self-triggered*, control con mínima atención, control con atención en cualquier momento, entre otras.

A pesar de su reciente popularización, el muestreo basado en eventos no es en realidad una idea nueva. Sus orígenes se remontan a finales de los años 50 cuando Ellis (1959) planteaba que el método de muestreo más apropiado consiste en transmitir solamente datos cuando existe un cambio significativo en la señal que justifique la adquisición de una nueva muestra. Más tarde, en las décadas de los 60 y los 70, se popularizó un método heurístico llamado *muestreo adaptativo*. El objetivo que persigue es reducir el número de muestras, sin que se produzca una degradación en la respuesta del sistema, evaluando en cada intervalo el periodo de muestreo (Dormido et al., 2008).

Más recientemente, en Arzén (1999) se implementa el control basado en eventos (CBE) en controladores PID mostrando que el número de acciones de control puede reducirse sin afectar al rendimiento del sistema. En Heemels et al. (1999) se utiliza el control por cruce de nivel para controlar el giro de un motor con sensores de baja resolución.

Los primeros resultados analíticos se obtuvieron para sistemas estocásticos de primer orden en Åström y Bernhardsson (2003), mostrando que bajo ciertas condiciones el CBE tiene un mejor rendimiento que el control periódico. Pero el gran impulso que ha recibido el CBE en estos últimos años tiene su motivación en la aplicación a los sistemas de control en red (SCR) o *networked control systems* (NCSs) en inglés.

* Autor en correspondencia

Correos electrónicos: mguinaldo@dia.uned.es (María Guinaldo),
jsanchez@dia.uned.es (José Sánchez), sdormido@dia.uned.es
(Sebastián Dormido)

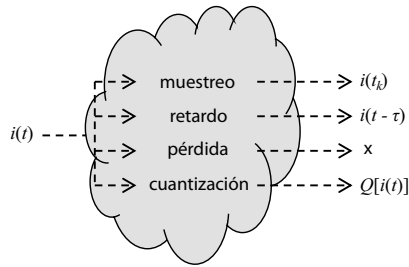


Figura 1: Los diferentes problemas que afectan a una señal $i(t)$ transmitida a través de la red.

En los SCR los diferentes elementos de un lazo de control (sensores, actuadores y controladores) se conectan a través de un medio de comunicación digital con ancho de banda limitado. El uso de redes compartidas de propósito múltiple para conectar elementos que están espacialmente distribuidos tiene varias ventajas que, por otro lado, son el principal motivo de su éxito:

- Ofrecen arquitecturas flexibles, haciendo más fácil reconfigurar partes del sistema o añadir nuevos elementos, y posibilitando la implementación de estrategias de control distribuidas.
- En general, se reducen los costes de instalación y mantenimiento, debido a la reducción del cableado que se requiere en una arquitectura punto a punto.
- Como consecuencia de ello, resulta más fácil el diagnóstico y la detección de fallos.

Los SCR han abierto también un nuevo espectro de posibles aplicaciones en el mundo real, como redes de sensores móviles (Wang et al., 2010; Zhong y Cassandras, 2011), sistemas de energía distribuidos o inteligentes o *smart grids* (Farhangi, 2010; Bolognani y Zampieri, 2011), sistemas de transporte inteligentes (Zhang et al., 2011), formación de vehículos autónomos (Millán et al., 2014), vigilancia (Basesggio et al., 2010; Kumar et al., 2015), operaciones quirúrgicas teleasistidas (Meng et al., 2004; Martini et al., 2013), entre otras.

Sin embargo, el uso de redes introduce nuevos retos y hace que el análisis y diseño de los SCR resulte más complejo. Las teorías de control convencional plantean una serie de hipótesis como la sincronización en el control, la ausencia de retardos en la medida y en la actuación y un ancho de banda ilimitado. Sin embargo, estas hipótesis han de replantearse antes de poder aplicarse a los SCR. Tratar de mejorar las redes de comunicaciones y los protocolos de red son sólo una solución parcial al problema. Por tanto, es necesario diseñar nuevos algoritmos de control con el fin de hacer frente a las imperfecciones de la comunicación y sus restricciones (Heemels, 2010), las cuales pueden resumirse en (véase la Figura 1):

- **Ancho de banda limitado:** Cualquier red puede transmitir una cantidad de información finita por unidad de tiempo, y esto puede tener graves consecuencias en los sistemas de control. En la mayoría de las redes digitales, la información es transmitida en forma de unidades atómicas

o paquetes. Dichos paquetes pueden dividirse en dos partes, el área de datos o *payload* y la cabecera, que contiene la información de control requerida para la transmisión. El máximo tamaño del área de datos depende del protocolo, y varía de los 1500 bytes en Ethernet a los 8 bytes de algunos protocolos por radio frecuencia.

- **Retardos variables en la comunicación:** La transferencia de un paquete de un nodo a otro de la red no es instantánea sino que puede suponer una cantidad variable de tiempo, que depende de condiciones variables de la red como son la congestión, la calidad del canal o el protocolo. Esto puede afectar al rendimiento del lazo de control de varias maneras. Por un lado, la información transmitida se recibe con retraso. Por otro lado, los retardos pueden suponer que el sistema sea muestreado a intervalos variables de tiempo. Un número significativo de resultados (véase Hespanha et al. (2007) y referencias de ese artículo) han tratado de caracterizar la cota superior del tiempo inter-muestreo para la cual la estabilidad del sistema está garantizada.
- **Pérdidas de paquetes:** Un paquete puede perderse por errores en la transmisión a nivel físico entre enlaces de la red, por la congestión del canal o por la corrupción de los paquetes en tránsito. En aplicaciones de control un paquete puede ser descartado si contiene información desactualizada, es decir, debido al desorden en la recepción de paquetes. Por tanto, esta cuestión ha de tenerse en cuenta en el control de flujo del tráfico de paquetes, normalmente a través de una marca de tiempo o *time stamp*.
- **Cuantización:** Un cuantizador es una función que transforma una función real en una función a tramos con un número finito de valores constantes. Aunque no se trata de un problema propio de los SCR sino de los sistemas de discretización, el tamaño finito de los paquetes puede introducir errores en las señales transmitidas al tener que reducir adicionalmente la longitud de los datos.

Estas imperfecciones son más severas en las redes inalámbricas, donde, además, hay que tener en cuenta el consumo energético, ya que muchos dispositivos están alimentados por baterías.

Dentro de las estrategias que pueden utilizarse para hacer frente a estos problemas, distinguimos dos líneas principales, las cuales no son incompatibles. Por un lado, los SCR pueden modelarse como sistemas de control en red convencionales con ciertas restricciones en la comunicación. Normalmente esto supone el re-diseño de técnicas convencionales que, además, pueden desembocar en diseños conservadores (Zhao et al., 2009). Por otro lado, puede abordarse el problema a través del *codiseño* del controlador y el protocolo o la política de comunicación. En este sentido, el control basado en eventos juega un papel importante.

En lo que respecta a la línea relativa al análisis y síntesis de controladores en red, caben destacar las siguientes soluciones:

- **Adaptar soluciones de sistemas con retardo a SCR:** Aunque el retardo inducido en un sistema en red tiene una na-

turalidad diferente a los sistemas con retardo, en algunos casos se han podido usar las herramientas para sistemas lineales muestreados para ciertas clases de SCR lineales (Nešić y Teel, 2004). En Naghshtabrizi et al. (2008), el sistema se modela a través de ecuaciones diferenciales (impulsivas) con retardo, y se obtienen condiciones de estabilidad basadas en LMI como resultado de la extensión de funcionales de Lyapunov-Krasovskii a plantas y controladores lineales.

- *Técnicas de control robusto:* El retardo y los intervalos inter-muestreo representan la incertidumbre del sistema, en la que se basa para derivar las condiciones de estabilidad a través de un LMI (Donkers et al., 2011; Millán et al., 2012).
- *Técnicas de estimación:* Para sistemas muestreados y cuando en la transmisión de información puede haber pérdidas de paquetes, se puede estimar el estado de la planta remota para paliar este efecto. En muchos casos, el control equivalente no es óptimo pero incluso así tiene un gran interés práctico (Sinopoli et al., 2004). Estas técnicas de estimación también pueden utilizarse en presencia de retardos, siendo ésta la idea del control anticipativo (Naghshtabrizi y Hespanha, 2006; Guinaldo et al., 2011).

La opción de hacer un diseño conjunto del control y la política de comunicación ha adquirido especial relevancia con la proliferación de redes inalámbricas en multitud de aplicaciones de control. En muchos casos, había un vacío de estándares en los protocolos de comunicación, lo que ha hecho proliferar múltiples propuestas para intentar alcanzar la robustez y la escalabilidad que un sistema de control con una arquitectura distribuida requiere, aunque aún no se ha dado con la solución definitiva. Por ejemplo, para el control de procesos industriales existen soluciones comerciales como WirelessHART, desarrollado por varias empresas como ABB y Siemens, entre otras, o OneWireless Network de Honeywell.

Distintos autores han abordado el problema del codiseño control-comunicación desde el punto de vista de la planificación de las tareas de muestreo y control a través de distintas técnicas de optimización. En Walsh y Ye (2001), se demostró como la planificación dinámica permitía alcanzar mejores resultados que una estática. El concepto de planificación por re-alimentación (Cervin et al., 2002) ha sido considerada tanto en sistemas deterministas como estocásticos (Marti et al., 2004; Henriksson y Cervin, 2005; Cervin et al., 2011). La planificación fuera de línea tanto de la ley de control como del acceso al medio de comunicación de múltiples plantas también ha sido analizada en Lincoln y Bernhardsson (2002); Zhang y Hristu-Varsakelis (2006); Dai et al. (2010).

Otros trabajos se han centrado en la idea del compromiso existente entre el desempeño y la tasa promedio de transmisión, y se han propuesto distintas técnicas que buscan un mejor aprovechamiento del ancho de banda reduciendo el número de envíos a través de la red a la vez que se mantiene un correcto rendimiento del sistema. En este sentido, se ha demostrado que

las técnicas basadas en eventos pueden conseguir mejores resultados que el envío periódico de información y que, al reducirse el número de envíos a través de la red, se palián en cierto sentido problemas derivados de la congestión de la red como latencias o pérdidas. Algunos ejemplos significativos son los siguientes:

- *Control send-on-delta o por cruce de nivel:* El error se define como la diferencia entre el estado en el instante de la última muestra y el estado actual de la planta y el límite o umbral de disparo es constante

$$\|e(t)\| = \|x(t) - x(t_k)\| \leq c,$$

donde t_k hace referencia al instante de tiempo del último evento y t es el tiempo presente. El valor del umbral de disparo c determina el rendimiento del sistema y la región en torno al equilibrio en la que el estado permanece confinado. Trabajos relevantes en esta categoría son (Heemels et al., 2008; Miskowicz, 2006; Sandee, 2006). De manera experimental, y en el contexto de la teleoperación, se propone en Hinterseer et al. (2008) una modificación de la política send-on-delta, introduciendo una cota relativa a la última medida.

- *Métodos basados en funciones de Lyapunov:* La estrategia anterior no asegura la estabilidad asintótica del sistema. Por este motivo, se han investigado otro tipo de reglas de disparo que permitan alcanzar esta propiedad. Un ejemplo se presenta en Tabuada (2007), en el que el error es acotado por el estado del sistema:

$$\|e(t)\| = \|x(t) - x(t_k)\| \leq \sigma \|x(t)\|.$$

Esta estrategia permite que el sistema sea asintóticamente estable pero presenta el inconveniente de que el tiempo entre eventos disminuye a medida que el sistema se acerca al equilibrio. En Tabuada (2007) se demuestra que un valor mínimo para dicho intervalo está garantizado sólo bajo ciertas condiciones. Este aspecto es importante en el control basado por eventos ya que el efecto Zeno, es decir, la ocurrencia de infinitos eventos en tiempo finito, debe evitarse. Para ello, se diseña el parámetro σ de acuerdo con ciertas propiedades de la función de Lyapunov. Otros autores han explotado la idea de usar la función de Lyapunov $V(x, t)$ directamente en la función de disparo (Mazo Jr et al., 2009): Un evento se produce cuando $V(x, t)$ alcanza una cierta cota de desempeño $S(x, t)$.

- *Funciones de disparo variables con el tiempo:* Recientemente, se han propuesto como alternativa para la estabilidad asintótica funciones de disparo variables con el tiempo. Así, en Guinaldo et al. (2013a); Seyboth et al. (2013) éstas han sido utilizadas para sistemas lineales interconectados y sistemas multi-agente, respectivamente, acotando el error de la siguiente forma

$$\|e(t)\| \leq ce^{-\alpha t}, \quad \alpha > 0,$$

que además garantiza una cota inferior para el tiempo entre eventos. Nótese que la cota del error tiende a cero

cuando $t \rightarrow \infty$. Sin embargo, se puede excluir el efecto Zeno aun bajo condiciones de la red no ideales.

- *Self-triggering*: Las redes de sensores son un caso especial de SCR en las que el consumo de energía tiene un papel fundamental. Por tanto, el muestreo basado en eventos resulta conveniente para reducir el número de emisiones. Sin embargo, tal y como se analiza en (Anta y Tabuada, 2010b; Araujo, 2014), la mayor parte de la energía consumida por el sensor se debe a la monitorización de la señal y no tanto a la transmisión. Las reglas de disparo analizadas hasta ahora requieren de la medida continua del estado. Por este motivo, una nueva estrategia conocida como *self-triggering*, o auto-disparo, ha emergido en los últimos años. Las políticas basadas en *self-triggering* calculan el tiempo t_{k+1} en el cual tendrá lugar la siguiente ejecución de las tareas relacionadas con el control en función de la medida del último estado x_k . De esta manera, el sensor no monitoriza el proceso hasta que es despertado en el instante temporal t_{k+1} , adquiere la medida, la transmite, y recalcula el tiempo de la próxima ejecución. El concepto de *self-triggering* fue sugerido por primera vez en (Velasco et al., 2003). El control por auto-disparo puede interpretarse como una emulación basada en software del control basado en eventos. Un problema inherente a este esquema es el efecto de, por ejemplo, la existencia de incertidumbre en el modelo o la aparición de perturbaciones externas desconocidas.
- *Técnicas de optimización para el tiempo/intervalo de muestreo*: Dentro de esta categoría destacamos dos propuestas. El *control por mínima atención* (MAC, de *Minimum Attention Control*) maximiza el siguiente tiempo de ejecución, mientras que garantiza un cierto nivel en el rendimiento del sistema (Anta y Tabuada, 2010a; Donkers et al., 2012). La política es similar a la de *self-triggering* en el sentido de que el objetivo es que las tareas de control sean ejecutadas lo mínimo posible pero sin utilizar técnicas de emulación. A pesar de que en Donkers et al. (2012) se propone un diseño que permite aliviar el problema de carga computacional que presentaba Anta y Tabuada (2010a), MAC es mucho menos robusto frente a retardos y perturbaciones que el control basado en eventos. El mismo tipo de problemas presenta el *control con atención en cualquier momento* (AAC, de *Anytime Attention Control*) (Greco et al., 2011; Gupta, 2009). El diseño que se propone en Anta y Tabuada (2010a) asume que después de cada ejecución de la tarea de control, la señal de control no puede ser recalculada durante un cierto intervalo de tiempo determinado por otro elemento del sistema (el planificador), calculando así la señal de control que maximiza el rendimiento del sistema en lazo cerrado.
- *Control periódico basado en eventos*: Esta estrategia, como su propio nombre indica, combina el control periódico y el control basado en eventos, y evita la continua monitorización de las salidas del sistema como en el caso del *self-triggering*, evaluando la condición de disparo única-

mente en ciertos instantes de tiempo dados por un periodo de muestreo T_s . Este diseño fue propuesto por Heemels et al. (2013), y usa funciones de disparo basadas en funciones de Lyapunov, donde como parámetros de diseño, además de σ , está T_s . Nótese que el efecto Zeno queda excluido, ya que se garantiza que el intervalo entre eventos es de, al menos, un periodo de muestreo.

- *Control basado en modelo y disparado por eventos*: Todas las estrategias descritas hasta ahora consideran que el actuador mantiene constante la señal de control entre eventos (ZOH, *Zero Order Hold*). A pesar de que bajo esta consideración de “no hacer nada” se simplifica el análisis, se ha demostrado que, si existe un modelo preciso de la planta, un generador de la señal de control puede tratar de emular el comportamiento de un sistema convencional de control en lazo cerrado para obtener un mejor rendimiento que con ZOH (Lunze y Lehmann, 2010). La idea de sacar ventaja del conocimiento de un modelo de la planta en SCR se introduce por primera vez en Montestruque y Antsaklis (2003), aunque las ejecuciones son periódicas y no basadas en eventos. No obstante, este tipo de estrategias, como la propuesta en (Lunze y Lehmann, 2010), requieren sincronización de todos los elementos en el lazo de control, lo que la hace difícilmente extensible a sistemas de control remoto y a esquemas distribuidos. Este mismo problema se presenta también en Heemels et al. (2013), aunque en este artículo se considera un marco más general con presencia de perturbaciones y sin necesidad de realimentación completa del estado.

Algunas de estas estrategias han surgido, como se ha comentado previamente, a partir de las necesidades específicas de algunas aplicaciones como las redes de sensores. En este sentido, los SCR permiten arquitecturas más flexibles, y han permitido que las arquitecturas distribuidas sean una realidad en sistemas relativamente complejos o de gran escala. Estos sistemas se pueden modelar como una colección de subsistemas heterogéneos que interactúan entre sí y que cuentan con un controlador local y capacidad de comunicación con otros. Se suelen distinguir dos tipos de interacciones: un acoplamiento físico que representa el acoplamiento de las dinámicas de un subsistema con los otros, y un acoplamiento a través de la ley de control que permite a los subsistemas o agentes alcanzar un objetivo común. A este último tipo de sistemas se le conoce como Sistemas Multi-Agente, y tienen multitud de aplicaciones (Fax y Murray, 2004) en aquellos problemas en los que se requiere una cooperación entre los agentes, como, por ejemplo, el control de formaciones en misiones de rescate; la detección, localización y neutralización de minas en el mar por vehículos autónomos subacuáticos; la exploración y el reconocimiento de áreas a través de *drones*; o la robótica móvil. En algunos de estos escenarios, por ejemplo, bajo el agua, las restricciones como los retardos, la fiabilidad y el ancho de banda son muy exigentes y, por tanto, también lo es el diseño del control y la comunicación.

En este trabajo se analizan algunas de las estrategias de control basado en eventos para sistemas de control en red. En primer lugar, se describen algunas soluciones de control en red

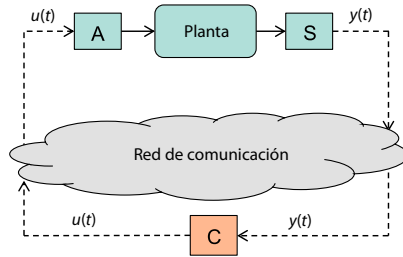


Figura 2: Sistema de control en red centralizado en la configuración de control remoto. El sensor S transmite las medidas al controlador C a través de la red, y C al actuador A.

centralizado para sistemas de pequeña dimensionalidad con un número reducido de salidas. A continuación, se revisan posibles diseños de esquemas de control distribuido para sistemas de gran escala. Finalmente, se presenta un ejemplo de la aplicación de estas técnicas al control de formaciones.

2. Sistemas de control centralizados

Desde sus orígenes, prácticamente todas las técnicas de control y estimación han sido desarrolladas para esquemas centralizados. En estos esquemas, todos los sensores y actuadores de la planta están conectados a una unidad central que recibe y procesa toda la información.

Las ventajas de las implementaciones centralizadas se han explotado durante décadas por los ingenieros de sistemas. Cuando un único agente reúne toda la información disponible del sistema, las tareas de monitorización y de control pueden alcanzar un elevado rendimiento. Además, el conocimiento existente y la gran variedad de técnicas de control desarrolladas para sistemas centralizados, hace que a la hora de realizar la implementación los profesionales puedan elegir aquella que mejor se ajuste a las necesidades del sistema.

En los SCR centralizados, existen varias configuraciones posibles dependiendo de dónde se encuentren los nodos sensores, actuadores y controladores respecto a la red. Así, el controlador puede recibir medidas directamente de los sensores y transmitir las señales de control a los actuadores a través de la red, puede conectarse directamente a los actuadores y recibir las medidas de los sensores a través de la red, o puede funcionar como un controlador remoto, donde tanto la recepción como la transmisión tienen lugar mediante el medio de comunicación. Obviamente, el caso más general corresponde a este último.

En la Figura 2 se ilustra el esquema de control remoto en el caso de tener una entrada y una salida, y cuya dinámica viene dada por

$$\dot{x}(t) = Ax(t) + Bu(t) + w(t) \quad (1)$$

$$y(t) = Cx(t) + v(t). \quad (2)$$

De esta manera, tanto la salida y como la acción de control u pueden sufrir las imperfecciones de la comunicación. Además, dado que el controlador se puede encontrar en una localización muy diferente a la de la planta, se añade otro problema adicional al control: la falta de sincronización, es decir, el reloj que mide

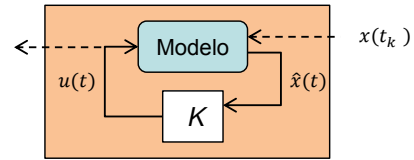


Figura 3: Esquema de un controlador basado en modelo.

el tiempo en la planta no tiene por qué ser el mismo que el del controlador. Obviamente, el efecto de este fenómeno sólo es apreciable cuando tenemos dinámicas suficientemente rápidas para que se vean influidas por el desfase en los relojes.

Sin embargo, desde el punto de vista práctico, estas limitaciones suponen un reto mayor que cuando, por ejemplo, tenemos el controlador colocado bien con el sensor, bien con el actuador. Así, en Lunze y Lehmann (2010); García y Antsaklis (2011) el esquema considerado, en el cual el controlador se comunica directamente con el actuador, permite considerar la sincronización de todos los elementos del sistema de control y, por tanto, en caso de existir retardos en la comunicación entre el sensor y el controlador, estos pueden medirse para así tenerlos en cuenta en el cálculo de las acciones de control. Algo similar sucede cuando el controlador se encuentra situado junto al sensor (Epstein et al., 2007; Quevedo et al., 2008), ya que el actuador puede conocer el retardo entre el controlador y él mismo, y seleccionar dentro de una secuencia de valores la acción de control que se ajuste a ese valor. Dentro de las posibles estrategias que se han presentado en la introducción (adaptación de técnicas existentes, control robusto y estimación), nos vamos a centrar en aquellos que utilizan técnicas de estimación, ya que ofrecen resultados menos conservadores y han sido la línea de investigación de los autores de este artículo.

2.1. Diseño de controladores

En algunos de los trabajos anteriormente citados, se utilizan técnicas basadas en modelo para estimar el estado de la planta y calcular la señal de control de manera continua en base a esa estimación:

$$\hat{x}(t) = (A + BK)x(t_k), \quad t \in [t_k, t_{k+1}) \quad (3)$$

$$u(t) = K\hat{x}(t), \quad (4)$$

donde t_k representa el último tiempo de muestreo, \hat{x} es el estado estimado y K es la ganancia por realimentación. La mayor diferencia entre los trabajos de Lunze y Lehmann (2010) y García y Antsaklis (2011) radica en cómo se determinan los tiempos de muestreo t_k del estado. En ambos casos se compara el estado actual $x(t)$ con la estimación $\hat{x}(t)$ para calcular la función de error. En el caso de Lunze y Lehmann (2010) se utiliza una variante del control send-on-delta, y se produce un evento cuando $\|x(t) - \hat{x}(t)\|$ supera una cota c , mientras que en García y Antsaklis (2011) la regla de disparo es del tipo $\|e(t)\| \geq \sigma\|x(t)\|$.

La estrategia utilizada para compensar el retardo del sensor al controlador τ_{sc} (variable con el tiempo) consiste en estimar, también en base al modelo, el estado $x(t_k + \tau_{sc})$ a partir de la

medida $x(t_k)$, y sustituir $x(t_k)$ por $\hat{x}(t_k + \tau_{sc})$ en (3). De manera similar se puede estimar el estado cuando se pierden de datos. La estrategia utilizada para compensar el retardo del sensor al controlador τ_{sc} (variable con el tiempo) consiste en estimar, también en base al modelo, el estado $x(t_k + \tau_{sc})$ a partir de la medida $x(t_k)$, y sustituir $x(t_k)$ por $\hat{x}(t_k + \tau_{sc})$ en (3). De manera similar se puede estimar el estado cuando se pierden de datos.

La extensión de estas técnicas al esquema de control remoto no es trivial. La falta de sincronización hace que no podamos medir los retardos τ_{sc} y τ_{ca} de manera independiente, y, al ser un controlador remoto, la comunicación entre el controlador y actuador no puede ser continua. Por tanto, para poder adaptar técnicas de estimación basadas en modelo, es necesario adaptar las soluciones propuestas anteriormente. Considérese la versión discreta de la planta (1)-(2) muestreada a un periodo T_s :

$$x(k+1) = Ax(k) + Bu(k) + w(k) \quad (5)$$

$$y(k) = Cx(k) + v(k). \quad (6)$$

Para mantener la estrategia de comparar las predicciones dadas por el modelo con el estado actual de la planta, es necesario que la información calculada por el controlador basado en modelo se encuentre disponible en el actuador. Para ello, se puede aprovechar la característica que presentan muchas redes tipo Internet, en las que el tamaño del paquete es muy superior a un único dato de control. De esta manera, podemos generar predicciones del estado de la planta así como futuras acciones de control y enviar esa información en un único paquete. Esa es la idea de lo que se conoce como *control en red basado en paquetes*, y que ha sido utilizado en Millán et al. (2008); Quevedo et al. (2008); Guinaldo et al. (2011).

Se define una secuencia de futuras acciones de control \mathcal{U}_k como un conjunto de N_u valores futuros de control calculados en base a un modelo de la planta (5)-(6), cuyo primer elemento se calcula en base al estado medido por el sensor a tiempo k , $x(k)$. Asimismo, la secuencia de futuros estados de la planta \mathcal{X}_k es un conjunto de N_u predicciones calculadas en base a un modelo de la planta (5)-(6). El elemento j de \mathcal{X}_k corresponde al estado correspondiente después de aplicar el elemento j de \mathcal{U}_k .

Como se ha comentado antes, cuando existen imperfecciones en la comunicación y hay falta de sincronización, no podemos medir los retardos absolutos, sino que lo que podemos estimar es lo que se conoce como *Round Trip Time* (RTT), que denotaremos como τ en general y que se define como $\tau = \tau_{sc} + \tau_{ca} + \tau_p$, donde τ_p es el tiempo de procesamiento empleado por el controlador. Este tiempo en algunos casos puede considerarse despreciable.

Existen varias estrategias en la literatura sobre cómo compensar retardos desconocidos a priori. En Guinaldo et al. (2011), se considera su valor mínimo a la hora de estimar el estado de la planta y generar así el primer elemento de \mathcal{U} y \mathcal{X} . Una vez se obtiene el valor real de RTT en el actuador, se compara con el valor mínimo y se descartan tantos elementos de \mathcal{U} y \mathcal{X} como correspondan. Otra estrategia, propuesta en Quevedo et al. (2008) consiste en generar tantas secuencias \mathcal{U} y \mathcal{X} como valores pueda tomar el retardo, en términos de periodos de muestreo. Obviamente, dado que el tamaño del paquete es fijo, la

longitud máxima que pueden tener \mathcal{U} y \mathcal{X} se reducirá en función del número de posibilidades del retardo existentes. Otra alternativa se presenta en Cuenca et al. (2011), donde se propone un predictor-observador no uniforme. Se demuestra que existe una relación entre la capacidad de predecir correctamente el retardo y el desempeño del sistema, y la incertidumbre en el retardo influye en la robustez de la solución.

El uso de control basado en paquetes permite disminuir el número de envíos a través de la red, especialmente si se combina con sensores basados en eventos (Guinaldo et al., 2014). Como alternativa, el mecanismo basado en eventos puede estar en la salida del controlador, como en Epstein et al. (2007), o en ambos elementos (Heemels y Donkers, 2013).

2.2. Análisis de estabilidad

El análisis de la estabilidad de los esquemas anteriormente presentados se basa, en general, en tres aspectos fundamentales:

- Por un lado, el control basado en eventos introduce una función de error pero, por su propia definición, está acotado, al menos en el caso de que no exista retardo.
- Los posibles retardos de la red pueden ser variables en el tiempo pero permanecen acotados. Esta cota está relacionada con cuánto crece la función de error adicionalmente.
- La pérdida de paquetes, a efectos de estabilidad, puede tratarse como un retardo adicional bajo la hipótesis de que, cuando no se recibe confirmación de recepción de un dato en un intervalo de tiempo dado por el protocolo, se retransmite.

Por ejemplo, en Lehmann y Lunze (2012) se trabaja bajo la hipótesis de que el máximo retardo de la red está acotado por el tiempo mínimo entre eventos. De esta manera, si el sensor muestrea en base a una regla de tipo send-on-delta, es decir, se detecta un evento cuando $\|e(t)\| \geq c$, bajo la hipótesis anterior se garantiza que el error permanece acotado por

$$\|e(t)\| < 2c, \quad \forall t. \quad (7)$$

Con este resultado, se puede demostrar que el estado del sistema converge a un entorno del punto de equilibrio, es decir, el sistema es *Globally Ultimately Uniformly Bounded* (GUUB). El tamaño de esta región es proporcional a la cota (7), de manera equivalente al caso en el que se considera $\tau = 0$.

Bajo hipótesis similares, en Guinaldo (2013) se demuestra que el máximo valor de τ (τ_{max}) que garantiza (7) se puede calcular usando la desigualdad $\|e^{A\tau_{max}} - I\| < 1$. Nótese que τ_{max} no depende de c , sino de la dinámica libre del sistema.

Para funciones de disparo del tipo $\|e(t)\| \geq \sigma \|x(t)\|$, el análisis es más complicado y hay que hacer un estudio de la dinámica de $\|e(t)\|/\|x(t)\|$. Bajo ciertas hipótesis, se puede obtener una solución numérica para τ_{max} (García y Antsaklis, 2011).

2.3. Influencia de las perturbaciones

En condiciones ideales, con un controlador basado en modelo bastaría un único evento para que el sistema evolucionase

hasta el equilibrio. Bajo condiciones ideales se considera la no existencia de retardos y pérdidas en la red, la ausencia de perturbaciones $w(t)$ en (1) (o $w(k)$ en el caso discreto (5)) y que el modelo reproduce idénticamente la dinámica del sistema. El primer caso ya se ha abordado en la sección anterior, por lo que a continuación nos centraremos en los otros dos aspectos.

Un método que ha sido utilizado en la literatura (Lehmann y Lunze, 2011) y que permite estimar las perturbaciones, asume que éstas permanecen constantes en el tiempo entre dos eventos consecutivos. Para ello, se incorpora en el modelo un término que incluye la estimación del modelo. Por ejemplo, en el caso discreto, la dinámica del modelo es

$$\hat{x}(k+1) = (A + BK)\hat{x}(k) + \hat{w}(k).$$

De esta manera, la dinámica del error viene dada por $e(k+1) = \hat{x}(k+1) - x(k+1) = Ae(k) + w(k) - \hat{w}(k)$. Realizando algunas transformaciones y utilizando las propiedades recursivas de los sistemas discretos, se puede obtener una estimación de las perturbaciones para el tiempo entre eventos (suponiendo que han transcurrido q periodos de muestreo) como

$$\hat{w}(k+q) = \hat{w}(k) + \left(\sum_{j=0}^{q-1} A^j \right)^{-1} (A^q e(k) - e(k+q)).$$

Esta estimación no atenúa las perturbaciones pero, sin embargo, dado que se mejora la emulación del estado del sistema, se consigue así reducir el número de eventos, es decir, el número de transmisiones a través de la red.

En el caso de que exista incertidumbre en el modelo dado por (A, B) , que se denota como $(\Delta A, \Delta B)$, su efecto es similar al caso de la existencia de perturbaciones externas, es decir, el número de eventos será mayor y no se podrá alcanzar el mismo desempeño. Sin embargo, en este caso, el incremento en la dinámica del error es dependiente del estado:

$$e(k+1) = \hat{x}(k+1) - x(k+1) = Ae(k) + (\Delta A + \Delta BK)\hat{x}(k).$$

Se puede estimar el tiempo mínimo entre eventos asumiendo una cota máxima de la incertidumbre en el modelo (García y Antsaklis, 2011).

2.4. Ejemplo

Considérese el esquema de experimentación mostrado en la Figura 4, basado en una arquitectura cliente-servidor. La planta está conectada a un equipo de computación limitada que envía y recibe comandos a través de la red. Al otro lado de la red, un controlador se encarga de realizar el procesamiento y enviar nuevas órdenes de control a la planta, bien a través de una red cableada o inalámbrica.

La planta utilizada para la experimentación es un brazo flexible de la casa Quanser (Quanser, 2016) que, para pequeñas deformaciones del brazo, puede aproximarse como el siguiente sistema lineal:

$$\begin{pmatrix} \dot{x}_1 \\ \dot{x}_2 \\ \dot{x}_3 \\ \dot{x}_4 \end{pmatrix} = \begin{pmatrix} 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 \\ 0 & 592 & 32 & 0 \\ 0 & -947,3 & -32 & 0 \end{pmatrix} \begin{pmatrix} x_1 \\ x_2 \\ x_3 \\ x_4 \end{pmatrix} + \begin{pmatrix} 0 \\ 0 \\ -56,2 \\ 56,2 \end{pmatrix} u, \quad (8)$$

$$\begin{pmatrix} y_1 \\ y_2 \end{pmatrix} = \begin{pmatrix} 1 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 \end{pmatrix} \begin{pmatrix} x_1 \\ x_2 \\ x_3 \\ x_4 \end{pmatrix}. \quad (9)$$

El sistema se muestrea con un periodo $T_s = 10$ ms. La siguiente ganancia de realimentación

$$K = (17,3205 \quad -24,7388 \quad 1,7164 \quad 0,5007),$$

fija los polos para el sistema en $(-48,13 \quad -35,34 \quad -8,43 + 11,50i \quad -8,43 - 11,50i)$.

La estrategia de control remoto basado en paquetes presentada anteriormente ha sido evaluada sobre esta planta y comparada con otros dos esquemas, es decir, se comparan estas tres situaciones:

1. Control local: La ganancia de realimentación se coloca en el lado del proceso, es decir, no hay comunicación a través de la red. El desempeño de este esquema es el que queremos alcanzar.
2. Control remoto con realimentación de estado: El control remoto está constituido por la ganancia de realimentación diseñada para el caso anterior, y envía una única señal de control al proceso. El sensor transmite las medidas con una estrategia basada en eventos del tipo send-on-delta con $c=0.1$, y en el cliente se utiliza una estrategia basada en modelo, pero solo para compensar los retardos.
3. Control remoto anticipativo: En este caso, se envía una secuencia de acciones de control de longitud $N_u = 20$, y el resto de parámetros iguales al caso anterior.

En la Tabla 1 se resume el desempeño en $y_1 = x_1$ de estos tres esquemas para un experimento de 3 s, y en la Figura 5 se muestran la salida y la señal de control. Se puede observar como en el segundo caso (en verde) la respuesta se vuelve más lenta porque cuando el actuador no recibe a tiempo nuevas acciones de control aplica $u(k) = 0$. Por otro lado, la sobreelongación y el tiempo de asentamiento en los casos 1 y 3 son similares. Si comparamos el número de eventos por el segundo y el tercer caso, vemos que el control anticipativo reduce el número de eventos en un 64%. Esto es debido a que el uso del control anticipativo disminuye la necesidad de comunicación al enviar secuencias de futuras acciones de control.

Se ha calculado también la integral del error absoluto (IAE, *Integral Absolute Error*) en las tres situaciones como $IAE = \int_{t_0}^{t_f} |e_{sp}(t)| dt$, donde $e_{sp}(t) = y_{sp} - y(t)$ e y_{sp} es la referencia de la salida. El valor del IAE aumenta con estrategias basadas en eventos debido al error estacionario que depende del valor de c del mecanismo de disparo.

3. Sistemas de control distribuidos

Las arquitecturas centralizadas requieren conectar cada dispositivo al nodo central. Esto en muchos sistemas no resulta adecuado, especialmente en el contexto de sistemas de gran escala, para los que es complicado modelar todas las interacciones y, además, requeriría una red muy potente para transmitir

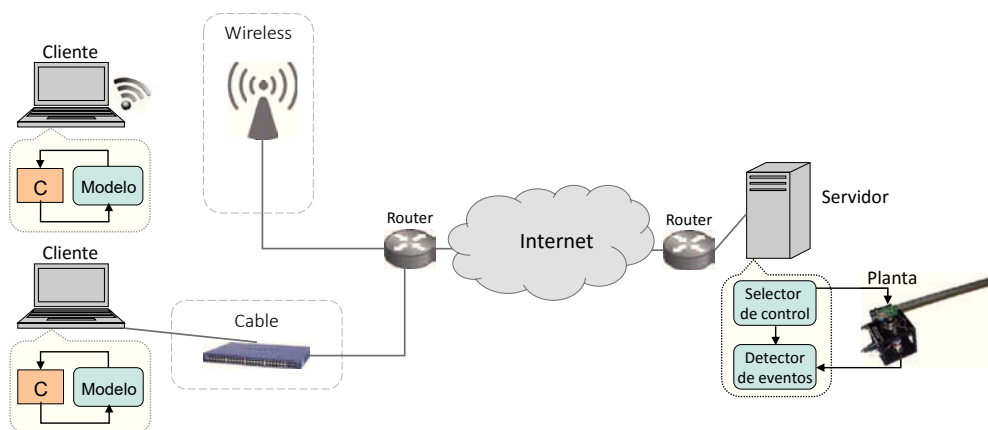


Figura 4: Esquema de experimentación basado en la arquitectura cliente-servidor. La planta de experimentación se encuentra en el lado del servidor, y el controlador en la parte del cliente.

Tabla 1: Parámetros del desempeño de los tres casos estudiados y del experimento de la Figura 5 para $y_1 = x_1$

Caso	T. subida (s)	T. asentamiento (s)	Sobreeelongación (%)	IAE	Nº Eventos
1	0.13	0.91	40.01	0.063	300
2	0.29	0.43	0.00	0.114	182
3	0.17	0.71	46.64	0.086	65

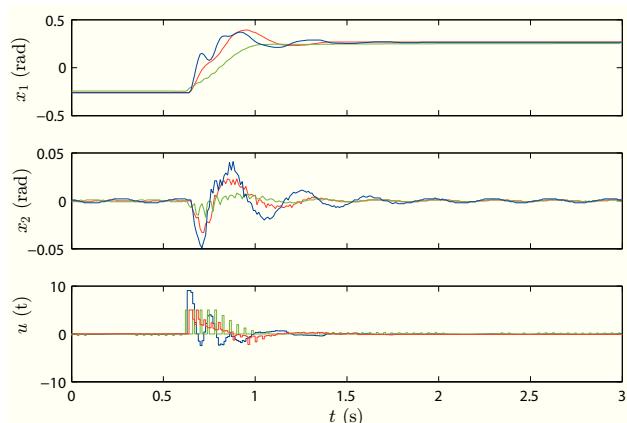


Figura 5: Comparativa del desempeño de un controlador local (azul), control remoto con $N_u = 1$ (verde) y control remoto con $N_u = 20$ (rojo).

y recibir información a y desde todos los dispositivos, respectivamente. También presenta problemas de seguridad, robustez y requiere un alto poder computacional en el controlador central. Por otro lado, un esquema descentralizado no presenta los problemas anteriores pero obvia las interacciones entre los subsistemas en los que se descompone el sistema a la hora de diseñar los controladores locales. El control distribuido supone una solución intermedia en la que existen varios controladores locales que pueden compartir información. De esta manera, se evitan los riesgos de centralizar la computación y además se puede tener en cuenta la dependencia dinámica entre los subsistemas.

Hay diferentes maneras de modelar un sistema de control distribuido. En la Figura 6 se muestra un esquema, en el que el

sistema se descompone en diferentes nodos, los cuales se comunican entre sí. Cada nodo representa tanto la dinámica del subsistema como un controlador local, que cuenta además con las funciones de comunicación con los nodos cercanos o vecinos. En un esquema basado en eventos, esta información será transmitida a instantes de tiempo determinados por la condición de eventos de manera local, es decir, el agente i monitoriza variables locales (el estado x_i , el error e_i , etc.) y, en base a ello, determina cuándo enviar información a su entorno. En general, se denotan estos instantes de tiempo locales como t_k^i con $k \in \mathbb{N}$.

Como se ha mencionado anteriormente, la interconexión entre los subsistemas puede ser fruto del acoplamiento físico (sistemas interconectados), es decir, la dinámica de un subsistema influye directamente sobre otro u otros subsistemas, o a través de la ley de control para alcanzar objetivos comunes (sistemas multi-agentes). En ambos casos, la descomposición del problema de diseño de control en problemas locales ayuda a resolverlo de manera más eficiente.

A continuación haremos una breve comparativa de los sistemas interconectados y los sistemas multi-agentes. Después se analizarán distintos mecanismos de disparo de eventos para estos sistemas y cómo se puede estudiar la estabilidad del sistema. Finalmente, se describirá brevemente cómo adaptar el formalismo a situaciones en las que retardos y pérdidas de datos pueden afectar al sistema.

3.1. Sistemas interconectados vs sistemas multi-agente

Considérese un sistema de gran escala que puede descomponerse dinámicamente de la siguiente forma

$$\dot{x}_i(t) = A_{ii}x_i(t) + B_iu_i(t) + \sum_{j \in \mathcal{N}_i} A_{ij}x_j(t), \quad i = 1, \dots, N, \quad (10)$$

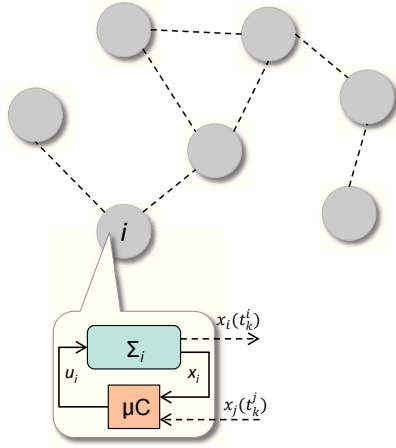


Figura 6: Esquema de un sistema de control descentralizado.

donde $x_i \in \mathbb{R}^n$ es el estado del subsistema i , $u_i \in \mathbb{R}^m$ es la señal de control local, A_{ii} y B_i representan la dinámica del agente aislado y A_{ij} define el acoplamiento físico con los nodos vecinos $j \in \mathcal{N}_i$. La información de quién está interconectado con quién se suele representar a través de un grafo \mathcal{G} , en el que los vértices \mathcal{V} del grafo son los nodos del sistema y las aristas \mathcal{E} representan precisamente esos enlaces.

La dinámica del sistema global vendrá definida de la siguiente forma

$$\dot{x}(t) = Ax(t) + Bu(t), \quad (11)$$

donde $x = (x_1^T \dots x_N^T)^T$, $u = (u_1^T \dots u_N^T)^T$, A es una matriz definida por bloques $A = \{A_{ij}\}$ y B es una matriz diagonal por bloques $B = \text{diag}(B_1, \dots, B_N)$.

El objetivo consiste en definir la ley de control local u_i de tal forma que se alcance el objetivo de control global para el sistema (11), esto es:

- Alcanzar el equilibrio en sistemas interconectados, donde $A_{ij} \neq 0$ si $j \in \mathcal{N}_i$, es decir, si el subsistema j tiene acoplamiento físico con el subsistema i . Nótese que en general esta propiedad es recíproca y, por tanto, se da la simetría $A_{ij} = A_{ji}$.
- Alcanzar un objetivo común como puede ser el consenso, la sincronización, la formación deseada, etc., en sistemas multi-agente. Se suele considerar, para simplificar el problema, que no existe acoplamiento físico ($A_{ij} = 0$ si $i \neq j$) y que las dinámicas son idénticas para todos los agentes, de tal forma que $A = I_N \otimes A_1$ y $B = I_N \otimes B_1$, donde I_N representa la matriz identidad de dimensión $N \times N$. En la Figura 7 se muestra un ejemplo de sistema que alcanza el consenso (Figura 7a), un ejemplo de sincronización de osciladores (Figura 7b) y una formación de cinco vehículos en dos dimensiones (Figura 7c).

Para ello, analizaremos ambos casos por separado:

- En sistemas interconectados la ley de control local al agente i viene dada por

$$u_i(t) = K_{ii}x_i(t) + \sum_{j \in \mathcal{N}_i} K_{ij}x_j(t), \quad (12)$$

donde K_{ii} es la ganancia de realimentación y K_{ij} representan ganancias de desacoplo. La ganancia K_i se calcula de manera que se garantice que $A_{K,i} = A_{ii} + B_i K_{ii}$ sea Hurwitz, y con K_{ij} se trata de cancelar el efecto del acoplamiento del sistema. Sin embargo, en muchas situaciones esto no es posible por la incertidumbre de estos términos o, simplemente, por propiedades algebraicas del sistema como, por ejemplo, que B_i no sea una matriz de rango completo. La diferencia entre el desacoplo perfecto y el obtenido lo denotaremos como $\Delta_{ij} = A_{ij} + B_i K_{ij}$, y la dinámica del sistema (10) con la ley de control (12) será:

$$\dot{x}_i(t) = A_{K,i}x_i(t) + \sum_{j \in \mathcal{N}_i} \Delta_{ij}x_j(t), \quad i = 1, \dots, N, \quad (13)$$

y la del sistema global $\dot{x}(t) = (A_K + \Delta)x(t)$, donde $A_K = \text{diag}(A_{K,1}, \dots, A_{K,N})$ y $\Delta = \{\Delta_{ij}\}$.

- En sistemas multi-agentes la ley de control también suele separarse en dos partes, la parte autónoma ($u_a(t)$) que solo depende del estado del agente i y la parte cooperativa ($u_c(t)$) en la que intervienen los estados de los agentes con los que i intercambia información. La parte cooperativa es la que nos va a permitir alcanzar el objetivo de control, y puede realimentar las diferencias de los estados (Olfati-Saber et al., 2007), de las salidas (Kim et al., 2011) o dotar al controlador de una estructura más compleja que haga uso de estas variables (Su y Huang, 2012). Consideremos el primer caso:

$$u_{c,i}(t) = K_c \sum_{j \in \mathcal{N}_i} (x_i(t) - x_j(t)), \quad (14)$$

donde K_c representa una ganancia que nos permite ajustar la velocidad de convergencia. Con esta ley de control, se demuestra que bajo ciertas condiciones todos los agentes convergen a un mismo estado (Olfati-Saber et al., 2007), y podemos hablar en función del caso de consenso o sincronización. Nótese que cuando esto ocurre, la señal de control cooperativa se hace cero, y el sistema se regirá por la dinámica en lazo abierto. Si se quiere corregir esto, se puede utilizar la parte autónoma u_a , por ejemplo, $u_{a,i} = K_a x_i$ para alcanzar el comportamiento deseado. De acuerdo con las definiciones de estados ampliados x y u , la ley de control se escribe de forma compacta como

$$u(t) = (I_N \otimes K_a)x(t) + (\mathcal{L} \otimes K_c)x(t) = (I_N \otimes K_a + \mathcal{L} \otimes K_c)x(t),$$

donde \mathcal{L} es la matriz laplaciana del grafo que define la topología. De esta forma, el sistema en lazo cerrado se puede escribir como

$$\dot{x}(t) = (I_N \otimes A_{K,a} + \mathcal{L} \otimes B_1 K_c)x(t),$$

donde $A_{K,a} = A_1 + B_1 K_a$.

Este desarrollo puede ser extendido para el control de formaciones, tal y como se verá en la Sección 4.

Tanto para sistemas acoplados como para sistemas multi-agente, se puede observar que la dinámica en lazo cerrado depende de

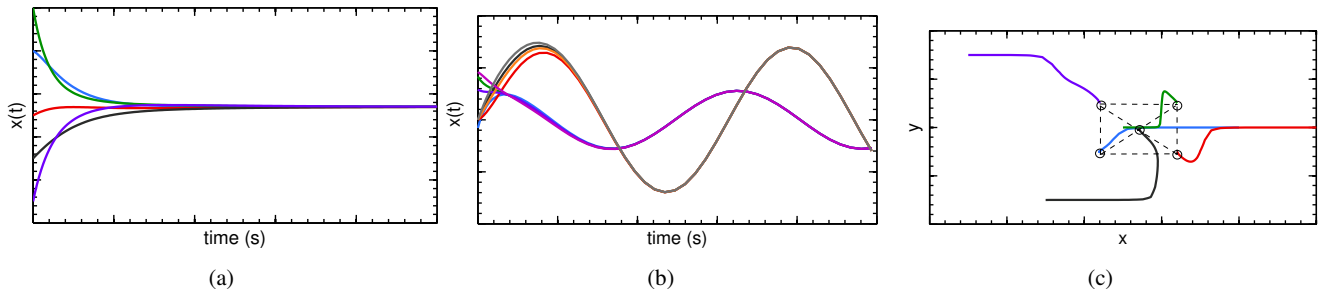


Figura 7: Ejemplos de tres objetivos de control diferentes en sistemas multi-agente.

las propiedades algebraicas de las matrices $A_K + \Delta$ y $I_N \otimes A_{K,a} + \mathcal{L} \otimes B_1 K_c$, respectivamente. Para alcanzar el comportamiento deseado, se deben ajustar las ganancias correspondientes.

Nótese que, tal y como se han definido las señales de control, éstas requieren el conocimiento de los estados en todo instante de tiempo. En un sistema en red, esto no es admisible, por lo que es necesario definir unos protocolos de comunicación que fijen los instantes de tiempo en los que esta información se envía. Lógicamente esto puede causar que el sistema no se comporte exactamente igual que cuando el estado se realimenta de manera continua. A continuación analizaremos cómo influyen las diferentes funciones de disparo de protocolos basados en eventos en el comportamiento del sistema.

3.2. Control distribuido basado en eventos

Las principales técnicas de control basadas en eventos se han presentado ya en la introducción. Éstas se pueden adaptar al control distribuido, para lo que es necesario definir una función de error local para cada agente:

- La manera más inmediata es definir el error $e_i(t) = x_{b,i}(t) - x_i(t)$, donde $x_{b,i} = x_i(t_k^i)$, $\forall t \in [t_k^i, t_{k+1}^i)$ representa el estado enviado por el agente i a sus vecinos en los tiempos de eventos t_k^i . De esta manera $x_{b,i}$ es una función constante a tramos y el error se monitoriza de manera continua y permanece acotado de acuerdo con la función de disparo.
- De manera parecida a como en el caso centralizado, se puede tratar de aumentar el tiempo entre eventos a través de un modelo. En ese caso existen dos alternativas que preservan la propiedad distribuida: utilizar sólo un modelo de la dinámica local a cada nodo, es decir, obviando los acoplamientos con otros subsistemas, o asumir que cada agente tiene además un modelo de cómo influye el resto del sistema en la dinámica del agente (Stocker y Lunze, 2013). En este último caso se requiere que, cada vez que se produce un evento en el subsistema i , se haga una petición a su entorno para que le envíe su estado actual, aumentando así el número de transmisiones a través de la red. Como en el caso centralizado, se producirá un evento cuando la estimación comparada con la medida real alcance la cota dada por la función de disparo.
- Otra posible definición de la función de error es a través de la ley de control, siendo $e_i(t) = u(t_k^i) - u(t)$, $\forall t \in$

$[t_k^i, t_{k+1}^i)$. Mientras que en sistemas centralizados la definición del error en la variable de entrada o en la salida puede ser equivalente para controladores con realimentación de estados, en el caso de sistemas distribuidos no lo es, ya que en el cálculo de la señal de control intervienen los estados de otros subsistemas. Sin embargo, si se consideran leyes de control del tipo (12) o (14), su monitorización continua requiere el conocimiento en todo instante de tiempo de los estados x_j , haciendo que en la práctica no sea viable. Como alternativa, se puede considerar un esquema basado en eventos en dos capas: una primera capa que monitoriza un error $e_{x,i}$ en la variable de estado x_i y con una secuencia de tiempos de eventos denotados por t_k^i , y una segunda capa en la que se define una función de error $e_{u,i}$ en la señal de control, que se evalúa únicamente en los tiempos de eventos del nodo i o cuando se recibe información de los vecinos. De esta forma, hay una segunda secuencia de tiempos de eventos que será un subconjunto de t_k^i . Así se evitan, además, los posibles casos de la actualización demasiado frecuente de la ley de control en nodos que tienen un número elevado de interconexiones (Guinaldo et al., 2013b).

En cualquiera de los casos anteriores, el mecanismo de control basado en eventos introduce una perturbación respecto a la dinámica del sistema continuo que se traduce en añadir nuevas restricciones al diseño para garantizar la estabilidad del sistema. Veamos esta idea con un ejemplo. Considérese el caso de sistemas interconectados y variables de error basadas en el estado. En ese caso, la dinámica del sistema global es

$$\dot{x}(t) = (A_K + \Delta)x(t) + B_K e(t),$$

donde $e(t) = (e_1^T(t) \dots e_N^T(t))^T$ y B_K es una matriz por bloques donde los bloques diagonales son $B_i K_{ii}$ y los de fuera de la diagonal se corresponden con $B_i K_{ij}$. Nótese que se cumple que $\|e(t)\| \leq \sqrt{N} \max\{\|e_i(t)\|\}$.

Por ejemplo, para condiciones de eventos del tipo $\|e_i(t)\| > c e^{-\alpha t}$, se demuestra que el sistema resulta asintóticamente estable y se garantiza un tiempo mínimo entre eventos si se cumple que el parámetro α de la función de disparo está restringido como $\alpha < |\lambda_{\max}(A_K) - \kappa(V)| \|\Delta\|$, donde $\lambda_{\max}(A_K)$ es la parte real del autovalor de A_K más próxima a cero y $\kappa(V) = \|V\| \|V^{-1}\|$ es el número de condición de la matriz V de autovectores de A_K (Guinaldo et al., 2013a).

Condiciones similares se pueden establecer para sistemas multi-agentes (Seyboth et al., 2013; Guinaldo et al., 2016). Si se consideran otras funciones de disparo, como las del tipo $\|e_i(t)\| > \sigma \|x_i(t)\|$, también se obtienen restricciones sobre el valor de σ (Wang y Lemmon, 2008, 2011; De Persis et al., 2013).

3.3. Efectos de la red

En el análisis anterior se ha considerado que el tiempo de detección de un evento, la transmisión y recepción de datos se hace con un retardo despreciable y que la fiabilidad de la red es del 100 %. Estas hipótesis se ajustan poco a la realidad sobre todo en los sistemas inalámbricos. En los sistemas distribuidos, además de los problemas comentados para los sistemas centralizados, tiene lugar lo que se conoce como *inconsistencia* de la información. Dado que los subsistemas comparten su estado con los nodos vecinos y que esta información se utiliza en la ley de control, puede ocurrir que la versión del último estado transmitido $x_{b,i}$ no sea la misma en todos los nodos, ya que cada transmisión puede sufrir un retardo diferente, puede que unos datos sean recibidos correctamente y otros no, etc. Para controlar estas inconsistencias, es necesario diseñar protocolos de comunicación que implementen un control de flujo.

A priori, existen dos posibilidades:

- Un protocolo que garantice que la información utilizada por un nodo y sus vecinos es la misma. Para ello, cada vez que el nodo i genera un evento y produce un envío, espera a recibir confirmación de la recepción (señal ACK de *acknowledgement*) durante un cierto tiempo de espera T_W . En caso de no haber recibido la confirmación de todos los vecinos, retransmite de nuevo, y así sucesivamente hasta que haya recibido confirmación de que el dato se recibió correctamente $\forall j \in \mathcal{N}_i$. Una vez esto ocurre, el agente i envía una señal que habilita el dato (denotada por PERM y que se asume es recibida con un retardo despreciable), de tal manera que tanto i como sus vecinos actualizan el valor de $x_{b,i}$ en la ley de control de manera simultánea. De este modo, se puede aplicar el mismo formalismo que para redes ideales, aunque haya que tener en cuenta que ya no se puede garantizar que el error permanece acotado según la función de disparo, sino que el error habrá crecido más cuanto mayor haya sido el tiempo entre la detección del evento hasta el envío de la señal PERM.
- Asumir que puede haber una discrepancia entre nodos ya que cada receptor actualiza la ley de control cada vez que recibe nueva información, pero que el emisor es capaz de discernir a qué nodos tiene que retransmitir y a cuáles no en función de si ha recibido o no la señal ACK. De esta manera, se admite que el valor de $x_{b,i}$ pueda no ser el mismo en todos los nodos de $j \in \mathcal{N}_i$ y, por tanto, es necesario definir variables de error locales adicionales $e_{ij}(t) = x_{b,ij}(t) - x_i(t)$, donde $x_{b,ij}(t)$ representa el último valor de $x_{b,i}$ que fue enviado por i y recibido satisfactoriamente por j . Para este protocolo el formalismo se complica, sin embargo los resultados obtenidos son menos conservadores, es decir, en general, el error total crecerá menos que en el caso anterior.

En cuanto a la influencia de la estabilidad, como es lógico, la existencia de retardos y pérdidas de datos introduce condiciones más restrictivas en el diseño. Por ejemplo, en el caso analizado anteriormente, se demuestra que hay una correlación entre el retardo máximo de la red y los parámetros de la función de disparo c y α según una ecuación del tipo

$$\left(\frac{k_1}{c} + \frac{k_2}{c} + \frac{k_3}{c}(1 + e^{\alpha\tau_{max}})\right)\tau_{max} = e^{-\alpha\tau_{max}},$$

donde k_1 , k_2 y k_3 son constantes que dependen de las propiedades dinámicas del sistema y el diseño de control (Guinaldo et al., 2014).

4. Ejemplo de aplicación: control de formaciones de robots móviles

El control de formaciones distribuido es una aplicación del control cooperativo, en el que los distintos agentes (en este caso agentes físicos) intercambian información relativa a su posición para alcanzar la formación deseada. La manera de describir el objetivo a alcanzar se puede dar en forma de posiciones relativas a un punto o de posiciones relativas entre los agentes.

Tal y como se presentó en la sección anterior, leyes del tipo (14) hacen converger al sistema a un estado común a partir de unas condiciones iniciales. Esta ley se puede generalizar para que, en lugar de alcanzar un estado común, se alcance la formación deseada. La manera más sencilla es introducir un vector de constantes r_{ij} , que representan las distancias relativas entre los agentes i y j que se desean alcanzar:

$$u_{c,i}(t) = K_c \sum_{j \in \mathcal{N}_i} (x_i(t) - x_j(t) - r_{ij}). \quad (15)$$

De esta forma, la ley de control se hace cero cuando la diferencia entre x_i y x_j alcanza r_{ij} , $\forall j$.

Sin embargo, la dinámica de la mayoría de los vehículos móviles y, por tanto, la de los robots, presenta no linealidades y se hace necesario adaptar el formalismo para este tipo de sistemas. A continuación, se hará una breve descripción del modelo considerado para los robots autónomos y cómo conseguir este objetivo, para posteriormente presentar algunos resultados experimentales sobre un entorno real.

4.1. Modelo no-holónico para robots móviles

En robótica, las restricciones holónicas hacen referencia a la relación entre las variables controladas y el número total de grados de libertad. Si el número de grados de libertad controlables es inferior al número total, el vehículo es no-holónico.

El esquema de un robot no-holónico se muestra en la Figura 8. La distancia entre las ruedas traseras y las delanteras se denota por d . En este caso, se asume una única rueda delantera. Las ecuaciones dinámicas vienen dadas por

$$\begin{aligned} \dot{x}_i &= v_i \cos \theta_i \\ \dot{y}_i &= v_i \sin \theta_i \\ \dot{\theta}_i &= \omega_i, \end{aligned}$$

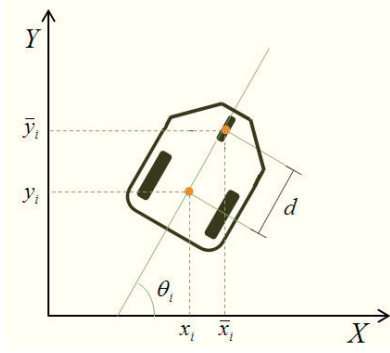


Figura 8: Esquema de de un vehículo no-holónico.

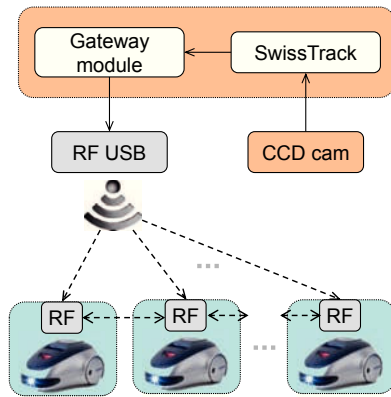


Figura 9: Diagrama de bloques del marco experimental.

donde (x_i, y_i) es la posición en el plano del agente i , θ_i representa la orientación, v_i la velocidad lineal, ω_i la velocidad angular, F_i la fuerza, T_i la torsión, m_i la masa y J_i el momento de inercia.

Se puede definir un cambio de variables $\bar{x}_i = x_i + d \cos \theta_i$, $\bar{y}_i = y_i + d \sin \theta_i$ (véase la Figura 8), y con las leyes de control adecuadas, se puede escribir la dinámica del sistema sin las restricciones holonómicas.

Así por ejemplo, considérese el caso simplificado de tomar v_i y ω_i como entradas de control. Si la ley de control viene dada por

$$\begin{pmatrix} v_i \\ \omega_i \end{pmatrix} = \begin{pmatrix} \cos \theta_i & -d \sin \theta_i \\ \sin \theta_i & d \cos \theta_i \end{pmatrix}^{-1} \begin{pmatrix} \sum_{j \in \mathcal{N}_i} (\bar{x}_j - \bar{x}_i - (r_{x,j} - r_{x,i})) \\ \sum_{j \in \mathcal{N}_i} (\bar{y}_j - \bar{y}_i - (r_{y,j} - r_{y,i})) \end{pmatrix},$$

la dinámica del robot en lazo cerrado en términos de las nuevas coordenadas es

$$\begin{pmatrix} \dot{\bar{x}}_i \\ \dot{\bar{y}}_i \end{pmatrix} = \begin{pmatrix} \sum_{j \in \mathcal{N}_i} (\bar{x}_j - \bar{x}_i - (r_{x,j} - r_{x,i})) \\ \sum_{j \in \mathcal{N}_i} (\bar{y}_j - \bar{y}_i - (r_{y,j} - r_{y,i})) \end{pmatrix},$$

que es equivalente a las evolución de un integrador con el control por consenso generalizado a formaciones (15).

4.2. Resultados

Marco experimental. El marco experimental utilizado es parte de un laboratorio remoto desarrollado para enseñar robótica, cuya descripción detallada puede encontrarse en Fabregas et al.

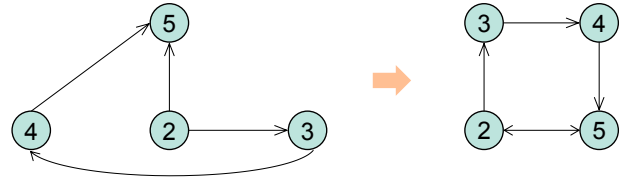


Figura 10: Esquema de la topología de comunicación, formación inicial (izquierda) y formación deseada (derecha).

(2015). En la Figura 9 se muestra un esquema del laboratorio, cuyos componentes principales son los siguientes:

- **Robots Moway:** Se trata de pequeños robots autónomos de bajo coste que cuentan con un microcontrolador, dos servomotores, una batería y distintos sensores como un acelerómetro, de luz, de infrarrojos, etc. Otro componente importante es el módulo de radio frecuencia (RF) que permite la comunicación inalámbrica con otros dispositivos de RF.
- **Cámara CCD:** Se encuentra situada en el techo del laboratorio y captura el vídeo que será procesado por un software para obtener las posiciones de los robots.
- **SwissTrack:** Esta aplicación de código abierto, desarrollado por la Escuela Politécnica de Lausana (EPFL), para hacer el seguimiento de uno o varios objetos cuando recibe como entrada las imágenes procedentes de una cámara. De esta manera, se obtienen los valores de x_i , y_i y θ_i para cada robot, que serán enviados via TCP/IP.
- **Módulo Gateway:** Esta aplicación corre en el mismo ordenador que *SwissTrack* y se encarga del procesamiento de las medidas y de su transmisión a través del dispositivo USB de RF a cada robot.

Descripción del experimento. Considérese un sistema con cuatro robots, etiquetados como 2, 3, 4 y 5. En la Figura 10 se muestra la topología de la comunicación así como la formación inicial y la deseada. A pesar de que el grafo tiene enlaces dirigidos, se puede alcanzar el consenso ya que existe un camino que conecta cualquier par de nodos (Olfati-Saber et al., 2007).

Las condiciones iniciales vienen dadas por $x(0) = (0 \ 0 \ -50 \ 0)^T$, $y(0) = (0 \ 0 \ 0 \ 40)^T$ y $\theta(0) = (198 \ 280 \ 262 \ 189)^T$, y las componentes del vector de posiciones relativas deseadas son

$$\begin{aligned} r_x &= (-20 \ -20 \ 20 \ 20)^T \\ r_y &= (-20 \ 20 \ 20 \ -20)^T. \end{aligned} \quad (16)$$

Comunicación basada en eventos frente a comunicación periódica. El experimento descrito previamente se ha realizado en dos escenarios: cuando la comunicación entre los robots es periódica con $T_s = 250$ ms y cuando está basada en eventos. El sistema de medida y de comunicación impone una restricción sobre T_s y no debe ser inferior a 200 ms. La función de disparo es de tipo constante con $c = 4$ cm. Un valor mayor produciría un error excesivo, y un valor inferior estaría reñido con la precisión dada por la cámara.

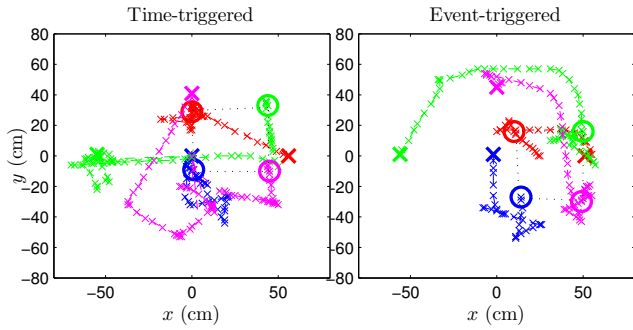


Figura 11: Representación en el plano de las trayectorias de cada robot para comunicaciones periódicas (izquierda) y basadas en eventos (derecha). Agente 2 (azul), agente 3 (rojo), agente 4 (verde), agente 5 (magenta).

Tabla 2: Número de eventos generados por cada robot.

Robot	n° eventos	n° envíos
2	26	52
3	36	36
4	54	54
5	49	49
Total:	165	191

Los resultados para ambas estrategias se muestran en la Figura 11. En ambos casos, se alcanza la formación pero el centro de la misma es diferente a pesar de que las condiciones iniciales son las mismas. Esto ilustra el efecto de las comunicaciones reales ya que las trayectorias se ven afectadas por los retardos y las pérdidas de conexión o de datos.

Si comparamos el número de comunicaciones necesitadas para alcanzar la formación en uno y otro caso, se obtienen unos resultados claramente mejores en el caso del disparo por eventos. Los resultados se muestran en la Tabla 2 para este último caso. El número de envíos hace referencia a que cuando se produce un evento, el robot realizará tantas transmisiones del mismo dato como vecinos tenga. Para el caso de comunicación periódica, hay que tener en cuenta el tiempo transcurrido hasta que se alcanza la formación y el número de robots y de enlaces existentes, es decir:

$$\left(\frac{t_f - t_0}{T_s} + 1\right) \times (n^{\circ} \text{ robots} + n^{\circ} \text{ enlaces}) = 89 \times (4 + 5) = 801,$$

donde t_0 es el tiempo inicial y t_f es el tiempo final.

Análisis del consumo energético. El número de transmisiones está relacionado con el consumo de energía. El módulo de RF de los robots Moway tiene las siguientes características: Corriente de transmisión I_t de 11.3 mA; corriente de recepción I_r de 12.3 mA; voltage medio V_{RF} de 2.75 V; duración estimada de la transmisión/recepción δt_{RF} de 10 ms.

Nótese que el número de recepciones y transmisiones han de ser consideradas de manera independiente para calcular el consumo energético, ya que cuando se produce un evento, se transmite un paquete del módulo de USB al robot, y luego este

dato será a su vez transmitido a los vecinos. Por tanto, el consumo energético se puede estimar como

$$\begin{aligned} E_{ET} &= V_{RF} \delta t_{RF} (n^{\circ} \text{ eventos} \times I_r + n^{\circ} \text{ envíos} \times (I_r + I_t)) \\ &= 2,75[\text{V}] \cdot 0,01[\text{s}] \cdot (165 \cdot 0,123[\text{A}] \\ &\quad + 191 \cdot (0,123[\text{A}] + 0,113[\text{A}])) = 1,8[\text{J}]. \end{aligned}$$

De acuerdo con lo anterior, para el caso periódico el consumo energético viene dado por

$$\begin{aligned} E_{TT} &= 2,75[\text{V}] \cdot 0,01[\text{s}] \cdot (801 \cdot 0,123[\text{A}] + 89 \cdot 5 \cdot 0,113[\text{A}]) \\ &= 4,1[\text{J}]. \end{aligned}$$

Por tanto, la energía consumida se reduce en un 56 % en el caso de comunicaciones basadas por eventos.

Sin embargo, cabe plantearse la siguiente cuestión: ¿Puede ocurrir que una reducción produzca un aumento en el consumo de energía por las señales de control aplicadas? Las v_i y ω_i obtenidas en ambos casos se muestran en la Figura 12. Las señales de control calculadas son escaladas y transformadas por las librerías propias de los robots antes de aplicar los voltajes que correspondan sobre los servomotores del robot. Por tanto, es complicado hacer un cálculo exacto del consumo energético, pero sí es posible comparar la eficiencia de las dos estrategias. Para ello, se calculan los siguientes parámetros:

$$W_{v_i} = \int_{t_0}^{t_f} |v_i(t)| dt, \quad W_{\omega_i} = \int_{t_0}^{t_f} |\omega_i(t)| dt. \quad (17)$$

En la Tabla 3 se resumen los resultados obtenidos para cada uno de los robots. Como era de esperar, las comunicaciones basadas en eventos no suponen, en general, un incremento de W_{v_i} y W_{ω_i} . Además, como las señales de control se calculan con menos frecuencia, se produce además una reducción en la carga del microprocesador, reduciendo este aspecto también el consumo energético.

5. Conclusiones

En este artículo se han revisado distintas estrategias de control basado en eventos y su aplicabilidad a los sistemas de control en red. Para esquemas centralizados con un único controlador remoto, se ha visto como se puede reducir la frecuencia de transmisión y hacer frente a fenómenos de la red como los retardos y pérdidas de datos. Para sistemas de larga escala, en los que la aproximación centralizada no es robusta y en muchos casos tampoco viable, se han estudiado algunas estrategias de comunicación y control basadas en eventos desde dos enfoques diferentes: los sistemas interconectados y el control cooperativo en sistemas multi-agente. Para este último caso, se ha ilustrado con un ejemplo de aplicación real la eficiencia de estas técnicas en cuanto a la reducción de comunicación y el ahorro en el consumo energético se refiere.

English Summary

Event-based control for networked systems: From centralized to distributed approaches

Robot	Periódico		Basado en eventos	
	W_{v_i}	W_{ω_i}	W_{v_i}	W_{ω_i}
2	404.43	35.35	285.50	22.28
3	119.68	11.47	166.55	13.89
4	658.92	66.16	360.01	36.85
5	375.92	35.29	332.80	44.80

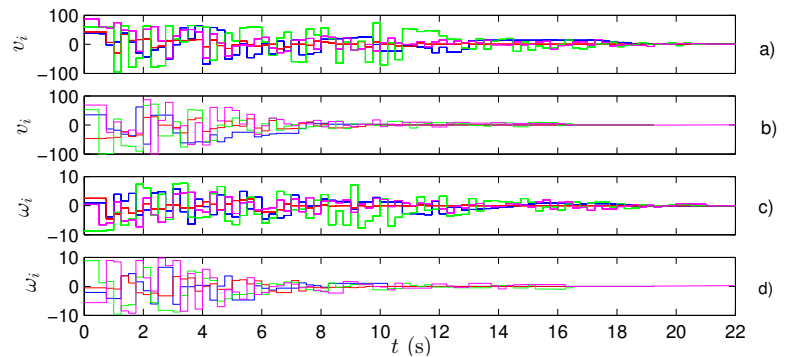


Tabla 3 & Figura 12: W_{v_i} y W_{ω_i} con comunicaciones periódicas y basadas en eventos para cada robot basadas en las señales de control que se muestran en la Figura: a) v_i caso periódico, b) v_i basado en eventos, c) ω_i caso periódico, d) ω_i basado en eventos. Agente 2 (azul), agente 3 (rojo), agente 4 (verde), agente 5 (magenta).

Abstract

Networked control systems (NCSs) are spatially distributed systems in which sensors, actuators and controllers exchange information through a communication channel or network. The reduction in the amount of transmitted information has a relevant impact over the system's performance. In this regard, non-conventional communication rules, such as the event-based control, have been demonstrated to be effective. In this paper, some of these strategies are reviewed. We first focus on centralized NCSs, and then the distributed control for large-scale systems is studied. Finally, some of the results are applied to the formation control problem and implemented over an experimental setup.

Keywords: Networked control, event-based control, distributed control, large-scale system, multi-agent system, formation control, mobile robot

Agradecimientos

Este trabajo ha sido financiado por el Ministerio de Economía y Competitividad bajo los proyectos DPI2012-31303 y DPI2014-55932-C2-2-R, y por la UNED a través del proyecto 2014-007-UNED-PROY.

Referencias

- Anta, A., Tabuada, P., 2010a. On the minimum attention and anytime attention problems for nonlinear systems. En: 49th IEEE Conference on Decision and Control. Atlanta, pp. 3234–3239.
- Anta, A., Tabuada, P., 2010b. To sample or not to sample: self-triggered control for nonlinear systems. IEEE Transactions on Automatic Control 55 (9), 2030–2042.
- Araujo, J., 2014. Design, implementation and validation of resource-aware and resilient wireless networked control systems. Ph.D. thesis, KTH Royal Institute of Technology.
- Årzén, K., 1999. A simple event-based pid controller. En: IFAC World Congress. Beijing, pp. 423–428.
- Baseggio, M., Cenedese, A., Merlo, P., Pozzi, M., Schenato, L., 2010. Distributed perimeter patrolling and tracking for camera networks. En: 49th IEEE Conference on Decision and Control. Atlanta, pp. 2093–2098.

- Bolognani, S., Zampieri, S., 2011. A gossip-like distributed optimization algorithm for reactive power flow control. En: IFAC World Congress. Milano, pp. 5700–5705.
- Cervin, A., Eker, J., Bernhardsson, B., Årzén, K.-E., 2002. Feedback-feedforward scheduling of control tasks. Real-Time Systems 23 (1-2), 25–53.
- Cervin, A., Velasco, M., Martí, P., Camacho, A., 2011. Optimal online sampling period assignment: Theory and experiments. Control Systems Technology, IEEE Transactions on 19 (4), 902–910.
- Cuenca, A., García, P., Albertos, P., Salt, J., 2011. A non-uniform predictor-observer for a networked control system. International Journal of Control, Automation and Systems 9 (6), 1194–1202.
- Dai, S.-L., Lin, H., Ge, S. S., 2010. Scheduling-and-control codesign for a collection of networked control systems with uncertain delays. Control Systems Technology, IEEE Transactions on 18 (1), 66–78.
- De Persis, C., Sailer, R., Wirth, F., 2013. Parsimonious event-triggered distributed control: A zeno free approach. Automatica 49 (7), 2116–2124.
- Donkers, M., Heemels, W., van de Wouw, N., Hetel, L., 2011. Stability analysis of networked control systems using a switched linear systems approach. IEEE Transactions on Automatic Control 56 (9), 2101–2115.
- Donkers, M., Tabuada, P., Heemels, W., 2012. Minimum attention control for linear systems: A linear programming approach. Discrete Event Dynamic Systems Theory and Applications DOI 10.1007/s10626-012-0155-x.
- Dormido, S., Sánchez, J., Kofman, E., 2008. Muestreo, control y comunicación basados en eventos. Revista Iberoamericana de Automática e Informática Industrial IRAI 5 (1), 5–26.
- Ellis, P., 1959. Extension of phase plane analysis to quantized systems. IRE Transactions on Automatic Control 4 (2), 43–54.
- Epstein, M., Shi, L., Di Cairano, S., Murray, R. M., 2007. Control over a network: Using actuation buffers to reduce transmission frequency. En: European Control Conference. Kos.
- Fabregas, E., Farias, G., Dormido-Canto, S., Guinaldo, M., Sánchez, J., Dormido-Bencomo, S., 2015. Platform for teaching mobile robotics. Journal of Intelligent & Robotic Systems, 1–13.
- Farhangi, H., 2010. The path of the smart grid. Power and energy magazine, IEEE 8 (1), 18–28.
- Fax, J., Murray, R., 2004. Information flow and cooperative control of vehicle formations. IEEE Transactions on Automatic Control 49 (9), 1465–1476.
- García, E., Antsaklis, P. J., 2011. Model-based event-triggered control with time-varying network delays. En: 50th IEEE Conference on Decision and Control. Orlando, pp. 1650–1655.
- Greco, L., Fontanelli, D., Bicchi, A., 2011. Design and stability analysis for anytime control via stochastic scheduling. IEEE Transactions on Automatic Control 56 (3), 571–585.
- Guinaldo, M., 2013. Contributions to networked and event-triggered control of linear systems. Ph.D. thesis, ETSI Informática, UNED.
- Guinaldo, M., Dimarogonas, D., Johansson, K., Sánchez, J., Dormido, S., 2013a. Distributed event-based control strategies for interconnected linear systems. IET Control Theory & Applications 7 (6), 877–886.
- Guinaldo, M., Lehmann, D., Sanchez, J., Dormido, S., Johansson, K. H., 2013b.

- Reducing communication and actuation in distributed control systems. En: Decision and Control (CDC), 2013 IEEE 52nd Annual Conference on. IEEE, pp. 5288–5293.
- Guinaldo, M., Sánchez, J., Dormido, S., 2011. A co-design strategy of NCS for treacherous network conditions. *IET Control Theory & Applications* 5 (16), 1906–1915.
- Guinaldo, M., Sanchez, J., Dormido, S., 2014. Anticipative control design for output measurement in internet-like networks. En: Control Conference (ECC), 2014 European. pp. 1193–1198.
- Guinaldo, M., Sánchez, J., Dormido, S., 2016. Distributed adaptive control of linear multi-agent systems with event-triggered communications. *Applied Mathematics and Computation* 274, 195–207.
- Gupta, V., 2009. On an anytime algorithm for control. En: 47th IEEE Conference on Decision and Control. Cancun, pp. 6218–6223.
- Heemels, W.P.M.H. & van de Wouw, N., 2010. *Networked Control Systems*. Vol. 406. Springer-Verlag, Ch. Stability and stabilization of networked control systems, pp. 203–253.
- Heemels, W., Donkers, M., 2013. Model-based periodic event-triggered control for linear systems. *Automatica* 49 (3), 698–711.
- Heemels, W., Donkers, M., Teel, A. R., 2013. Periodic event-triggered control for linear systems. *Automatic Control, IEEE Transactions on* 58 (4), 847–861.
- Heemels, W. P. M. H., Gorter, R. J. A., van Zijl, A., van den Bosch, P. P. J., Weiland, S., Hendrix, W. H. A., Vonder, M. R., 1999. Asynchronous measurement and control: a case study on motor synchronization. *Control Engineering Practice* 7 (12), 1467–1482.
- Heemels, W. P. M. H., Sandee, J., van den Bosch, P., 2008. Analysis of event-driven controllers for linear systems. *International Journal of Control* 81 (4), 571–590.
- Henriksson, D., Cervin, A., 2005. Optimal on-line sampling period assignment for real-time control tasks based on plant state information. En: Decision and Control, 2005 and 2005 European Control Conference. CDC-ECC'05. 44th IEEE Conference on. pp. 4469–4474.
- Hespanha, J. P., Naghshtabrizi, P., Y., X., 2007. A survey of recent results in networked control systems. *Proceedings of the IEEE* 95 (1), 138–162.
- Hinterseer, P., Hirche, S., Chaudhuri, S., Steinbach, E., Buss, M., 2008. Perception-based data reduction and transmission of haptic data in telepresence and teleaction systems. *IEEE Transactions on Signal Processing* 56 (2), 588–597.
- Kim, H., Shim, H., Seo, J. H., 2011. Output consensus of heterogeneous uncertain linear multi-agent systems. *Automatic Control, IEEE Transactions on* 56 (1), 200–206.
- Kumar, N., Lee, J.-H., Rodrigues, J., 2015. Intelligent mobile video surveillance system as a bayesian coalition game in vehicular sensor networks: learning automata approach. *Intelligent Transportation Systems, IEEE Transactions on* 16 (3), 1148–1161.
- Lehmann, D., Lunze, J., 2011. Extension and experimental evaluation of an event-based state-feedback approach. *Control Engineering Practice* 19 (2), 101–112.
- Lehmann, D., Lunze, J., 2012. Event-based control with communication delays and packet losses. *International Journal of Control* 85 (5), 563–577.
- Lincoln, B., Bernhardsson, B., 2002. Lqr optimization of linear system switching. *Automatic Control, IEEE Transactions on* 47 (10), 1701–1705.
- Lunze, J., Lehmann, D., 2010. A state-feedback approach to event-based control. *Automatica* 46 (1), 211–215.
- Marti, P., Lin, C., Brandt, S., Velasco, M., Fuertes, J. M., et al., 2004. Optimal state feedback based resource allocation for resource-constrained control tasks. En: Real-Time Systems Symposium, 2004. Proceedings. 25th IEEE International. pp. 161–172.
- Martini, M., Hewage, C. T., Nasralla, M., Smith, R., Jourdan, I., Rockall, T., 2013. 3-d robotic tele-surgery and training over next generation wireless networks. En: Engineering in Medicine and Biology Society (EMBC), 2013 35th Annual International Conference of the IEEE. IEEE, pp. 6244–6247.
- Mazo Jr, M., Anta, A., Tabuada, P., 2009. On self-triggered control for linear systems: Guarantees and complexity. En: Control Conference (ECC), 2009 European. pp. 3767–3772.
- Meng, C., Wang, T., Chou, W., Luan, S., Zhang, Y., Tian, Z., 2004. Remote surgery case: robot-assisted teleneurosurgery. En: IEEE International Conference on Robotics and Automation. Vol. 1. Barcelona, pp. 819–823.
- Millán, P., Jurado, I., Vivas, C., Rubio, F. R., 2008. Networked predictive control of systems with data dropouts. En: 47th IEEE Conference on Decision and Control. pp. 2704–2709.
- Millán, P., Orihuela, L., Bejarano, G., Vivas, C., Áamo, T., Rubio, F., 2012. Design and application of suboptimal mixed H_2/H_∞ controllers for networked control systems. *IEEE Transactions on Control Systems Technology* 20 (4), 1057–1065.
- Millán, P., Orihuela, L., Jurado, I., Rubio, F. R., 2014. Formation control of autonomous underwater vehicles subject to communication delays. *IEEE Transactions on Control Systems Technology* 22 (2), 770–777.
- Miskowicz, M., 2006. Send-on-delta concept: an event-based data reporting strategy. *sensors* 6 (1), 49–63.
- Montestruque, L. A., Antsaklis, P., 2003. On the model-based control of networked systems. *Automatica* 39, 1837–1843.
- Naghshabrizi, P., Hespanha, J., 2006. Anticipative and non-anticipative controller design for network control systems. *Networked Embedded Sensing and Control* 331, 203–218.
- Naghshabrizi, P., Hespanha, J. P., Teel, A. R., 2008. Exponential stability of impulsive systems with application to uncertain sampled-data systems. *Systems & Control Letters*, 378–385.
- Nešić, D., Teel, A. R., 2004. Input-to-state stability of networked control systems. *Automatica* 40 (12), 2121–2128.
- Olfati-Saber, R., Fax, J., Murray, R., 2007. Consensus and cooperation in networked multi-agent systems. *Proceedings of the IEEE* 95 (1), 215–233.
- Quanser, 2016. Rotary flexible link. Website, http://www.quanser.com/products/rotary_flexible_link.
- Quevedo, D. E., Silva, E. I., Goodwin, G. C., 2008. Control over unreliable networks affected by packet erasures and variable transmission delays. *IEEE Journal on Selected Areas in Communications* 26 (4), 672–685.
- Åström, K., Bernhardsson, B., 2003. Systems with lebesgue sampling. En: Rantzer, A., Byrnes, C. (Eds.), *Directions in Mathematical Systems Theory and Optimization*. Vol. 286 of Lecture Notes in Control and Information Sciences. Springer Berlin / Heidelberg, pp. 1–13.
- Sandee, J., 2006. Event-driven control in theory and practice. Ph.D. thesis, Technische Universiteit Eindhoven.
- Seyboth, G., Dimarogonas, D., Johansson, K., 2013. Event-based broadcasting for multi-agent average consensus. *Automatica* 49 (1), 245–252.
- Sinopoli, B., Schenato, L., Franceschetti, M., Poolla, K., Jordan, M., Sastry, S. S., et al., 2004. Kalman filtering with intermittent observations. *Automatic Control, IEEE Transactions on* 49 (9), 1453–1464.
- Stocker, C., Lunze, J., 2013. Distributed event-based control of physically interconnected systems. En: Decision and Control (CDC), 2013 IEEE 52nd Annual Conference on. IEEE, pp. 7376–7383.
- Su, Y., Huang, J., 2012. Cooperative output regulation of linear multi-agent systems. *Automatic Control, IEEE Transactions on* 57 (4), 1062–1066.
- Tabuada, P., 2007. Event-triggered real-time scheduling of stabilizing control tasks. *IEEE Transactions on Automatic Control* 52 (9), 1680–1685.
- Velasco, M., Martí, P., Fuertes, J. M., 2003. The self triggered task model for real-time control systems. En: 24th IEEE Real-Time Systems Symposium (RTSS03). Cancun.
- Walsh, G. C., Ye, H., 2001. Scheduling of networked control systems. *Control Systems, IEEE* 21 (1), 57–65.
- Wang, J., Ghosh, R. K., Das, S. K., 2010. A survey on sensor localization. *Journal of Control Theory and Applications* 8 (1), 2–11.
- Wang, X., Lemmon, M. D., 2008. Event-triggered broadcasting across distributed networked control systems. En: American Control Conference, 2008. IEEE, pp. 3139–3144.
- Wang, X., Lemmon, M. D., 2011. Event-triggering in distributed networked control systems. *Automatic Control, IEEE Transactions on* 56 (3), 586–601.
- Zhang, J., Wang, F.-Y., Wang, K., Lin, W.-H., Xu, X., Chen, C., 2011. Data-driven intelligent transportation systems: A survey. *Intelligent Transportation Systems, IEEE Transactions on* 12 (4), 1624–1639.
- Zhang, L., Hristu-Varsakelis, D., 2006. Communication and control co-design for networked control systems. *Automatica* 42 (6), 953–958.
- Zhao, Y., Liu, G., Rees, D., 2009. Design of a packed-based control framework for networked control systems. *IEEE Transactions on Control Systems Technology* 17 (5), 859–865.
- Zhong, M., Cassandras, C., 2011. Distributed coverage control and data collection with mobile sensor networks. *Automatic Control, IEEE Transactions on* 56 (10), 2445–2455.