



UNIVERSIDAD NACIONAL
AUTÓNOMA DE
MÉXICO

UNIVERSIDAD NACIONAL AUTÓNOMA DE MEXICO

**PROGRAMA DE MAESTRIA Y DOCTORADO EN
INGENIERIA**

**CODISEÑO DE SISTEMAS DE CONTROL EN RED
COMPENSANDO IMPERFECCIONES ACOTADAS DE TIEMPO
INDUCIDAS POR LA RED**

T E S I S

QUE PARA OPTAR POR EL GRADO DE

DOCTOR EN INGENIERIA

ELECTRICA – CONTROL

P R E S E N T A :

PAUL ERICK MÉNDEZ MONROY

TUTOR:
DR. HÉCTOR BENÍTEZ PÉREZ

2012

JURADO ASIGNADO :

Presidente: Dr. Álvarez Icaza Longoria Luis Agustín

Secretario: Dr. Fridman Leonid

Vocal: Dr. Benítez Pérez Héctor

1er Suplente: Dr. Álvarez Ramírez José de Jesús

2do Suplente: Dr. Ortega Arjona Jorge Luis

Lugar donde se realizó la tesis:

IIMAS, Ciudad Univesitaria, México, Distrito Federal

TUTOR DE TESIS:

Dr. Héctor Benítez Pérez

FIRMA

Índice General

1.	INTRODUCCIÓN	1
1.1	MOTIVACIÓN	1
1.2	OBJETIVO.....	3
1.3	ALCANCES	3
1.4	METAS.....	3
1.5	CONTRIBUCIÓN.....	4
1.6	METODOLOGÍA	4
1.7	ESTRUCTURA DE LA TESIS.....	5
2.	ANTECEDENTES	7
2.1	SISTEMAS CONTROLADOS POR COMPUTADORA.....	7
2.2	REDES DE COMUNICACIÓN	14
2.3	CONTROL DIFUSO	21
2.4	CONCLUSIONES.....	¡ERROR! MARCADOR NO DEFINIDO.
3.	ESTADO DEL ARTE	29
3.1	RETARDOS EN NCS.....	30
3.2	PÉRDIDA DE PAQUETES.....	38
3.3	MÚLTIPLES IMPERFECCIONES DE RED	40
3.4	CONCLUSIONES.....	¡ERROR! MARCADOR NO DEFINIDO.
4.	CASOS DE ESTUDIO	65
4.1	CASO SISO. LEVITACIÓN MAGNÉTICA	65
4.2	CASO MIMO. HELICÓPTERO 2 DOF.....	67
4.3	CONCLUSIONES.....	70
5.	MODELO DE IMPERFECCIONES DE RED.....	43
5.1	MODELO DEL RETARDO IDA Y VUELTA (<i>ROUND TRIP TIME</i> , RTT).....	43
5.2	MODELO DE RETARDOS DE TIEMPO Y PÉRDIDA DE PAQUETES	46
5.3	MODELO DE IMPERFECCIONES DE TIEMPO.....	48
5.4	CONCLUSIONES.....	¡ERROR! MARCADOR NO DEFINIDO.
6.	CODISEÑO DIFUSO PARA NCS.....	53
6.1	MODELO DIFUSO	53
6.2	CONTROLADOR DIFUSO	57
6.3	PLANIFICADOR DIFUSO.....	60
6.4	CONCLUSIONES.....	63
7.	RESULTADOS.....	71
7.1	SISTEMA DE LEVITACIÓN MAGNÉTICA	71
7.2	SISTEMA HELICÓPTERO 2-DOF CONTROLADO POR TIEMPO.....	81
7.3	CODISEÑO DEL SISTEMA HELICÓPTERO 2-DOF	91
7.4	CONCLUSIONES.....	105
8.	CONCLUSIONES	107
9.	REFERENCIAS	110

Lista de Figuras

2.1. SISTEMA DE CONTROL EN RED BÁSICO	10
2.2. REPRESENTACIÓN DEL RETARDO DE TIEMPO EN UN NCS.....	10
2.3. REPRESENTACIÓN DE PÉRDIDA DE PAQUETES EN UN NCS.....	12
2.4. REPRESENTACIÓN DEL EFECTO DEL MUESTREO VARIANTE EN EL TIEMPO EN UN NCS.	13
2.5. REPRESENTACIÓN DEL EFECTO DE CUANTIZACIÓN EN UN NCS.....	13
2.6. TRAMA DEL PROTOCOLO CAN.....	16
2.7. TRAMA DEL PROTOCOLO CONTROLNET.....	17
2.8. FORMATO DE TRAMA PARA PROTOCOLO CSMA/CD (ETHERNET)	20
4.1. CONFIGURACIÓN DEL SISTEMA MAGLEV EN RED ETHERNET.	66
4.2. ESQUEMA Y DIAGRAMA DE CUERPO LIBRE DEL MAGLEV.....	67
4.3. HELICÓPTERO CONTROLADO EN RED CON TRÁFICO.....	68
5.1. CONFIGURACIONES NCS COMUNES CON MEDICIÓN DE RETARDOS RTT CON RED ETHERNET.	44
5.2. DIAGRAMA DE TIEMPOS DE PAQUETES ENVIADOS EN UN NCS.	47
5.3. MODELO ONE-SHOT.....	50
6.1. CONCEPTO BÁSICO PARA LA ELECCIÓN DE LOS MODELOS DISCRETOS CON MÚLTIPLES PERÍODOS.	57
6.2. FUNCIONES DE MEMBRESÍA CON CUATRO PERÍODOS DE MUESTREO.	¡ERROR! MARCADOR NO DEFINIDO.
6.3. CONFIGURACIÓN DE CODISEÑO	61
7.1. RETARDO RTT PARA EL EXPERIMENTO SIN TRÁFICO EXTERNO.	72
7.2. RETARDOS RTT PARA LA COMUNICACIÓN DEL NCS Y TRÁFICO LIGERO CON PAQUETES DE 64 BYTES CADA 1 MS.	73
7.3. RETARDOS RTT PARA EL NCS CON TRÁFICO MEDIO, PAQUETES DE 64 Y 512 BYTES CADA 1 MS.	74
7.4. HISTOGRAMA DE LOS RETARDOS RTT MEDIDOS PARA LOS TRES ESCENARIOS.	75
7.5. RETARDO RTT (+) Y ESTIMACIÓN RTT (•) PARA LOS TRES EXPERIMENTOS.....	76
7.6. FUNCIONES DE MEMBRESÍA NORMALIZADAS PARA EL MODELO DIFUSO MAGLEV.	77
7.7. COMPARATIVO ENTRE EL MODELO DIFUSO Y LA SALIDA MEDIDA (SUPERIOR), ERROR DE MODELADO (INFERIOR)	78
7.8. DESEMPEÑO DEL SISTEMA LIBRE DE TRÁFICO EN LA RED (ARRIBA), ERROR DE CONTROL (ABAJO).....	79
7.9. DESEMPEÑO DEL SISTEMA CON TRÁFICO LIGERO EN LA RED (ARRIBA), ERROR DE POSICIÓN (ABAJO).	80
7.10. DESEMPEÑO DEL SISTEMA CON TRÁFICO MEDIO EN LA RED (ARRIBA), ERROR DE POSICIÓN (ABAJO).....	80
7.11. PERÍODO DE MUESTREO VS RETARDOS DE TIEMPO	82
7.12. RETARDO DE TIEMPO VS TRÁFICO.....	82
7.13. PERÍODO DE MUESTREO VS GRADO DE DISPARO NORMALIZADO PARA EL HELICÓPTERO 2DOF	84
7.14. POSICIÓN PITCH DEL HELICÓPTERO 2DOF SIN TRÁFICO EN LA RED.	85
7.15. RETARDOS DE TIEMPO (A) Y TIEMPO PERDIDO (B) DEL NCS SIN TRÁFICO.	86
7.16. RESPUESTA DE PITCH CON 1.17% DE PAQUETES PERDIDOS Y RETARDOS DE TIEMPO DE [0.0005 – 0.035]	87
7.17. RETARDOS DE TIEMPO (A) Y PÉRDIDA DE PAQUETES (B) CON 6 NODOS DE TRÁFICO.	88
7.18. RESPUESTA DE LA POSICIÓN DE PITCH CON 12 NODOS DE TRÁFICO.	89
7.19. RETARDOS DE TIEMPO Y PÉRDIDA DE PAQUETES EN LA RED CON 12 NODOS DE TRÁFICO.	89
7.20. DESEMPEÑO IAE PARA EL CONTROL DIFUSO (—) Y CONTROL RETROALIMENTADO (...)	90
7.21. PERIODOS DE MUESTREO VARIABLE PARA TRAFICO A) LIGERO, B) MEDIO Y C) SEVERO.....	91
7.22. DESEMPEÑO DEL CODISEÑO Y SEÑAL DE CONTROL SIN TRÁFICO EXTERNO A LA RED.	93
7.23. COMPORTAMIENTO DE LA RED SIN TRÁFICO EXTERNO PARA EL CODISEÑO NCS.....	94
7.24. EAM, PLAZOS PERDIDOS Y NUEVO PERÍODO PARA EL NCS SIN TRÁFICO EXTERNO.	95
7.25. DESEMPEÑO DEL CODISEÑO DIFUSO Y CONTROL HIBRIDO CON TRÁFICO LIGERO (50 – 80 s).	96
7.26. SEÑAL DE CONTROL DEL CODISEÑO DIFUSO Y CONTROL HIBRIDO CON TRÁFICO LIGERO.....	97
7.27. RETARDOS DE TIEMPO Y PÉRDIDA DE PLAZOS CON TRÁFICO LIGERO DE 50 – 80 s.	98
7.28. DESEMPEÑO SISTEMA – RED, Y COMPORTAMIENTO DEL PERÍODO DE MUESTREO ANTE TRÁFICO LIGERO.	99
7.29. DESEMPEÑO DEL CODISEÑO DIFUSO Y CONTROL HIBRIDO CON TRÁFICO MEDIO DE 50 – 80 s.	100
7.30. SEÑAL DE CONTROL DEL CODISEÑO DIFUSO Y CONTROL HIBRIDO CON TRÁFICO MEDIO.	100
7.31. RETARDOS DE TIEMPO Y PÉRDIDA DE PLAZOS CON TRÁFICO MEDIO DE 50 – 80 s.	101
7.32. DESEMPEÑO DEL SISTEMA – RED CON TRÁFICO MEDIO Y PERÍODO DE MUESTREO DEL PLANIFICADOR DIFUSO... ..	102
7.33. DESEMPEÑO DEL CODISEÑO DIFUSO Y CONTROL HIBRIDO CON TRÁFICO SEVERO DE 50 – 80 s.	103

7.34. SEÑAL DE CONTROL DEL CODISEÑO DIFUSO Y CONTROL HIBRIDO CON TRÁFICO SEVERO.	103
7.35. RETARDOS DE TIEMPO Y PÉRDIDA DE PLAZOS CON TRÁFICO SEVERO DE 50 – 80 s.....	104
7.36. DESEMPEÑO DEL SISTEMA – RED CON TRÁFICO SEVERO Y PERÍODO DE MUESTREO DEL PLANIFICADOR DIFUSO.	105

Lista de Tablas

2.1. COMPARATIVA DE LAS REDES DE CONTROL MÁS EMPLEADAS.....	21
7.1. ESTADÍSTICAS DEL RETARDO RTT PARA MÚLTIPLES ESCENARIOS.	74
7.2. COMPARATIVA DIFUSO VS RETROALIMENTADO	91

Lista de Publicaciones

REVISTAS INDEXADAS

- Quiñones-Reyes, P., **Méndez Monroy E.**, Ortega-Arjona J., Benitez-Perez H. & Duran-Chavesti A.; “*Fuzzy Control Design for a Class of Nonlinear Network Control System, considering a Helicopter Case Study*”; International Journal of Computers, Communications and Control, Vol. 7(2), pp. 363-374, 2012. ISSN: 1841-9836. (0.65)
- **Méndez-Monroy P.**, and Benitez-Perez H.; “*Neuro-Fuzzy Control with Time Delay Estimation for Nonlinear Networked Control Systems*”; International Journal of Innovative Computing, Information and Control (IJICIC), Vol.7(7), pp. 4435-4450, 2011. ISSN: 1349-4198. (1.667)
- **Méndez-Monroy E.**, and Benitez-Perez H.; “*Fuzzy Control with Estimated Variable Sampling Period for Nonlinear Networked Control Systems*”; Transactions of the Institute of Measurement and Control, 21 de Septiembre 2011. ISSN: 0142-3312. doi: 10.1177/0142331211411505. (0.462)

REVISTAS ARBITRADAS

- O. Esquivel-Flores, H. Benitez-Perez, **P. E. Méndez-Monroy**, Antonio Menendez; “*Frequency Transition for scheduling management using dynamic system approximation for a kind of NCS*”; Innovative Computing, Information and Control, Express Letters B, Vol. 1(1), pp. 93-98, 2010. ISSN: 2185-2766.
- **Méndez-Monroy E.** and Benitez-Perez H.; “*Supervisory Fuzzy Control for Networked Control Systems*”; Innovative Computing, Information and Control, Express Letters, Vol. 3(2), pp. 233-240, 2009. ISSN: 1881-803X.

CONFERENCIAS NACIONALES

- **Méndez-Monroy P.E.**, Benítez-Pérez H., Velasco M., Fuertes J. M.; “*Fuzzy control of network control Systems with periodic actuation tasks*”; Congreso 2011 Asociación de México de Control Automático (AMCA); Saltillo, México, 2011.
- **Méndez-Monroy P.E.**, and Benitez-Perez H.; “*Control Difuso con Estimación del Retardo de Tiempo para Sistemas de Control en Red*”; Congreso 2010 Asociación de México de Control Automático (AMCA); Puerto Vallarta, México, 2010.

CONFERENCIAS INTERNACIONALES

- **Méndez-Monroy P.**, and Benitez-Perez H.; “*fuzzy Observer based Fault Detection for Network Control Systems with Periodic Actuation Tasks*”; 8th IFAC Symposium on Fault Detection, Supervision and Safety of technical Process (SAFEPROCESS); Ciudad de México, México, 2012.
 - **Méndez-Monroy P.**, and Benitez-Perez H.; “*Codesign Strategy based upon Fuzzy Control for Networked Control Systems and a Scheduling Algorithm*”; IEEE International Conference on Networking, Sensing and Control (ICNSC); Delft, Holanda, pp. 221-226, 2011, ISBN: 978-1-4244-9570-2.
 - **Méndez-Monroy P.E.**, Benítez-Pérez H., Quiñones-Reyes P., Cárdenas-Flores F., García-Nocetti F.; “*Reconfigurable fuzzy networked control following LMI based on a structural reconfiguration algorithm considering time delays*”; IEEE International Conference on Industrial Technology (ICIT); Victoria, Australia , pp. 78-83, 2009. ISBN: 978-1-4244-3506-7.
 - **Méndez-Monroy P.E.**, Benítez-Pérez H.; “*Codesign strategy based upon supervisory fuzzy control for networked control systems and real time computing*”; 8th IFAC International conference on Fieldbuses and Networks in Industrial and Embedded Systems, vol. 8(1); Hanyang Univesity, Corea del Sur, 2009.
-

-
- **Méndez-Monroy P.E.**, Benítez-Pérez H.; “*Fuzzy Control with Time Delay for Networked Control Systems*”; 10th European Control Conference (ECC); Budapest, Hungaria, 2009.
 - **Méndez-Monroy P.E.**, Benítez-Pérez H.; “*Fuzzy Control with Time Delay estimation for Networked Control Systems within a single network segment*”; 6th International Conference on Electrical Engineering Computing Science and Automatic Control (CCE); Toluca, México, pp. 285-290, 2009. ISBN: 978-1-4244-4688-9.
-

Lista de Abreviaturas

A/D	COVERTIDOR ANALÓGICO / DIGITAL
ACK	AGRADECIMIENTO (<i>ACKNOWLEDGEMENT</i>)
AMP	ARBITRAJE POR PRIORIDAD DE PAQUETES (<i>ARBITRACIÓN ON MESSAGE PRIORITY</i>)
BEB	RETROCESO EXPONENCIAL BINARIO (<i>BINARY EXPONENTIAL BACKOFF</i>)
C.D.	CORRIENTE DIRECTA
CAN	RED DE ÁREA DE CONTROLADOR (<i>CONTROLLER AREA NETWORK</i>)
CD	DETECCIÓN DE COLISIÓN (<i>COLLISION DETECTION</i>)
CPU	UNIDAD CENTRAL DE PROCESAMIENTO (<i>CENTRAL PROCESSING UNIT</i>)
CRC	CHEQUEO REDUNDANTE CÍCLICO (<i>CYCLIC REDUNDANCY CHECK</i>)
CSMA	ACCESO MÚLTIPLE POR DETECCIÓN DE PORTADORA (<i>CARRIER SENSE MULTIPLE ACCESS</i>)
D/A	CONVERTIDOR DIGITAL / ANALÓGICO
DCS	SISTEMA DE CONTROL DISTRIBUIDO (<i>DISTRIBUTED CONTROL SYSTEM</i>)
DDCS	SISTEMA DE CONTROL DIGITAL DIRECTO (<i>DIRECT DIGITAL CONTROL SYSTEM</i>)
DOF	GRADOS DE LIBERTAD (<i>DEGREES OF FREE</i>)
EAM	ERROR ABSOLUTO MEDIO
EOF	FIN DE TRAMA (<i>END OF FRAME</i>)
FIFO	PRIMERO QUE ENTRA – PRIMERO QUE SALE (<i>FIRST IN – FIRST OUT</i>)
GBPS	GIGABITS POR SEGUNDO
I.I.D	INDEPENDIENTE E IDENTICAMENTE DISTRIBUIDO
I/O	ENTRADA / SALIDA (<i>INPUT / OUTPUT</i>)
IAE	INTEGRAL DEL ERROR ABSOLUTO (<i>INTEGRAL ABSOLUTE ERROR</i>)
ID	IDENTIFICADOR (<i>IDENTIFICATOR</i>)
IEEE	INSTITUTO DE INGENIEROS ELECTRICOS Y ELECTRONICOS (<i>INSTITUTE OF ELECTRICAL AND ELECTRONICS ENGINEERS</i>)
INT	INTERRUPCIÓN (<i>INTERRUPTION</i>)
IP	PROTOCOLO DE INTERNET (<i>INTERNET PROTOCOL</i>)
ISO	ORGANIZACIÓN INTERNACIONAL PARA LA ESTANDARIZACIÓN (<i>INTERNATIONAL ORGANIZATION FOR STANDARIZATION</i>)
LQG	REGULADOR CUADRÁTICO LINEAL (<i>LINEAR QUADRATIC REGULATOR</i>)
LMI	DESIGUALDAD MATRICIAL LINEAL (<i>LINEAR MATRIX INEQUALITY</i>)
MAGLEV	SISTEMA DE LEVITACIÓN MAGNÉTICA
MAP	PROTOCOLO PARA AUTOMATIZACIÓN DE MANUFACTURA (<i>MANUFACTURING AUTOMATION PROTOCOL</i>)
MAC	CONTROL DE ACCESO AL MEDIO (<i>MEDIA ACCESS CONTROL</i>)
MBPS	MEGABITS POR SEGUNDO
MIMO	MÚLTIPLES ENTRADAS – MÚLTIPLES SALIDAS (<i>MULTIPLE INPUTS – MULTIPLE OUTPUTS</i>)
NBA	ARBITRAJE CON DETECCIÓN DE BIT NO DESTRUCTIVO (<i>NONDESTRUCTIVE BITWISE ARBITRATION</i>)
NCS	SISTEMAS DE CONTROL EN RED (<i>NETWORKED CONTROL SYSTEM</i>)
NUT	TIEMPO DE TRAMA (<i>START OF FRAME</i>)
PC	COMPUTADORA PERSONAL (<i>PERSONAL COMPUTER</i>)
PI	PROPORCIONAL INTEGRAL
QoC	CALIDAD DE CONTROL
QoS	CALIDAD DE SERVICIO
RAM	MEMORIA DE ACCESO ALEATORIO (<i>RANDOM ACCESS MEMORY</i>)
RM	TASA MONOTÓNICA (<i>RATE MONOTONIC</i>)
RTT	RETARDO IDA Y VUELTA (<i>ROUND TRIP TIME</i>)
SISO	UNA ENTRADA – UNA SALIDA (<i>SINGLE INPUT – SINGLE OUTPUT</i>)
TCP	PROTOCOLO DE CONTROL DE TRANSMISIÓN (<i>TRANSMISSION CONTROL PROTOCOL</i>)
TSK	TAKAGI-SUGENO-KANG

UDP
WCET

PROTOCOLO DE DATAGRAMA DE USUARIO (*USER DATAGRAM PROTOCOL*)
TIEMPO DE EJECUCIÓN DEL PEOR CASO (*WORST CASE EXECUTION TIME*)

Resumen

Un Sistema de Control en Red (*NCS, Networked Control System*) es un sistema de control donde los sensores, controladores y actuadores intercambian información en una red de comunicación. Este documento presenta el codiseño control – planificación de una clase de NCSs utilizando redes no confiables.

El Codiseño es realizar de manera concurrente e integral el diseño de dos o más metodologías ó elementos de un sistema a fin de cumplir objetivos específicos. Así, El Codiseño control – planificación es el diseño concurrente e integral de control y planificación con el objetivo de cumplir requerimientos del sistema y/o especificaciones técnicas. El Codiseño control – planificación tiene como objetivo principal compensar las imperfecciones de la red a través del controlador y disminuir las transmisiones dentro de la red a través del planificador.

El codiseño inicia con el análisis y modelado de las imperfecciones de la red, que proporciona información para el diseño integral del controlador y el planificador. El controlador y planificador se diseñan a través de sistemas difusos, la parte antecedente de manera integral genera reglas difusas que dividen el rango de las imperfecciones. La parte consecuente se diseña de manera concurrente, el controlador difuso diseña un controlador óptimo para cada intervalo, mientras que el planificador difuso diseña un planificador dinámico para cada intervalo.

Para mostrar la aplicación del codiseño NCS se presentan dos casos de estudio: el primero es un sistema de levitación magnética, con un electro magneto como actuador, y dos sensores que miden la posición de una esfera de acero y la corriente del electro magneto, donde el objetivo es mantener levitando la esfera en una posición deseada; el segundo sistema es un simulador de helicóptero de dos grados de libertad (2-DOF), donde el objetivo es mantener el ángulo deseado de *pitch* (cabeceo) y *yaw* (viraje), a través de dos hélices con dos motores como actuadores.

Ambos casos de estudio presentan comparativas con algunas estrategias presentes en la literatura.

Capítulo 1

1. Introducción

1.1	MOTIVACIÓN
1.2	OBJETIVO
1.3	ALCANCES
1.4	METAS
1.5	CONTRIBUCIÓN
1.6	METODOLOGÍA
1.7	ESTRUCTURA DE LA TESIS

1.1 Motivación

En las últimas décadas se ha incrementado el desarrollo de sistemas de alta tecnología, como automóviles, robots autónomos, procesos industriales, etc, para cubrir las necesidades con una mayor funcionalidad y alto desempeño a bajo costo pero aumentando su complejidad. Para este desarrollo es necesario la interacción entre diferentes disciplinas, como la ingeniería mecánica, eléctrica y de software. Durante el proceso de diseño, muchas decisiones influyen en el resultado de etapas posteriores de diseño y/o el resultado final. Así, sin la interacción entre las disciplinas involucradas en el diseño, se corre el riesgo de hacer suposiciones que repercuten en el desempeño final del sistema, principalmente si son propiedades y características establecidas en etapas anteriores por disciplinas diferentes.

Una de las grandes tendencias incorporadas en los sistemas de alta tecnología es incluir redes de comunicación para el intercambio de información entre los diferentes sistemas y subsistemas a fin de controlar, monitorear, proteger, etc. el sistema completo. Bajo esta tendencia se ha desarrollado el concepto de Sistemas de Control en Red (*Network Control Systems* ó *NCSs*) donde los sensores, actuadores, controladores, y otros elementos, se distribuyen geográficamente, intercambiando información a través de una red de comunicación con fines de control.

La característica más relevante de un NCS es que la información de control y monitoreo (señal de referencia, señal de control, salida de la planta, etc.) se comunica mediante la red entre los componentes del sistema NCS. Las ventajas de un NCS son, principalmente, reducción de

cableado, fácil diagnóstico, mantenimiento, funcionalidad, adaptabilidad y bajo costo.

Sin embargo, el análisis y diseño de NCSs tienen desventajas, debido a que la interacción entre disciplinas genera un aumento en su complejidad. Analizando un NCS, se presentan cinco imperfecciones principales generadas por la incorporación de una red de comunicación. Cuatro son imperfecciones de tiempo y una es imperfección de cuantización:

- i) *Los retardos de red* es el tiempo que se tarda un paquete de información desde un componente a otro, y principalmente se compone del retardo de almacenamiento y transmisión.
- ii) *La pérdida de paquetes* es cuando el componente destino no recibe el paquete de información. Esto por razones como fallas en las conexiones de comunicación, desbordamiento de un buffer, gran cantidad de retransmisiones, etcétera. Generalmente se presentan cuando la red no es confiable.
- iii) *El intervalo de muestreo variante* es generado cuando el nodo sensor tiene más tareas a ejecutarse en el procesador y/o la conversión A/D tiene un consumo considerable en tiempo que limitan la periodicidad de muestreo.
- iv) *Las limitaciones de la red* son impuestas por el tipo de red. Estas restringen la información disponible en cada nodo o componente, por ejemplo, transmitir solo un paquete a la vez en el canal de comunicación.
- v) *El error de cuantización* es generado por las limitaciones en la longitud de bytes para representar una variable dentro de un paquete. Es específico de cada protocolo de comunicación y red.

Desde la perspectiva de Ingeniería de Control, se crea la necesidad de considerar las características y limitaciones de cada uno de los componentes del sistema para el análisis y diseño del controlador, mientras que en la perspectiva de Ingeniería de Software en Tiempo Real, es necesario considerar los requisitos del sistema y el controlador para el análisis y diseño del planificador, tanto de cada uno de los componentes del sistema, como de la red de comunicación.

Así, el objetivo del presente trabajo es compensar las imperfecciones inducidas por la red de comunicación a través del codiseño de dos disciplinas principales controlador – planificador para obtener un mejor desempeño del sistema.

1.2 Objetivo

El objetivo es desarrollar una estrategia de codiseño control – planificación para NCS, tal que se cumplan los requerimientos del sistema ante perturbaciones del sistema e imperfecciones generadas por la red.

El objetivo se divide en dos objetivos particulares, el primero es modelar, analizar y diseñar un controlador incorporando información sobre las imperfecciones de la red y compensarlas ajustando las leyes de control. El segundo es diseñar un planificador ajustando la ejecución del controlador en función de las imperfecciones de la red.

1.3 Alcances

El Codiseño propuesto, involucra, obtener un modelo de las imperfecciones de tiempo de la red Ethernet (los retardos de tiempo, la pérdida de paquetes y los muestreos variantes) con criterios suficientes para efectos de control, diseñar un modelo NCS que incorpore las imperfecciones en un rango acotado, diseñar un controlador que compense las imperfecciones de la red dentro de un rango a fin de evitar la degradación del desempeño, y diseñar un planificador que mejore la calidad de servicio de la red sin degradar el desempeño. Esto utilizando las tecnologías existentes sin más modificaciones que el software de los componentes de control. La red Ethernet considera que no tiene imperfecciones de cuantización, ni limitaciones de comunicación.

1.4 Metas

El trabajo se concentra en derivar un modelo NCS en tiempo discreto, para controlar un proceso en tiempo continuo a través de redes no confiables. El modelo incluye retardos de tiempo mayores al período de muestreo, la pérdida de paquetes e intervalo de muestreo variante en el tiempo, teniendo la capacidad de aplicarse a una variedad de redes de comunicación. El modelo propuesto asume que el tiempo entre muestras toma valores en un conjunto acotado $\tau_k \in [\tau_{\min}, \tau_{\max}]$. El efecto de rechazo de paquetes es incluido en el controlador propuesto y se implementa a través de estampado de tiempo. El modelo NCS propuesto, es aplicable tanto a sistemas manejados por tiempo como por evento.

Se pretende realizar el codiseño de un sistema de control en red, utilizando información tanto del sistema de control como de la red a fin de compensar las imperfecciones de tiempo generadas por la red. La metodología tiene como objetivo emplear información tanto del sistema de comunicación como del proceso con cambios mínimos en la estructura y configuración del NCS. Se pretende su aplicabilidad en múltiples configuraciones de NCS.

1.5 Contribución

Durante el desarrollo del proyecto se presentaron múltiples logros, que se pueden dividir en logros de concepto y de aplicación. Dentro de los logros de concepto se obtuvo la compensación de tres imperfecciones de tiempo de la red siendo compensados los retardos de tiempo, la pérdida de paquetes y la variabilidad del período de muestreo, se obtuvo una metodología aplicable a múltiples configuraciones de NCS y para sistemas manejados por tiempo o por evento, se obtuvo un modelo NCS difuso que incorpora información de las imperfecciones de la red a fin de compensarlos.

En los logros de aplicación se emplearon dos casos de estudio para las pruebas de concepto obtenidas, el primero un sistema de levitación magnética que es un sistema no lineal, variante en el tiempo, una entrada – una salida, el segundo un simulador de Helicóptero de dos grados de libertad, sistema no lineal, variante en el tiempo, dos entradas – dos salidas. Ambos casos de estudio están incorporados de manera distribuida a través de una red Ethernet de 100 Mbps, cada elemento de control es un nodo con un sistema operativo en tiempo real que permite el buen desempeño del sistema.

1.6 Metodología

La metodología inicia con el conocimiento de los efectos inducidos por la red al incorporarse en un lazo de control, para esto se establecen las clasificaciones de las imperfecciones y como afecta estas en la degradación del sistema.

Una vez que se tiene la clasificación y el registro de las imperfecciones para una red Ethernet, se elige un modelo probabilístico que permite modelar el comportamiento de las imperfecciones.

Una vez con los modelos de las imperfecciones se presenta un modelo difuso que incorpora

estos modelos para modelar las dinámicas del proceso ante estas imperfecciones. A partir del modelo difuso se diseña un controlador difuso que compensa las imperfecciones de tiempo en un rango acotado.

La conclusión de la metodología es codiseñar el controlador y planificador de acuerdo al modelo difuso del sistema con el fin de compensar y minimizar los efectos de las imperfecciones en el sistema NCS y red de comunicación respectivamente.

1.7 Estructura de la tesis

El esquema de la tesis es como sigue. El capítulo 2 da algunos aspectos básicos en control por computadora, redes de comunicación y control difuso, que serán usados en el capítulo 5,6 y 7.

El capítulo 3 discute el modelo de NCS y las diversas imperfecciones generadas al incluir una red de comunicación en un lazo de control. Presenta algunos modelos de imperfecciones como retardos de tiempo, paquetes perdidos, etc. disponibles en literatura. Adicionalmente, presenta ventajas y desventajas de los modelos presentados.

El capítulo 4 presenta los sistemas que serán utilizados como casos de estudio, presenta un sistema de levitación magnética que es inestable a lazo abierto, una entrada – una salida, que permitirá verificar las estrategias de control difuso propuestas. Por otro lado, se presenta un helicóptero simulador de dos grados de libertad que es inestable a lazo abierto, dos entradas – dos salidas, destinado a verificar las estrategias de codiseño difuso propuestas.

El capítulo 5 presenta el modelado dinámico de imperfecciones para una red Ethernet con diversas configuraciones del NCS, estos modelos serán empleados en los capítulos 6 y 7.

El capítulo 6 presenta el tema central del trabajo, se diseña el controlador difuso para compensar imperfecciones variantes de la red, además de presentar un planificador difuso para mejorar la calidad de servicio de la red sin degradar el desempeño del sistema, complementado con el controlador difuso se obtiene el codiseño NCS.

El capítulo 7 es destinado a proporcionar veracidad al codiseño, se realizan experimentos con los casos de estudio variando las condiciones de tráfico y se comparan con el desempeño de trabajos disponibles en la literatura.

El capítulo 8 establece las conclusiones y recomendaciones para investigación futura.

Capítulo 2

2. Antecedentes

2.1	SISTEMAS CONTROLADOS POR COMPUTADORA
2.2	REDES DE COMUNICACIÓN
2.3	CONTROL DIFUSO
2.4	CONCLUSIONES

2.1 Sistemas controlados por computadora

Con el desarrollo de las tecnologías computacionales, los sistemas controlados por computadora son cada vez más populares. Esto bajo la tendencia de realizar operaciones complejas de manera remota, integrando la computación, redes de comunicación y control en tiempo real. Los sistemas controlados por computadora generalmente son clasificados en tres tipos: *Sistemas de Control Digital Directo (DDCS)*, *Sistemas de Control Distribuido (DCS)* y *Sistemas de Control en Red (NCS)*[].

Los sistemas de control digital directo (DDCS) realizan el control de un proceso a través de las computadoras, que conectan los elementos de control punto a punto. Se utilizan tarjetas de adquisición de datos para obtener información del proceso y elementos de potencia para aplicar las acciones de control; la señal de control es calculada a través de una computadora de manera directa sin etapas de comunicación. A principios del desarrollo de la computación, el uso de las computadoras digitales como un componente del sistema de control era limitado, debido a que las computadoras eran lentas respecto al proceso, poco confiables, voluminosas y de alto consumo de potencia [17]. Principalmente, se utilizaban como guía de los operadores o como control de la referencia, pero los controladores analógicos eran imprescindibles. Esta situación cambió conforme las computadoras digitales eran más rápidas y confiables.

El objetivo general del DDCS es desempeñar las tareas en tiempo real de sensado, calcular la señal de control dentro de la computadora digital y aplicar la señal a través de la etapa de potencia.

En un Sistema de Control Distribuido (DCS), el objetivo es monitorear el sistema de

control a gran escala, utilizando dispositivos llamados de proceso que realizan tareas de control digital directo de un subsistema, pero comparten información en la estación de monitoreo al operador, a través de una red de comunicación. La mayoría de las tareas de control en tiempo real son desempeñadas por los dispositivos de proceso, es decir, los lazos de control de los subsistemas son cerrados localmente. La información tal como monitoreo, señales de referencia, señales de alarma entre otras, son transmitidas también sobre la red. Las aplicaciones más comunes de estos sistemas son la teleoperación y el control supervisado. La red es usada para mantener informado al operador (monitoreo) y el envío de comandos de usuario para acciones correctivas en el caso de emergencias.

La teleoperación es una aplicación de un DCS, donde el objetivo general es que un operador conduzca un sistema usando un control local y un control remoto para desempeñar varias tareas en ambientes peligrosos o de alto riesgo. En teleoperación, el operador depende de la retroalimentación desempeñada por el sensor, que a su vez genera la señal de control local antes de enviarla por la red al control remoto para aplicarla al sistema. Para el diseño este tipo de aplicación se consideran imperfecciones clave. Primero, el retardo de tiempo, el ancho de banda limitado, y las señales de ruido que obstaculizan la comunicación entre el operador y el sistema afectando su desempeño. Otro efecto es la dificultad de obtener un modelo con una exactitud suficiente para que el desempeño del sistema no sea afectado por la diferencia entre el sistema real y el modelo.[38]

El control supervisado es otra aplicación de un DCS. Éste tiene una arquitectura cliente-servidor, donde el usuario en la estación cliente puede dar instrucciones remotamente a un servidor que está directamente conectado al sistema en lugar de guiar remotamente al sistema como se realiza en la teleoperación. Esta aplicación tiene efectos similares a la teleoperación, pero sólo en la señal de referencia, debido a que es lo único que modifica el usuario, pero con la desventaja de no poder realizar cambios en el controlador.

Los sistemas de Control en Red (NCSs) tienen como característica la distribución geográfica de los componentes del lazo de control, mediante el empleo de una red de comunicación gracias a la capacidad de los sensores y actuadores de incorporar un dispositivo de comunicación [1][5]. Esta distribución permite localizar al controlador en un lugar de fácil acceso, de manera que en caso de mal funcionamiento de algún componente del sistema, el

operador tome acciones correctivas de manera casi inmediata, sin correr riesgo alguno. Sin embargo, el éxito de este esquema de control depende de la habilidad de compensar las imperfecciones generadas por insertar la red de comunicación.

Estas imperfecciones pueden ser compensadas de dos maneras: la primera es la mejora en las redes de comunicación empleadas y los protocolos, resultando en un incremento en la fiabilidad, la reducción de las latencias y pérdida de paquetes. Sin embargo, las compensaciones no pueden ser desempeñadas sólo por las mejoras en la infraestructura de comunicación. La segunda manera toma la perspectiva de control para compensar estas imperfecciones diseñando algoritmos de control que puedan contrarrestar estas imperfecciones y evitar que degraden el desempeño del sistema de control. Las imperfecciones presentes con la inserción de una red en un NCS han sido reconocidas ampliamente en la comunidad de control, siendo categorizadas en cinco tipos[]:

- i) *Retardos de comunicación en red*
- ii) *Pérdida de paquetes*
- iii) *Intervalo/período de muestreo variante*
- iv) *Limitaciones de la red*
- v) *Error de cuantización.*

La introducción de una red de comunicación en el lazo de control modifica las señales externas (u , y) del proceso y su controlador dadas estas cinco imperfecciones. Es decir, la entrada \hat{u} que se aplica al proceso, no siempre es la entrada necesaria u que se genera en el controlador; y la salida medida y del proceso no es exactamente conocida por el controlador que tiene solo acceso a la versión discreta \hat{y} . Cada imperfección inducida por la red tiene su efecto en la diferencia de las señales $e_x = x - \hat{x}$ y $e_u = u - \hat{u}$, gráficamente se puede observar en la figura(). La presencia de estas imperfecciones pueden degradar el desempeño del sistema significativamente, incluso llevarlo a la inestabilidad.

La naturaleza compleja de las imperfecciones crea la necesidad de analizar los efectos que tienen las imperfecciones en la estabilidad y el desempeño del sistema NCS, preferentemente de una manera cuantitativa, para poder compensarlas. Generando la tarea de analizar y diseñar NCS que incluya todas o parte de estas imperfecciones.

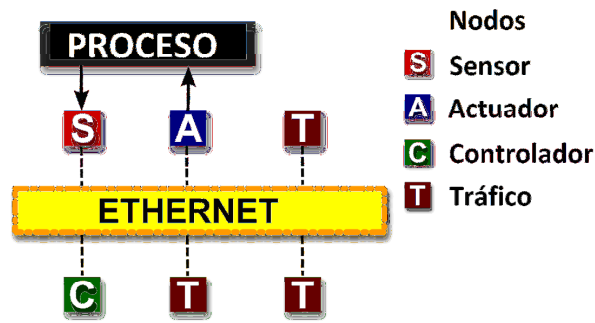


Figura 2.1. Sistema de Control en Red básico

Retardos de tiempo: Una de las imperfecciones en NCSs es el retardo inducido por la red. Estos retardos ocurren cuando los sensores, actuadores y controladores intercambian paquetes de datos a través de la red. Estos retardos pueden degradar el desempeño del sistema de control si es diseñado sin considerar el retardo, e incluso hacer inestable el sistema. El efecto se genera cuando los paquetes se envían a través de la red y, dependiendo del tipo de red, el paquete tiene que esperar a que el bus esté libre para transmitir (Ethernet), o espere su ranura asignada (ControlNet) o el nodo tenga la prioridad para transmitir (DeviceNet).[]

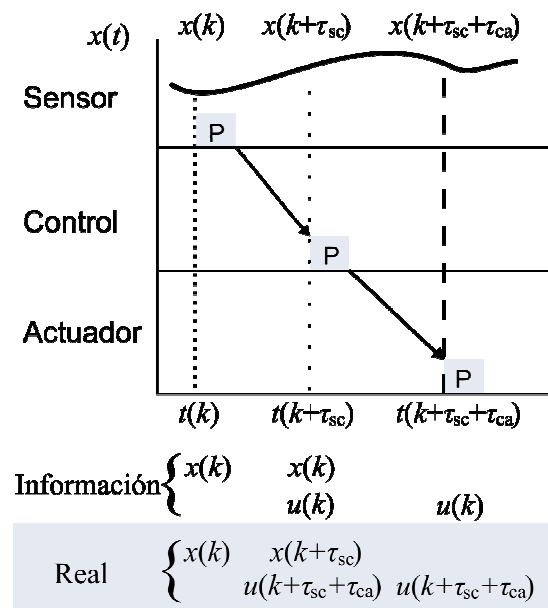


Figura 2.2. Representación del retardo de tiempo en un NCS.

El efecto del retardo de tiempo se muestra en la Figura 2.2. Los retardos de tiempo desde el sensor al controlador τ_{sc} y del controlador al actuador τ_{ca} , afectan la diferencia de las mediciones

e_x y la señal de control e_u , degradando el desempeño del sistema:

$$e_x = x(k) - x(k + \tau_{sc})$$

$$e_u = K \left[x(k) - x(k + \tau_{sc} + \tau_{ca}) \right]$$

donde $x(k)$ es la señal medida en el instante k con $k=kh$, $x(k + \tau_{sc})$ es la señal real del sistema cuando arriba el paquete al dispositivo, y h es el período de muestreo. Los retardos de tiempo de acuerdo al tipo de red y al protocolo se pueden clasificar en retardos constantes o variantes en el tiempo, y pueden ser menores o mayores al período de muestreo.[]

Pérdida de paquetes: Los paquetes perdidos ocurren cuando uno o más paquetes, viajando a través de la red, fallan en alcanzar su destino. Esto puede ser causado por varios factores, incluyendo degradación de la señal sobre el bus debido a múltiples trayectorias, arrastre de paquetes debido a la congestión de la red, paquetes corruptos, ruido en la señal, desbordamiento de buffer, y la distancia entre el transmisor y el receptor.[]

Si un paquete es perdido, el nodo destino genera su tarea con la última información recibida. El efecto de la pérdida de paquetes se muestra en la Figura 2.3, donde existen dos casos de pérdida: el primero es del controlador al actuador y el segundo del sensor al controlador. Estableciendo que d_1 y d_2 son el número de paquetes perdidos en la transmisión sensor – controlador y controlador – actuador, respectivamente, después de la última transmisión exitosa. En el caso de la comunicación controlador – actuador si d_2 paquetes son perdidos significa que no se realizaron d_2 acciones de control, hasta que arriba la acción de control $u(k + d_2h)$ en el instante $k + d_2h + \tau_{sc} + \tau_{ca}$, cuando la señal de actuación se debe realizar en el instante $k + \tau_{sc} + \tau_{ca}$. Lo mismo ocurre si d_1 paquetes son perdidos en la comunicación sensor – controlador: el controlador calcula la señal de control $u(k + d_1h)$ con la señal medida $x(k + d_1h)$ en el instante $k + d_1h + \tau_{sc}$, cuando debe ser calculada en el instante $k + \tau_{sc}$ con la información $x(k)$. Las diferencias entre los datos medidos y los reales causan degradación, e incluso inestabilidad en el sistema.

$$e_x = x(k) - x(k + d_1h + \tau_{sc})$$

$$e_u = K \left[x(k) - x(k + d_1h + d_2h + \tau_{sc} + \tau_{ca}) \right]$$

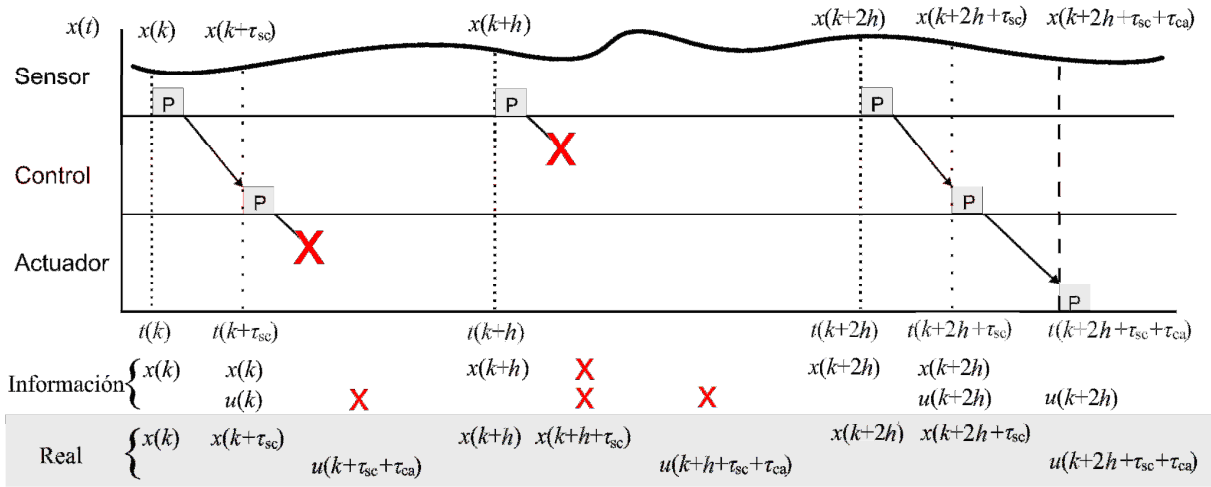


Figura 2.3. Representación de pérdida de paquetes en un NCS.

Intervalos de muestreo variantes: Las técnicas de diseño clásicas para sistemas digitales suponen que el muestreo de la variable medida es periódico, situación que en dispositivos de tiempo real no siempre se asegura, debido a diferentes factores, como baja velocidad del procesador para el muestreo de la variable, otras tareas que desempeña el procesador.

El efecto del período de muestreo variante se muestra en la Figura 2.4, donde una variación en el muestreo (jitter) α_k con $k=1,2,\dots$, causa que la señal de control aplicada al sistema $u(k+\alpha_k)$ tenga una diferencia con la señal de control que dicta la teoría de control a ser periódica $u(k)$, sin considerar los retardos presentes en el sistema.[]

$$e_u = K [x(k+\alpha_k) - x(k)]$$

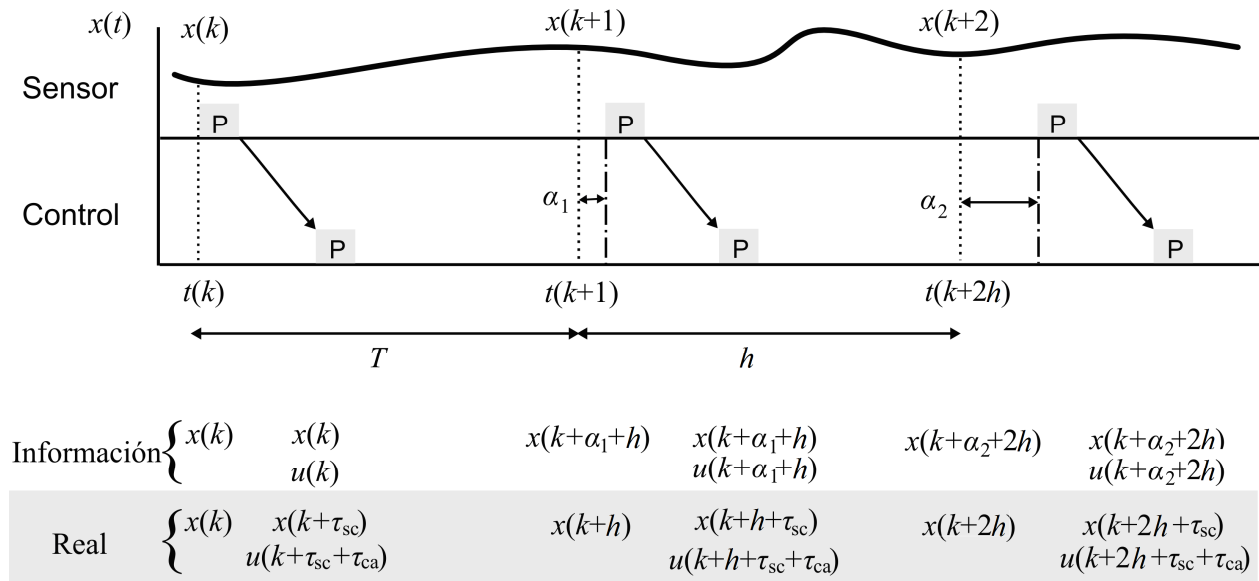


Figura 2.4. Representación del efecto del muestreo variante en el tiempo en un NCS.

Errores de cuantización: El error de cuantización se debe al truncamiento en el número de bits para representar las variables discretizadas. Esto es una limitante impuesta por el procesador, el tipo de paquetes en la red y/o el protocolo utilizado. Al representar un dato o variable con una representación finita de bits para realizarle operaciones dentro de un procesador o encapsularlo dentro de una trama para su transmisión, se genera un error de cuantización. []

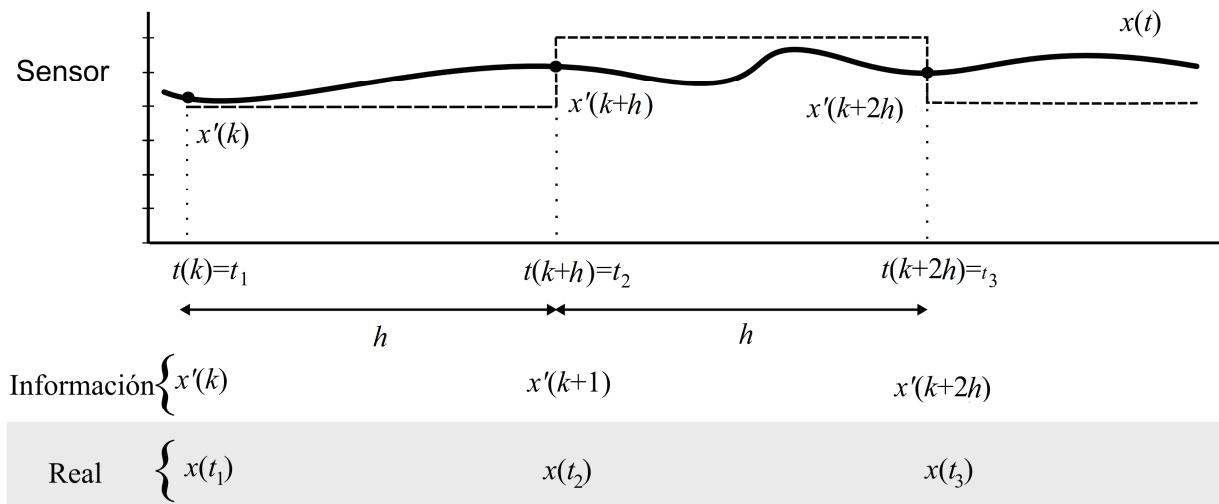


Figura 2.5. Representación del efecto de cuantización en un NCS.

Limitaciones de la red: Son restricciones definidas en el análisis de la red y el protocolo de

transmisión. Entre algunas esta el ancho de banda limitado, el ruteo, y el tamaño del paquete de datos definido por el protocolo. La principal limitante de la red a considerar en el diseño de NCS es la imposibilidad de transmitir múltiples paquetes al mismo tiempo sobre el bus de comunicación (tamaño del paquete). El efecto es más notorio cuando se tienen sistemas con múltiples entradas – múltiples salidas (MIMO) donde las transmisiones de sensores o controladores son retrazadas con tiempos diferentes debido a que cada sensor o actuador tiene su propia tarjeta de red.

2.2 Redes de comunicación

El diseño e implementación de un NCS requiere una apropiada integración del sistema de control y el sistema de comunicación de red. Elegir una red de comunicación apropiada para control es esencial para el diseño del NCS. Es necesario entender las características que presenta cada una de las redes de comunicación, así como sus protocolos de planificación que definen propiedades como la prioridad de los paquetes, la característica de los retardos de tiempo, pérdida de paquetes, etc. Los parámetros de tiempo que inevitablemente influyen las aplicaciones de control, son afectados por la tasa de transmisión de datos, el tamaño del paquete, y el protocolo de comunicación. Comparado con las redes de datos, las redes empleadas para control deben cumplir con ciertas características:[]

Períodos de muestreo fijos estrictos. La mayoría de los lazos de control en su diseño hacen la suposición de períodos de muestreo constantes para el análisis de estabilidad. En general los períodos de muestreo son cortos y, sumado a una gran cantidad de lazos de control, requiere una tasa de transmisión alta.

Longitud de paquete corto. La longitud de un paquete está determinada por el tipo de red y protocolo empleado, y varía de unos cuantos bytes a miles de bytes. Para sistemas de control el tamaño de los paquetes es relativamente corto.

Requerimientos de tiempo real crítico. Algunos requerimientos son confiabilidad, latencia, a prueba de fallas, etc. son más críticos en un NCS que en redes de datos.

Las redes de control que emplean retroalimentación generalmente están basadas en los siguientes protocolos de red: Ethernet (IEEE 802.3), Token Bus (IEEE 802.4), Token Ring (IEEE 802.5), y DeviceNet (CAN - *Controller Area Network*) (ISO 11898). Las características de estas

redes de control son resumidas a continuación, con su influencia en el desempeño de NCS. Al término de la sección, se realiza una comparación más detallada entre los protocolos Ethernet, Token Bus y CAN.

2.2.1 DeviceNet

DeviceNet es un protocolo de la capa de aplicación con considerable aceptación en aplicaciones de manufactura. Las especificaciones DeviceNet se basan en el estándar de comunicación serial CAN [34][41], desarrollado principalmente para aplicaciones en la industria automotriz, con buen desempeño en otras aplicaciones industriales de tiempo crítico. El estándar CAN es optimizado para paquetes cortos y usa el control de acceso al medio por prioridad de paquetes (CSMA/AMP). El protocolo DeviceNet es orientado a paquetes, y cada paquete tiene una prioridad específica que es usada para moderar el acceso al bus en caso de transmisión simultánea.

La trama de un paquete CAN inicia con un 'bit' de inicio para sincronización, y el arbitraje es realizado en el siguiente identificador de paquete, donde un '0' lógico domina sobre un '1' lógico. Un nodo que quiere transmitir un paquete espera hasta que el bus esté libre. Entonces inicia a enviar el identificador de su paquete bit por bit. Los conflictos por acceso al bus son resueltos durante la transmisión por un proceso de arbitraje a nivel de bit en el campo de arbitraje, que es la parte inicial de cada trama. Con esto, si dos nodos quieren transmitir sus paquetes al mismo tiempo, ambos continúan enviando sus paquetes y escuchando la red. Si uno de ellos recibe un bit diferente de sus bits enviados, éste pierde el derecho a continuar transmitiendo su paquete. Con este método, una transmisión en curso nunca colisiona.

En una red basada en el estándar CAN, los datos son transmitidos y recibidos usando trama de paquetes que transporta datos desde un nodo transmisor hasta uno o más nodos receptores. Los datos transmitidos no necesariamente contienen direcciones del transmisor y receptor en el paquete. En su lugar, cada paquete es etiquetado por un identificador que es único dentro de la red. Todos los otros nodos en la red reciben el paquete y lo aceptan o descartan, dependiendo de la configuración de los filtros aplicados al identificador. Este modo de operación es conocido como Multicast (Múltiples repartos).

El formato de trama se muestra en Figura 2.6. El encabezado completo es de 47 bits, que

incluye los campos de inicio de trama (SOF), 11 bits de arbitraje (Identificador), control, chequeo de redundancia cíclica (CRC), reconocimiento (ACK), fin de trama (EOF), y pausa (INT). El tamaño del campo de datos esta entre 0 y 8 bytes.

DeviceNet es un protocolo determinista, optimizado para paquetes cortos, principal ventaja para efectos de control. Se puede establecer la prioridad del paquete en el campo de arbitraje: los paquetes con más alta prioridad siempre gana el acceso al medio durante el arbitraje. Así, el retardo de transmisión para los paquetes con la prioridad más alta puede garantizar una cota superior. Otro aspecto importante es que la pérdida de paquetes no afecta a los paquetes con prioridad alta.

La mayor desventaja de DeviceNet comparado con otras redes es su baja tasa de transmisión (Máximo 1 Mbps), lo que limita el rendimiento comparado con otras redes de control. También presenta un bajo rendimiento cuando la cantidad de datos a transmitir es grande, aunque soporta fragmentación de datos que son mayores de 8 bytes. El problema es que para cada dato de 8 bytes, es necesario enviar el encabezado de 47 bytes.

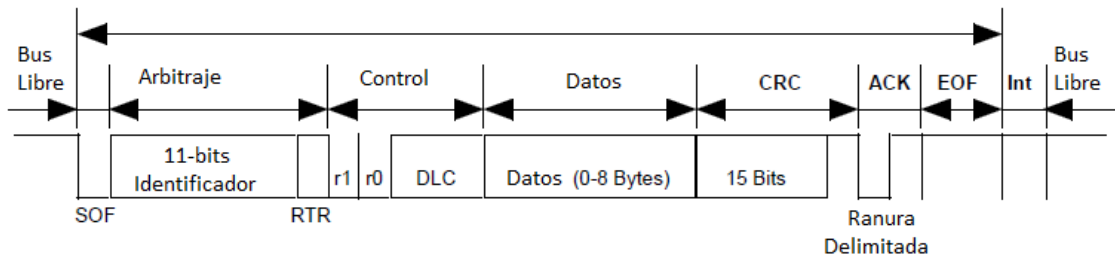


Figura 2.6. Trama del protocolo CAN

En el estándar CAN los datos son transmitidos con tramas de paquetes se muestra en la Figura 2.6, y donde un paquete puede ser transmitido periódicamente, esporádicamente o por demanda. A cada paquete se le asigna una prioridad que determina el acceso a la red.

2.2.2 ControlNet

Los protocolos MAP (*Manufacturing Automation Protocol*), PROFIBUS (*PROcess Field BUS*), y ControlNet son ejemplos típicos de redes de control con bus de paso por estafeta (Token bus). Todas son redes deterministas, debido a que el tiempo de espera máximo antes de enviar un paquete puede ser caracterizado por el tiempo de rotación de la estafeta. El protocolo Token bus (IEEE 802.4) permite una topología linear, multipunto, árbol o segmentada [64].

Con respecto a la topología, los nodos con protocolo Token bus son arreglados en un anillo, y en el caso específico de la red ControlNet, cada nodo conoce la dirección de su predecesor y sucesor. Durante la operación de la red, el nodo con la estafeta transmite sus tramas hasta que agota sus tramas o el tiempo asignado alcanza su límite. El nodo entonces transmite la estafeta a su sucesor en la red. Si un nodo no tiene tramas para enviar, éste únicamente pasa la estafeta a su sucesor. La localización física del sucesor no es importante, debido a que la estafeta es enviada al vecino lógico. La colisión de tramas no ocurre, ya que sólo un nodo puede transmitir a la vez. El protocolo también garantiza un tiempo máximo para el acceso a la red a cada nodo, y genera la estafeta si el nodo que la tenía deja de transmitir y no pasa la estafeta a su sucesor. Los nodos también pueden ser añadidos dinámicamente al bus y ser retirados del anillo lógico.

El formato de trama de ControlNet se muestra en la Figura 2.7. El encabezado total es de 7 bytes, incluye el preámbulo, delimitador de inicio, identificador fuente MAC, chequeo redundante cíclico (CRC) y delimitador final, además de la trama de datos, también llamado paquete de conexión (Lpacket). La trama puede incluir varios Lpackets que contienen campos de tamaño, control, etiqueta, datos con un tamaño de trama total entre 0 y 510 bytes. La dirección destino individual es especificada con el campo etiqueta. El campo tamaño especifica el número de pares de bits (3-255) contenidos en un Lpacket individual, incluyendo los campos tamaño, control, etiqueta y datos de conexión.

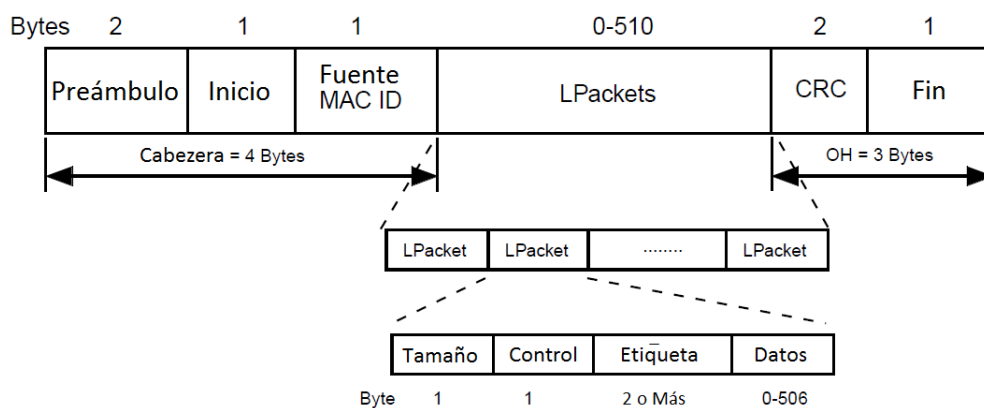


Figura 2.7. Trama del protocolo ControlNet

El protocolo ControlNet es un mecanismo con paso de estafeta implícito y asigna un ID MAC único (1-99) a cada nodo. En general, el nodo con la estafeta puede enviar datos. Sin embargo, no hay un pase de estafeta real alrededor de la red. En su lugar, cada nodo monitorea el

ID MAC fuente de cada trama recibida. Al finalizar una trama, cada nodo establece un registro de la estafeta implícita de la fuente recibida más 1 (ID MAC fuente + 1). Si el registro de estafeta implícito es igual al ID MAC propio, este nodo puede ahora transmitir paquetes. Todos los nodos tienen el mismo valor en sus registros, previniendo colisiones en el medio. Si un nodo no tiene datos para enviar, éste sólo envía un campo Lpacket vacío, llamado traza nula.

La longitud de un ciclo, llamado tiempo de actualización de red (NUT) en ControlNet, es dividido en tres partes: programada, no programada, y banda de seguridad. Durante la parte programada, cada nodo puede transmitir datos de tiempo crítico/programados para obtener la estafeta implícita desde el nodo 0 hasta S . Durante la parte no programada, los nodos del 0 al U comparten la oportunidad de transmitir datos no críticos en una forma round-robin, hasta que la asignación no programada expire. Cuando el tiempo para la banda de seguridad está listo, todos los nodos paran de transmitir, y solo el nodo con el ID MAC más bajo, llamado el moderador. Puede transmitir un paquete de mantenimiento, llamado paquete moderador, llevando a cabo la sincronización de todos los temporizadores de cada nodo y la publicación de parámetros de conexión crítica tal como el NUT, tiempo del nodo, S , U , etc. Si el paquete del moderador no es escuchado por dos NUTs consecutivos, el nodo con el ID MAC más bajo comienza transmitiendo el paquete del moderador en la parte de banda de seguridad del tercer NUT. Además, si el moderador observa que otro nodo tiene un ID MAC más bajo que el propio, inmediatamente cancela su papel como moderador.

Token Bus es un protocolo determinista que provee excelente rendimiento y eficiencia en redes con carga alta [44][64]. Durante la operación, Token Bus puede adicionar o remover nodos dinámicamente. Esto contrasta con el caso Token Ring, donde los nodos forman físicamente un anillo y no pueden adicionarse o removerse. Las partes programada y no programada en cada ciclo NUT hacen de la red como ControlNet que emplean Token Bus, accesible para paquetes de tiempo crítico y de tiempo no crítico.

Una desventaja de ControlNet (Token Bus) a pesar de su eficiencia y determinismo a tráfico de red alto, es que en canales con bajo tráfico su desempeño es equivalente a los protocolos de contención []. En general, cuando hay muchos nodos en un anillo lógico, un gran porcentaje del tiempo de la red es usado para pasar la estafeta entre nodos aún cuando el tráfico de datos es ligero [44], otra desventaja es la limitante de solo poder manejar 48 nodos dentro de

la red o 99 si se tienen tres repetidores.

2.2.3 Ethernet

Otra red muy usada como red de control es Ethernet, que usa el mecanismo de control de acceso múltiple por detección de portadora con detección de colisión (CSMA/CD) IEEE 802.3 para resolver el acceso en el medio de comunicación. Cuando un nodo quiere transmitir, éste escucha la red. Si la red está ocupada, espera hasta que la red esté libre. En otro caso éste transmite inmediatamente. Mientras un nodo transmite, también escucha el bus para detectar una colisión. Si dos o más nodos escuchan que el bus está libre y deciden transmitir simultáneamente, los paquetes colisionan y se dañan. En caso de detectar una colisión, los nodos paran de transmitir y esperan un tiempo aleatorio para retransmitir. Este tiempo aleatorio es determinado por el algoritmo de retroceso exponencial binario estándar (BEB) donde el tiempo de retransmisión es aleatoriamente elegido entre 0 y $(2^i - 1)$ ranuras de tiempo, donde i indica el i -ésimo evento de colisión detectado por el nodo, y una ranura de tiempo es el tiempo mínimo necesario para una transmisión de ida y vuelta. Sin embargo, después de 10 colisiones consecutivas detectadas, el intervalo es establecido a un máximo de 1023 ranuras. Después de 16 colisiones, el nodo para de transmitir y reporta una falla al procesador. La recuperación del nodo puede ser realizada en capas superiores [59].

El formato de trama de Ethernet se muestra en la Figura 2.8. El encabezado esta formado por 22 bytes, el fin de trama de 4 bytes y el tamaño de datos esta entre 46 y 1500 bytes. El tamaño de la trama mínima es de 72 bytes que incluye 46 bytes de datos, 12 bytes de las direcciones destino y fuente, 4 bytes del chequeo de suma, además de 8 bytes para el preámbulo y el inicio delimitador. Si el tamaño de los datos transmitidos en una trama es menor de 46 bytes, el campo de relleno es usado para completar el tamaño de datos mínimo. Hay dos razones para la existencia de un tamaño mínimo. Primero, facilita la distinción entre tramas válidas y basura. Cuando un transmisor detecta una colisión, trunca la trama actual, lo que significa que se pierden bits y partes de la trama. Segundo, previene que un nodo complete la transmisión de una trama corta antes de que el primer bit alcance el final del cable, donde puede colisionar con otra trama. Así, el tamaño de paquete mínimo dicta la longitud máxima del cable de red. Para Ethernet de 10-Mbps con una longitud máxima de 2500 m y cuatro repetidores, la ranura de tiempo es de 51.2

μs , que es el tiempo requerido para transmitir 64 bytes a 10 *Mbps* [59].

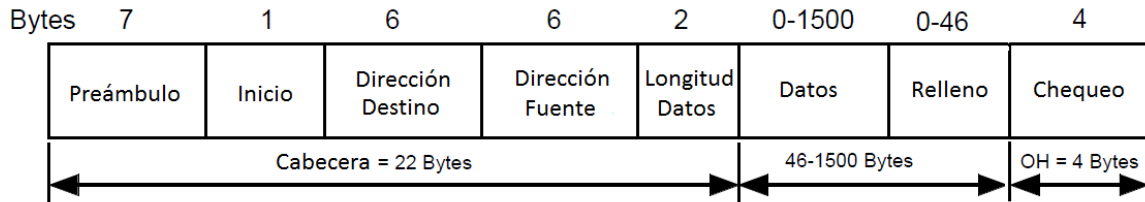


Figura 2.8. Formato de trama para protocolo CSMA/CD (Ethernet)

Debido a la baja sobrecarga de acceso al medio, Ethernet utiliza un algoritmo simple para la operación de la red y casi no tiene retardos a cargas bajas en la red [63]. El mínimo ancho de banda es usado para gana acceso a la red comparado con el protocolo Token Bus o Token Ring. El tipo de Ethernet comúnmente usado como red de control es el estándar 10 *Mbps* (Modbus/TCP); y el Ethernet de alta velocidad (100 *Mbps* o 1 *Gbps*) es principalmente usado en redes de datos [36][42][59], pero también se ha usado como redes de control [59].

La desventaja más importante de Ethernet es ser un protocolo no determinista, además de no soportar alguna prioridad de paquetes. Para cargas altas en la red, el gran problema son las colisiones porque afectan el rendimiento y el retardo de tiempo, que pueden ser no acotados Además de la pérdida de paquetes [64]. Otra desventaja es el efecto que provoca el algoritmo BEB, donde un nodo transmite paquetes de manera exclusiva por un tiempo prolongado mientras otros nodos esperan por el acceso al medio, causando degradación en el desempeño [41]. Basado en el algoritmo BEB, un paquete puede ser descartado después de una serie de colisiones, además, la comunicación de extremo a extremo no es garantizada. Debido al requerimiento de un tamaño de trama mínimo válido, Ethernet usa un tamaño de paquete grande aún para transmitir una cantidad pequeña de datos.

Varias soluciones se han propuesto para mejorar la utilización de Ethernet en aplicaciones de control. Por ejemplo, cada paquete puede estampar el tiempo antes de ser enviado. Esto requiere sincronización de reloj, tarea que no es fácil realizar, especialmente en este tipo de red [36]. Varios esquemas basados en retardos de retransmisión determinista para los paquetes colisionados de un protocolo CSA/CD resultan en un retardo acotado para todos los paquetes. Sin embargo, ésto se logra a expensas de minimizar el desempeño de CSMA/CD moderando la utilización del canal en términos del rendimiento del retardo [40]. Otras soluciones (LonWorks)

proporciona un nivel de prioridad dentro de CSMA/CD para mejorar la respuesta en tiempo de paquetes críticos [23]. Usar Ethernet conmutado subdividiendo la arquitectura de la red es otra forma de incrementar su eficiencia [85].

2.2.4 Comparativa

Un resumen de las características de los tres tipos de redes para control se muestra en la Tabla 2.1, donde se hacen ciertas conclusiones.

Tabla 2.1. Comparativa de las redes de control más empleadas

	Ethernet	ControlNet	DeviceNet
Tasa de transmisión (<i>Mbps</i>)	10 – 1000	5	0.125 – 1
Longitud máxima (<i>m</i>)	2500 – 550	1000	500 – 100
Dato máximo (<i>bytes</i>)	1500	504	8
Paquete máximo (<i>bytes</i>)	72	7	55
Número máximo de nodos	1024	48 – 99 con repetidores	64

Basado en la información mostrada en la Tabla 2.1 y las características de cada protocolo la elección de una red para control depende de ciertas condiciones del sistema. Por ejemplo, si se tiene alta carga en la red, pero se requiere un ambiente determinista con pocos nodos, se puede utilizar una red tipo DeviceNet. Pero si el ambiente es determinista, con una mayor tasa de transmisión y mayor número de nodos, se puede optar por una red ControlNet. Por otro lado, si las distancias entre los nodos y/o el servidor son grandes y se tiene una gran cantidad de nodos para la transmisión, con un tamaño de datos mayor, Ethernet es la mejor opción; esto porque al aumentar el tamaño del paquete a transmitir, las otras redes (DeviceNet, ControlNet) aumentan el tráfico en la red, debido a que tienen que dividir el paquete y agregar el encabezado a cada subpaquete. Otra ventaja del uso de Ethernet es la posibilidad de contrarrestar los efectos de su inclusión en el lazo de control por técnicas de control o planificación de los nodos.

Así, Ethernet es potencialmente la red de control más práctica debido a su bajo costo, disponibilidad, alta tasa de transmisión, aunque no es optimizada para paquetes cortos con requerimientos de tiempo real.

2.3 Control Difuso

La mayoría de los algoritmos de control convencional requieren un modelo matemático del

sistema dinámico a ser controlado. El modelo matemático es usado para construir un controlador. En muchas situaciones prácticas, no siempre es posible tener un modelo matemático exacto del sistema a controlar. Es aquí donde el *sistema difuso*, al tener la propiedad de aproximador universal [74], ofrece una forma para resolver algunos problemas de modelado, mediante implementaciones lingüísticas. El objetivo es capturar la conducta dinámica en la forma de reglas IF-THEN. Cada regla IF-THEN define el conjunto de factores para los cuales es verdad (parte antecedentes IF) ante un conjunto de acciones (parte consecuente THEN) a ejecutarse.

La parte antecedente se conforma de las variables de entrada, las funciones de membresía, y la función del conjunto difuso. Las variables de entrada ‘ x ’ son aquellas posibles variables que representen un cambio en la validez de la parte consecuente. Una función de membresía ‘ μ ’ es aquella que mapea parte del intervalo de una variable de entrada $[a,b]$ a un intervalo $[0,1]$; mientras que el conjunto difuso ‘ ω ’ mapea todas las variables de entrada a un intervalo $[0,1]$.

$$\begin{aligned}\mu_j(x) &= \{0 \leq \mu_j \leq 1 : x_j \in V \subset U\} \\ \omega(x) &= \{0 \leq \omega \leq 1 : x \in U\}\end{aligned}\tag{2.1}$$

donde U es el conjunto de rangos válidos para las variables de entrada, y V es un subconjunto de una variable de entrada de U .

La parte consecuente se forma con la máquina de inferencia y la defusificación. La máquina de inferencia es la función de salida $x(k+1)$ que se pondera por el resultado de la función de defusificación α . El método del *centro promedio* es utilizado generalmente para la defusificación.

Los modelos difusos pueden ser construidos tanto en tiempo continuo como en tiempo discreto, aprovechando las propiedades de cada uno. El método para modelado de sistemas dinámicos en tiempo continuo mediante modelos difusos ha sido propuesto por Takagi y Sugeno [74] y desarrollados por Takagi-Sugeno-Kang [75]. Este tipo de modelos es llamado normalmente Takagi-Sugeno-Kang (TSK). Para sistemas dinámicos en tiempo discreto, el modelo difuso TSK es analizado por Tanaka y Sugeno [76], y Tanaka et. al. [77][78]. A continuación, se presenta el modelo difuso TSK en tiempo discreto que sirve como referencia para cumplir el objetivo principal de este trabajo.

En principio se puede construir un modelo TSK [] si la descripción del sistema esta en

términos de ‘ r ’ modelos lineales locales de la forma:

$$x(k+1) = \mathbf{A}_i x(k) + \mathbf{B}_i u(k) \quad i=1,2,\dots,r \quad (2.2)$$

donde el vector de estado $x(k) \in \mathfrak{R}^n$, el vector de control $u(k) \in \mathfrak{R}^m$, y las matrices \mathbf{A}_i y \mathbf{B}_i son de dimensiones apropiadas obtenidas de un modelo lineal discreto. La información anterior es utilizada para describir cada una de las ‘ r ’ reglas IF-THEN, donde la i -ésima regla tiene la siguiente forma:

$$\begin{aligned} \text{Rule } i: \text{ IF } x_1(k) \text{ is } \mu_1^i \text{ and } \dots \text{ and } x_n(k) \text{ is } \mu_n^i \\ \text{THEN } x(k+1) = \mathbf{A}_i x(k) + \mathbf{B}_i u(k) \end{aligned} \quad (2.3)$$

donde μ_j^i $j=1,2,\dots,n$ es la j -ésima función de membresía de la i -ésima regla con n estados del sistema.

El modelo difuso resultante es construido como el promedio pesado de los modelos locales y tiene la forma:

$$x(k+1) = \frac{\sum_{i=1}^r \omega^i (\mathbf{A}_i x(k) + \mathbf{B}_i u(k))}{\sum_{i=1}^r \omega^i} \quad (2.4)$$

$$= \sum_{i=1}^r \alpha_i (\mathbf{A}_i x(k) + \mathbf{B}_i u(k))$$

$$\text{con } \alpha_i = \frac{\omega^i}{\sum_{i=1}^r \omega^i} \quad (2.5)$$

$$\text{y } \omega^i = \omega^i(x) = \prod_{j=1}^n \mu_j^i(x_j) \quad (2.6)$$

donde α_i es el conjunto difuso (grado de disparo normalizado) (2.5), y ω^i el grado de disparo de la i -ésima regla (2.6).

Haciendo notar que α_i tiene las siguientes propiedades:

$$0 \leq \alpha_i \leq 1 \text{ y } \sum_{i=1}^r \alpha_i = 1 \quad (2.7)$$

El modelo difuso dado por (2.4), (2.5) y (2.6) es también referido en la literatura como *sistema politópico* [79].

Si consideramos que el sistema es no forzado donde no se aplica un vector de entrada, es decir $u = 0$, se pueden establecer condiciones suficientes para garantizar la estabilidad asintótica

del sistema. Para esto, primero se muestra un lema de desigualdades matriciales, útil en el análisis de estabilidad.

$$x(k+1) = \sum_{i=1}^r \alpha_i (\mathbf{A}_i x(k)) \quad (2.8)$$

Lema 1. Sea $\mathbf{A}, \mathbf{B}, \mathbf{P} \in \mathfrak{R}^{n \times n}$ matrices cuadráticas, más si la matriz $\mathbf{P} = \mathbf{P}^T > 0$ es positiva definida y cumple que:

$$\mathbf{A}^T \mathbf{P} \mathbf{A} - \mathbf{P} < 0 \text{ y } \mathbf{B}^T \mathbf{P} \mathbf{B} - \mathbf{P} < 0$$

Entonces la siguiente desigualdad es establecida:

$$\mathbf{A}^T \mathbf{P} \mathbf{B} + \mathbf{B}^T \mathbf{P} \mathbf{A} - 2\mathbf{P} < 0 \quad (2.9)$$

Usando el lema 1, se probarán las condiciones suficientes para la estabilidad asintótica global del modelo de la ecuación (2.8).

Teorema 1. El estado de equilibrio, $x_e = 0$ del modelo (2.8) es global asintóticamente estable si existe una matriz definida positiva $\mathbf{P} = \mathbf{P}^T > 0$ tal que para $i = 1, 2, \dots, r$, con r matrices \mathbf{A}_i , se cumple que:

$$\mathbf{A}_i^T \mathbf{P} \mathbf{A}_i - \mathbf{P} < 0 \quad (2.10)$$

Prueba: Suponiendo que exista una matriz definida positiva simétrica $\mathbf{P} = \mathbf{P}^T > 0$ tal que para $i = 1, 2, \dots, r$, desigualdades dadas por (2.10) son satisfechas. Se considera una función candidata de Lyapunov:

$$V(x(k)) = x^T(k) \mathbf{P} x(k)$$

Para probar el teorema, es suficiente mostrar que:

$$\Delta V(x(k)) < 0$$

Se tiene:

$$\begin{aligned} \Delta V(x(k)) &= V(x(k+1)) - V(x(k)) \\ &= x(k+1)^T \mathbf{P} x(k+1) - x(k)^T \mathbf{P} x(k) \\ &= \left(\sum_{i=1}^r \alpha_i x^T \mathbf{A}_i^T \right) \mathbf{P} \left(\sum_{i=1}^r \alpha_i \mathbf{A}_i x \right) - x^T \mathbf{P} x \\ &= x^T \left(\left(\sum_{i=1}^r \alpha_i \mathbf{A}_i^T \right) \mathbf{P} \left(\sum_{i=1}^r \alpha_i \mathbf{A}_i \right) - \mathbf{P} \right) x \end{aligned} \quad (2.11)$$

Por la propiedad de (2.7), se puede establecer que:

$$\sum_{i=1}^r \alpha_i \sum_{j=1}^r \alpha_j = \sum_{i=1}^r \sum_{j=1}^r \alpha_i \alpha_j = 1 \quad (2.12)$$

Sustituyendo (2.12) dentro (2.11) se obtiene:

$$\begin{aligned} \Delta V(x) &= x^T \left(\left(\sum_{i=1}^r \alpha_i \mathbf{A}_i^T \right) \mathbf{P} \left(\sum_{i=1}^r \alpha_i \mathbf{A}_i \right) - \sum_{i=1}^r \sum_{j=1}^r \alpha_i \alpha_j \mathbf{P} \right) x \\ &= x^T \left(\sum_{i=1}^r \sum_{j=1}^r \alpha_i \alpha_j (\mathbf{A}_i^T \mathbf{P} \mathbf{A}_j - \mathbf{P}) \right) x \end{aligned} \quad (2.13)$$

con α_i, α_j Arreglando (2.13) se obtiene

$$\begin{aligned} \Delta V(x) &= \sum_{i=1}^r \alpha_i^2 x^T (\mathbf{A}_i^T \mathbf{P} \mathbf{A}_i - \mathbf{P}) x + \sum_{i=1}^r \sum_{\substack{j=1 \\ j \neq i}}^r \alpha_i \alpha_j x^T (\mathbf{A}_i^T \mathbf{P} \mathbf{A}_j - \mathbf{P}) x \\ &= \sum_{i=1}^r \alpha_i^2 x^T (\mathbf{A}_i^T \mathbf{P} \mathbf{A}_i - \mathbf{P}) x + \sum_{i=1}^r \sum_{j>i}^r \alpha_i \alpha_j x^T (\mathbf{A}_i^T \mathbf{P} \mathbf{A}_j - \mathbf{A}_j^T \mathbf{P} \mathbf{A}_i - 2\mathbf{P}) x \end{aligned} \quad (2.14)$$

El primer término en (2.14) es “negativo definido” por la suposición inicial y las propiedades (2.7) y (2.12) de los α_i . El segundo término en (2.14) es “negativo definido” en virtud del lema 1. Así, $V = x^T \mathbf{P} x$ es una función de Lyapunov para el modelo difuso (2.8), y por el teorema de Lyapunov el estado de equilibrio $x_e = 0$ del modelo es global y asintóticamente estable.

El teorema 1 da una condición suficiente para la inexistencia de una \mathbf{P} común que satisfice (2.10).

A continuación se presenta una condición suficiente para la estabilidad asintótica global del modelo (2.4)-(2.6), suponiendo que se tiene una excitación a través de una ley de control (2.15) por retroalimentación de estado difuso:

$$u = - \sum_{j=1}^r \alpha_j \mathbf{K}_j x \quad (2.15)$$

El sistema en lazo cerrado tiene la forma:

$$x(k+1) = \sum_{i=1}^r \sum_{j=1}^r \alpha_i \alpha_j (\mathbf{A}_i - \mathbf{B}_i \mathbf{K}_j) x \quad (2.16)$$

Suponiendo que las matrices de ganancia \mathbf{K}_j fueron elegidas tal que las matrices

$\mathbf{A}_i - \mathbf{B}_i \mathbf{K}_i$, $i=1,2,\dots,r$ tienen sus eigenvalores en el círculo unitario del plano z . Esto es, las matrices son asintóticamente estables. Así, las condiciones suficientes para la inexistencia de una matriz \mathbf{P} común deben satisfacer las desigualdades (2.17) y (2.18) que se establecen en el teorema 2.

Teorema 2. El sistema en lazo cerrado de las ecuaciones (2.4)-(2.6) es global asintóticamente estable si existe una matriz positiva definida común $\mathbf{P} = \mathbf{P}^T > 0$ tal que:

$$\Lambda_{ii}^T \mathbf{P} \Lambda_{ii} - \mathbf{P} < 0 \quad (2.17)$$

$$(\Lambda_{ij} + \Lambda_{ji})^T \mathbf{P} (\Lambda_{ij} + \Lambda_{ji}) - 2\mathbf{P} < 0 \quad (2.18)$$

con $\Lambda_{ij} = \mathbf{A}_i - \mathbf{B}_i \mathbf{K}_j$ $i=1,\dots,r$ $j=1,\dots,r$

Prueba: suponiendo que existe una matriz definida positiva común tal que (2.17) y (2.18) son satisfechas. Se considera una función candidata de Lyapunov como:

$$\mathbf{V}(k) = x(k)^T \mathbf{P} x(k)$$

Es suficiente con mostrar que $\Delta \mathbf{V}(x(k)) < 0$ para probar el teorema.

$$\Delta \mathbf{V}(k) = \mathbf{V}(k+1) - \mathbf{V}(k) \quad (2.19)$$

Substituyendo $\mathbf{V}(k+1)$ y $\mathbf{V}(k)$ en () se tiene:

$$\begin{aligned} \Delta \mathbf{V}(k) &= x(k+1)^T \mathbf{P} x(k+1) - x(k)^T \mathbf{P} x(k) \\ &= \left(\sum_{i=1}^r \sum_{j=1}^r \psi_i \psi_j \Lambda_{ij} x(k) \right)^T \mathbf{P} \left(\sum_{i=1}^r \sum_{j=1}^r \psi_i \psi_j \Lambda_{ij} x(k) \right) - x(k)^T \mathbf{P} x(k) \end{aligned} \quad (2.20)$$

Aplicando las propiedades (2.12) a (2.20):

$$= x(k)^T \left(\sum_{i=1}^r \sum_{j=1}^r \sum_{k=1}^r \sum_{l=1}^r \psi_i \psi_j \psi_k \psi_l (\Lambda_{ij}^T \mathbf{P} \Lambda_{kl} - \mathbf{P}) \right) x(k)$$

$$\begin{aligned}
&= \frac{1}{4} x^T \left(\sum_{i=1}^r \sum_{j=1}^r \sum_{k=1}^r \sum_{l=1}^r \psi_i \psi_j \psi_k \psi_l \left((\Lambda_{ij} + \Lambda_{ji})^T \mathbf{P} (\Lambda_{ij} + \Lambda_{ji}) - 4\mathbf{P} \right) \right) x \\
&\leq \frac{1}{4} x^T \left(\sum_{i=1}^r \sum_{j=1}^r \psi_i \psi_j \left((\Lambda_{ij} + \Lambda_{ji})^T \mathbf{P} (\Lambda_{ij} + \Lambda_{ji}) - 4\mathbf{P} \right) \right) x \\
&+ x^T \left(\sum_{i=1}^r \sum_{j=1}^r \psi_i \psi_j \left(\frac{(\Lambda_{ij} + \Lambda_{ji})^T}{2} \mathbf{P} \frac{(\Lambda_{ij} + \Lambda_{ji})}{2} - \mathbf{P} \right) \right) x \\
\Delta \mathbf{V}(k) &\leq x^T \left(\sum_{i=1}^r \psi_i^2 (\Lambda_{ii}^T \mathbf{P} \Lambda_{ii} - \mathbf{P}) \right) x \\
&+ x^T \left(\sum_{i=1}^r \sum_{j>i} \psi_i \psi_j \left((\Lambda_{ij} + \Lambda_{ji})^T \mathbf{P} (\Lambda_{ij} + \Lambda_{ji}) - 2\mathbf{P} \right) \right) x
\end{aligned} \tag{2.21}$$

El primer término en (2.21) es negativo definido por la suposición de (2.17), mientras que por la suposición de (2.18); el segundo termino es negativo definido. Así, la función de Lyapunov propuesta asegura la estabilidad global asintótica del sistema con control difuso ().

2.4 Resumen

El presente capítulo establece el marco de referencia de los sistemas de control en red. Éstos se derivan de los sistemas de control por computadora, al incorporar una red de comunicación para el cálculo de la señal de control, y se distinguen de los sistemas de control distribuido en que monitorean la información o el manejo de las variables de referencia del sistema. Otra derivación son los sistemas de control basados en red (NCS), que intercambian información entre los diferentes componentes del control en lazo cerrado.

Se presenta una clasificación de las imperfecciones inducidas por la red al incorporarla en el control a lazo cerrado, y se define cómo afectan estas imperfecciones el desempeño de control del sistema.

Se hace una revisión de las redes de comunicación más comúnmente empleadas en sistemas NCS, donde Ethernet presenta un mayor reto para los sistemas NCS, ya que por su naturaleza no determinística, presenta casi la totalidad de las imperfecciones dependiendo de su configuración. Esto marca la pauta para establecer un codiseño con este tipo de red, que pudiera ser aplicado a las distintas redes empleadas para control que presentan menos imperfecciones inducidas por la red, sin grandes modificaciones en el codiseño.

Por último, se presenta una revisión de un modelo y control difusos, a fin de establecer su aplicabilidad como aproximador y controlador universal, con la capacidad para modelar no linealidades o fenómenos que se presenten dentro del sistema NCS. Este modelo difuso en particular tiene la ventaja de estar definido en el dominio de tiempo discreto y en forma de subsistemas en variables de estado, lo que permite emplearse con facilidad para modelar el comportamiento del sistema NCS. En cuanto al controlador difuso, tiene la capacidad de establecer un análisis de estabilidad, propiedad necesaria para efectos de control.

Capítulo 3

3. Estado del arte

3.1	RETARDOS EN NCS
3.2	PÉRDIDA DE PAQUETES
3.3	MÚLTIPLES IMPERFECCIONES DE RED
3.4	CONCLUSIONES

Los sistemas de control en red (NCS), en sus inicios, emplearon técnicas de control clásico como control lineal, adaptable, robusto, etc. con algunas modificaciones para compensar algunas imperfecciones inducidas por la red [80]. Como es común en la teoría de control clásica, se dividen en técnicas en el dominio continuo [49] y en el dominio discreto [52][55][57].

Este tipo de técnicas para diseñar NCSs dejan a un lado los conceptos de diseño establecidos por las redes de comunicación, haciendo algunas suposiciones sobre algunas características inducidas por la red como el retardo máximo. Su derivada, o la pérdida de paquetes, así como suposición en la ejecución de los nodos estrictamente periódicos y sincronización de los nodos.

En los últimos años, el concepto de *codiseño* se ha utilizado para denominar la aplicación de múltiples áreas del conocimiento en conjunto, con el objetivo de resolver un problema común. El codiseño, más común para NCS, es el control-comunicación, cuyo objetivo es diseñar un NCS a través de un controlador contemplando características de la red y un planificador para la transmisión de los nodos contemplando aspectos de control.

Como parte del codiseño, es de especial importancia considerar las múltiples imperfecciones inducidas por la red, tanto para el diseño de control y de planificación. Por ejemplo, reducir un error de cuantización (transmitiendo paquetes con más bits) resulta normalmente en retardos de transmisión más largos que se tienen que compensar, ya sea en la red por el planificador, o en el proceso a través del controlador. Realizar el codiseño supone la compensación del NCS completo (proceso, controlador, planificador) de una manera integral, haciendo necesario el uso de herramientas para obtener información cuantitativa a partir del

sistema de control y la red.

Aunque el campo de los NCSs es relativamente joven, varias líneas de investigación se han desarrollado. La mayoría consideran solo alguna imperfección inducida por la red. Los resultados presentes en la literatura deben ser extendidos e integrados para obtener un marco de referencia y estudiar las imperfecciones inducidas por la red simultáneamente. Este capítulo tiene el objetivo de presentar una revisión literaria sobre NCS, enfocándose en métodos que compense varias imperfecciones, haciendo una distinción entre la metodología de diseño.

En principio, el capítulo presenta algunas técnicas que compensan solo una de las imperfecciones, siendo las más comunes los retardos de tiempo y la pérdida de paquetes, que degradan en mayor medida el desempeño del sistema. Posteriormente se presentan los métodos que combinan algunas imperfecciones, presentando ventajas y desventajas de cada uno.

3.1 Retardos en NCS

Dependiendo de la configuración del NCS, la red empleada y el protocolo de comunicación entre otros factores del sistema, los retardos de tiempo pueden ser constantes, variantes en el tiempo o aleatorios [7]. El retardo es formado por tres componentes: el tiempo de procesamiento de los nodos, el tiempo de transmisión del paquete y el tiempo de espera del paquete en el regulador (*buffer*) de los elementos de comunicación (tarjetas de red, ruteadores, conmutadores, servidores) [32][56]. Generalmente, el tiempo de espera es mucho mayor al tiempo de transmisión de un paquete, por lo que en ocasiones son considerados como uno solo.

El retardo de red en muchos casos es un prerequisite para el diseño de control dependiendo del tipo de red y protocolo usado, etc. La clasificación más común del retardo se muestra por Nilsson [7], donde tres diferentes modelos se discuten: retardo constante, retardo aleatorio independiente de transmisiones anteriores, y retardo aleatorio con distribuciones de probabilidad gobernadas por cadenas de Markov.

El modelo de retardo constante es adoptado cuando las dinámicas del proceso son mucho más lentas que las dinámicas de red, y los retardos son significativamente más pequeños que las constantes de tiempo y que los retardos del proceso. En este caso, los retardos de la red serían siempre menores que el período de muestreo y, si el NCS es manejado por tiempo, la variación del retardo no tiene efecto.

Por otro lado, el modelo del retardo aleatorio independiente es justificado cuando hay varios eventos en la red, que pueden causar una conducta asíncrona en la comunicación. Esto se debe a que en un medio compartido, no todos los nodos pueden transmitir simultáneamente, generando la necesidad de esperar a que la red este desocupada. En determinadas redes cuando hay colisión de paquetes es necesario esperar un tiempo aleatorio, para poder intentar transmitir otra vez. Todos estos sucesos hacen que los retardos varíen en una forma aleatoria, posiblemente de acuerdo a alguna distribución de probabilidad.

Sin embargo, los modelos del retardo aleatorio independiente no capturan el efecto de tráfico fuerte de paquetes en la red (gran cantidad de paquetes en relación al ancho de banda), que a veces sigue a correlaciones entre los retardos aleatorios, donde un valor del retardo es dependiente de valores de retardos previos. Si la red experimenta tráfico fuerte, es probable que todos los paquetes sufran retardos de transmisión largos, hasta que la carga disminuya. Si la carga de red varía considerablemente, una solución para modelarla es usar distribuciones del retardo gobernadas por una cadena de Markov [7].

De acuerdo al sección (), las redes que presentan una mayor complejidad de diseño para su uso dentro de un NCS son las redes no confiables, ya que presentan la mayoría de las imperfecciones inducidas por una red. Las distribuciones del retardo más utilizadas para redes no confiables, como Ethernet e Internet, son las distribuciones exponencial y gaussiana. La distribución exponencial es utilizada con la suposición de distancias entre nodos relativamente cortas (sólo un ruteador) y la distribución gaussiana para distancias relativamente largas (múltiples ruteadores). Para distancias cortas, la distribución de los retardos puede dividirse en dos partes: una constante y otra variable. La parte constante puede ser definida por el retardo de comunicación debido a la longitud física del cableado, el tiempo de procesamiento de los nodos involucrados, mientras que la parte variable se ve mayormente afectada por el apilado de los paquetes en los elementos como conmutadores y ruteadores. Al ser distancias cortas, estos elementos son pocos, por lo que afectan en menor medida la cantidad del retardo, razón para considerar el retardo con una distribución exponencial. La distribución exponencial generalizada también ha sido propuesta para modelar retardos en redes IP [62].

Con respecto a la distribución gaussiana, se considera cada nodo en la red con un regulador FIFO (Primero que entra – Primero que sale, *First In, First Out*) con arribos aleatorios y tiempo

de servicio exponencial. Al considerar distancias largas hay más elementos (conmutadores, ruteadores, ect.) que regulan y destruyen los paquetes en el camino, y el retardo total es una suma de múltiples variables aleatorias independientes distribuidas exponencialmente, resultando en una distribución gaussiana. Esto en condiciones normales de operación [54].

Para modelar el retardo de tiempo de una red con tráfico fuerte y variante, el modelo de red podría necesitar tener un estado. El efecto de tráfico variante de red puede ser modelado por una cadena de Markov, haciendo una transición cada vez que hay una transferencia en la red de comunicación, postulando distribuciones de probabilidad para los retardos sensor – controlador (τ^{sc}) y controlador – actuador (τ^{ca}) en cada estado.

En algunos casos del NCS, en los cuales el retardo del proceso, o planta tiene dinámicas lentas, los esquemas de control basados en identificación del retardo (constante) o por compensación pueden ser usados. Por ejemplo, en control de procesos químicos. La estimación del retardo puede estar basada en modelos de series de tiempo, donde por lo general el orden del polinomio del numerador de la función de transferencia del proceso y el retardo están acoplados. Koivo *et. al.* [43] basan la estimación del retardo en un modelo autoregresivo controlado con promedio móvil integrado (CARIMA), donde se presenta un control auto-sintonizado para procesos variantes en el tiempo. En Sinopoli *et. al.* [58], dos métodos para estimación del retardo de tiempo en procesos no lineales son considerados, formulándose un problema de programación no lineal, y una red neuronal puede ser entrenada para estimar el retardo. El método se basa en la suposición de que el retardo puede ser dividido en una parte entera y una fraccionaria. El problema de programación no lineal, calcula el gradiente de la salida de un modelo neural, describiendo el proceso con respecto a la parte fraccionaria del retardo. Este método puede, en cierta medida, ser aplicado a NCSs.

Diversas metodologías se han presentado en la literatura haciéndose una división en áreas: comunicación, control, y codiseño.

3.1.1 Metodologías de Comunicación

Las metodologías de comunicación, utilizan las ciencias de la computación para diseñar la red del NCS, y generar algoritmos de planificación de los nodos y la red, a fin de mantener la estabilidad del NCS. Se pueden distinguir dos líneas de investigación: la primera, encargada de

caracterizar y diseñar la red de comunicaciones [3][2][12][4]; y la segunda, ocupada en diseñar algoritmos de planificación para la transmisión de paquetes en la red [14] [15][26][60]. Uno de los objetivos de esta investigación es no modificar la estructura del NCS, estableciendo una red Ethernet sin modificación, y centrándose en los métodos de planificación.

En cuestión de planificación, uno de los pioneros fue Hong [14] que propone un *algoritmo de planificación* para determinar los tiempos de muestreo, utilizando el concepto de ventana de transmisión. Los datos muestreados desde los componentes en el NCS comparten un número limitado de ventanas, tal que los requerimientos de desempeño de cada lazo de control son satisfechos, así como la utilización de los recursos de la red son considerablemente incrementados. Esta metodología establece los retardos de tiempo en un rango y evita colisiones de paquetes, siendo aplicable a sistemas Polling y Token Passing.

Posteriormente Pegden *et al.* [15] presentan un *algoritmo de asignación del ancho de banda* aplicable al protocolo CAN para NCS, utilizando el mecanismo CSMA/NBA (Carrier Sense Multiple Access with Nondestructive Bitwise Arbitration) [81]. Cada nodo observa el estado del bus antes de la transmisión de datos. Los nodos aplazan la transmisión de datos hasta que el bus esté libre. Cuando el bus está libre, cualquier nodo puede iniciar transmisión. Si más de un nodo transmite su paquete al mismo tiempo, el bus entra en conflicto que requiere resolver comparando sus identificadores. El algoritmo de asignación del ancho de banda satisface los requerimientos de retardo de tiempo para el control en tiempo real y datos por eventos, maximiza la utilización del ancho de banda de red y es diseñado fuera de línea. Una de las desventajas es la incapacidad de modificar la asignación de la transmisión de datos dinámicamente, mientras el sistema de control está en ejecución, otra desventaja es contemplar retardos menores al período de muestreo.

Tarn *et. al* [30] proponen una *metodología basada en eventos*. Ésta usa el movimiento del sistema como referencia, en vez de decidir el siguiente instante de muestreo-actuación por tiempo, el instante es establecido por un evento disparado por el estado o error del sistema. El método mapea el espacio de tiempo en espacio de evento y la estabilidad del sistema ya no depende del tiempo. Así, los retardos inducidos por la red que son menores a los tiempos de respuesta del proceso no desestabilizan el sistema, pero si los retardos son muy grandes o hay pérdida de paquetes, el desempeño del sistema se degrada.

La *metodología de colas* usa observadores y predictores para compensar los retardos y asegurar la invarianza en el tiempo del NCS. La metodología usa la teoría de colas para forzar a que los retardos aleatorios tengan una conducta determinística (constante). El método presentado por Luck y Ray [8] usa un observador para estimar los estados de la planta y un predictor para computar el control predictivo basado en mediciones de salida pasadas. El control y las mediciones de salida pasadas son almacenados como colas FIFO en buffers. Estos son localizados antes y después del controlador en el lazo de control. Primero, las mediciones pasadas son usadas para estimar los estados de la planta en $k - \theta + 1$, donde θ es el tamaño del registro entre el sensor y el observador. Luego, usando las estimaciones previas, se predice el estado de la planta en $k + \mu$, donde μ es el tamaño del registro después del controlador. La señal de control predictivo $u(k + \mu)$ es calculada y almacenada en el registro. El observador y el predictor son basados en modelo, por lo que el desempeño del sistema es altamente dependiente de la exactitud del modelo.

3.1.2 Metodologías de Control

Mientras el área de comunicaciones pretende minimizar, compensar o eliminar los retardos inducidos por la red a través de la asignación ordenada de transmisión y maximizar la utilización de sus recursos, en el área de control se pretende hacer intrínsecas las imperfecciones de la red modificando las técnicas de control establecidas para control digital o continuo. Los métodos contemplan el control óptimo, robusto, adaptable, difuso, estocástico, entre otros.

Un método común es obtener un modelo aumentado en tiempo discreto que incorpora los retardos de red como estado del sistema [9]. También se ha considerado al retardo de tiempo como estocástico y diseñar un control óptimo [7]. Suponer el retardo como una perturbación del sistema y compensarla a través técnicas de control con perturbaciones [21], es otra idea muy recurrente. Lo mismo sucede al utilizar el control robusto para diseñar las leyes de control poco sensibles a los retardos [37]. Adaptar la señal de control de acuerdo al efecto de red que se quiere compensar es un método empleado por el control difuso y el control adaptable, para modificar las ganancias del controlador. La descripción de estos métodos con algunas evoluciones se describen a continuación.

Helavi y Ray [9] proponen el *método del modelo aumentado* en espacio de estados en

tiempo discreto. Los retardos de la red son manejados por el controlador usando las señales de control aplicadas anteriormente como estados internos para calcular la nueva señal de control en el instante k , lo que genera un nuevo modelo con los estados del sistema y las entradas anteriores como nuevos estados. La estabilidad para retardos periódicos es probada basándose en los eigenvalores de la matriz de transición del sistema aumentado.

Nilsson [7] desarrolla una *método de control óptimo* para NCS. En este trabajo, el retardo se asume aleatorio, pero menor que un período de muestreo. Después, Lincoln [48] extendió el método para retardos más grandes que un período de muestreo. La idea básica de este método es formular el problema como un sistema LQG (Linear Quadratic Gaussian). Los sistemas dinámicos son dados en espacio de estado, y la ganancia del controlador óptimo se resuelve como un problema LQG usando programación dinámica. Resolver el problema requiere la información del retardo pasado y los estados completos.

El *método de perturbación* [21] propuesto por Walsh considera la diferencia entre los valores de salida actual y los valores de la salida transmitida más reciente como una perturbación en el sistema, y busca los límites para el error. La estabilidad se obtiene usando el método de Lyapunov sobre las dinámicas del error. Varias suposiciones se establecen, incluyendo comunicaciones libre de errores, muestreo rápido y observaciones sin ruido, pero el proceso y el controlador pueden ser no lineales y variantes en el tiempo. En este método la red sólo transmite del sensor al controlador y no del controlador al actuador.

El *método de control robusto* propuestos por Goktas *et. al.* [37], los retardos inducidos por la red son tratados como perturbaciones del sistema nominal, y el diseño de control es realizado en el dominio de la frecuencia usando teoría de control robusta. La mayor ventaja de este método es que no hay necesidad de saber con exactitud la distribución del retardo por adelantado. Los retardos de la red son asumidos acotados, y son modelados como perturbaciones multiplicativas simultáneamente. Así puede ser aplicado el diseño H^∞/μ síntesis, eligiendo ciertos pesos multiplicativos de incertidumbre tal que el sistema NCS sea poco sensible a los retardos desconocidos.

El *método de control difuso* toma ventaja de la teoría difusa para actualizar las ganancias del controlador basándose en la señal de error entre la referencia y la salida actual del sistema, y la salida del controlador [31]. El método actualiza la ganancia de un controlador PI basado en el

error y la salida del controlador, se incluye el diseño de las funciones de membresía en línea y fuera de línea para optimización. Por otro lado, Zhang *et. al.* [69] obtienen el retardo por maximizar la función de correlación cruzada de la salida de la planta estimada sin el retardo y la salida medida. Este esquema de estimación usa un predictor Smith difuso que compensa el retardo constante, y el controlador difuso es usado para adaptar los parámetros cambiantes en el sistema.

El *método de control adaptable* se basa en la habilidad de medir las condiciones de tráfico de la red, y adaptar los parámetros del controlador. En éste método, el controlador puede solicitar y actualizar las condiciones de la calidad de servicio (QoS) de la red, y si los requerimientos de QoS deseados no son satisfactorios, los parámetros del controlador son ajustados con el objetivo de mejorar en lo posible el desempeño [61].

3.1.3 Metodologías de Codiseño

El codiseño tiene como objetivo mezclar las ventajas de dos o más áreas del conocimiento, y minimizar las desventajas de dichas áreas presentes por separado, a fin de resolver adecuadamente un problema común. En NCS, el codiseño control-comunicación comúnmente integra el control retroalimentado y las comunicaciones en tiempo real. Se basa en el principio de que el desempeño del sistema depende del diseño de los algoritmos de control, y también recae en el desempeño de planificación de los recursos de comunicación compartidos. Desafortunadamente, el diseño NCS se basa muy frecuentemente en el principio de *separación de preocupación* [27]. Este principio de separación supone que los controladores retroalimentados pueden ser modelados e implementados como tareas periódicas que tienen un período fijo, con consumo conocido ej. WCET (Worst Case Execution Time) y un deadline (tiempo límite) rígido; éstas suposiciones son ampliamente adoptadas por la comunidad de control y desarrolladas por la teoría de sistemas muestreados. Aunque éstas suposiciones permiten a la comunidad de control enfocarse en sus propios problemas sin preocuparse sobre como la planificación de tareas en los nodos, y paquetes en la red se realizan, la comunidad de comunicación no siente la necesidad de entender como la planificación impacta el desempeño del sistema.

La tarea de control no siempre utiliza los recursos de cómputo disponibles en una forma

óptima, y las suposiciones del modelo de tarea simple son demasiado restrictivas con respecto a las características de muchos sistemas de control. Por ejemplo, muchos tiempos límites no son siempre rígidos, es decir, muchos sistemas de control prácticos pueden tolerar ocasionalmente la pérdida de su deadline.

Para hacer frente a la limitación de recursos en un NCS, el codiseño es necesario a diferentes niveles, por ejemplo, codiseño hardware / software ó el codiseño mecánico / eléctrico. Aquí se aborda el codiseño control / planificación. La metodología de codiseño control / comunicación se puede dividir en dos categorías: control retroalimentado de sistemas computacionales y control en tiempo real.

La categoría de control de sistemas computacionales es también llamada *planificación retroalimentada* [27]. La idea básica es tratar al problema de planificación como un problema de control retroalimentado. Un lazo de control retroalimentado es introducido dentro del manejador de recursos en los sistemas de cómputo. El objetivo de planificación retroalimentada es incrementar la flexibilidad con respecto a las incertidumbres en la utilización de recursos. En lugar de preasignar recursos basados en análisis fuera de línea, los recursos son asignados en línea dinámicamente, basados en retroalimentación de la utilización del recurso actual.

La segunda categoría de codiseño se enfoca en control en tiempo real. Uno de los primeros en establecer que los algoritmos de planificación y sistemas de control no pueden ser diseñados por separado fue Branisky *et al.* [28], quienes trabajan el problema de estabilización de la planta en un problema de planificación óptima contemplando las limitaciones de planificabilidad RM (Rate Monotonic)[29] y las limitaciones de estabilización del NCS, contemplando retardos inducidos por la red, paquetes perdidos, y múltiple transmisión de paquetes.

Branisky considera un conjunto de NCSs con plantas lineales con transmisión únicamente de sensor a controlador con período $h_i(t)$ igual a su tiempo límite, con un tiempo de consumo c_i . Aplicando el algoritmo de planificación RM, se asignan prioridades estáticas a cada planta controlada, a una planta con dinámicas más rápidas se le asigna una prioridad más alta por tener una tasa de transmisión más alta, que a una planta con dinámicas más lentas. El conjunto de N tareas es *factible* si se cumple que el factor de la utilización de la red U es menor que uno.

Para la optimización de la planificabilidad, se asume que cada NCS es asociado con una

función medida de desempeño $J_i(h_i)$, que da el costo de control como una función del período de transmisión h_i . La selección de la función de desempeño es crucial para el problema de optimización. Normalmente se elige un costo cuadrático o exponencial. Branisky analiza el porcentaje de paquetes perdidos permitidos que aseguren el desempeño de un NCS. El NCS con paquetes perdidos es modelado como un Sistema dinámico asíncrono. Se asume que el período de muestreo es constante y que el NCS puede tolerar una cierta cantidad de datos de retroalimentación perdidos. Aún cuando se emplea el concepto de codiseño de NCS, el método tiene algunas desventajas debido a que solo se contemplan la transmisión de datos de sensores al controlador, y se supone un retardo constante para el análisis de paquetes perdidos.

Sename *et al.* [71] utilizan el concepto de control retroalimentado de recursos computacionales para planificar los recursos del CPU y la carga de trabajo. Proponen dos esquemas para el manejo de sistemas con una tarea de control o multitarea. El diseño del sistema de control toma en cuenta los retardos desconocidos, dado a las incertidumbres temporales que son inevitables. Así mismo, presentan un nuevo método de diseño de control por retroalimentación de estado para sistemas con retardo en tiempo discreto como una formulación LMI. Para la planificación retroalimentada, contemplan los retardos de comunicación principalmente la latencia entrada – salida, con el objetivo de ajustar en línea el período de muestreo del controlador, para cumplir los requerimientos de los recursos computacionales. Se contemplan sistemas en tiempo discreto donde el retardo de comunicación es la suma del retardo inducido por la red y el retardo del costo computacional para el cálculo de la entrada de control.

3.2 Pérdida de Paquetes

La pérdida de paquetes es otra imperfección de red con alta prioridad a considerar. En la literatura en general, los paquetes perdidos son modelados por dos procesos: como un proceso aleatorio de Bernoulli, o como un proceso típico de cadena de Markov. Los resultados que aplican procesos de Bernoulli asumen que los paquetes perdidos son independientes e idénticamente distribuidos (i.i.d.) [5,21], mientras que aquellos resultados que usan cadenas de Markov asumen que los paquetes perdidos son ráfagas y ocurren de acuerdo a una cadena de Markov [11,17,18]. La mayoría de los resultados existentes diseñan filtros NCS para estimar los

estados del sistema, con la característica de establecer la entrada a cero, cuando las mediciones son pérdidas. Con esta configuración, la estimación con paquetes perdidos es llamada estimación con observaciones intermitentes o pérdidas [5,17,18,21].

Adaptar un *filtro de Kalman* para estimación, es tomado como uno de los métodos más populares y útiles para problemas de filtrado. Tiene la desventaja de suponer el conocimiento de un modelo preciso del sistema, y que la información estadística de ruidos externos es conocida. En [5], se propone el diseño de un filtro de Kalman contemplando ambos modelos de paquetes perdidos, modelo de Bernoulli i.i.d. y modelo de Gilbert-Elliot (cadenas de Markov discretas con dos estados). Otro resultado que emplea un filtro de Kalman para estimar los estados del sistema para sistemas lineales estocásticos [21] considera el modelo de Bernoulli para modelar los paquetes perdidos, además de considerar incertidumbres del sistema. Smith y Seiler [17] emplearon un filtro de Kalman variante en el tiempo simple para estimar los estados del sistema, considerando un modelo de cadenas de Markov para los paquetes perdidos. En [18] se emplea un filtro de Kalman discreto para paquetes perdidos modelados por una distribución de probabilidad binaria. El análisis y diseño del filtro proporciona una cota en la tasa de arribo de las mediciones.

Otra metodología para estimación es utilizar *filtros H_∞* , que proveen garantía de atenuación de ruido y robustez sobre las incertidumbres del modelo [1]. Los filtros H_∞ para NCS han aparecido recientemente [1,15,22,24]. En [15], el filtro H_∞ se diseña para una clase de NCS estocásticos con pérdida de paquetes consecutivos, separando la tasa de pérdidas para la comunicación sensor – controlador y controlador - actuador.

En cuestión de controladores óptimos, Gupta et. al. [35] desarrollan un controlador LQG para un enlace con un solo paquete perdido entre un sensor y un controlador. Se asume un proceso lineal en tiempo discreto y sin retardos inducidos por la red dentro del lazo de control. El controlador no requiere información del modelo estadístico del evento del paquete perdido. Ambos trabajos, Imer *et al.* [40] y Sinopoli *et al.* [57] desarrollan un controlador LQG sobre enlaces de comunicación no fiables y comparan el desempeño de los protocolos TCP y UDP. Ambos estudios modelan los paquetes perdidos entre los enlaces sensor – controlador y controlador – actuador como procesos de Bernoulli i.i.d.

3.3 Imperfecciones de red

En NCS como se estableció en la sección (), las imperfecciones inducidas por la red son categorizadas en cinco tipos:

- i) *Error de cuantización.*
- ii) *Pérdida de paquetes.*
- iii) *Intervalo/periodo de muestreo variante.*
- iv) *Retardos de comunicación.*
- v) *Limitaciones de comunicación.*

La presencia de cualquiera de estas imperfecciones puede degradar el desempeño del sistema incluso a la inestabilidad. En base a esto, es importante entender cómo estos fenómenos influyen la estabilidad del sistema y las propiedades de desempeño. Desafortunadamente, la mayoría de la literatura disponible se enfoca en algunas de éstas, mientras ignoran las otras. Por ejemplo, métodos sistemáticos que analizan la estabilidad de NCSs considerando solo alguna imperfección. Los errores de cuantización son estudiados en [84][91][97][99][104][111][118]. La pérdida de paquetes en [35][40][57][116][117]. Los intervalos y retardos de muestreo variantes en [107][93] y [89][95][100][103][110][123] respectivamente. Las limitaciones de comunicaciones en [83][90][102][115].

De una manera práctica, casi todos los fenómenos de imperfección de red están presentes en un NCS, lo que enfrenta la necesidad de métodos de análisis y síntesis que incluyan la mayoría de estas imperfecciones para compensarlas. Menos resultados se han reportado que integren múltiples imperfecciones. Algunas de las referencias que consideran dos tipos de imperfección son: [88][96][106] estudian (ii) y (iv), [85][92][101][105][114][120][121][122] consideran (iii) y (v), [98] (i) y (iv), con tres imperfecciones están [112] (i), (iii), (v), [51][87][109] de tipo (ii), (iii), (iv), [113] (ii), (iii) y (v), [94] incorpora (i),(ii),(iv), [86] (iii), (iv) y (v).

Además, algunos de los métodos mencionados que estudian imperfecciones (iii) y (iv) pueden estudiar (ii) como una extensión. Chaillet y Bicchi [6] usan un método de compensación del retardo basado en enviar un paquete de control más grande a la planta, que contiene no solo un valor de control para un instante de tiempo en particular, sino considera una señal de control válida para un horizonte de tiempo futuro. El esquema de control provee cotas sobre el retardo y

el intervalo de transmisión tolerable tal que la estabilidad del NCS es garantizado.

3.4 Resumen

El capítulo presenta una revisión de la literatura para NCS, donde los retardos de tiempo y la pérdida de paquetes son las principales imperfecciones de la red estudiadas en la literatura. Para los retardos de tiempo se han considerado como constantes, aleatorios y estocásticos, presentando herramientas para modelado como funciones de probabilidad y cadenas de Markov. En cuestión de los retardos de tiempo, varias ramas han abordado el problema. Para la rama de comunicaciones, es común contemplan la modificación del protocolos de comunicación en tiempo real para cumplir con los requerimientos de control y del sistema. En la rama de control, la historia es similar donde se desarrollan las metodologías de control continuo y digital, modificados para cubrir los retardos de tiempo. El contemplar el diseño por ramas separadas de NCS representa considerar ciertas suposiciones de diseño que pueden no cumplirse para mantener o mejorar el desempeño del sistema, como contemplar cotas o comportamientos del retardo, ó suponer que no hay pérdidas de paquetes sin previa caracterización de la red al diseñar un controlador. Al diseñar un planificador, suponer que el ancho de banda es suficiente para el sistema, ó que las variaciones en el muestreo son aceptables para efectos de comunicación y que no degradan el desempeño del sistema.

Para sistemas con pérdida de paquetes, en la literatura han proliferados dos modelos de comportamiento de los paquetes perdidos: el modelo de Bernoulli que hace la suposición que los sucesos de paquetes perdidos son independientes e idénticamente distribuidos, y los modelos como cadenas de Markov, que hacen la suposición de que los paquetes perdidos son en ráfagas y tiene una probabilidad de ocurrencia regido por una cadena de Markov. Con cualquier modelo de pérdida de paquetes, en la literatura, un objetivo común es diseñar un filtro para estimar los estados del sistema a través de dos técnicas. La técnica del filtro de Kalman, que hace la suposición de conocer el modelo y la información estadística del ruido de sistema, y la técnica del filtro H_∞ , que supone incertidumbres en el modelo y garantiza la atenuación del ruido y la robustez ante las incertidumbres.

A medida que se intenta abarcar una mayor cantidad de imperfecciones para su compensación, los resultados presentes en la literatura son escasos, restringiéndose a utilizar

técnicas de modelos híbridos, control óptimo, control H_∞ y control estocástico. Esto genera un interés en el desarrollo de técnicas tanto de planificación como codiseño, para compensar el mayor número de imperfecciones inducidas por la red, imperfecciones que normalmente están presentes en NCS de manera inherente.

Capítulo 4

4. Modelo de imperfecciones de red

4.1	MODELO DEL RETARDO IDA Y VUELTA (<i>ROUND TRIP TIME</i> , RTT)
4.2	MODELO DE RETARDOS DE TIEMPO Y PÉRDIDA DE PAQUETES
4.3	MODELO DE IMPERFECCIONES DE TIEMPO
4.4	CONCLUSIONES

En la literatura, existen múltiples técnicas para compensar las imperfecciones de la red. Pero con respecto a la estimación de las imperfecciones, es necesario establecer el tipo de red y protocolo a emplear para definir una estrategia en el modelado de las imperfecciones [54][55][62].

Para el presente trabajo se utiliza la red Ethernet para la comunicación entre los nodos a través de paquetes UDP. Ethernet es una red no confiable y con el envío de paquetes UDP no se tiene certeza sobre el arribo de los paquetes a su destino. Sin embargo, tiene la ventaja de un mayor ancho de banda, debido a que la transmisión de los paquetes se reduce a la mitad al no tener un paquete de agradecimiento enviado del receptor al servidor, y al no esperar por el reenvío de un paquete que se ha perdido.

Este tiene como objetivo la compensación de tres imperfecciones de la red: los retardos de tiempo, paquetes perdidos y muestreo variante. Haciendo la suposición que la resolución en bits para la representación de los datos no afecta el desempeño del sistema, lo que descarta parte de la imperfección de la red por cuantización. Así, también se configura el envío de todos los sensores en un solo paquete, y la aplicación de las señales de control a todos los actuadores al mismo tiempo que corrige en parte la imperfección de la red por restricciones de la comunicación.

El modelo de retardos de tiempo es tomado de la literatura y es acondicionado para obtener el modelo de los retardos con pérdida de paquetes, y el modelo de imperfecciones de tiempo.

4.1 Modelo del retardo ida y vuelta (*Round Trip Time*, RTT)

En configuraciones del NCS, el retardo RTT (Tiempo de ida y vuelta – Round Trip Time)

es el tiempo que tarda un paquete del nodo sensor al nodo actuador. El nodo sensor transmite un paquete con información del proceso hacia el nodo controlador, el nodo controlador calcula la señal de control y la transmite al actuador. El retardo RTT se puede dividir en dos tipos de retardo: retardo sensor – controlador τ_{sc} y retardo controlador – actuador τ_{ca} . Este tipo de modelo del retardo es útil para múltiples configuraciones (Figura 4.1), En A) los sensores y actuadores se encuentran en el mismo nodo, teniendo el nodo controlador la capacidad de recibir mediciones del retardo RTT anteriores para estimar el actual; B) Si se tienen los componentes de control completamente distribuidos, el nodo controlador tiene la capacidad de medir el retardo τ_{sc} y estimar τ_{ca} ; en C) los sensores y el controlador se localizan en el mismo nodo por lo que $\tau_{sc} = 0$ y τ_{ca} es estimado; finalmente en D) el controlador y los actuadores se localizan en el mismo nodo facilitando el cálculo RTT debido a que τ_{sc} puede ser medido y $\tau_{ca} = 0$.

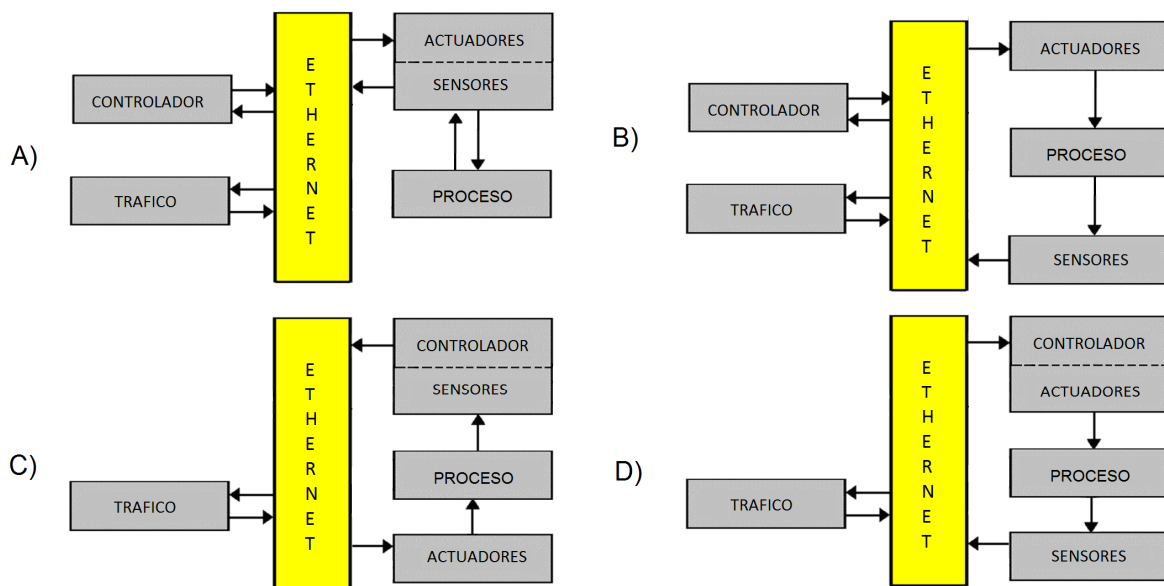


Figura 4.1. Configuraciones NCS comunes con medición de retardos RTT con red Ethernet.

De las configuraciones anteriores, Las configuraciones A y B presentan ambos retardos τ_{sc} y τ_{ca} , y serán utilizadas para aplicar las metodologías de este trabajo. En estas configuraciones se puede calcular el retardo τ_{sc} directamente en el controlador pero es necesario estimar el retardo τ_{ca} . De acuerdo a la clasificación del retardo definida en la sección (), se consideran los tres tipos

de retardo: constante, aleatorio o estocástico.

Se considera que el retardo de red no puede ser descrito por un modelo de retardo constante que es comúnmente un modelo simple invariante en el tiempo en un cierto lapso de tiempo. En este trabajo se emplea un modelo de retardo aleatorio para estimar los retardos RTT. El modelo de retardo aleatorio es un modelo que supone al retardo como un proceso independiente y distribuido idénticamente, que se ajusta a una cierta distribución de probabilidad. La utilización de un modelo estadístico tiene la ventaja que el modelo del retardo es simple de calcular, con fácil implementación para predicción en tiempo real. Además, los parámetros pueden ser dinámicamente adaptados, de acuerdo a diferentes tipos de red y diferentes condiciones de tráfico, que reduce significativamente los errores de predicción.

El algoritmo de distribución exponencial generalizada ha sido usado como modelo del retardo estadístico, donde los parámetros que caracterizan el retardo son (η, φ) , y forman una función de densidad de probabilidad:

$$P[\tau_k] = \begin{cases} \frac{1}{\varphi} e^{-(\tau_k - \eta)/\varphi}, & \tau_k \geq \eta \\ 0, & \tau_k < \eta. \end{cases} \quad (4.1)$$

donde η es la media del retardo y φ es la desviación estándar.

El algoritmo de distribución exponencial se emplea como modelo del retardo RTT. Esta distribución se calcula sobre una ventana de tiempo con w mediciones previas del retardo $\{\tau_{k-w+1}, \dots, \tau_{k-1}, \tau_k\}$, donde el valor esperado del retardo RTT es $E[\tau_{k|w}] = \varphi + \eta$ y con varianza $\sigma^2 = \varphi^2$ [].

El diseño de modelo inicia eligiendo η siendo una elección factible la media de la ventana, que representa el retardo esperado en la ventana de tiempo. Se calcula la distribución a cada uno de los datos de la ventana $\{\tau_{k-w+1}, \dots, \tau_{k-1}, \tau_k\}$. Así, $P[\tau = \hat{\eta}_{k+1|w}]$ es el retardo con la máxima probabilidad, Con este valor se calculan los valores esperados de la media $\hat{\varphi}_{k+1}$ y su desviación estándar $\hat{\eta}_{k+1}$, que forman el retardo estimado, por lo que:

$$\eta_{k+1|w} = w_i | P_{\max} [w] \quad (4.2)$$

$$\hat{\phi}_{k+1lw} = \sqrt{\text{var}\{\tau_{k-w+1}, \dots, \tau_{k-1}, \tau_k\}} \quad (4.3)$$

Siendo el retardo RTT estimado:

$$\hat{\tau}_{k+1lw} = \hat{\phi}_{k+1lw} + \hat{\eta}_{k+1lw} \quad (4.4)$$

La elección de modelo del retardo RTT debe cumplir algunos criterios para ser implementado en un NCS: debe proporcionar robustez al sistema de control, exactitud y eficiencia, además de bajo costo computacional para su estimación en tiempo real. Por esto el valor de los datos de la ventana debe ser suficiente para una buena estimación sin una carga computacional elevada.

4.2 Modelo de retardos de tiempo y pérdida de paquetes

Los retardos de tiempo y pérdida de paquetes presentes en un NCS son los principales problemas a tomar en cuenta para diseñar un controlador [47]. Contrario a los sistemas digitales, un NCS con un período de muestreo pequeño puede en algunos casos mejorar el desempeño del sistema, pero está limitado por el ancho de banda de la red. Por lo que es importante analizar su conducta y, de ser posible, obtener un modelo que permita una acción que minimicé sus efectos.

Basado en una configuración completamente distribuida (Figura 4.1B), donde los sensores son manejados por tiempo, y el controlador y actuadores manejados por evento, se hace la suposición de un reloj común. Este tipo de configuración permite conocer el comportamiento de los paquetes desde el sensor al controlador, pero para efectos de cálculos en el nodo controlador, se tiene la necesidad de estimar el comportamiento de los paquetes del controlador al actuador.

En este caso los retardos de tiempo one way (un solo trayecto) y pérdida de paquetes son tomados como un período de muestreo variable (\hat{h}_v). Para su cálculo en el nodo controlador, primero se calcula el retardo sensor – controlador (τ^{sc}) (ecuación (4.6)) y los paquetes perdidos sensor – controlador (τ^{lsc}) (ecuación (4.7)), y se estima el retardo entre el controlador – actuador ($\hat{\tau}^{ca}$), obteniendo el período de muestreo variable estimado \hat{h}_v .

$$\hat{h}_v = \tau^{sa} + \hat{\tau}^{ca} \quad (4.5)$$

Los paquetes perdidos sensor – controlador (τ_i^{lsc}) sirven para estimar los paquetes perdidos controlador – actuador, que se mide en el tiempo sin información para calcular la señal de

controlador. Éste modifica la estimación del retardo controlador – actuador ($\hat{\tau}^{ca}$). Suponiendo que se tienen en el nodo controlador los estampados de tiempo del arribo de paquetes desde el sensor (c_i) y el estampado de tiempo de envío del sensor (s_i):

$$\begin{aligned}\tau_i^j &= (c_i - \bar{c}_{i-1}) - (s_i - s_{i-1}) \\ \tau^{sc} &= \tau_0^{sc} + \tau_i^j\end{aligned}\quad (4.6)$$

$$\begin{aligned}\bar{c}_i &= c_i - \tau_i^j \\ \tau_i^{lsc} &= ((s_i - s_{i-1}) - T_i)\end{aligned}\quad (4.7)$$

donde τ_i^j es el jitter (variación del retardo). Éste forma el retardo actual sensor – controlador τ^{sc} , junto con el retardo inicial desconocido τ_0^{sc} . \bar{c}_i es el tiempo de ejecución en el controlador c_i sin jitter, con s_i y c_i como los tiempos estampados de los paquetes en el sensor y controlador respectivamente. τ_i^{lsc} son los paquetes perdidos sensor – controlador en cada instante i , y T_i es el período de muestreo en el instante i (Figura 4.2).

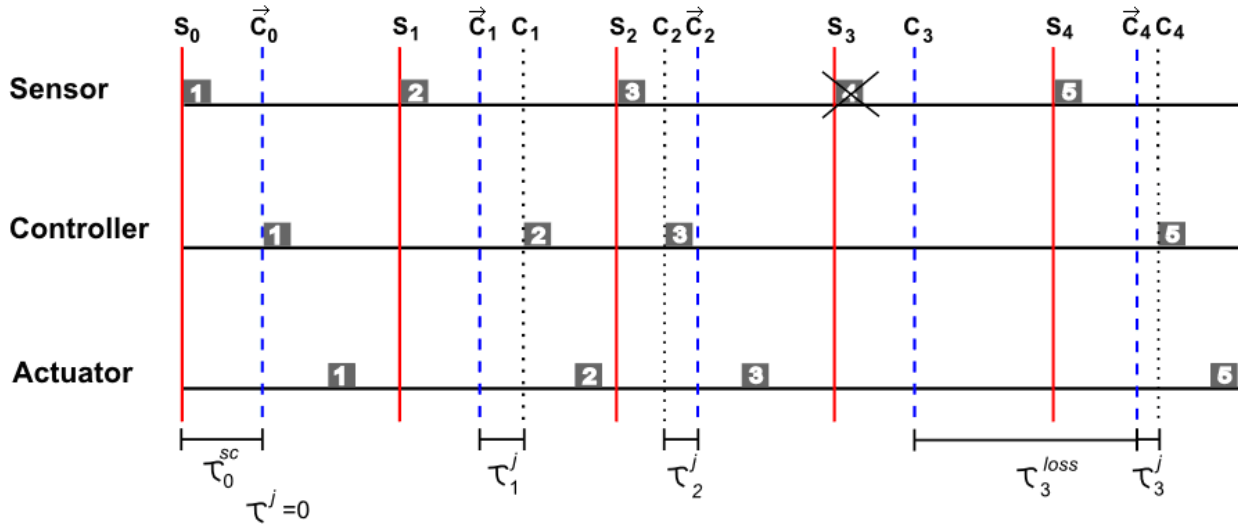


Figura 4.2. Diagrama de tiempos de paquetes enviados en un NCS.

Las Ecuaciones (4.6) y (4.7) se calculan en línea cada vez que un nuevo paquete llega al nodo controlador. Para estimar el tiempo de arribo de la señal de control en el nodo actuador y utilizarlo en el control difuso. A través de mediciones de tiempo entre el sensor - actuador fuera de línea, con múltiples escenarios de tráfico en el nodo actuador, se obtienen los parámetros de la distribución exponencial. Se realiza el análisis estadístico y se caracteriza la media y la

desviación estándar $q_E = [\eta, \varphi]^T$ del retardo sensor – actuador τ^{sa} . Estos son usados para formar la función densidad de probabilidad[]:

$$\mathbf{P}[\tau^{sc}] = \begin{cases} \frac{1}{\varphi} e^{-(\tau^{sc}-\eta)/\varphi}, & \tau_k \geq \eta \\ 0, & \tau_k < \eta \end{cases} \quad (4.8)$$

Esta función $P[\tau^{sc}]$ es calculada para la ventana de tiempo con los w retardos τ^{sc} previos $\{\tau_{k-w+1}^{sc}, \dots, \tau_{k-1}^{sc}, \tau_k^{sc}\}$, donde el valor esperado del retardo $\hat{\tau}^{sa}$ es $E[\tau_{klw}^{sc}] = \eta_{k+llw} + \varphi_{k+llw}$. Primero, se obtiene la nueva media η_{k+llw} que es el retardo previo con el máximo de la función de probabilidad para la ventana de tiempo w (4.9) y la nueva desviación estándar φ_{k+llw} es la raíz cuadrada de la varianza de w (4.10).

$$\eta_{k+llw} = w_i | P_{\max} [w] \quad (4.9)$$

$$\varphi_{k+llw} = \sqrt{\text{var}\{\tau_{k-w+1}^{sc}, \dots, \tau_{k-1}^{sc}, \tau_k^{sc}\}} \quad (4.10)$$

Así, el retardo controlador – actuador τ^{ca} estimado es:

$$\hat{\tau}_k^{ca} = \varphi_{k+llw} + \eta_{k+llw} + \tau_{klw}^{lca} \quad (4.11)$$

donde τ_{klw}^{lca} es la probabilidad de pérdida del paquete controlador – actuador, el retardo controlador – actuador $\hat{\tau}^{ca}$ es utilizado para estimar el período de muestreo \hat{h}_v del NCS (4.5).

4.3 Modelo de imperfecciones de tiempo

A fin de incorporar las imperfecciones de tiempo mencionadas en la sección (), se establece un solo modelo basado en la configuración one-shot propuesta para sistema empotrados [135]. Esta configuración es aplicada a una arquitectura completamente distribuida NCS (Figura 4.1B), donde los nodos sensor, controlador y actuador intercambian datos vía Ethernet; pero es fácilmente aplicable al resto de las configuraciones presentadas. El nodo sensor es manejado por tiempo con período no estricto; el nodo controlador es manejado por evento; y el actuador es manejado por tiempo con período estricto. Manejar el nodo sensor con un período no estricto permite al nodo realizar otras tareas, además de un mayor rango en el consumo de tiempo de la

tarea de adquisición de datos. Manejar el nodo controlador por evento permite calcular la señal de control en el instante que arriba la medición desde el nodo sensor sin necesidad de tiempo de espera. Para el nodo actuador manejarlo por período estricto permite manejar la periodicidad de la señal de control aplicada, disminuyendo los efectos de variaciones en el período de muestreo y en los retardos de tiempo.

La configuración estándar para el diseño de NCS, considera el nodo sensor manejado por tiempo, mientras el controlador y actuador manejados por evento. Utilizando un modelo en espacio de estado de un sistema lineal en tiempo discreto invariante en el tiempo con un período de muestreo h [17]

$$\begin{aligned}x_{k+1} &= \Phi_h x_k + \Gamma_h u_k \\ y_k &= Cx_k\end{aligned}\tag{4.12}$$

donde x_k es el estado de la planta, u_k y y_k son las entradas y salidas de la planta, la matriz $C \in \mathbb{R}^{p \times n}$ es la matriz de salida y las matrices Φ_h y Γ_h son obtenidas usando:

$$\Phi_h = e^{Ah}, \quad \Gamma_h = \int_0^h e^{As} B ds\tag{4.13}$$

con $A \in \mathbb{R}^{n \times n}$ y $B \in \mathbb{R}^{n \times m}$ son las matrices del sistema y entrada del modelo en forma continua:

$$\begin{aligned}\dot{x}(t) &= Ax + Bu(t) \\ y(t) &= Cx(t)\end{aligned}\tag{4.14}$$

Para la operación en lazo cerrado estándar de (4.12), el comando de control u_k es dado por:

$$u_k = Lx_k, \quad L \in \mathbb{R}^{1 \times n}\tag{4.15}$$

donde L es la ganancia de retroalimentación de estado obtenida usando diseño de control estándar desde las matrices Φ_h y Γ_h .

La técnica de control retroalimentado estándar [17] supone el instante de muestreo y la aplicación del control a la planta al mismo tiempo. Esto es físicamente imposible, debido a que el sistema de control tiene una plataforma basada en procesador y el cómputo del comando de control toma tiempo.

La ecuación (4.12) puede ser aumentada para incorporar el modelo del retardo de tiempo y latencias I/O que aparecen dado al cómputo del algoritmo de control o dado a la inserción de la

red dentro del lazo de control, en el caso de NCS. El modelo estándar que incorpora el retardo de tiempo τ , con $\tau < h$, es [17] :

$$x_{k+1} = \Phi_h x_k + \Phi_{h-\tau} \Gamma_\tau u_{k-1} + \Gamma_{h-\tau} u_k \quad (4.16)$$

El modelo (4.17) es la forma más simple utilizada para analizar y diseñar controladores en NCS. Este modelo asume un tiempo de referencia dado por los instantes de muestreo con un retardo de tiempo fijo desde el muestreo a la actuación. Pero si el retardo de tiempo es variante, más grande que el período de muestreo y/o el intervalo de muestreo es variante, este modelo es poco viable para NCS[.]

Bajo el modelo One-Shot [135] a diferencia del control retroalimentado estándar, el modelo sincroniza la operación de cada lazo cerrado a través de los instantes de actuación y no sobre los instantes de muestreo como se realiza con el controlador retroalimentado estándar [17]. Así, el tiempo transcurrido entre instantes de actuación consecutivos, etiquetados t_{k-1} y t_k , es el período de actuación h . Dentro de este intervalo, el estado del sistema es muestreado, etiquetado por $x_{s,k} \in (t_{k-1}, t_k)$, y el tiempo de muestreo es registrado, $t_{s,k}$. La diferencia entre este tiempo y el siguiente instante de actuación t_k es:

$$\tau_k = t_k - t_{s,k} \quad (4.18)$$

y es usado para estimar el estado en el instante de actuación:

$$\hat{x}_k = \Phi_{\tau_k} x_{s,k} + \Gamma_{\tau_k} u_{k-1} \quad (4.19)$$

Finalmente, haciendo uso de \hat{x}_k , el comando de control es computado como:

$$u_k = L \hat{x}_k, \quad L \in \mathbb{R}^{1 \times n} \quad (4.20)$$

donde L es la matriz de retroalimentación que puede calcularse con diversas técnicas de control clásico. El comando de control u_k es establecido contante durante instantes de actuación.

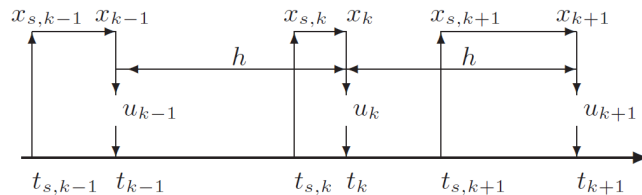


Figura 4.3. Modelo One-shot

Una estrategia de control usando las ecuaciones (4.18)-(4.20) se basa en la referencia de

tiempo dada por los instantes de actuación. Por lo que, las muestras no son requeridas a ser estrictamente periódicas así que τ_k en (4.18) puede variar en cada operación [134].

En cada actuación, el nodo actuador, después de aplicar la señal de control a la planta, e.g. u_{k-1} , genera el siguiente instante de actuación t_k , que es enviado al sensor. El sensor, recibiendo este paquete o llegado el siguiente período de muestreo; muestrea la planta $x_{s,k}$ y registra el tiempo absoluto $t_{s,k}$. Este último, junto con t_k es usado para computar τ_k . Ambos $x_{s,k}$ y τ_k son enviados al nodo controlador. Con la recepción de este paquete, el nodo controlador estima el estado de la planta que se tendrá en el tiempo t_k (4.19) con el siguiente retardo de tiempo estimado τ_{k+1} , y computa el comando de control u_k (4.20). El último es enviado al actuador que aplicará a la planta en el instante de actuación siguiente.

Con la utilización del modelo One-Shot, la variabilidad del retardo de tiempo, paquetes perdidos y el intervalo de muestreo son transformados a retardos múltiples del período de actuación. Por lo que la estimación del retardo de tiempo se centra sólo a múltiplos del período de actuación. El retardo es estimado en el nodo controlador, cuando éste recibe un nuevo paquete, que contiene el estado $x_{s,k}$ y el retardo de tiempo τ_{k-1} . El retardo τ_{k-1} se utiliza en el algoritmo de distribución exponencial presentado en la sección () para estimar el retardo.

4.4 Resumen

El capítulo presenta tres modelos de imperfecciones inducidas por la red. Se presenta la configuración y el método probabilístico para la estimación de cada uno, y presentando una función de distribución exponencial generalizada.

En configuraciones donde sea posible tener una referencia de tiempo común, es posible implementar métricas de las imperfecciones de tiempo con viaje completo, como es el caso de retardo RTT, permitiendo obtener estimaciones del retardo en función de mediciones anteriores de su comportamiento.

En configuraciones donde no se emplee algún algoritmo de sincronización o referencia de tiempo común, es posible implementar métricas de un solo envío, como es el caso de retardos one way, pérdida de paquetes o intervalos de muestreo variantes. Permitiendo obtener estimaciones

del tiempo generado por las imperfecciones de la red en todo el lazo de control, a través de la información desde el sensor – controlador. Todo esto dentro del nodo controlador para calcular la señal de control que controle el sistema y compense las imperfecciones de la red.

Capítulo 6

5. Codiseño Difuso para NCS

6.1	MODELO DIFUSO
6.2	CONTROLADOR DIFUSO
6.3	PLANIFICADOR DIFUSO
6.4	CONCLUSIONES

El codiseño control – planificación tiene como objetivo que el sistema o conjunto de sistemas, que integran elementos de comunicación con recursos computacionales limitados, sean controlados. Diseñar un conjunto de controladores y planificar sus tareas y transmisiones tal que el desempeño de control de todos los sistemas sea optimizado[.]

Bajo este objetivo se propone un codiseño difuso que permite incorporar algunas de las imperfecciones de la red, diseñando leyes de control que garanticen la estabilidad del sistema ante estas imperfecciones, y planificando las transmisiones en base a estas imperfecciones y el desempeño del sistema.

Los modelos de imperfecciones del Capítulo 4, son la base inicial del codiseño difuso. Con estos modelos se diseña de manera integral la parte antecedente de los sistemas difusos utilizados para el codiseño. Estos sistemas se dividen en: un controlador difuso basado en un modelo difuso y un planificador difuso. El modelo difuso modifica su conducta de acuerdo al comportamiento de la red de comunicación. El controlador difuso se diseña para compensar las imperfecciones de la red, manteniendo desempeño óptimo ante imperfecciones estáticas. Por último el planificador difuso observa las imperfecciones de la red y el error del sistema para modificar la transmisión de paquetes, a fin de optimizar el desempeño del NCS.

5.1 Parte Antecedente Difusa

Un sistema difuso captura la conducta dinámica de un sistema en forma de reglas IF-THEN. Cada regla IF-THEN define el conjunto de factores para los cuales se cumple la regla (parte antecedentes IF) generando un conjunto de acciones (parte consecuente THEN) a

ejecutarse.

La parte antecedente es común a los sistemas difusos propuestos, ésta incorpora la información proporcionada por el modelo de imperfecciones.

El objetivo de la parte antecedente es dividir el rango de tiempo de las imperfecciones.

$$\alpha_j(\tau) = \exp\left(\frac{-(\tau - \rho_j)^2}{\sigma_j^2}\right) \quad (5.1)$$

donde $\alpha_j(\tau)$ es la función de membresía Gaussiana de la j -ésima regla difusa y tiene las propiedades:

$$\psi_j = \frac{\alpha_j}{\sum_{s=1}^r \alpha_s} \quad \psi_j \geq 0 \quad \sum_{j=1}^r \psi_j = 1 \quad (5.2)$$

Donde ψ_j es el grado de disparo normalizado.

Mediante el comportamiento de las imperfecciones de la red a compensar, se obtiene la cota máxima (\bar{h}) con la cual los sistemas difusos serán válidos. Con esta información se asignan los parámetros (ρ_j, σ_j) para cada una de las funciones de membresía, donde ρ_j son los centros y σ_j las desviaciones estándar de las gaussianas.

Los centros ρ_j se establecen por diseño y se busca una división equitativa del rango de tiempo de las imperfecciones. Mientras que para los parámetros σ_j se calcula de tal forma que no se traslapen más de dos funciones de membresía gaussiana.

La **¡Error! No se encuentra el origen de la referencia.** muestra un ejemplo de la división del tiempo de imperfecciones, se muestra el grado de disparo normalizado ψ_j (5.2) de cuatro gaussianas con una cota máxima en tiempo de 25 ms.

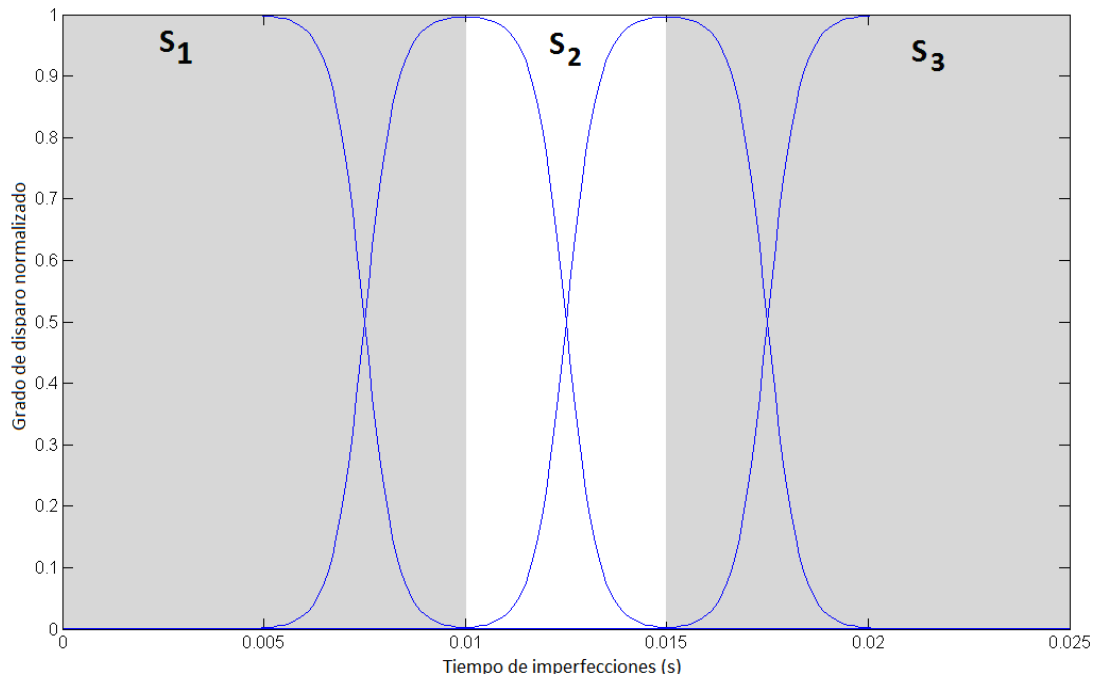


Figura 5.1. Funciones de membresía con cuatro períodos de muestreo.

5.2 Modelo Difuso

El modelo difuso inicia con el diseño de la parte antecedente descrita en la sección anterior, para completarlo se diseña la parte consecuente. Como parte consecuente se definen modelos discretos del sistema con diferentes períodos de muestreo igual al número de reglas generadas en la parte antecedente.

El diseño inicia con un modelo del sistema no lineal de la forma:

$$\begin{aligned}\dot{\mathbf{x}}(t) &= \mathbf{F}(\mathbf{x}, \mathbf{u}) \\ \mathbf{y}(t) &= \mathbf{G}(\mathbf{x})\end{aligned}\tag{5.3}$$

donde $F(\bullet)$ y $G(\bullet)$ son funciones no lineales que representan la dinámica interna y la salida del sistema, respectivamente. La linealización se realiza alrededor de un punto de operación (x_0, u_0) .

Las dinámicas del sistema resultante son descritas por:

$$\begin{aligned}\mathbf{x}(t) &= \mathbf{A}\mathbf{x}(t) + \mathbf{B}\mathbf{u}(t) \\ \mathbf{y}(t) &= \mathbf{C}\mathbf{x}(t)\end{aligned}\tag{5.4}$$

$$\mathbf{A} = \left. \frac{\partial \mathbf{F}}{\partial \mathbf{x}} \right|_{\substack{u=u_0 \\ x=x_0}} \quad \mathbf{B} = \left. \frac{\partial \mathbf{F}}{\partial \mathbf{u}} \right|_{\substack{u=u_0 \\ x=x_0}} \quad \mathbf{C} = \left. \frac{\partial \mathbf{G}}{\partial \mathbf{x}} \right|_{\substack{u=u_0 \\ x=x_0}} \quad (5.5)$$

donde $\mathbf{x}(t) \in R^n$, $\mathbf{u}(t) \in R^m$. $\mathbf{x}(t)$ son los estados del sistema, $y(t)$ la salida del sistema y $\mathbf{u}(t)$ la entrada controlada respectivamente. \mathbf{A} , \mathbf{B} y \mathbf{C} son matrices de dimensiones apropiadas.

La parte consecuente ésta compuesta por modelos lineales discretos con períodos de muestreo iguales a los centros de las funciones gaussianas $\rho_j()$. El modelo discreto con el período de muestreo h_j tiene la forma:

$$\begin{aligned} x_j(k+1) &= \Phi_j x(k) + \Gamma_j u(k) \\ \Phi_j &= \exp(\mathbf{A}h_j) \\ \Gamma_j &= \int_0^{h_j} e^{\mathbf{A}s} ds \mathbf{B} \end{aligned} \quad (5.6)$$

Definiendo r reglas difusas de la forma:

$$\begin{aligned} \text{if } \tau(k+1) \text{ is } \alpha_j(\tau) \text{ then } x_j(k+1) &= \Phi_j x(k) + \Gamma_j u(k) \\ j &= 1, \dots, r \end{aligned} \quad (5.7)$$

donde $\tau(k+1)$ es el tiempo estimación de las imperfecciones de red.

El sistema completo tiene la forma:

$$x(k+\tau) = \sum_{j=1}^r \psi_j (\Phi_j x(k) + \Gamma_j u(k)) \quad (5.8)$$

A manera de ejemplo, La Figura 5.2 muestra el concepto básico de la elección de modelos discretos con múltiples períodos de muestreo. Se muestra el estado de un sistema obtenido de 3 modelos discretos con períodos de muestreo $[0.001 \ 0.002 \ 0.004]$, En la cuarta gráfica se muestra la reconstrucción del estado con el conocimiento a priori del tiempo de las imperfecciones de red. El modelo difuso permite un cambio suave entre los diferentes modelos discretos, eligiendo el mejor modelo discreto para el comportamiento de tiempo de imperfecciones estimado en cada instante dado.

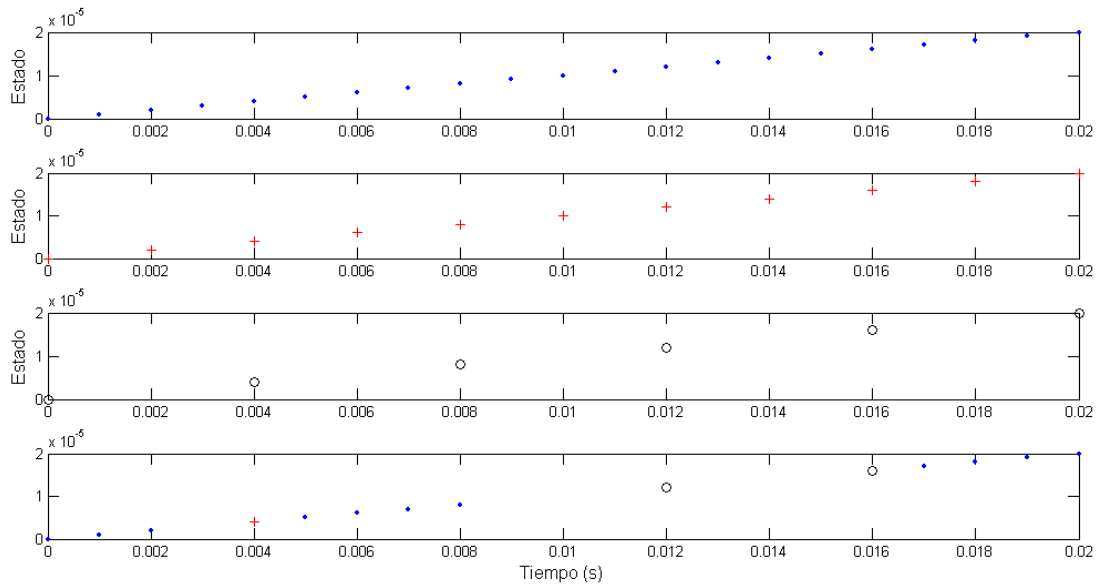


Figura 5.2. Concepto básico para la elección de los modelos discretos con múltiples períodos.

5.3 Controlador Difuso

Del mismo modo que el modelo difuso, el controlador difuso utiliza la parte antecedente de la sección () para definir sus reglas IF – THEN. En este caso los modelos lineales discretos del modelo difuso, permiten diseñar una ley de control por retroalimentación de estados difusa:

$$u(k) = -\sum_{j=1}^r \psi_j K_j x(k) \quad (5.9)$$

donde K_j es la matriz de retroalimentación del j -ésimo modelo discreto y se diseña por una estrategia LQR (Linear Quadratic Regulator) con el objetivo de minimizar el índice de desempeño cuadrático:

$$J_j(u) = \sum_{k=0}^{\infty} \{x^T(k) \mathbf{Q}_j x(k) + u^T(k) \mathbf{V}_j u(k)\} \quad (5.10)$$

donde $\mathbf{Q}_j = \mathbf{Q}_j^T \geq 0$ y $\mathbf{V}_j = \mathbf{V}_j^T \geq 0$. Se asume que los componentes del vector de control no están limitados. El diseño del control por LQR de los modelos discretos requiere la solución de las ecuaciones algebraicas de Riccati Discretas para las matrices \mathbf{W}_j [82].

$$\Phi_j^T \mathbf{W}_j \Phi_j - \mathbf{W}_j + \mathbf{Q}_j - \Phi_j^T \mathbf{W}_j \Gamma_j (\mathbf{V}_j + \Gamma_j^T \mathbf{W}_j \Gamma_j)^{-1} \Gamma_j^T \mathbf{W}_j \Phi_j = 0 \quad (5.11)$$

Así, las matrices de retroalimentación se calculan como:

$$K_j = (\mathbf{V}_j + \mathbf{\Gamma}_j^T \mathbf{W}_j \mathbf{\Gamma}_j)^{-1} \mathbf{\Gamma}_j^T \mathbf{W}_j \mathbf{\Phi}_j \quad j = 1, \dots, r \quad (5.12)$$

El sistema en lazo cerrado tiene la forma:

$$\begin{aligned} x(k+1) &= \sum_{i=1}^r \sum_{j=1}^r \psi_i \psi_j (\mathbf{\Phi}_i - \mathbf{\Gamma}_i K_j) x(k) \\ &= \sum_{i=1}^r \sum_{j=1}^r \psi_i \psi_j \mathbf{\Lambda}_{ij} x(k) \end{aligned} \quad (5.13)$$

con $\mathbf{\Lambda}_{ij} = \mathbf{\Phi}_i - \mathbf{\Gamma}_i K_j$

Para analizar el control difuso (Ecuación (5.13)), se consideran las siguientes propiedades de la parte antecedente:

$$\psi_i \psi_j \geq 0 \quad \sum_{i=1}^r \sum_{j=1}^r \psi_i \psi_j = 1 \quad \sum_{i=1}^r \psi_i^2 + 2 \sum_{i < j} \psi_i \psi_j = 1 \quad (5.14)$$

Para probar la estabilidad del control difuso es necesario apoyarse en el siguiente lema.

Lema 1. (Guan, 2004) Para matrices $\mathbf{A}_{ij}, \mathbf{B}_{kg}, \mathbf{P} > 0 \in \mathfrak{R}^{n \times n}$ for $i = 1, \dots, r$, se cumple la siguiente desigualdad

$$\begin{aligned} &2 \sum_{i=1}^r \sum_{j=1}^r \sum_{k=1}^r \sum_{g=1}^r \psi^i \psi^j \psi^k \psi^g \mathbf{A}_{ij}^T \mathbf{P} \mathbf{B}_{kg} \leq \\ &\sum_{i=1}^r \sum_{j=1}^r \psi^i \psi^j (\mathbf{A}_{ij}^T \mathbf{P} \mathbf{A}_{ij} + \mathbf{B}_{ij}^T \mathbf{P} \mathbf{B}_{ij}) \end{aligned} \quad (5.15)$$

Donde las propiedades del grado de disparo normalizado ψ tiene propiedades (5.14).

Teorema 1. El estado de equilibrio, $x_e = 0$ del sistema en lazo cerrado (Ecuación (5.13)), con entrada de control (Ecuación (5.9)), con a lo más dos funciones de membresía normalizadas traslapadas, es asintóticamente estable en el rango definido de las imperfecciones de la red, si existen μ matrices $\mathbf{P}_s = \mathbf{P}_s^T > 0$ tal que:

$$\mathbf{\Lambda}_{ii}^T \mathbf{P}_s \mathbf{\Lambda}_{ii} - \mathbf{P}_s < 0 \quad i \in S_s \quad (5.16)$$

$$(\mathbf{\Lambda}_{ij} + \mathbf{\Lambda}_{ji})^T \mathbf{P}_s (\mathbf{\Lambda}_{ij} + \mathbf{\Lambda}_{ji}) - 2\mathbf{P}_s < 0 \quad i, j \in S_s \quad i < j \in S_s \quad (5.17)$$

con $\mathbf{\Lambda}_{ij} = \mathbf{\Phi}_i - \mathbf{\Gamma}_i K_j$, donde $S = \{S_1, S_2, \dots, S_\mu\}$ son las μ son las regiones de traslape, S_x contiene los índices de las funciones de membresía que conforman la región x .

Demostración. Suponiendo que existen μ matrices $\mathbf{P}_s = \mathbf{P}_s^T > 0$ tal que (5.16) y (5.17) son

satisfechas. Se considera la siguiente función candidata de Lyapunov.

$$V(k) = x(k)^T \left(\sum_{s=1}^{\mu} \lambda_s \mathbf{P}_s \right) x(k) \quad (5.18)$$

donde

$$\lambda_s(\hat{t}_{k+1}) \begin{cases} 1 & \hat{t}_{k+1} \in S_s \\ 0 & \hat{t}_{k+1} \notin S_s \end{cases} \quad (5.19)$$

$$\sum_{s=1}^{\mu} \lambda_s(\hat{t}_{k+1}) = 1$$

Es fácil mostrar las dos primeras condiciones para que sea una función de Lyapunov

1. $\mathbf{V}(0) = 0$, $\mathbf{V}_k > 0$ para $x_k \neq 0$
2. $\mathbf{V}(\mathbf{x}) \rightarrow \infty$ como $\|x_k\| \rightarrow \infty$

Es suficiente mostrar que $\Delta \mathbf{V}(x_k) < 0$ para probar que \mathbf{V}_k es una función de Lyapunov y el sistema es estable. Se tiene que:

$$\Delta \mathbf{V}_k = \mathbf{V}_{k+1} - \mathbf{V}_k = \sum_{s=1}^{\mu} \lambda_s (x_{k+1}^T \mathbf{P}_s x_{k+1}) - \sum_{s=1}^{\mu} \lambda_s (x_k^T \mathbf{P}_s x_k)$$

$$\Delta \mathbf{V}_k = \sum_{s=1}^{\mu} \lambda_s \mathbf{L}_s \quad \mathbf{V}_k^s = x_k^T \mathbf{P}_s x_k$$

$$\mathbf{L}_s = \mathbf{V}_{k+1}^s - \mathbf{V}_k^s \quad (5.20)$$

Es suficiente mostrar que:

$$\mathbf{L}_s < 0 \quad s = 1, \dots, \mu$$

Sustituyendo \mathbf{V}_{k+1}^s y \mathbf{V}_k^s en (5.20) se tiene:

$$\mathbf{L}_s = \left(\sum_{i \in S_s} \sum_{j \in S_s} \psi_i \psi_j \Lambda_{ij} x \right)^T \mathbf{P}_s \left(\sum_{i \in S_s} \sum_{j \in S_s} \psi_i \psi_j \Lambda_{ij} x \right) - x^T \mathbf{P}_s x \quad (5.21)$$

Aplicando las propiedades (5.14) en (5.21) se tiene

$$= x^T \left(\sum_{i \in S_s} \sum_{j \in S_s} \sum_{k \in S_s} \sum_{g \in S_s} \psi_i \psi_j \psi_k \psi_g (\Lambda_{ij}^T \mathbf{P}_s \Lambda_{kg} - \mathbf{P}_s) \right) x$$

Por el lema 1(5.15), \mathbf{L}_s es:

$$\mathbf{L}_s \leq x^T \left(\sum_{i \in S_s} \sum_{j \in S_s} \psi_i \psi_j (\Lambda_{ij}^T \mathbf{P}_s \Lambda_{ij} - \mathbf{P}_s) \right) x \quad (5.22)$$

$$\leq x^T \left(\sum_{i \in S_s} \psi_i^2 (\Lambda_{ii}^T \mathbf{P}_s \Lambda_{ii} - \mathbf{P}_s) + \sum_{i \in S_s} \sum_{j \in S_s}^{j < i} \psi_i \psi_j \left((\Lambda_{ij} + \Lambda_{ji})^T \mathbf{P}_s (\Lambda_{ij} + \Lambda_{ji}) - 2\mathbf{P}_s \right) \right) x \quad (5.23)$$

$$\mathbf{L}_s < 0 \rightarrow \Delta \mathbf{V}_k < 0$$

El primer término en (5.23) es negativo definido por (5.16). El segundo término en (5.23) es negativo definido por (5.17). Así, la función cuadrática definida positiva (5.18) es una función de Lyapunov para el control difuso (5.9). Esto implica estabilidad asintótica en el rango de las imperfecciones de la red.

5.4 Planificador Difuso

En esta sección, la teoría de planificación es usada para diseñar un planificador retroalimentado que asigne en línea el tiempo de ejecución del NCS en base al desempeño del sistema de control y de las condiciones de carga de la red. El cambio del período de muestreo o actuación es realizado con un período de planificación. La idea básica es mantener la tasa de plazos perdidos y el desempeño del sistema en un nivel deseado ajustando el período de muestreo. Se propone un planificador difuso local para cada nodo sensor presente en la red de comunicación, que en base al tráfico externo, realice una planificación dinámica también llamada planificación por retroalimentación.

La planificación dinámica propuesta tiene la configuración de la Figura 5.3, cuyo comportamiento es el siguiente: el sensor envía paquetes con información del sistema y su período de ejecución al controlador con un período h . El controlador añade la señal de error y control en el paquete y lo envía al actuador. El actuador contiene calcula el desempeño del sistema y la tasa de límites perdidos cada período de planificación y lo envía al sensor. Finalmente, el sensor modifica el período h , en base a la información de la tasa de límites perdidos y del desempeño del sistema. Un plazo límite h_{\max} (*deadline*) ocurre cuando el paquete arriba al nodo actuador después de un tiempo límite o cuando el paquete se pierde.

En términos de control, la variable manipulada es el período de muestreo y las variables

controladas serán la tasa de plazos perdidos y el desempeño del sistema.

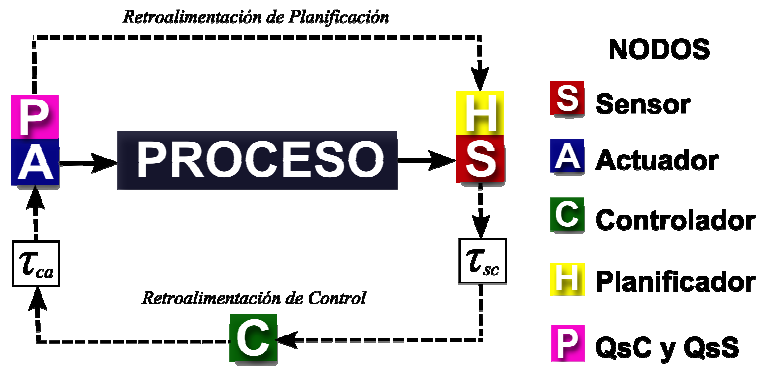


Figura 5.3. Configuración de Codiseño

El módulo del nodo actuador calcula el desempeño del sistema Δe a través del error absoluto medio (EAM) de “ n ” paquetes recibidos con un período de planificación δ . Mientras que la tasa de plazos perdidos Δh se calcula con la información del total de paquetes recibidos “ m ” que hayan excedido su plazo límite h_{\max} más los paquetes perdidos λ durante el período de planificación.

$$\Delta e = \frac{\sum_{k \in n} |e_k|}{n} \quad (5.24)$$

$$m = \{\forall k \in n \mid \tau_k > h_{\max}\} + \lambda \quad (5.25)$$

$$\Delta h = \frac{m}{\delta} \sum_{k \in n} h_k$$

Donde λ es el número de paquetes perdidos durante el período de planificación, τ_k es el retardo de tiempo sensor – actuador medido durante el período de planificación, y h_{\max} es el plazo límite para el paquete arribado.

La elección de la tasa de plazos perdidos como una variable controlada al ser una de las métricas más comunes para la calidad de servicio en red (QoS), y desde el punto de vista de tiempo real, es también un factor importante que afecta la calidad de control (QoC). El QoS mantiene controlado la tasa de plazos perdidos en un nivel bajo aceptable. Además, usar Δh como variable controlada puede direccionar simultáneamente los problemas de retardos de tiempo variables y paquetes perdidos.

Cuando Δh se mantiene en un nivel bajo, los retardos de la mayoría de los paquetes es menor al plazo y el número de paquetes perdidos es limitado. Como una consecuencia, el impacto

del retardo y los paquetes perdidos en el QsC es minimizado.

El período de muestreo afecta la tasa de plazos perdidos, con períodos de muestreo cortos se incrementa la utilización de la red, que inerentemente causa que las imperfecciones de la red se incrementen y viceversa. Con carga de tráfico fuerte, la probabilidad de colisiones entre los nodos es mayor, que potencialmente incrementa tanto el retardo de tiempo y los paquetes perdidos y a su vez la tasa de plazos perdidos. Por otro lado, una tasa grande de plazos perdidos generalmente puede ser reducida incrementando el período de muestreo, particularmente cuando la red esta en condiciones de sobrecarga.

De acuerdo a la teoría de control de datos muestreados, períodos de muestreo cortos genera un mejor QoC. En este contexto, la QoC puede ser mejorada incrementando la eficiencia de utilización de la red, ajustando el período de muestreo. Esto justifica la elección del período de muestreo como variable manipulada que es ajustada con respecto a las condiciones de la red, donde las variaciones en la carga de tráfico impredecible y dinámico en el NCS pueden ser compensados con efectividad.

Con la métrica del QsC (Δe) y QsS (Δh) se diseña el planificador difuso como un problema de control donde las métricas forman la parte antecedente y la parte consecuente son leyes de retroalimentación paralela. La i -ésima regla difusa tiene la forma de:

$$\text{Rule } i: \text{ IF } \Delta e \text{ is } \beta_1 \text{ and } \Delta h \text{ is } \beta_2 \text{ THEN } \bar{h} = Fz$$

donde β son las funciones de membresía y $z \in \mathfrak{R}^2$ con $z = [\Delta e \ \Delta h]$, F la matriz de retroalimentación de planificación. El planificador difuso completo queda como:

$$\bar{h} = \sum_{i=1}^M \prod_{j=1}^2 \beta_{ij} F_i z \quad (5.26)$$

donde M es el número de reglas difusas. El nuevo período de muestreo es asignado en el intervalo:

$$h^+ = \{h_{\min} \leq \bar{h} \leq h_{\max}\} \quad (5.27)$$

Dado a la ausencia de un modelo matemático que describa la relación entre la tasa de plazos perdidos, el EAM del sistema y el período de muestreo, los parámetros son determinados basados en simulación.

5.5 Conclusiones

En el capítulo se presenta la parte fundamental del trabajo de doctorado, esto es el modelo y controlador difuso para NCS compensando las imperfecciones de tiempo inducidas por la red.

El modelo difuso cumple la función de aproximador universal con el fin de incorporar el comportamiento del sistema ante imperfecciones de tiempo a través de la parte antecedente, el modelo puede ser aplicado para incorporar una o varias imperfecciones de tiempo debido a que el objetivo es seccionar el tiempo generado por las imperfecciones para la elección del mejor modelo que defina el comportamiento del sistema NCS. El modelo difuso puede ser diseñado para cualquier proceso modelado en variables de estado, por lo que es aplicable tanto al caso del estudio MAGLEV y el simulador helicóptero 2-DOF.

El control difuso empleado tiene la característica de utilizar la misma estructura del modelo difuso propuesto, aplicando una ley de control retroalimentada paralela, obteniendo las matrices de retroalimentación para cada submodelo discreto a través de un diseño LQR. El controlador difuso considera el desempeño del sistema y el comportamiento de la red. El objetivo de control es mantener el desempeño del sistema en un nivel deseado ante el comportamiento dinámico e impredecible de la red, teniendo la posibilidad de obtener la mejor respuesta del sistema ante imperfecciones de tiempo estáticas y robustez ante variaciones en las imperfecciones.

Con la presentación de un planificador difuso se completa el codiseño de una clase de NCSs, el planificador difuso incorpora información del desempeño del sistema y el comportamiento de la red para seleccionar el período de muestreo del nodo sensor. El objetivo de planificación es mantener la tasa de plazos perdidos y el EAM del sistema en niveles deseados adaptando el período de muestreo cada período de planificación.

Ambos el controlador y planificador propuestos contemplan tanto el desempeño del sistema y de la red con el objetivo final de mejorar y mantener el desempeño del sistema. El controlador difuso puede compensar tanto las variaciones en el desempeño de la red como el cambio en el período de muestreo adaptado por el planificador. Mientras que el planificador difuso puede compensar las variaciones de carga de la red y mejorar el desempeño del sistema cuando el controlador por su diseño no tenga esta capacidad, Ej. con período de muestreo largos aumenta la sensibilidad del sistema a perturbaciones.

Capítulo 6

6. Casos de estudio

4.1	CASO SISO. LEVITACIÓN MAGNÉTICA
4.2	CASO MIMO. HELICÓPTERO 2 DOF
4.3	CONCLUSIONES

Para mostrar la aplicabilidad del diseño de control propuesto para NCS, se presentan dos casos de estudio. El caso SISO (*Simple Input – Simple Output*, una entrada – una salida) es un sistema de levitación magnética, muestra el diseño para el control de seguimiento de un sistema inestable. El segundo caso es MIMO (*Multiple Input – Multiple Output*, Múltiples entradas – Múltiples salidas), es un sistema simulador de un Helicóptero de dos grados de libertad, muestra el control para sistemas interconectados con múltiples entrada y salidas.

Ambos casos de estudios intercambian información a través de una red Ethernet de 10 Mbps, en diferentes configuraciones donde los nodos pueden ser manejados por tiempo o por evento. El caso SISO tiene una configuración donde el actuador y los sensores están incorporados en el mismo nodo mientras que el controlador se encuentra en otro nodo, mostrará el desempeño ante seguimiento de trayectorias. El caso MIMO tiene dos configuraciones, ambas con los componentes de control físicamente distribuidos, la primera maneja el nodo sensor por tiempo y los nodos controlador y actuador son manejados por evento, se mostrará la estabilidad y desempeño ante perturbaciones. La segunda configuración maneja el nodo sensor y controlador por evento y el actuador manejado por tiempo, se mostrará estabilidad y desempeño al manejar el sistema completo por tiempo y por evento.

6.1 Caso SISO. Levitación Magnética

El caso de estudio SISO es un sistema de levitación magnética MAGLEV de Quanser donde los sensores, controlador y actuador están conectados a través de una red Ethernet (Figura 6.1). Los sensores y el actuador conforman un nodo y otro nodo el controlador. Los nodos son PC

Pentium 2 con 254 Mb en RAM y tarjetas de red INTEL 10/100 Mb, cada PC tiene un sistema operativo de tiempo real XPC target de MATLAB que esta conectado a través de un conmutador CISCO Catalyst 1658 con puertos 10/100 Mbps. Los sensores están incorporados a una tarjeta de adquisición de datos Q8 de Quanser, tienen un convertidor A/D de 10 bits de resolución con 8 canales y los actuadores utilizan una fuente de potencia UPM2405 de Quanser y un convertidor D/A de 8 bits de resolución de la tarjeta Q8 de Quanser. El sistema es no lineal variante en el tiempo e inestable a lazo abierto. El objetivo es estabilizar en una posición deseada una esfera de metal mediante la energización de un electroimán. Información detallada del sistema puede ser encontrada en [73].

El sistema NCS para este caso de estudio utiliza una configuración común, donde los sensores y actuador conforman un nodo, mientras que el controlador se incorpora en un nodo, esta configuración es empleada cuando se requiere una marca de tiempo común y no se cuenta con algún método de sincronización de relojes. La configuración permite conocer el estado de la red, ya que es posible establecer los retardos de tiempo, los paquetes perdidos entre otras imperfecciones inducidas por la red.

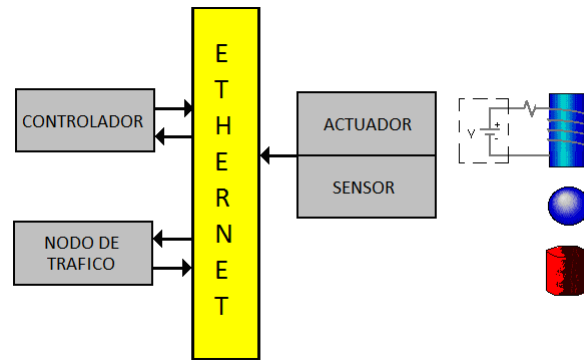


Figura 6.1. Configuración del sistema MAGLEV en red Ethernet.

El modelo del sistema de levitación se obtiene a partir del diagrama es (Figura 6.2) obteniendo las siguientes ecuaciones no lineales:

$$\begin{aligned} \frac{di(t)}{dt} &= \frac{1}{L} v(t) - \frac{R}{L} i(t) \\ \frac{d^2 a(t)}{dt^2} &= g - \frac{K_e}{m} \frac{i^2(t)}{a^2(t)} \end{aligned} \quad (5.28)$$

Donde g es la fuerza de gravedad, $i(t)$ es la corriente de la bobina, $a(t)$ es la distancia

desde la bobina hasta la posición de la esfera, $v(t)$ es el voltaje de la bobina, K_e es la constante de fuerza magnética, R la resistencia de la bobina, y L es la inductancia de la bobina.

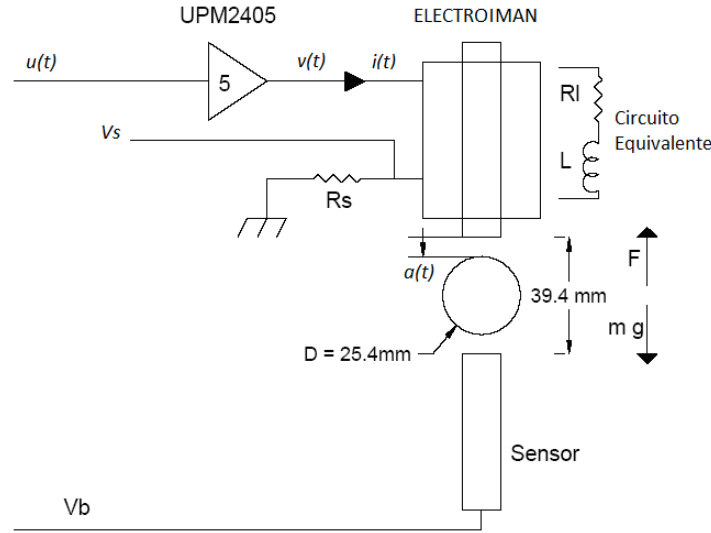


Figura 6.2. Esquema y diagrama de cuerpo libre del MAGLEV

Linealizando el sistema se obtiene un modelo lineal continuo desde el modelo no lineal (5.28) con la suposición que la esfera esta levitando alrededor de un punto de operación nominal (a_0, i_0) . El modelo de espacio de estados con el vector de estado $x(t) = (a(t), \dot{a}(t), i(t))$ es:

$$\begin{aligned} \dot{x}(t) &= \mathbf{A}x(t) + \mathbf{B}u(t) \\ &= \begin{bmatrix} 0 & 1 & 0 \\ K_e i_0^2 / ma_0^3 & 0 & -K_e i_0 / ma_0^2 \\ 0 & 0 & -R/L \end{bmatrix} x(t) + \begin{bmatrix} 0 \\ 0 \\ 1/L \end{bmatrix} u(t) \end{aligned} \quad (5.29)$$

El sistema MAGLEV incorporado a la red será empleado para evaluar el desempeño de la metodología propuesta ante el objetivo de control de seguimiento de una señal de referencia, con la configuración de la Figura 6.1. El modelo lineal continuo será utilizado para obtener un modelo difuso presentado en el capítulo 5.

6.2 Caso MIMO. Helicóptero 2 DOF

El caso de estudio MIMO es un helicóptero de dos grados de libertad 2-DOF (pitch y yaw)

de Quanser, cuyos elementos de control están distribuidos físicamente mediante una red Ethernet [73] (Figura 6.3). Los nodos son PC Pentium 2 con 254 Mb en RAM y tarjetas de red INTEL 10/100 Mb, cada PC tiene un sistema operativo XPC Target de MATLAB que esta conectado a través de un conmutador CISCO 1658 de 1 Gbps de velocidad con puertos 10/100 Mbps. Los sensores son dos enconders de 12 bits de resolución y los actuadores utilizan un convertidor D/A de 8 bits de resolución que forman parte de una tarjeta de adquisición de datos Q8 de Quanser, los actuadores son dos motores de corriente directa de 12 volts.

El sistema NCS para este caso de estudio utiliza una configuración físicamente distribuida, donde los sensores, controlador y actuadores son incorporados en nodos separados, esta configuración es empleada cuando cada uno de los componentes del lazo de control están separados físicamente lo que provoca no tener una referencia de tiempo del sistema. Para conocer el estado de la red en esta configuración requiere un análisis más profundo sobre el comportamiento de la red. Este sistema será empleado en dos formas de manejo distintas, la primera establece una base de tiempo en el nodo de los sensores que es manejado por tiempo mientras que los nodos controlador y actuadores serán manejados por evento cuando arribe un nuevo paquete, la segunda forma es manejar los nodos sensores y controlador por evento y el nodo actuador por tiempo. Esto con el fin de establecer junto con el sistema MAGLEV una amplia gama de configuraciones.

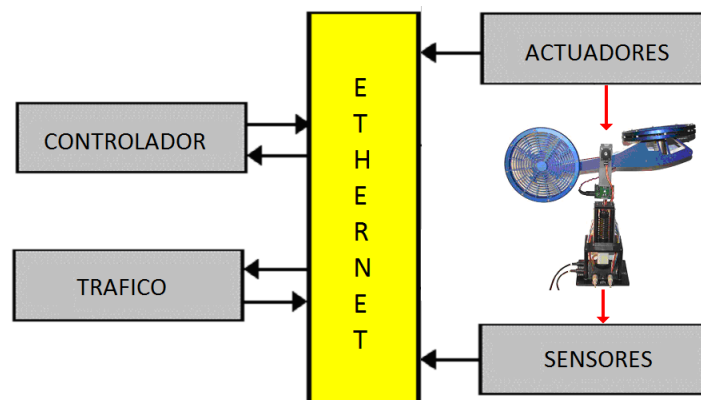


Figura 6.3. Helicóptero controlado en red con tráfico.

El helicóptero 2-DOF es no lineal variante en el tiempo, MIMO e inestable en lazo abierto. Se compone de dos hélices manejadas por motores de corriente directa. La hélice frontal controla

la elevación del la nariz del helicóptero sobre el eje de inclinación y la hélice trasera controla el movimiento lateral del helicóptero sobre el eje de dirección.

El método de Euler-Lagrange es usado para derivar las ecuaciones no lineales que describen el movimiento del helicóptero (5.30). Desde estas ecuaciones se obtiene el modelo no lineal en espacio de estados.

$$\begin{aligned}
 \mathbf{D}(q)\ddot{q} + \mathbf{C}(q, \dot{q})\dot{q} + \mathbf{g}(q) &= \alpha \\
 \mathbf{D}(q) &= \begin{bmatrix} J_p + ml^2 & 0 \\ 0 & J_y l^2 \cos^2 \theta \end{bmatrix} \\
 \mathbf{C}(q, \dot{q}) &= \begin{bmatrix} B_p & ml^2 \sin \theta \cos \theta \dot{\psi} \\ -2ml^2 \sin \theta \cos \theta \dot{\psi} & B_y \end{bmatrix} \\
 \mathbf{g}(q) &= \begin{bmatrix} mgl \cos \theta \\ 0 \end{bmatrix} \\
 \alpha &= \begin{bmatrix} K_{pp} V_p + K_{py} V_y \\ K_{yp} V_p + K_{yy} V_y \end{bmatrix}
 \end{aligned} \tag{5.30}$$

Donde K_{pp} es el torque aplicado por el motor de inclinación al eje de inclinación, K_{py} es el torque aplicado por el motor de inclinación al eje de desvio, K_{yp} es el torque aplicado por el motor de desvio al eje de inclinación, K_{yy} es el torque aplicado por el motor de desvio al eje de desvio, B_p es el factor de amortiguamiento viscoso equivalente sobre el eje de inclinación, B_y es el factor de amortiguamiento viscoso equivalente sobre el eje de desvio, l es la longitud al centro de masa del cuerpo del helicóptero, g es la fuerza de gravedad, m es la masa total del helicóptero, J_p es el momento total de inercia sobre el pivote de inclinación, J_y es el momento total de inercia sobre el pivote de desvio, V_p es el voltaje del motor de inclinación, V_y es el voltaje del motor de desvio, θ es el ángulo del eje de inclinación y ψ es el ángulo del eje de desvio.

El modelo lineal continuo se obtiene a partir de la ecuación no lineal (5.30) sobre el estado de equilibrio $\mathbf{x} = [\theta \ \psi \ \dot{\theta} \ \dot{\psi}]^T$ y entradas $\mathbf{u} = [V_p \ V_y]^T$, El modelo en espacio de estados es

$$\begin{aligned}
\dot{\mathbf{x}} &= \begin{bmatrix} 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 \\ 0 & 0 & -\frac{B_p}{J_p + ml^2} & 0 \\ 0 & 0 & 0 & -\frac{B_y}{J_y + ml^2} \end{bmatrix} \mathbf{x}(t) + \\
&\begin{bmatrix} 0 & 0 \\ \frac{K_{pp}}{J_p + ml^2} & \frac{K_{py}}{J_p + ml^2} \\ \frac{K_{yp}}{J_y + ml^2} & \frac{K_{yy}}{J_y + ml^2} \end{bmatrix} \mathbf{u}(t) \\
\mathbf{y} &= \begin{bmatrix} 1 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 \end{bmatrix} \mathbf{x}
\end{aligned} \tag{5.31}$$

Este modelo lineal continuo será empleado para obtener el modelo difuso propuesto con el fin observar el desempeño del sistema ante el objetivo de control de estabilización.

6.3 Conclusiones

Se presentan dos sistemas experimentales como casos de estudio para aplicar la metodología de codiseño propuesta, donde ambos sistemas son inestables a lazo abierto, estos sistemas permitirán aplicar diferentes configuraciones presentes en los sistemas en control en red.

La configuración del sistema de levitación magnética no es completamente distribuida al tener los sensores y el actuador manejados por el mismo nodo, además de manejar los sensores por tiempo, el controlador y actuador son manejados por el evento del arribo de un nuevo paquete.

El simulador de helicóptero 2-DOF tiene una configuración físicamente distribuida, además en el desarrollo de los experimentos los componentes del lazo de control pueden varias configuraciones, donde los sensores son manejados por tiempo, el controlador y actuadores por evento, o donde los sensores y controlador son manejados por evento y los actuadores manejados por tiempo.

Esta flexibilidad con respecto a las configuraciones permitirá establecer una amplia aplicabilidad de la metodología propuesta.

Capítulo 7

7. Resultados

7.1	SISTEMA DE LEVITACIÓN MAGNÉTICA
7.2	SISTEMA HELICÓPTERO 2-DOF CONTROLADO POR TIEMPO
7.3	CODISEÑO DEL SISTEMA HELICÓPTERO 2-DOF
7.4	CONCLUSIONES

Para mostrar la aplicabilidad de la metodología propuesta se presentan dos experimentos definidos en el capítulo 4, el sistema de levitación magnética tiene una configuración de la Figura 4.1A, el modelo de imperfecciones utilizado solo considera el retardo RTT para el cálculo de la señal de control a través del controlador difuso (5.9) – (5.12), mostrando su efectividad al seguimiento de trayectorias y robustez ante variaciones de tráfico sobre la red.

El sistema del helicóptero 2-DOF tiene una configuración de la Figura 4.1B completamente distribuido, este sistema contempla la aplicación de dos metodologías ambas consideran las tres imperfecciones de tiempo (retardos, paquetes perdidos y muestreo variante). La primera metodología solo considera utilizar el control difuso para compensar las imperfecciones, mientras que la segunda considera la estrategia de codiseño empleando el controlador y planificador difuso para compensar las imperfecciones.

7.1 Sistema de Levitación Magnética

El sistema de levitación magnética es un sistema inestable a lazo abierto, no lineal y variante en el tiempo, cuyo objetivo general es mantener levitando una esfera de metal a una distancia deseada. El objetivo particular es que la esfera siga una trayectoria deseada mostrando robustez a las imperfecciones de la red y manteniendo buen desempeño de seguimiento.

El diseño de un controlador que cumpla con el objetivo particular tiene tres etapas de diseño presentadas en el capítulo 5, diseñar el modelo probabilístico del retardo RTT, diseñar el modelo difuso del NCS, y diseñar el controlador difuso del NCS.

Para la etapa de diseño del modelo de retardo RTT, el primer paso es analizar el

comportamiento de la red de comunicación con diferentes cargas de tráfico y definir los parámetros que involucran el modelo del retardo RTT que asegure robustez y exactitud a la estimación.

El comportamiento del retardo RTT se obtiene enviando paquetes sobre la red de comunicación simulando el intercambio de información del lazo de control (sensor – controlador, controlador – actuador) sumado al envío de paquetes de tráfico para variar las condiciones de operación del sistema NCS.

Para el análisis del retardo RTT se tiene la configuración NCS de la Figura 4.1A, donde se realizan las mediciones del retardo RTT y el análisis de paquetes perdidos entre el nodo Sensor/Actuador y el nodo Controlador, con 1 ms como período de muestreo con paquetes de 64 bytes.

El analisis de comportamiento RTT se realiza en tres escenarios, sin tráfico, tráfico ligero (3 lazos) y tráfico medio (6 lazos). Para el escenario sin tráfico solo el lazo de control intercambia información sobre la red. Los retardos RTT generados son mostrados en la Figura 7.1, estos varían entre 2 ms – 10 ms, con una mayor periodicidad entre 2 ms – 3 ms

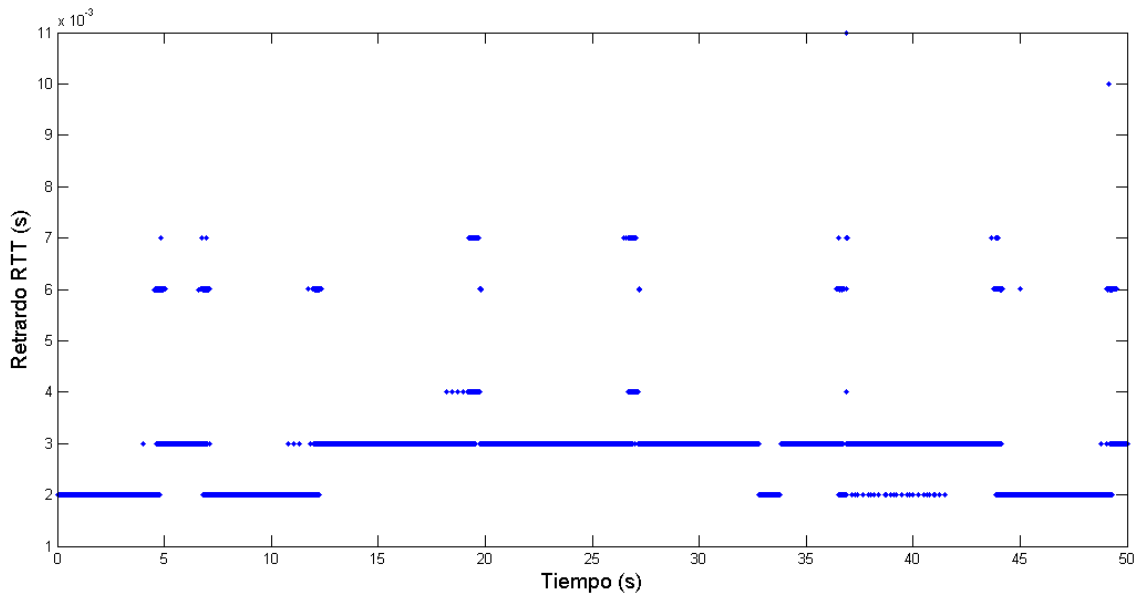


Figura 7.1. Retardo RTT para el experimento sin tráfico externo.

El segundo escenario mide los retardos RTT de paquetes de 64 bytes y período de 1 ms, más tráfico ligero con nodos transmitiendo paquetes UDP de 64 bytes cada 1 ms sobre la red que equivalen a 3 lazos de control externos, los retardos RTT resultantes son mostrados en la Figura

7.2. El rango del retardo RTT sufre una ligero aumento [$2\text{ ms} - 11\text{ ms}$] concentrándose en $2\text{ ms} - 4\text{ ms}$.

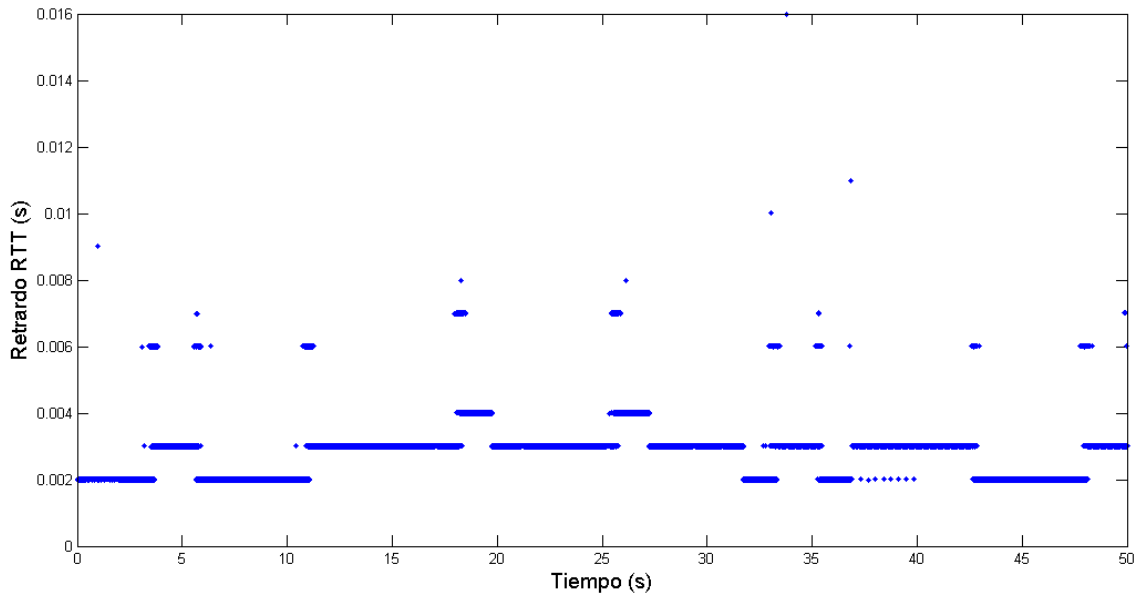


Figura 7.2. Retardos RTT para la comunicación del NCS y tráfico ligero con paquetes de 64 bytes cada 1 ms.

El tercer experimento mide los retardos RTT generados en la transmisión Sensor/Actuador y Controlador, más tráfico medio transmitiendo paquetes UDP de 64 bytes y 512 bytes a 1 ms equivalente a 6 lazos de control externos, los retardos RTT son mostrados en la Figura 7.3. En este escenario el rango del retardo RTT es [$2\text{ ms} - 15\text{ ms}$] concentrándose de $2\text{ ms} - 4\text{ ms}$.

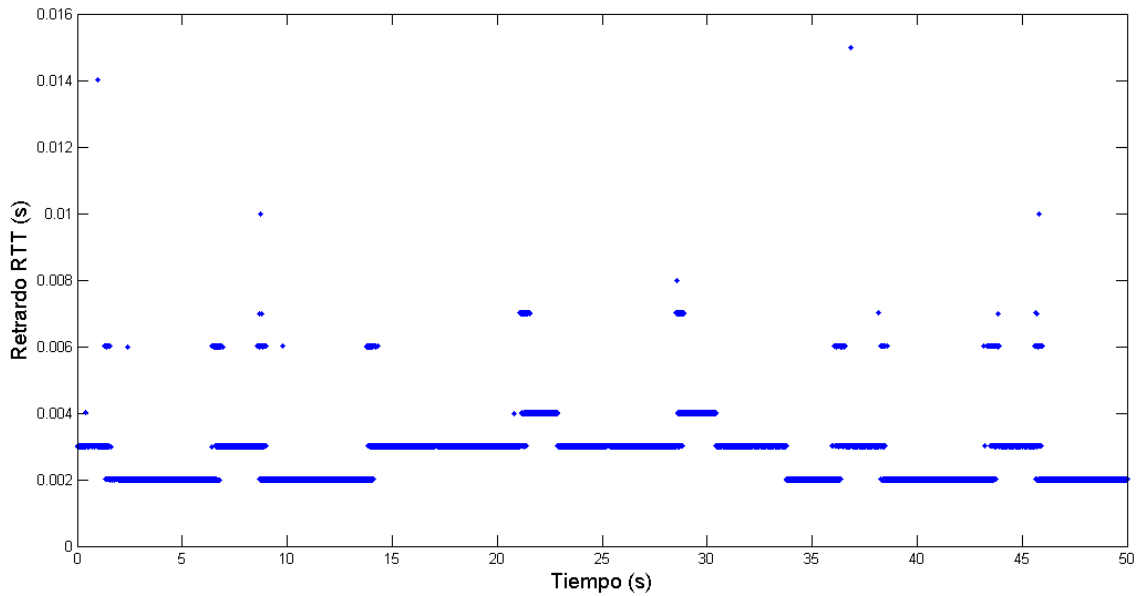


Figura 7.3. Retardos RTT para el NCS con tráfico medio, paquetes de 64 y 512 bytes cada 1 ms.

Una vez observado el comportamiento del retardo RTT a diferentes niveles de tráfico se realizan las estadísticas mostradas en la Tabla 7.1 donde se observa que los retardos RTT se mantienen en una media típica pero conforme aumentan los paquetes a través de la red aumenta la cantidad de paquetes perdidos, definiendo la variable α que es la media de tiempo que tarda en perderse un paquete. Lo que muestra que el aumento de tráfico en la red no solo incrementa el retardo RTT, sino el número de paquetes perdidos además de disminuir el tiempo entre paquetes perdidos. Otro comportamiento particular se presenta cuando hay un aumento considerable en el porcentaje de paquetes perdidos el retardo RTT máximo disminuye.

Tabla 7.1. Estadísticas del retardo RTT para múltiples escenarios.

Escenario	τ_{media}	$\tau_{mediana}$	τ_{max}	α perdidos	Perdidos
Sin tráfico	0.0024	0.003	0.011	20.89	4.79 %
Tráfico ligero	0.0027	0.003	0.016	20.08	4.98 %
Tráfico medio	0.0029	0.003	0.015	18.79	5.82 %

La Figura 7.4 muestra el histograma del retardo RTT medido en los tres experimentos, donde se muestra la periodicidad de los retardos RTT generados por la red. Los histogramas muestran que la probabilidad de un retardo RTT de 16 ms es más pequeña que la media μ que es

de 3 ms aproximadamente. Aunque el retardo RTT puede ser más grande que μ , este tiene poca probabilidad.

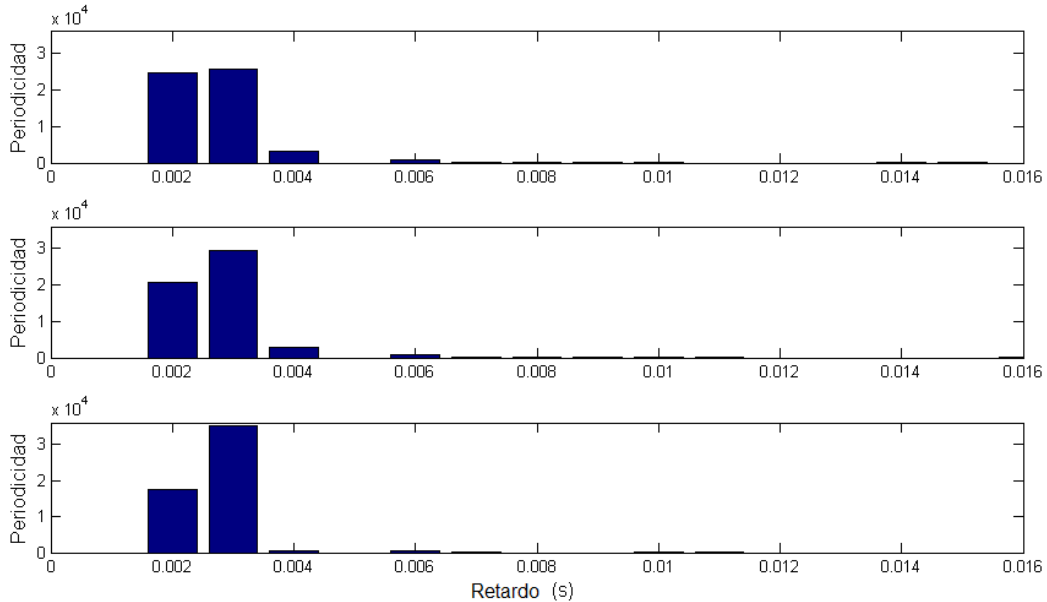


Figura 7.4. Histograma de los retardos RTT medidos para los tres escenarios.

Para fines de diseño se considera una cota de [2 ms – 20 ms] del retardo RTT, y analizando su comportamiento se utiliza el modelo de distribución exponencial para estimar el retardo RTT (4.4).

Utilizando el modelo de distribución exponencial generalizada expuesta en el capítulo 4 y las mediciones mostradas en la Tabla 7.1 se obtiene el modelo para la estimación del retardo RTT para los tres escenarios, los parámetros del modelo del retardo RTT son:

$$\begin{aligned}\eta &= 2 \times 10^{-3} \\ \varphi &= 1.56 \times 10^{-6} \\ w &= 10\end{aligned}$$

Con estos parámetros se muestra la estimación del retardo RTT (Figura 7.5), donde (\blacklozenge) es la medición del retardo RTT fuera de línea y (+) es el retardo RTT estimado, con un error cuadrático medio menor a 3.5×10^{-7} que para efectos del controlador es un buen modelo de estimación.

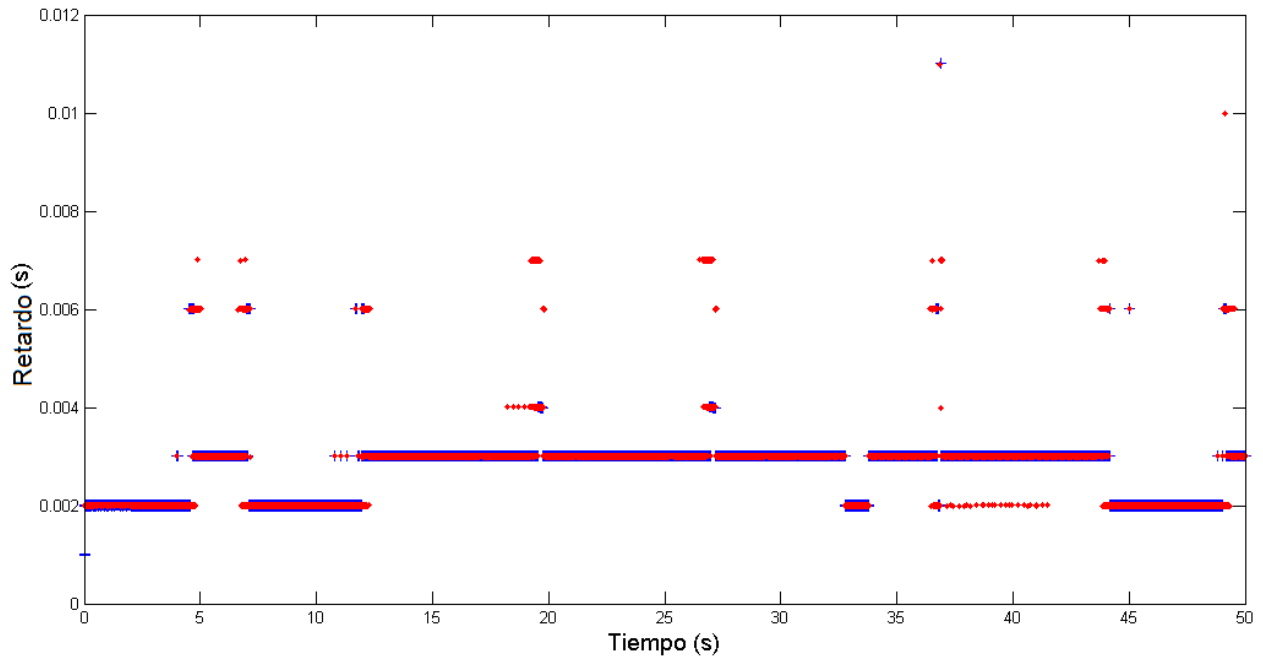


Figura 7.5. Retardo RTT (+) y estimación RTT (•) para los tres experimentos

Una vez obtenido el modelo RTT robusto y con una exactitud suficiente para efectos de control, se concluye la primera etapa de diseño.

La segunda etapa de diseño es obtener el modelo difuso del MAGLEV desde su modelo lineal continuo (5.29), definiendo las reglas difusas en función de los períodos de muestreo $h_j = (0.001, 0.005, 0.01, 0.015)$ que cubre el rango del retardo RTT $[0.002, 0.016]$ obtenido.

Cada regla difusa j se forma a partir del modelo lineal continuo (5.29) discretizado con el período de muestreo h_j que conforma la parte consecuente con $j = 1 \dots 4$.

El modelo lineal discreto para $h_1 = 0.001$ se muestra a continuación:

$$x_1^{k+1} = \begin{bmatrix} 1.0016 & 0.001 & 0 \\ 3.2718 & 1.0016 & -0.0225 \\ 0 & 0 & -0.9737 \end{bmatrix} x^k + \begin{bmatrix} 0 \\ 0 \\ 0.0024 \end{bmatrix} u^k$$

La parte antecedente es diseñada por la ecuación **¡Error! No se encuentra el origen de la referencia..** Tal que, los parámetros ρ_j y σ_j con $j = 1 \dots 4$ y cota superior del retardo RTT en $\bar{h} = 20ms$ es:

$$\rho_j = [0.001 \quad 0.005 \quad 0.01 \quad 0.015]$$

$$\sigma_j = [0.015 \quad 0.015 \quad 0.015 \quad 0.015]$$

La Figura 7.6 muestra las cuatro funciones de membresía normalizadas con los parámetros de diseño y una cota superior del retardo RTT de $\bar{h} = 20ms$.

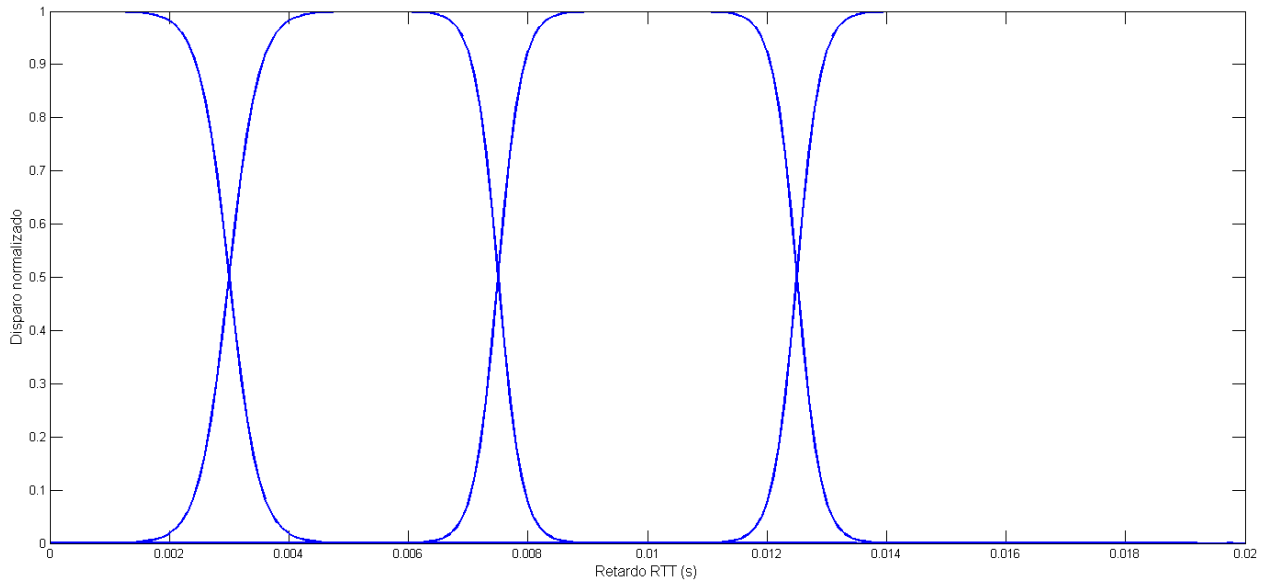


Figura 7.6. Funciones de membresía normalizadas para el modelo difuso MAGLEV.

El modelo difuso obtenido es formado por cuatro reglas que modelan el comportamiento del sistema NCS en un rango del retardo RTT de $[0.002, 0.020]$ definido en la parte antecedente y su correspondiente modelo discreto en la parte consecuente. Para mostrar la efectividad en la estimación del modelo difuso se realizan experimentos en el sistema MAGLEV con tráfico medio (6 lazos de control extra) donde el comportamiento del retardo RTT es similar a la Figura 7.3 que presenta retardos RTT en el rango de $[0.002, 0.016]$.

Se compara el desempeño del modelo difuso que estima la posición de la esfera con la posición medida por el sensor para el sistema controlado, la comparación es mostrada en la Figura 7.7 (superior) donde la eficiencia del modelado esta en función del error cuadrático medio que es de 0.551 mm, el error de modelado se muestra en la Figura 7.7 (inferior) con un 4% de error relativo máximo.

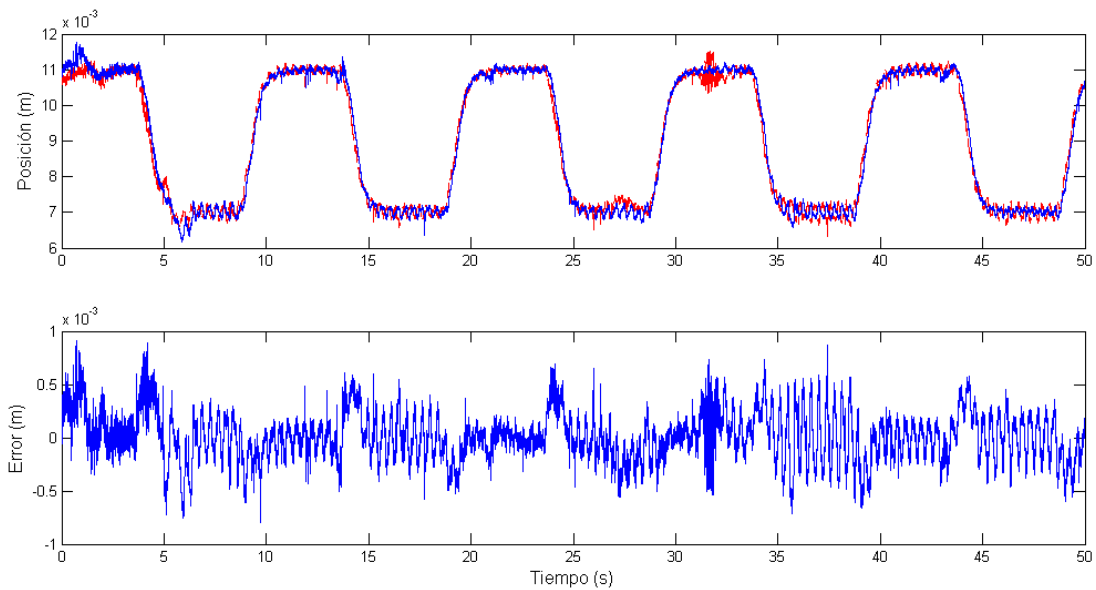


Figura 7.7. Comparativo entre el modelo difuso y la salida medida (superior), error de modelado (inferior)

Un vez diseñado y verificado el modelo difuso se procede a la etapa tres, que es el diseño del controlador difuso con matrices de retroalimentación por LQR de la forma (5.12) para cada una de las cuatro reglas del modelo difuso. La ley de retroalimentación para el primer modelo local es:

$$K_1 = [-38368 \quad -103.054 \quad 11.97]$$

El controlador difuso diseñado para el sistema de levitación magnética es verificado a tener robustez y estabilidad ante el rango del retardo RTT, se realizaron experimentos para los tres escenarios del retardo RTT (sin, ligero y medio) donde todas las pruebas mantuvieron el sistema estable. El error en estado estable para los tres escenarios se mantuvo por debajo del 3×10^{-4} por lo que el desempeño se mantiene en un margen del 2% respecto de la referencia que para efectos de control es aceptable.

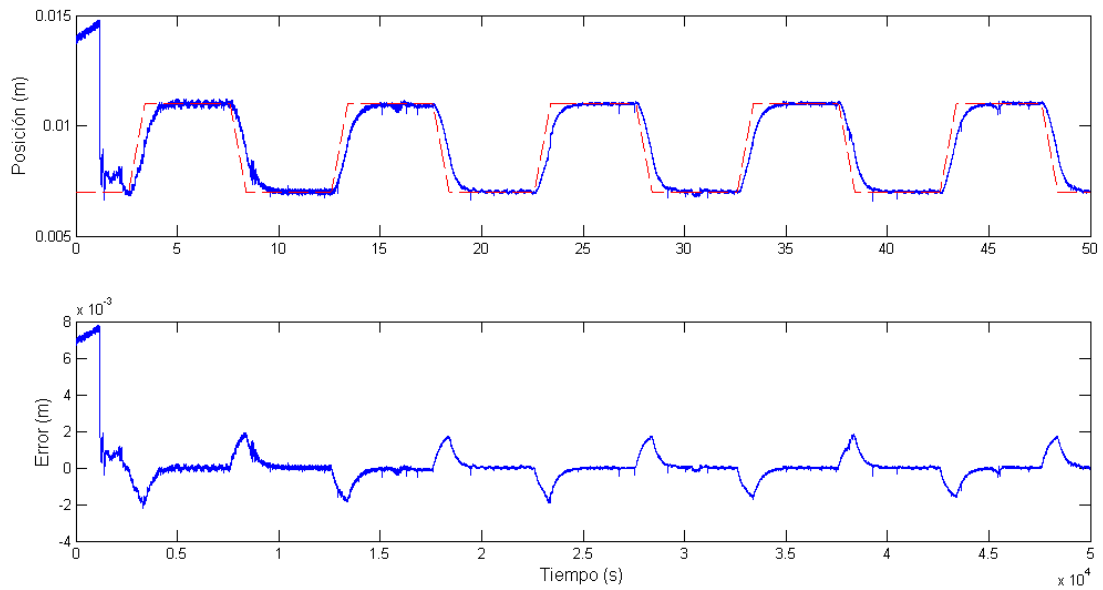


Figura 7.8. Desempeño del sistema libre de tráfico en la red (arriba), error de control (abajo).

Las oscilaciones en el sistema son muy pequeñas y son atribuidas a que el sensor de posición del sistema MAGLEV es un sensor de luz que lo hace sensible a perturbaciones. Además, al inicio de la prueba la posición es invertida, es decir, la esfera metálica sobre el poste es la distancia de 16 *mm* y cercano a la bobina es la posición 0 *mm*.

El comportamiento con tráfico ligero y medio el sistema se mantiene estable, con una mayor oscilación en el inicio de la prueba, pero conforme el tiempo transcurre este se mantiene en una oscilación mínima.

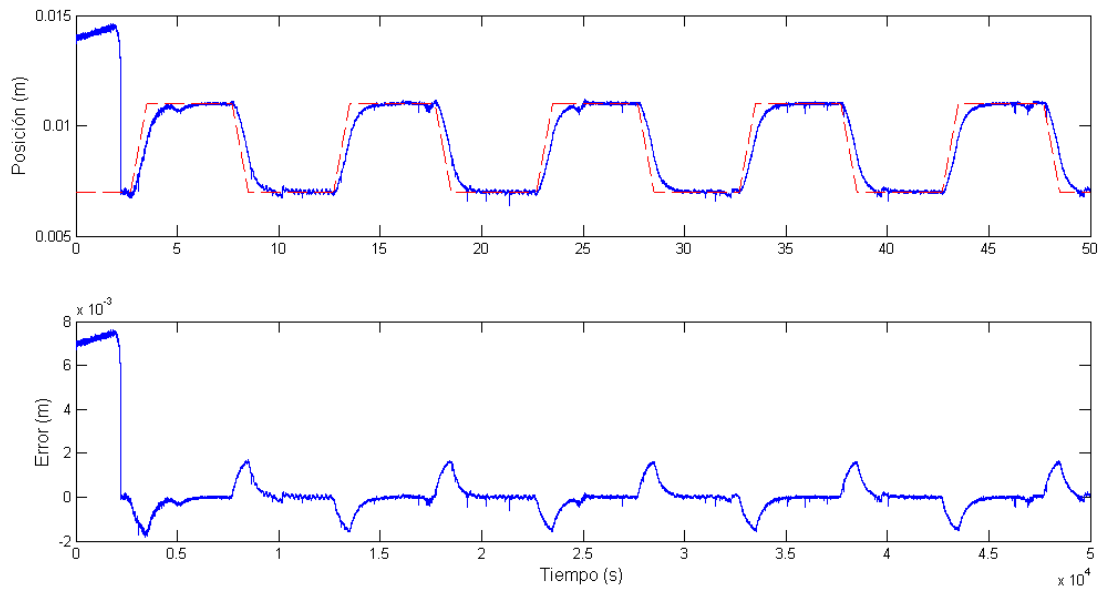


Figura 7.9. Desempeño del sistema con tráfico ligero en la red (arriba), error de posición (abajo).

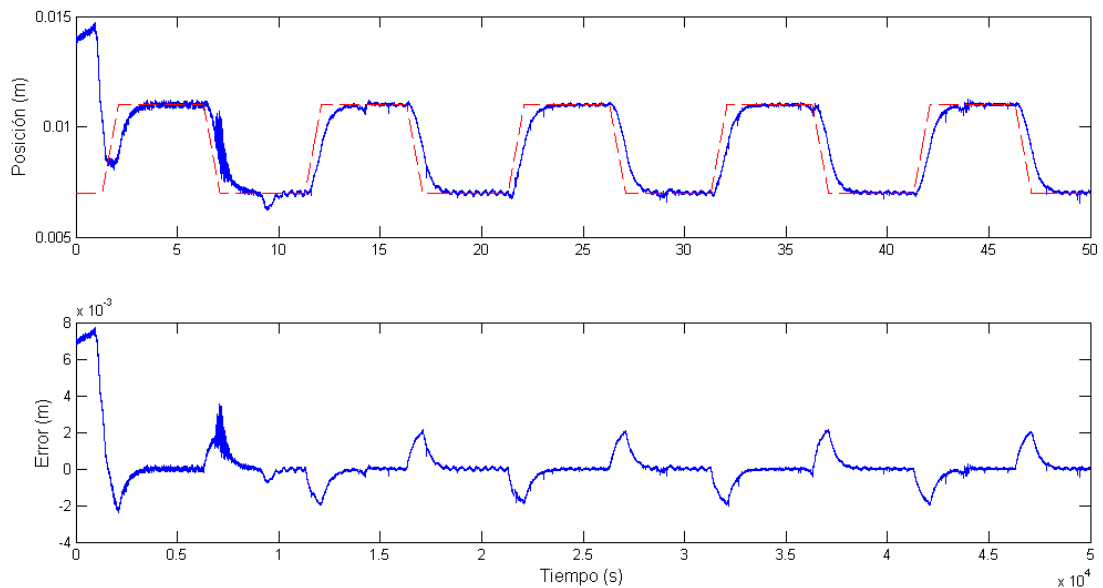


Figura 7.10. Desempeño del sistema con tráfico medio en la red (arriba), error de posición (abajo).

Con la obtención de las matrices de retroalimentación el diseño de un controlador para NCS es concuido, este controlador estabiliza el sistema con retardos RTT variantes en un rango de $[0.002, 0.016]$ que es obtenido a través de pruebas reales.

El diseño del controlador puede ser extendido para compensar variaciones en la pérdida de paquetes como en el intervalo de muestreo como se mostrará en la siguiente sección.

7.2 Sistema Helicóptero 2-DOF Controlado por Tiempo

El sistema del Helicóptero de 2-DOF es un sistema inestable a lazo abierto, no lineal y variante en el tiempo con dos entradas – dos salidas acopladas, cuyo objetivo general es mantener la posición de pitch y yaw en un valor deseado, alimentado dos motores de c.d. con hélices. El objetivo particular del sistema NCS es que las posiciones de pitch y yaw mantengan una posición deseada mostrando robustez a las imperfecciones de la red y manteniendo buen desempeño de estabilización.

La configuración NCS de este sistema se muestra en la Figura 4.1b, donde las imperfecciones a compensar son los retardos de tiempo desde el sensor al actuador, intervalos de muestreo variantes y pérdida de paquetes. El diseño del controlador que cumpla con el objetivo particular como en el caso anterior tiene tres etapas de diseño, diseñar el modelo probabilístico de las imperfecciones, diseñar el modelo difuso NCS, y por último diseñar el controlador difuso NCS.

Para la primera etapa del modelo de imperfecciones, se analizan los comportamientos de las tres imperfecciones por separado. Para el análisis del retardo se realizaron dos experimentos, el primero describe el comportamiento del retardo con respecto al período de muestreo y el segundo describe el comportamiento del retardo con respecto al tráfico para un período de muestreo estático.

Primero se describe el comportamiento del retardo en función del período de muestreo, se modifica el período de muestreo en un rango de $5\text{ ms} - 60\text{ ms}$, cada experimento se realizó 25 veces para tener una muestra estadística (Figura 7.11). El retardo de tiempo tiene un comportamiento que se ajusta a una función de distribución exponencial, cambiando ligeramente en su comportamiento cuando se tiene un menor período de muestreo debido a que genera mayor tráfico en la red y una mayor carga en los reguladores (*buffers*) de la tarjeta de red. Conforme el período de muestreo aumenta, la cantidad de paquetes disminuye y por lo tanto también el retardo de tiempo. Con este experimento se establecen la cota mínima del retardo de tiempo que es de 5 ms , esto debido a que el sistema trabaja con período base de 5 ms .

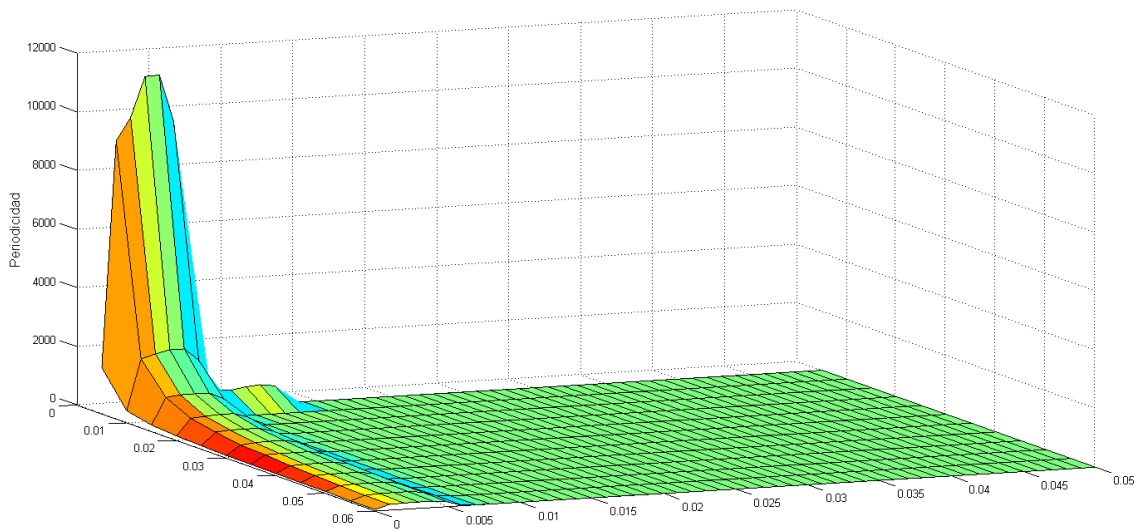


Figura 7.11. Período de muestreo vs Retardos de tiempo

Conforme a la gráfica anterior, se tiene que la mayor variabilidad del retardo de tiempo es cuando el sistema tiene un período de muestreo de 5 ms , por lo que en este caso se realizó una segunda prueba dejando fijo el período en 5 ms y modificando el número de nodos transmitiendo sobre la red Ethernet es decir aumentando el tráfico. Cada experimento se realizó 10 veces para poder hacer el análisis, donde se agregaron de 1 a 12 nodos. Cada nodo genera tráfico a través del intercambio de paquetes de 1024 bytes a un período de 0.5 ms . Los resultados son mostrados en la Figura 7.12, donde se muestra que a tráfico ligero el retardo de tiempo se mantiene en un rango de 0 a 5 ms , mientras que a tráfico severo se muestra una mayor variabilidad del retardo de tiempo estableciendo una cota superior para el retardo 30 ms .

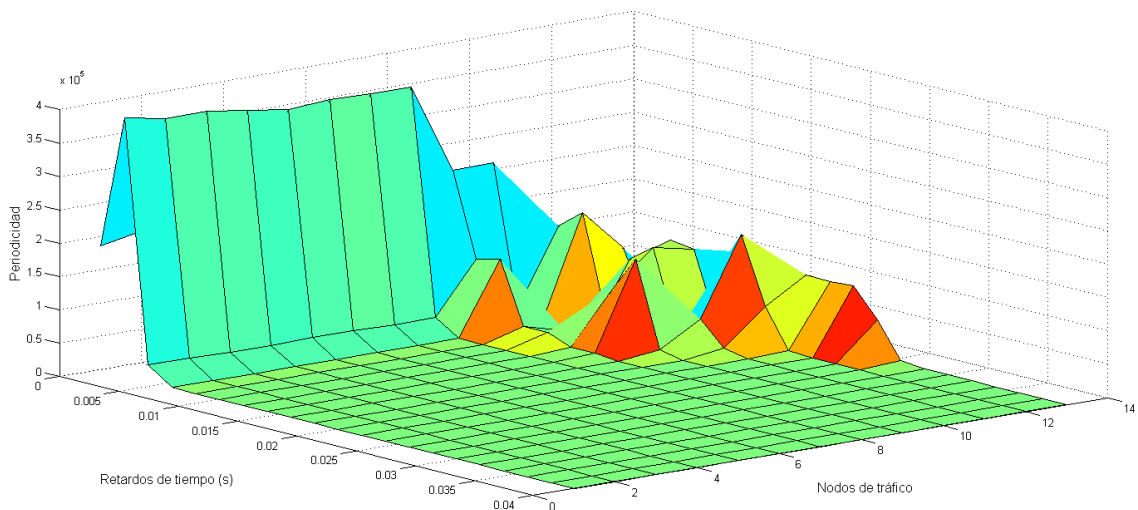


Figura 7.12. Retardo de tiempo vs Tráfico

Una vez terminada la etapa de análisis de imperfecciones y obtenido el modelo del período de muestreo variable, se procede con la segunda etapa, el diseño de un modelo difuso NCS.

El modelo difuso se obtiene a partir del modelo continuo lineal (5.31) con estados $\mathbf{x} = [\theta \ \psi \ \dot{\theta} \ \dot{\psi}]^T$ y entradas $\mathbf{u} = [V_p \ V_y]^T$. Para cada regla difusa j el modelo continuo lineal es discretizado con período de muestreo h_j $j=1\dots r$ definidos como $h_j = [0.005 \ 0.010 \ 0.015 \ 0.020]^T$, el primer modelo discreto para $h_1 = 0.005$ es:

$$\mathbf{x}_1(k+0.005) = \begin{bmatrix} 1 & 0 & 0.0048 & 0 \\ 0 & 1 & 0 & 0.0049 \\ 0 & 0 & 0.9547 & 0 \\ 0 & 0 & 0 & 0.9827 \end{bmatrix} \mathbf{x}(k) + \begin{bmatrix} 2.913\text{E}-5 & 9.722\text{E}-7 \\ 2.995\text{E}-6 & 9.834\text{E}-6 \\ 1.156\text{E}-2 & 3.859\text{E}-4 \\ 1.195\text{E}-3 & 3.922\text{E}-3 \end{bmatrix} \mathbf{u}(k)$$

Los parámetros de la parte antecedente son definidos por el usuario para cada regla j con una cota máxima del período de muestreo variable $h_v^{MAX} = 300ms$, esto permite definir las regiones de traslape (S_s).

$$\rho_j = [0.005 \ 0.010 \ 0.015 \ 0.020]$$

$$\sigma_j = [12 \times 10^{-4} \ 24 \times 10^{-4} \ 30 \times 10^{-4} \ 30 \times 10^{-4}]$$

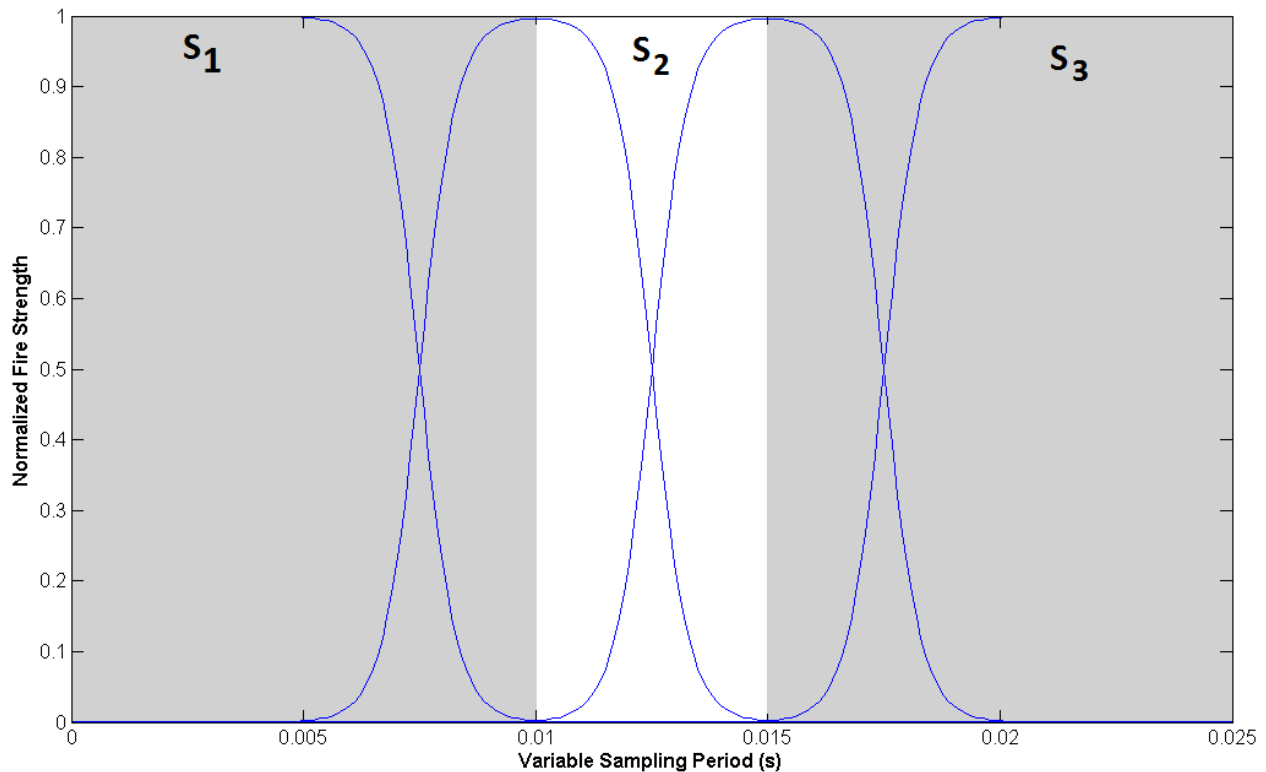


Figura 7.13. Período de muestreo vs grado de disparo normalizado para el Helicóptero 2DOF

Es de hacer notar que solo hay tres zonas de traslape con dos funciones de membresía, por lo que solo se requieren tres matrices P_s que cumplan las condiciones (5.16) - (5.17), para asegurar la estabilidad del NCS con el controlador difuso.

La última etapa es diseñar el controlador difuso a partir del conjunto de reglas obtenidas para el modelo difuso, diseñando una matriz de retroalimentación para cada modelo local. Todas las matrices de retroalimentación son diseñadas por LQR, siendo la primera matriz:

$$K_1 = \begin{bmatrix} 15.4 & 1.53 & 4.91 & 0.677 & 12.2 & 0.718 \\ -1.97 & 17.3 & -0.241 & 6.2 & -1.24 & 7.03 \end{bmatrix}$$

Una vez diseñado el controlador varios experimentos se realizan para medir el desempeño ante variaciones en las imperfecciones de la red. Se realizaron tres experimentos (sin, medio y alto tráfico) variando la cantidad de nodos extras a fin de generar tráfico en la red, el método propuesto es comparado con el control digital para el experimento sin tráfico en la red y el control discreto con un retardo constante, la comparativa es medida mediante la integral del error absoluto (IAE, por sus siglas en ingles).

El primer experimento solo incorpora el lazo de control sin tráfico dentro de la red, la señal

de referencia es la posición 0° del ángulo pitch (- -) y la posición inicial es de 41° (Figura 7.14). El experimento se realiza 25 veces, todos con respuesta estable con un error en estado estable promedio de $e_{ss} = 0.2^\circ$ y un sobrepaso de $\zeta = 1.3^\circ$, la conducta del control difuso en red (—) es comparado con el control digital (⋯) sin red obteniendo un desempeño similar con $e_{ss} = 0.2^\circ$ y $\zeta = 2^\circ$.

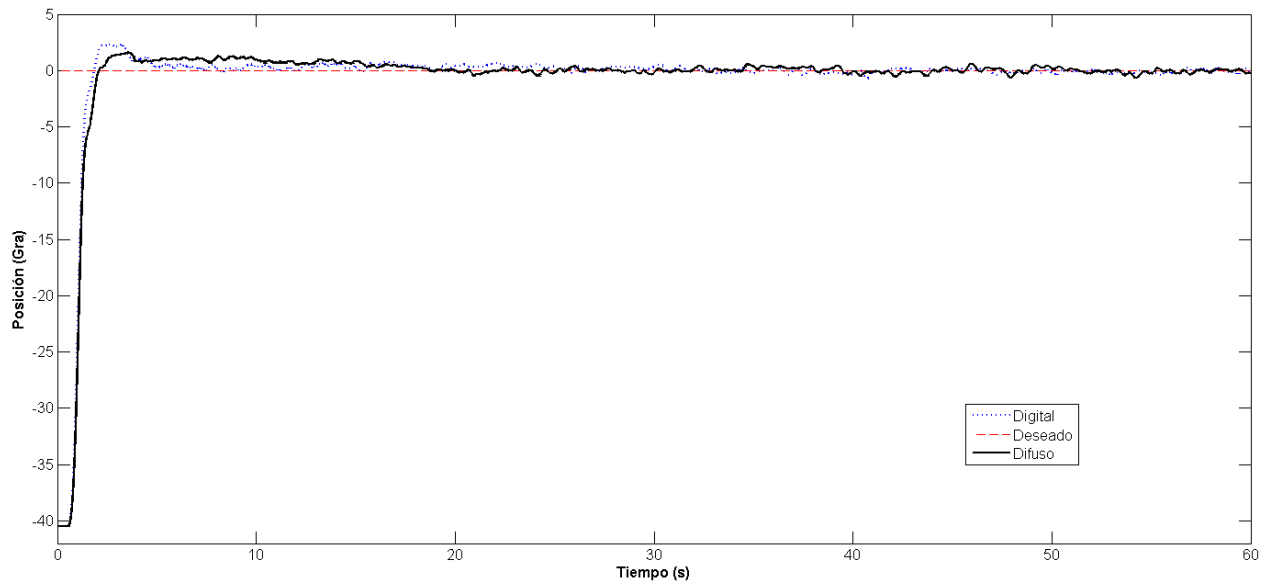


Figura 7.14. Posición pitch del Helicóptero 2DOF sin tráfico en la red.

En el experimento el sistema compensa retardos de tiempo en el rango de $[0.0005, 0.021]$ con una media de 7 ms (Figura 7.15a) y el tiempo sin información por paquetes perdidos (Figura 7.15b) con un porcentaje del 0.05% de paquetes perdidos. El IAE para el control difuso varía en el rango $[1.1287, 1.176]$ mientras que para el control digital sin red el mínimo es 1.0652 , siendo este el mejor desempeño cuando no se tienen imperfecciones de red.

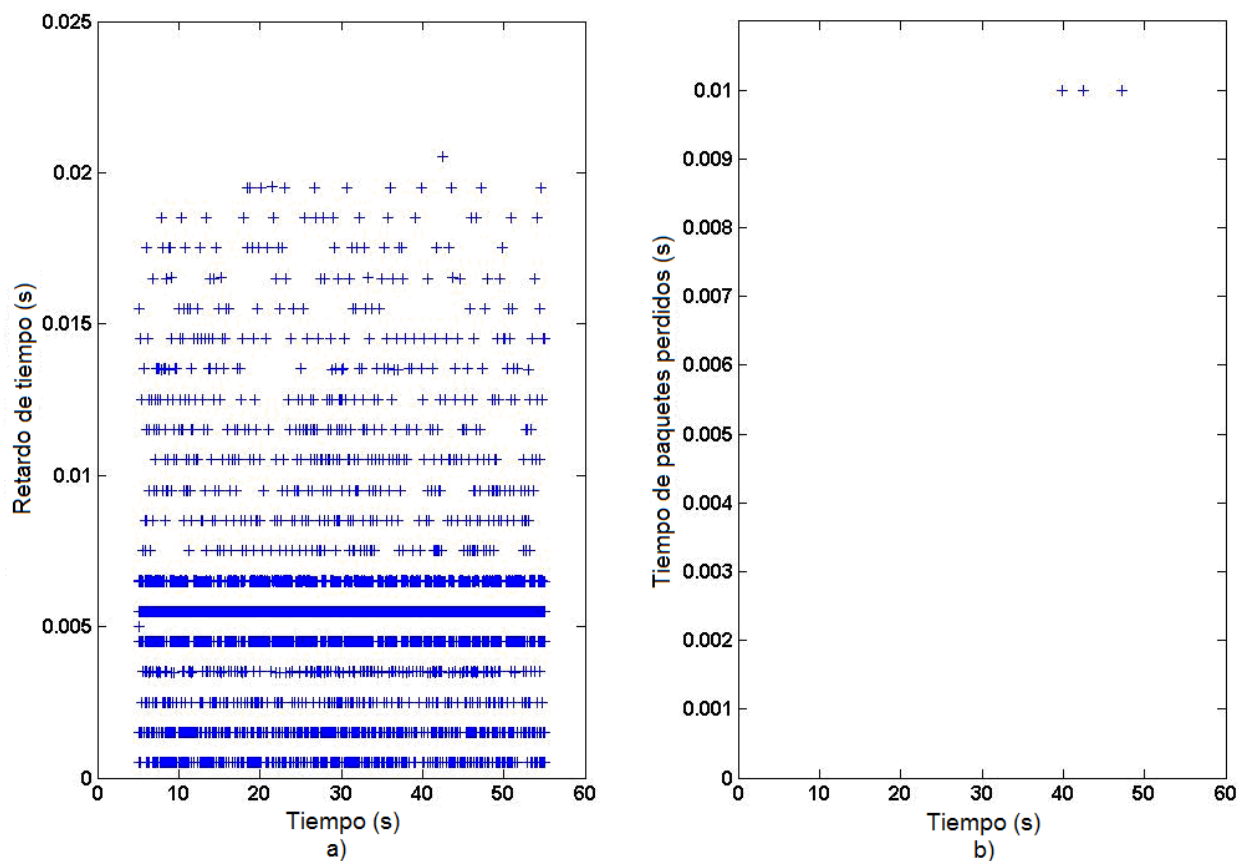


Figura 7.15 Retardos de tiempo (a) y tiempo perdido (b) del NCS sin tráfico.

El Segundo experimento muestra la conducta robusta del NCS con seis nodos de tráfico dentro de la red generando tráfico medio. Los paquetes de tráfico tienen una longitud de 254 bytes con un período de 1 *ms*.

En el experimento la conducta del NCS (Figura 7.16) con control difuso (—) tiene un comportamiento estable en todas las repeticiones. En general tiene pequeñas oscilaciones en el estado transitorio que se desvanecen con el tiempo, el error en estado estable promedio es de $e_{ss} = 0.5^\circ$ y un sobrepaso de $\zeta = 1.5^\circ$, esto aún con el comportamiento del retardo de tiempo y los paquetes perdidos (Figura 7.17).

Mientras que el desempeño del control retroalimentado con retardo constante (...) tiene un sobrepaso de $\zeta = 4^\circ$ y un error en estado estable promedio de $e_{ss} = 3^\circ$ con una mayor oscilación. Aquí se muestra que el controlador difuso tiene el mejor desempeño con un IAE en el rango de $[1.215, 1.2332]$ comparado con el desempeño del controlador retroalimentado que es de

1.8043.

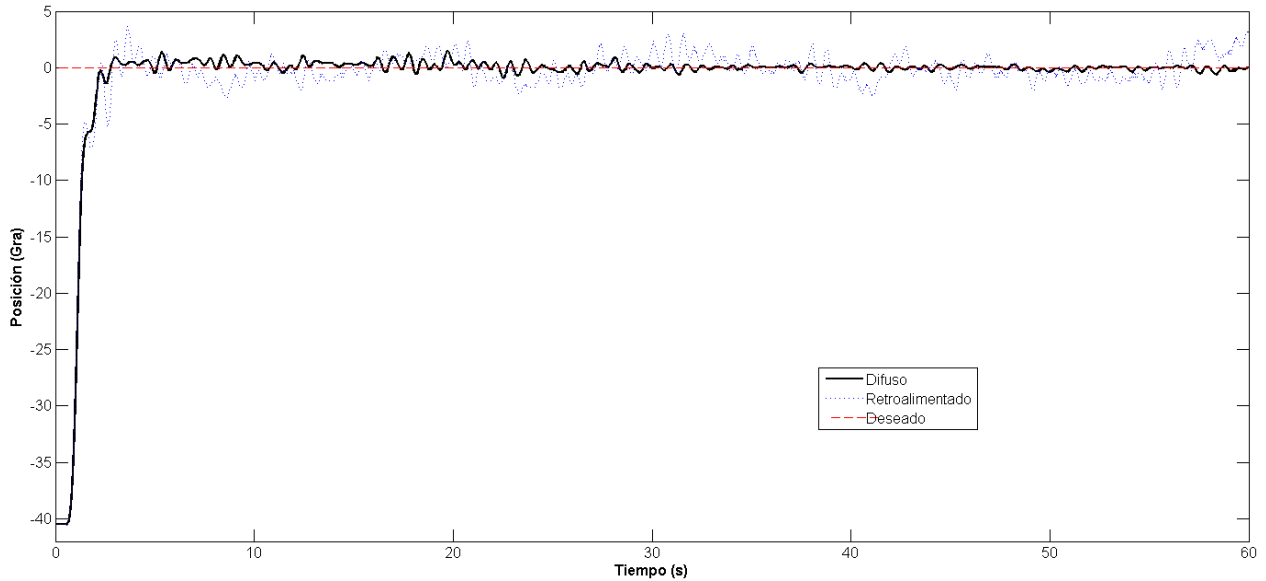


Figura 7.16 Respuesta de pitch con 1.17% de paquetes perdidos y retardos de tiempo de [0.0005 – 0.035]

Con respecto al comportamiento de la red, el retardo de tiempo presenta un máximo de 35 *ms* (Figura 7.17a) con una variación en el rango [0.0005,0.030] con un promedio de 9 *ms* y un porcentaje de paquetes perdidos máximo (Figura 7.17b) de 1.17% con alrededor de 140 paquetes perdidos de 12000 en un lapso de 60 segundos.

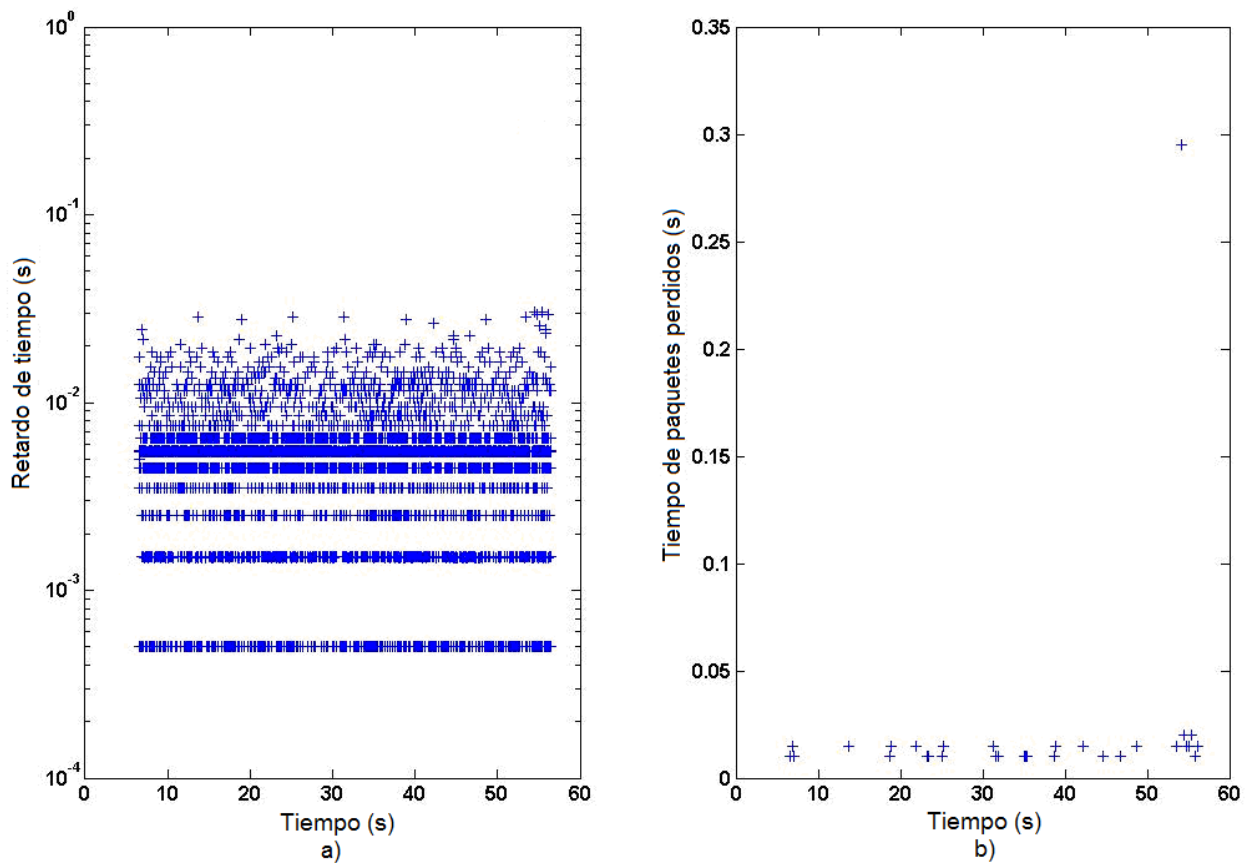


Figura 7.17. Retardos de tiempo (a) y pérdida de paquetes (b) con 6 nodos de tráfico.

El tercer experimento muestra la conducta del NCS con 12 nodos extras generando tráfico dentro de la red (Figura 7.18), donde la respuesta del NCS con el controlador difuso (—) tiene más oscilaciones en el estado transitorio pero se desvanecen al llegar al estado estable. El error en estado estable promedio es de $e_{ss} = 0.4^\circ$ con un sobrepaso máximo de $\zeta = 2.5^\circ$, mientras que el controlador retroalimentado (\cdots) tiene oscilaciones más grandes con un mal desempeño que no alcanza un estado estable. El desempeño IAE del NCS con control difuso es de 1.3463 mientras que para el control retroalimentado el IAE es 3.889 ya que no compensa variaciones de retardos de tiempo ni paquetes perdidos.

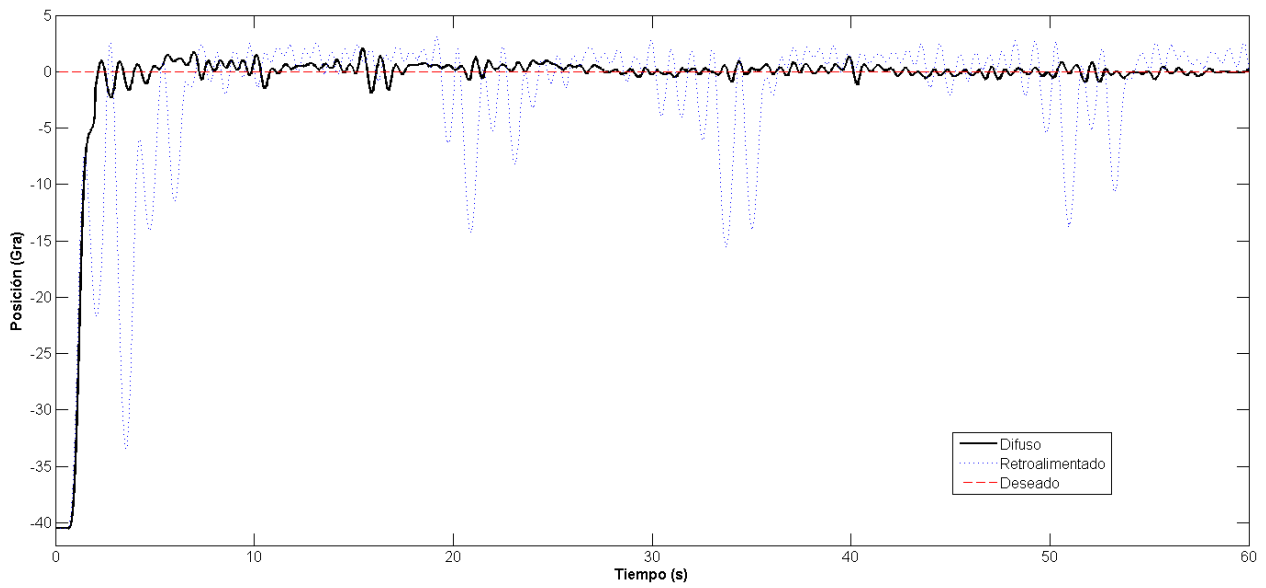


Figura 7.18. Respuesta de la posición de pitch con 12 nodos de tráfico.

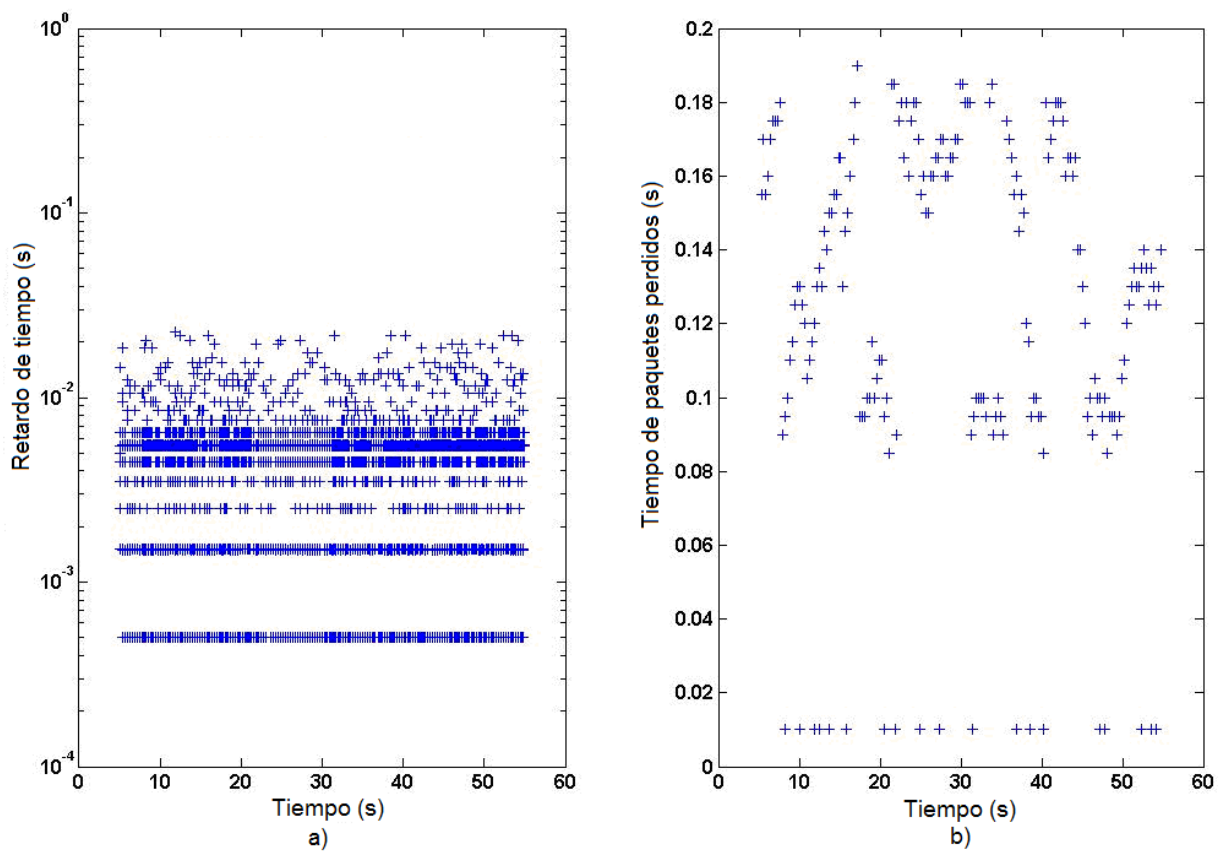


Figura 7.19. Retardos de tiempo y pérdida de paquetes en la red con 12 nodos de tráfico.

Con respecto a la red, el rango del retardo de tiempo (Figura 7.19a) es menor con respecto

al experimento anterior $[0.0005-0.022]$, pero en contraparte hay un aumento de los paquetes perdidos con un porcentaje del 38.5% (Figura 7.19b), 30 veces más que el experimento anterior.

Como se ha mostrado el desempeño IAE del controlador difuso es muy cercano al del control digital sin degradación por imperfecciones de red. La Figura 7.20 muestra la evolución del desempeño IAE en el tiempo de los tres experimentos tanto para el control difuso (línea sólida) como el control retroalimentado (línea punteada).

Siendo el control difuso más robusto a variaciones en los retardos de tiempo y paquetes perdidos, manteniendo estabilidad en el sistema con un incremento en el índice IAE y muy cercano al control digital, mientras que el control retroalimentado aumenta considerablemente el índice IAE conforme aumenta el tráfico de red.

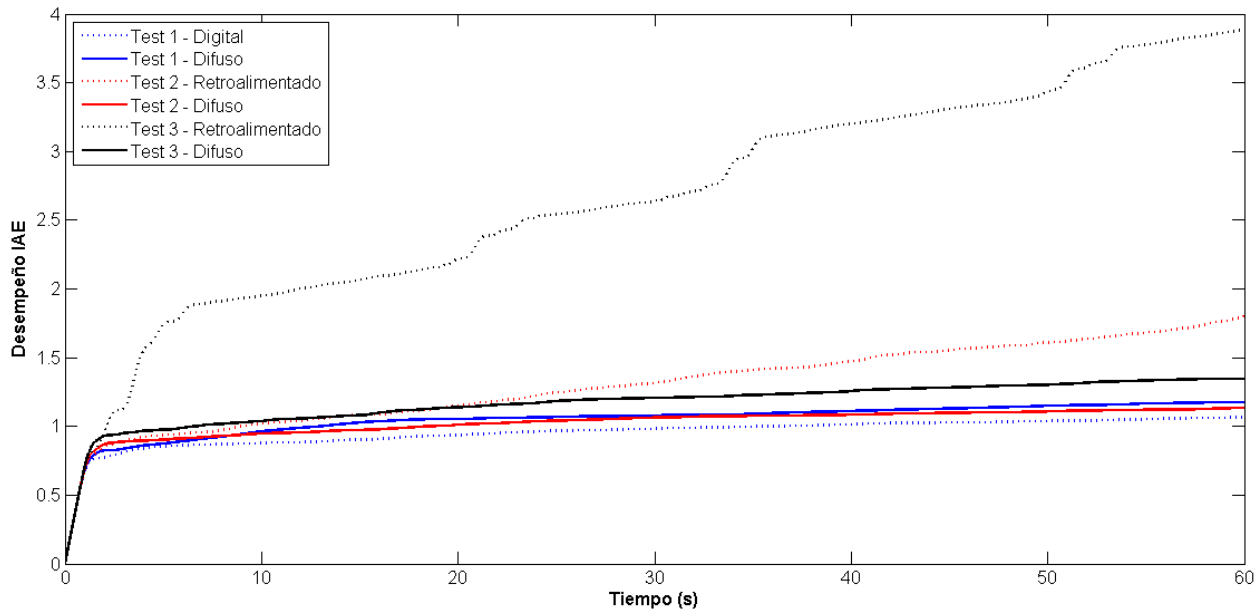


Figura 7.20. Desempeño IAE para el control difuso (—) y control retroalimentado (...)

En todos los experimentos se empleó la estimación del período de muestreo variable que es la combinación de los retardos de tiempo, paquetes perdidos e intervalos de muestreo variantes, en los experimentos no se presenta el comportamiento de los intervalos de muestreo variante debido a que es el mismo para los tres casos. Así para observar el comportamiento del período de muestreo variante se muestra en la Figura 7.21 para los tres niveles de tráfico, este es la suma en tiempo de las tres imperfecciones cuya información es complementaria con la Tabla 7.2.

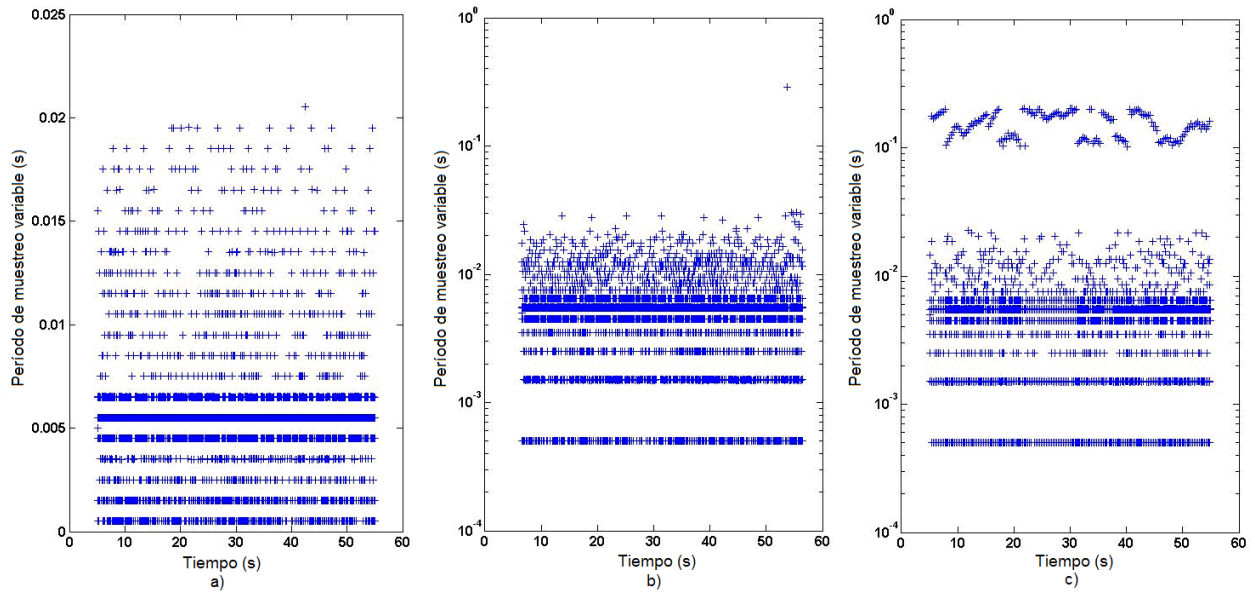


Figura 7.21. Periodos de muestreo variable para trafico a) ligero, b) medio y c) severo

La Tabla 7.2 muestra un resumen sobre el desempeño del NCS con el controlador difuso y el controlador retroalimentado, Aquí muestra en el caso del control difuso una mejor compensación de las imperfecciones de la red aún con tráfico severo, contrario al control retroalimentado que conforme aumenta el tráfico de red el desempeño del NCS se degrada. El controlador difuso selecciona el controlador local apropiado para mantener las mediciones de sobrepaso y error en estado estable en un nivel aceptable de acuerdo a las condiciones de la red mediante la medición de las imperfecciones de la red.

Tabla 7.2. Comparativa Difuso vs Retroalimentado

Control	Tráfico	τ_{sa} (ms)	λ_{sa} (%)	ζ	e_{ss}	IAE
Difuso	Ligero	0.5 – 7 – 21 0.5 – 9 – 35	0.05% 1.15%	1.3°	0.2°	1.1762
	Medio			1.5°	0.5°	1.2332
	Severo			2.5°	0.4°	1.3463
Retroalimentado	Ligero	0.5 – 5.5 – 22	38.5%	2°	0.2°	1.0652
	Medio			4°	3°	1.8043
	Severo			–	–	3.8890

7.3 Codiseño del sistema Helicóptero 2-DOF

En los experimentos anteriores se emplea información sobre el desempeño del sistema y de la red para que un controlador difuso compense las imperfecciones generadas por la red, sin embargo, una manera de ayudar a compensar las imperfecciones de la red es disminuir el tráfico.

Disminuir el tráfico requiere algunas tareas como planificar el envío de los paquetes, aumentar la velocidad del bus, disminuir la tasa de transmisión de los nodos, disminuir el tamaño de los paquetes, etc. De todos los aspectos anteriores el más sencillo de implementar y diseñar de manera local en cada uno de los nodos es disminuir la tasa de transmisión de los nodos. Una menor transmisión de paquetes por unidad de tiempo disminuye el tráfico y proporciona mayor tiempo para que el paquete llegue a su destino.

El planificador difuso mostrado en la sección 5.4 tiene el objetivo de modificar los periodos de muestreo (transmisión) del NCS en función del desempeño del sistema y de la red. El diseño del planificador difuso completa el codiseño controlador – planificador, donde el controlador fue diseñado para un Helicóptero simulador 2-DOF en la sección anterior. Para el diseño y validación del planificador difuso propuesto se emplea el mismo simulador con el controlador difuso diseñado en la sección anterior y la misma configuración de la Figura 4.1b que es completamente distribuida.

El objetivo general del codiseño es primero controlar la posición de pitch y yaw ante variaciones en las condiciones de la red y segundo planificar la transmisión del NCS. El objetivo particular del controlador difuso es generar la señal de control adecuada para compensar las imperfecciones variantes de la red. Mientras que el objetivo particular del planificador difuso es modificar el período de transmisión del NCS a fin de mejorar la calidad de servicio QoS sin degradar la calidad de control QoC.

El diseño del planificador difuso mostrado en la sección 5.4 se divide en dos etapas, la etapa de medición que implementa un modulo en el nodo actuador que mide el QoS y QoC cada período de planificación, y la etapa que implementa un modulo en el nodo sensor con el planificador difuso que calcula el siguiente período de muestreo del NCS de acuerdo a QoS y QoC. En el caso particular, el QoC es medido con el error absoluto medio (EAM) (5.24) y el QoS con el número de plazos perdidos (*deadlines*) (5.25).

De acuerdo a las mediciones del desempeño del sistema y de la red en las secciones previas y en específico del Helicóptero simulador 2-DOF y la red Ethernet utilizada, el rango del EAM para el diseño del planificador es $[0,1.4]$, mientras que para el porcentaje de plazos perdidos el rango es $[0,1]$ con un plazo máximo $h_{\max}=30\text{ ms}$, esto con un período de planificación

$\delta = 1s$. Para efectos de diseño se eligen cuatro membresías difusas para cada rango, con un rango de salida para el período de muestreo nuevo de $[0.005, 0.01, 0.015, 0.020]$, con esto se generan 16 reglas difusa cuya superficie es mostrada en la figura, donde las correspondencias de las reglas se generaron de manera heurística en base a conocimiento.

Para mostrar la aplicabilidad del codiseño, se diseñaron experimentos con el objetivo del seguimiento de una señal cuadrada para la posición de pitch, observando el comportamiento de las imperfecciones y el desempeño del sistema, cada experimento involucra la generación de tráfico en distintos niveles entre los tiempos $50 s$ y $80 s$. A fin comparativo se diseño un controlador híbrido que genera una ley de control por retroalimentación que asegura estabilidad ante las imperfecciones de red en el rango de la aplicación.

El primer experimento muestra el desempeño de codiseño con condiciones normales de operación sin tráfico dentro de la red. En este caso, el desempeño de ambos, el codiseño y el controlador híbrido cumplen los criterios de control. La posición de pitch para el codiseño difuso (Figura 7.22 sup.) mantiene un error en estado estable promedio $e_{ss} \approx 0.5$, con un tiempo de establecimiento $T_s = 3s$ y sin sobrepaso $\zeta = 0$. Siendo la respuesta del controlador híbrido muy similar y no se muestra en la gráfica.

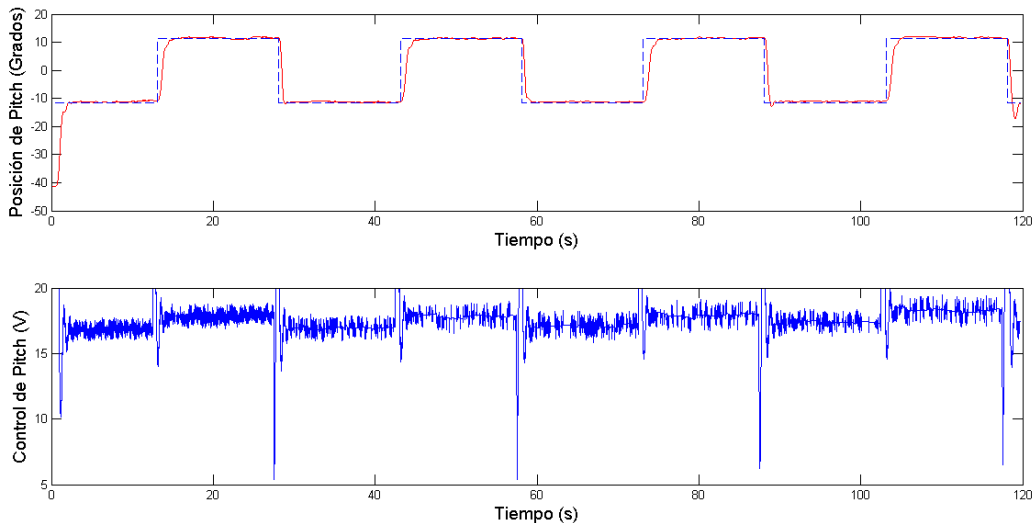


Figura 7.22. Desempeño del codiseño y señal de control sin tráfico externo a la red.

La Figura 7.22 (inf.) muestra la señal de control de pitch que mantiene un comportamiento

en rango normal de operación con variaciones en las imperfecciones de la red que son compensadas por el controlador difuso.

Con respecto al comportamiento de la red, sin tráfico externo solo se presentan retardos de tiempo (Figura 7.23) con un máximo de 4 ms y una media de 1 ms pero lo más importante sin pérdida de plazos límite.

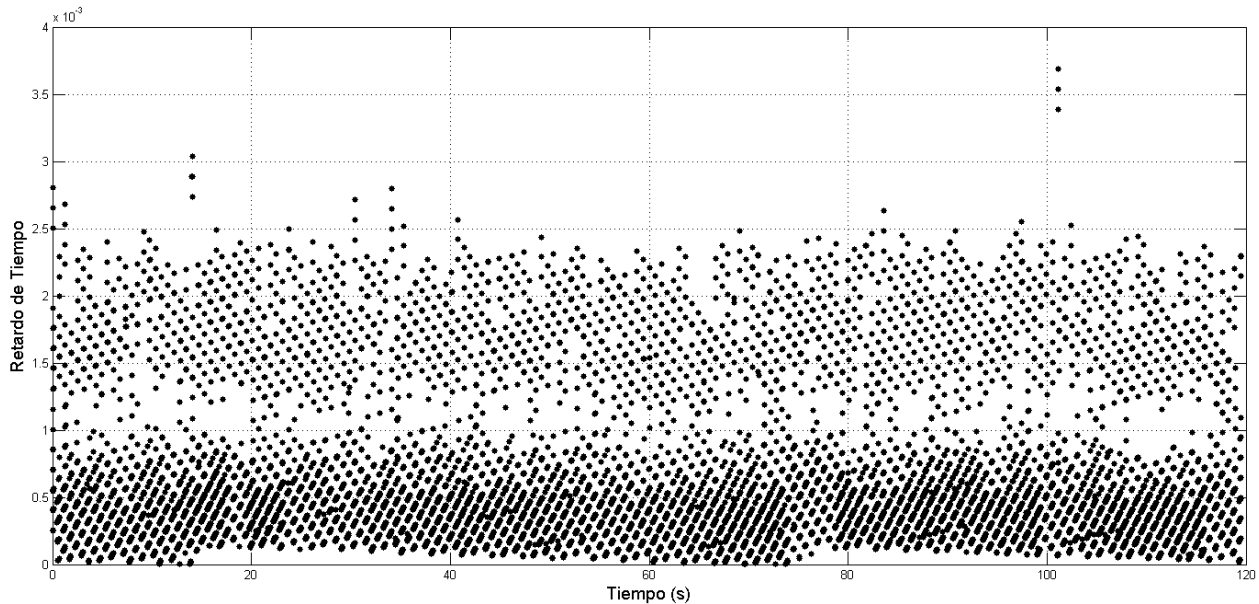


Figura 7.23. Comportamiento de la red sin tráfico externo para el codiseño NCS

Observado el comportamiento del sistema y la red, el complemento de codiseño NCS es decir el planificador difuso evalúa su desempeño cada período de planificación $\delta = 1s$ y definir el siguiente período de muestreo h , la Figura 7.24 muestra el desempeño del sistema que es medido a través del Error Absoluto Medio (EAM) (sup.) y el desempeño de la red que mide el porcentaje de plazos perdidos (med.), ambos cada período de planificación, también se muestra el comportamiento del planificador difuso que calcula el período de muestreo siguiente en un rango $[0.005, 0.02]$ segundos, disminuyendo cuando se tienen errores grandes y aumentando cuando el error es pequeño y requiere pocos cambios de control por unidad de tiempo.

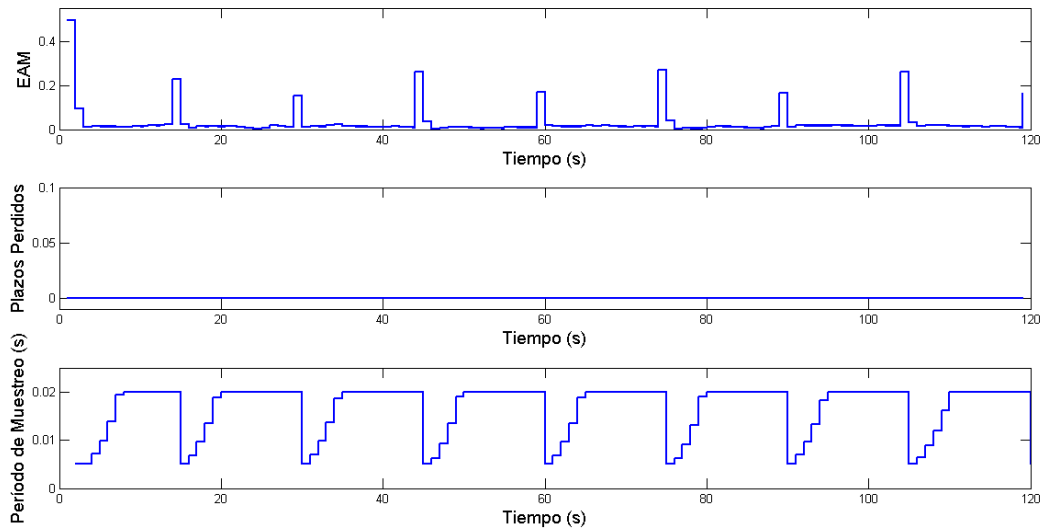


Figura 7.24. EAM, plazos perdidos y nuevo período para el NCS sin tráfico externo.

A fin de observar el comportamiento del codiseño NCS ante variaciones en las imperfecciones de la red se genera tráfico a distintos niveles observando el comportamiento del sistema con el controlador difuso, de la red y del período de muestreo con el planificador difuso.

Primero se aplica tráfico ligero en el rango de $[50,80] s$, el desempeño tanto del codiseño difuso y el control híbrido se mantienen estables (Figura 7.25), el codiseño (sup.) tiene un sobrepaso de $\zeta = 4^\circ$, con error en estado estable promedio $e_{ss} \approx 0.5$ y un tiempo de establecimiento $T_s = 3s$, pero los efectos de degradación del tiempo $50 - 80 s$ son mínimos; mientras que el control híbrido (inf.) degrada su desempeño ante la variación de tráfico sobre la red con un sobrepaso de $\zeta = 34^\circ$, un error en estado estable promedio $e_{ss} \approx 2$ y un tiempo de establecimiento $T_s = 10s$.

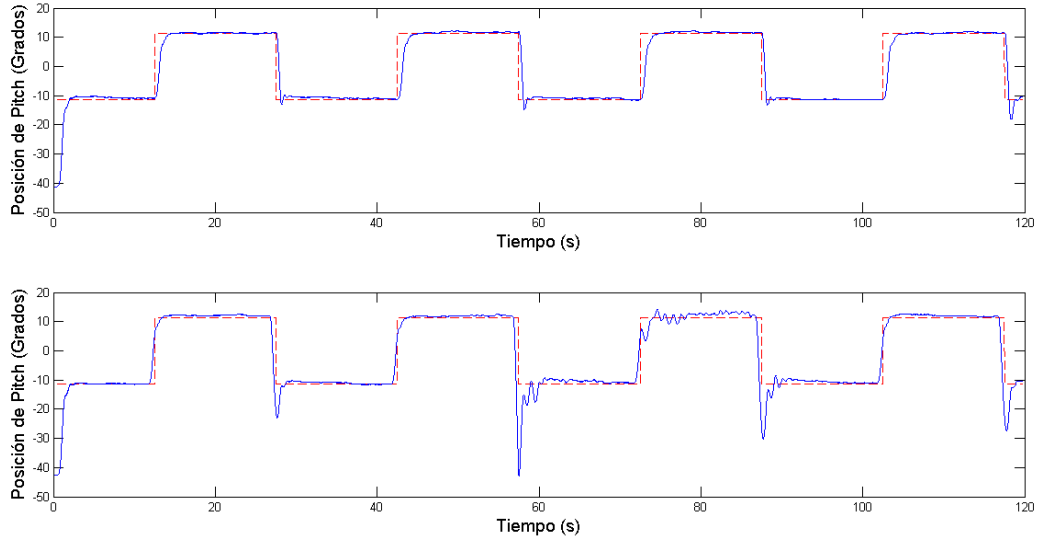


Figura 7.25. Desempeño del codiseño difuso y control híbrido con tráfico ligero (50 – 80 s).

El comportamiento de las señales de control de pitch se muestra en la Figura 7.26, el cambio de tráfico inicia en 50 s ambos sistemas no reflejan el cambio de tráfico en el desempeño ni la señal de control hasta que se presenta un cambio en la referencia que indica un incremento en el error en 58 s, la diferencia reside en que el codiseño difuso (sup.) realiza solo pequeños cambios en la señal de control, mientras que el control híbrido (inf.) realiza cambios abruptos debido a falta de información por los plazos perdidos que no contempla al calcular la señal de control.

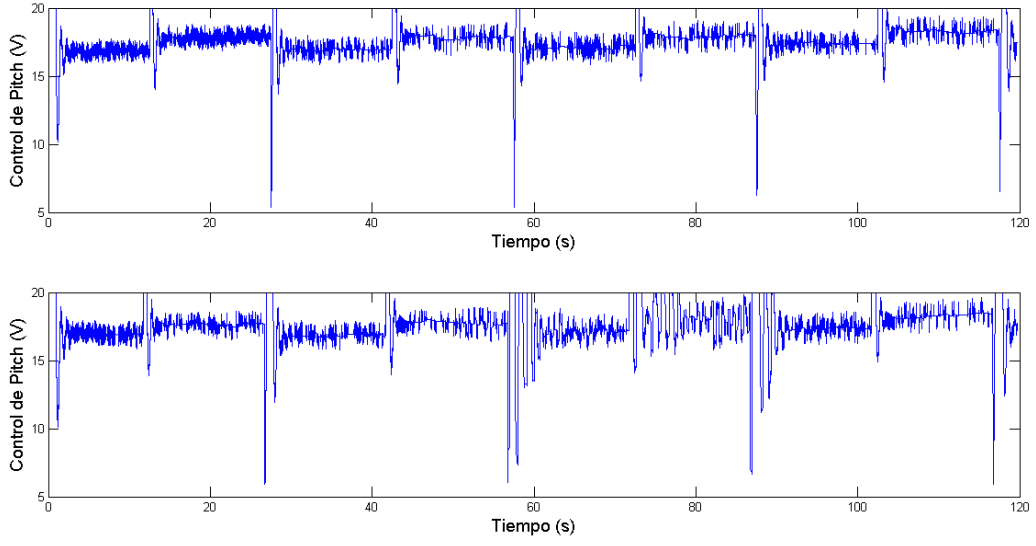


Figura 7.26. Señal de control del codiseño difuso y control híbrido con tráfico ligero.

El comportamiento de la red ante tráfico ligero genera un incremento en el comportamiento del retardo y la pérdida de plazos (Figura 7.27), El retardo de tiempo tiene un máximo de 27 ms, con una media de 10 ms con tráfico ligero, con un porcentaje de plazos perdidos 4.07% con un promedio de 146 ms entre plazos perdidos. La gráfica superior muestra el retardo de tiempo donde el tráfico ligero inicia en 50s sin embargo, el efecto inicia segundos después cuando debido a que la red se inunda de paquetes, por otro lado el tráfico termina en 80 s, pero su efecto se prolonga hasta los 90 s debido a que el switch sigue vaciando la cola de paquetes. Algo similar sucede en los plazos perdidos, donde los paquetes pierden sus plazos debido al encolado de los paquetes.

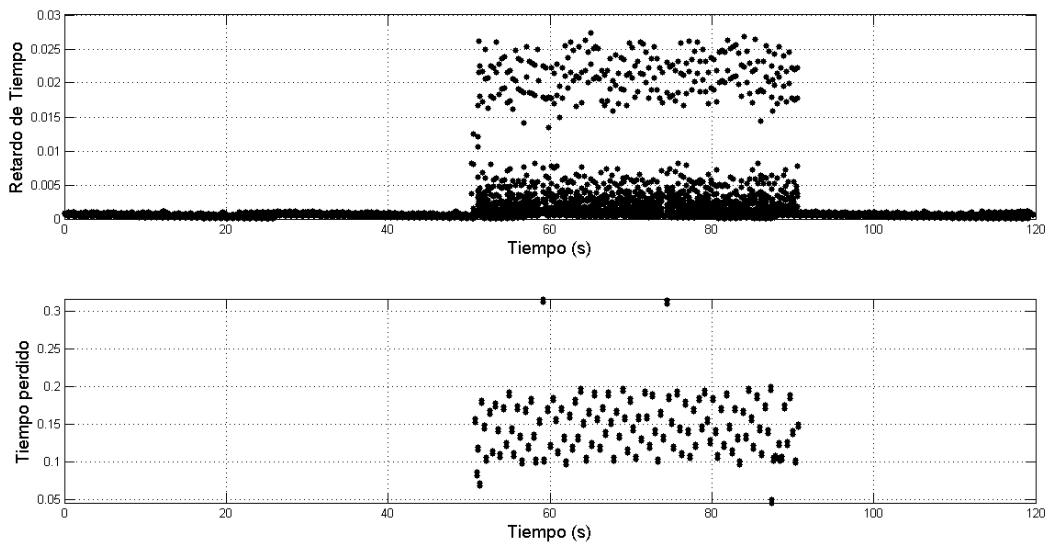


Figura 7.27. Retardos de tiempo y pérdida de plazos con tráfico ligero de 50 – 80 s.

Como el caso anterior, el planificador difuso evalúa el desempeño del sistema y de la red cada período de planificación $\delta = 1s$ y definir el siguiente período de muestreo h , al presentarse tráfico ligero el EAM (sup.) sufre ligeras variaciones que modifican el comportamiento del planificador, sumado a la presencia del 4% de plazos perdidos (med.), así el comportamiento del planificador difuso ante el tráfico ligero aumenta con una tasa de cambio menor el período de muestreo cuando se tiene un error pequeño a fin de no degradar el desempeño del sistema por la falta de información debida a la presencia de plazos perdidos.

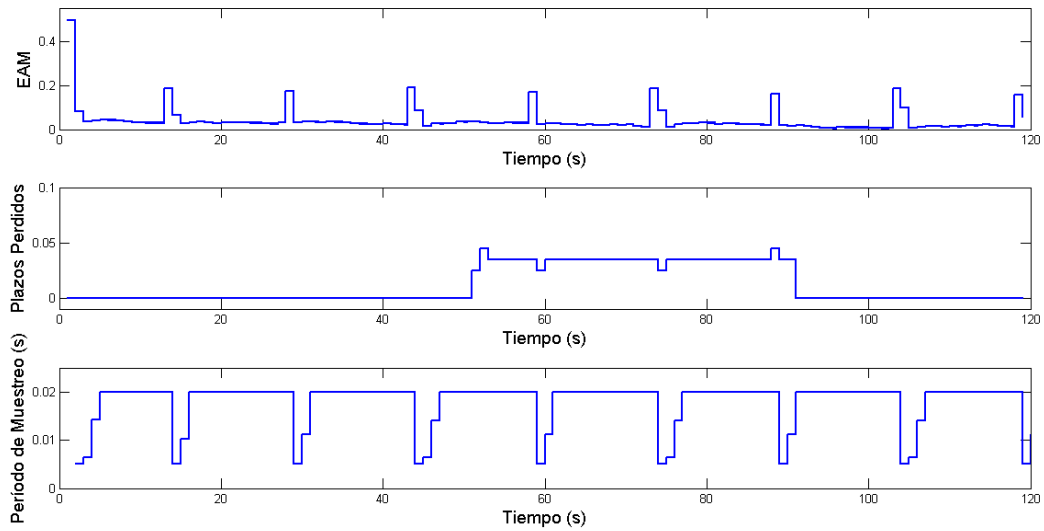


Figura 7.28. Desempeño sistema – red, y comportamiento del período de muestreo ante tráfico ligero.

El siguiente caso, observa el comportamiento del sistema aplicando tráfico medio en el rango de $[50,80]s$, donde se establece la diferencia entre el desempeño del codiseño difuso y el control híbrido, aunque ambos se mantienen estables (Figura 7.29), el desempeño de codiseño es el que cumple criterios de control, el codiseño (sup.) ante tráfico medio tiene un sobrepaso de $\zeta = 15^\circ$, con error en estado estable promedio $e_{ss} \approx 0.5$ y un tiempo de establecimiento $T_s = 4s$; mientras que el control híbrido (inf.) degrada su desempeño ante tráfico medio con un sobrepaso de $\zeta = 34^\circ$, un error en estado estable promedio $e_{ss} \approx 6$ y con oscilaciones. A este nivel de tráfico el control híbrido asegura estabilidad del sistema pero con un mal desempeño.

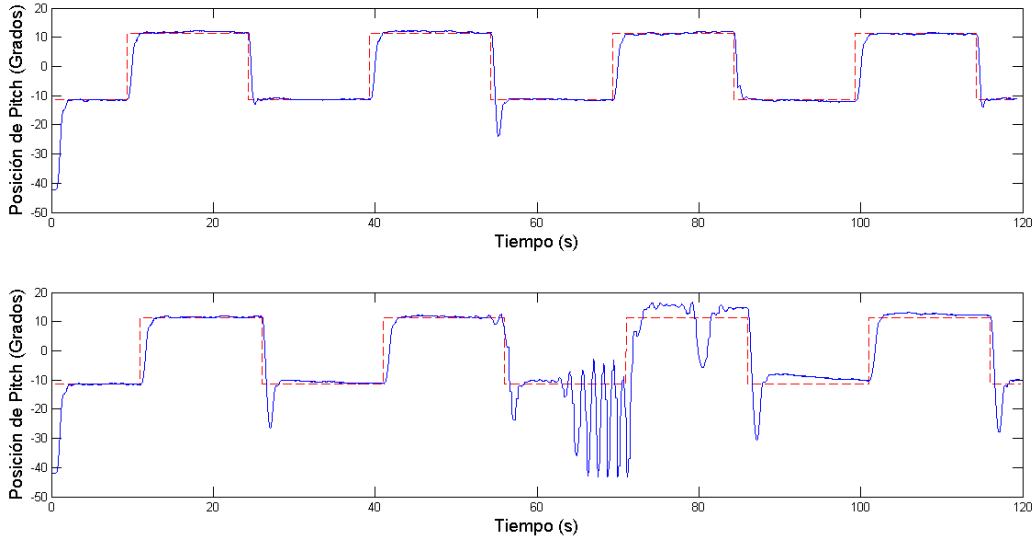


Figura 7.29. Desempeño del codiseño difuso y control híbrido con tráfico medio de 50 – 80 s.

El mal desempeño del sistema con señal de control híbrido (Figura 7.30) (inf.) se debe a cambios abruptos en un amplio rango de operación ante el tráfico que muestra una dinámica rápida del controlador, la diferencia reside en que el codiseño difuso (sup.) conmuta a una dinámica más lenta realizando pequeños cambios en la señal de control.

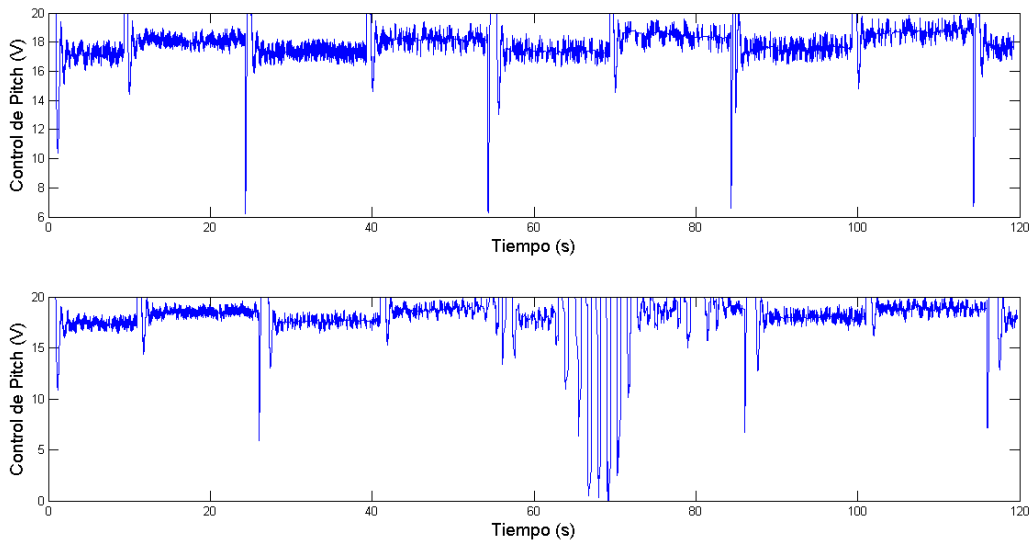


Figura 7.30. Señal de control del codiseño difuso y control híbrido con tráfico medio.

El comportamiento de la red ante tráfico medio genera un comportamiento peculiar en el

retardo y la pérdida de plazos (Figura 7.31), El retardo de tiempo se incrementa al inicio del tráfico con un máximo de 31 ms, con una media de 5 ms, pero conforme aumenta el tráfico los retardos se convierten en plazos perdidos con un porcentaje del 5.8% con un promedio de 132 ms entre plazos perdidos. La gráfica superior muestra el retardo de tiempo donde el tráfico medio inicia en 50s sin embargo, el efecto inicia segundos después y se decreta alrededor del segundo 55 dando paso a un incremento en los plazos perdidos al segundo 80, pero su efecto se prolonga hasta los 85s debido a que el switch sigue vaciando la cola de paquetes.

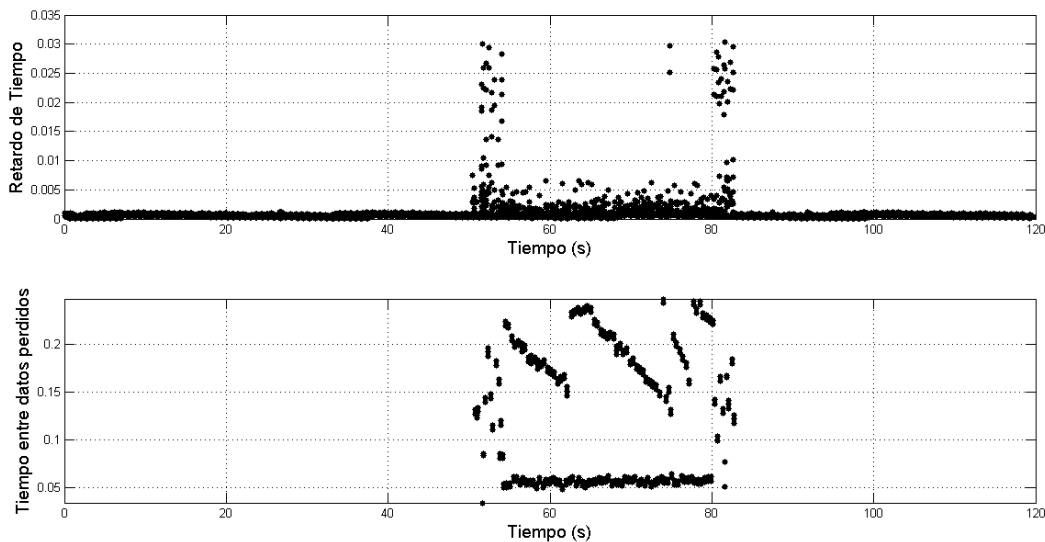


Figura 7.31. Retardos de tiempo y pérdida de plazos con tráfico medio de 50 – 80 s.

En este caso, el planificador difuso cambia la política de asignación del período de muestreo (Figura 7.32) (inf.), al presentarse tráfico medio el EAM (sup.) sufre variaciones mayores a 0.1 que representa un error en estado estable mayor al 2%, y sumado a la presencia del 5.8% de plazos perdidos (med.), el planificador difuso decide disminuir el período de muestreo lentamente pero sin sobrecargar la red y mejorar gradualmente el desempeño del sistema.

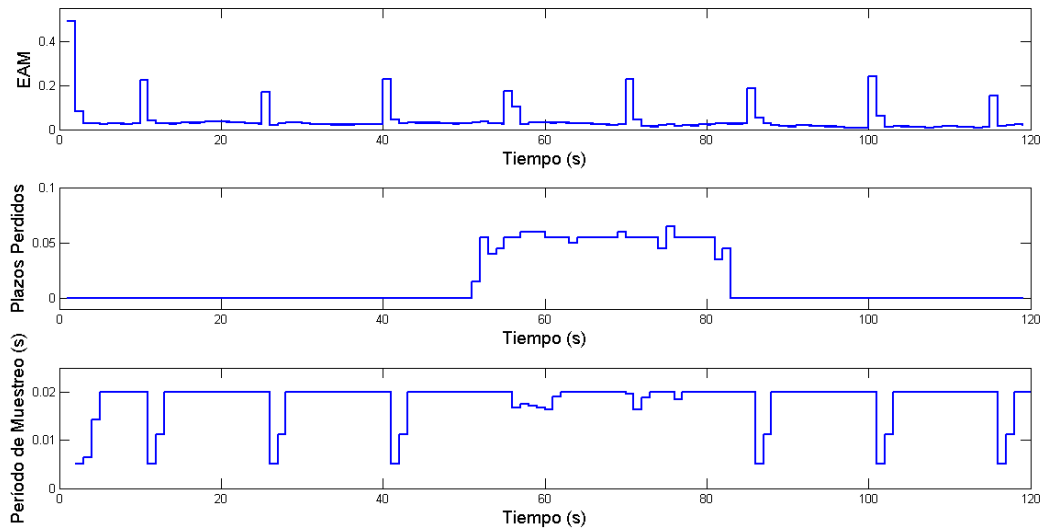


Figura 7.32. Desempeño del sistema – red con tráfico medio y período de muestreo del planificador difuso.

El caso con tráfico severo refleja un comportamiento errático con el controlador híbrido, a diferencia el desempeño del codiseño difuso se mantiene estable (Figura 7.33) cumpliendo los criterios de control, el codiseño (sup.) ante tráfico medio tiene un sobrepaso de $\zeta = 30^\circ$ pero un tiempo de duración muy corto (3s), con error en estado estable promedio $e_{ss} \approx 0.8$ y un tiempo de establecimiento $T_s = 4s$; mientras que el control híbrido (inf.) degrada su desempeño ante tráfico severo con un sobrepaso de $\zeta = 36^\circ$, un error en estado estable promedio $e_{ss} \approx 10$ y se mantiene el sistema con mal desempeño aún cuando el tráfico ha desaparecido. Con este caso, se muestra que el controlador difuso tiene mayor robustez ante mayores variaciones de las imperfecciones de la red que resultado recientes disponibles en la literatura como lo es el control híbrido.

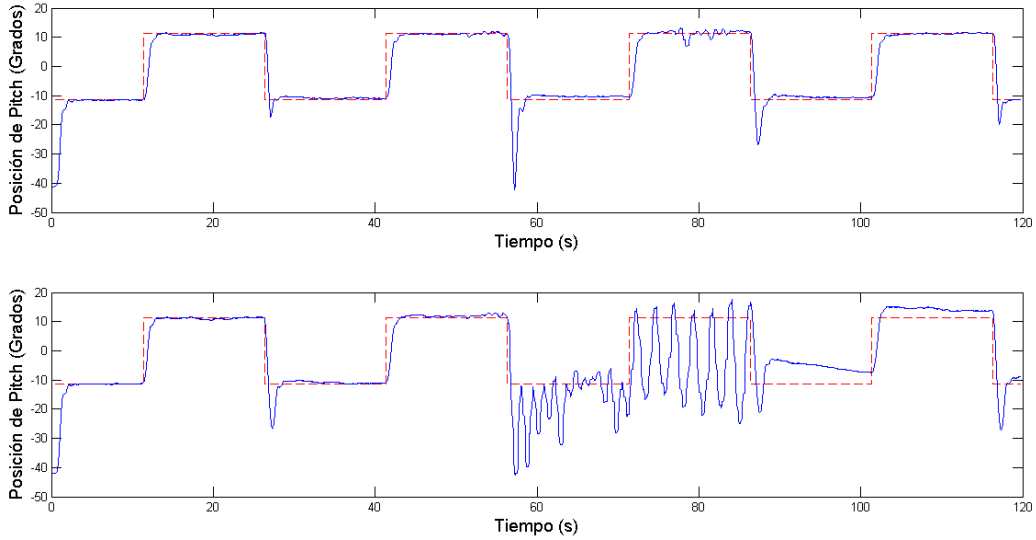


Figura 7.33. Desempeño del codiseño difuso y control híbrido con tráfico severo de 50 – 80 s.

El comportamiento de las señales de control de ambos controladores son muy similares al caso con tráfico severo (Figura 7.34), el controlador híbrido realiza cambios abruptos en un amplio rango de operación mostrando una dinámica rápida del controlador, mientras que el codiseño difuso (sup.) conmuta a una dinámica más lenta realizando pequeños cambios en la señal de control.

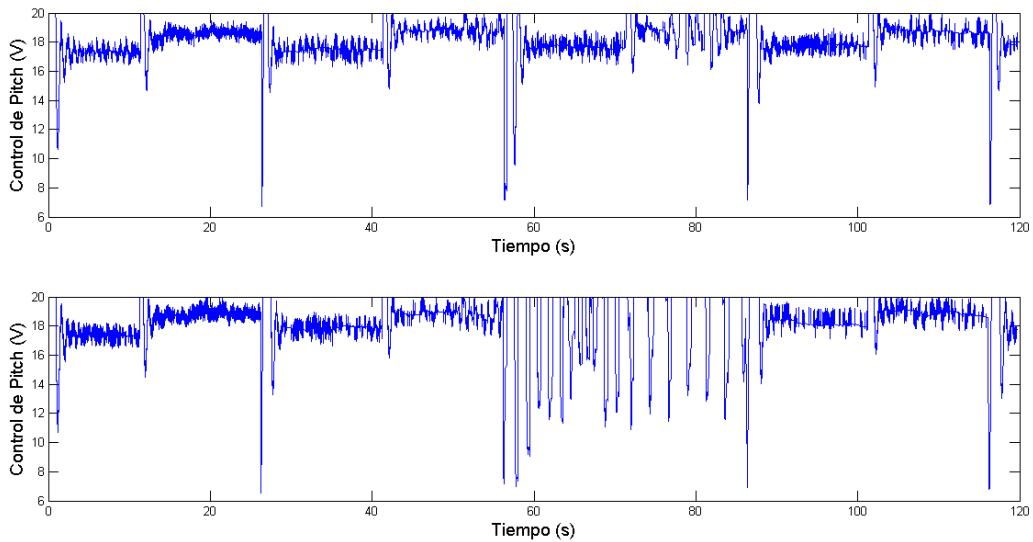


Figura 7.34. Señal de control del codiseño difuso y control híbrido con tráfico severo.

Ante tráfico severo, tanto el retardo de tiempo como los plazos perdidos se incrementan, retardo máximo es de 30 *ms*, con una media de 12 *ms*, y los plazos perdidos son un porcentaje de 7.3% con un tiempo promedio de 122 *ms* entre plazos perdidos. La gráfica superior muestra el retardo de tiempo donde el tráfico medio inicia en 50s sin embargo su efecto se prolonga hasta los 85s aún cuando el tráfico se termina a los 80s.

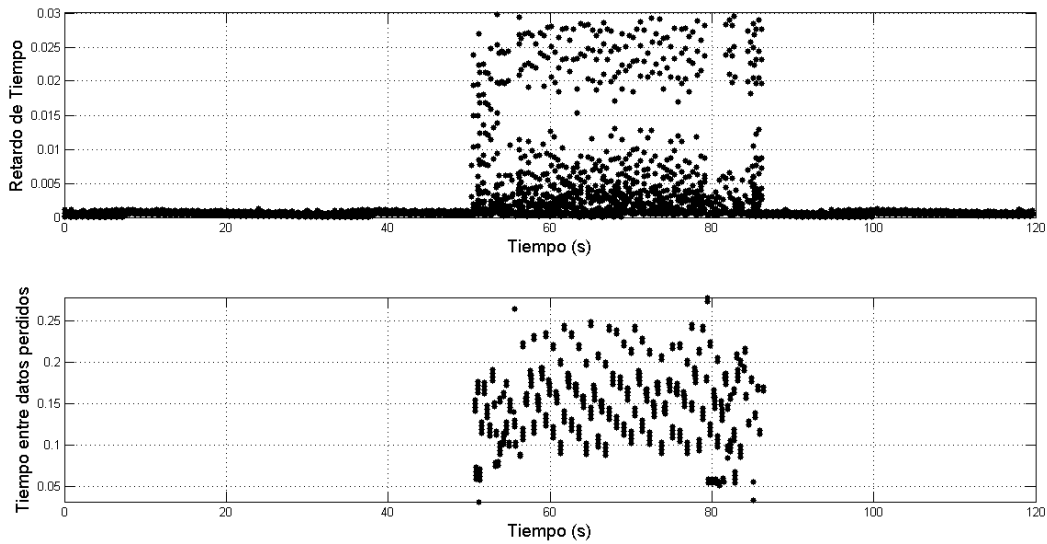


Figura 7.35. Retardos de tiempo y pérdida de plazos con tráfico severo de 50 – 80 s.

Ante tráfico severo, el planificador difuso cambia la política de asignación del período de muestreo (Figura 7.36) (inf.), esto debido a que el EAM (sup.) sufre variaciones mayores a 0.25 que representa un error en estado estable mayor al 5%, y sumado a la presencia del 7.8% de plazos perdidos (med.), el planificador difuso decide disminuir el período de muestreo lentamente a fin de corregir el error aún cuando se aumente el tráfico en la red.

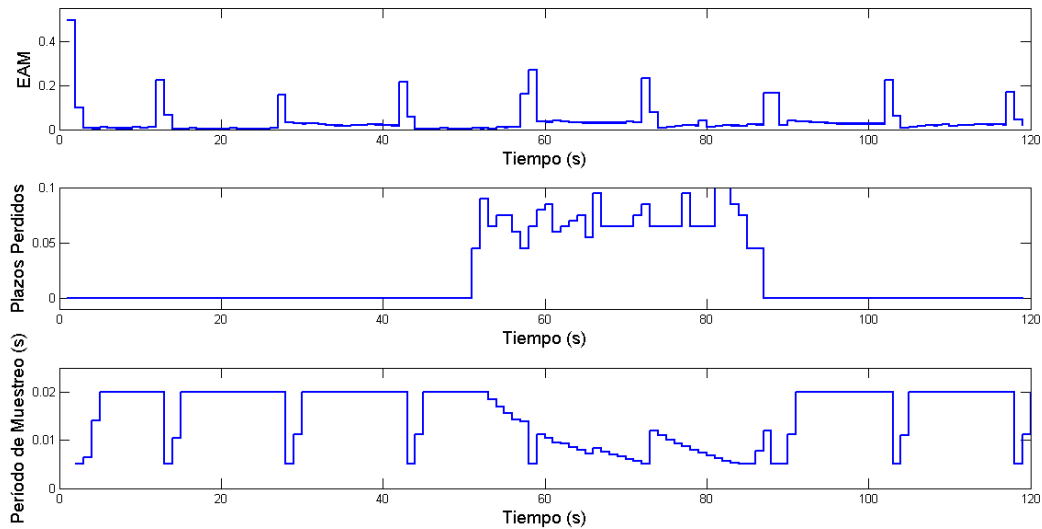


Figura 7.36. Desempeño del sistema – red con tráfico severo y período de muestreo del planificador difuso.

7.4 Conclusiones

En el capítulo se presentan las pruebas experimentales para validar el trabajo de doctorado, esto es el modelo, controlador y planificador difuso para NCS compensando las imperfecciones de tiempo inducidas por al red.

Se presentan tres configuraciones de NCS comunes en aplicaciones, y se aplican a dos sistemas inestables a lazo abierto, un MAGLEV SISO y un simulador helicóptero 2-DOF MIMO para mostrar la amplia aplicabilidad del método, para cada sección se emplean diferentes métricas para las imperfecciones que son mostradas en el capítulo 4.

Se diseña el modelo difuso que cumple la función de aproximador universal para ambos casos, SISO y MIMO incorporando el comportamiento de la red a través de la parte antecedente, en el caso del MAGLEV incorpora los retardo de tiempo ida y vuelta y en el simulador helicóptero 2-DOF incorpora los retardos de tiempo, paquetes perdidos e intervalos de muestreo variantes. El diseño del modelo se realiza a partir del modelo lineal continuo del sistema y del comportamiento en las imperfecciones de tiempo generadas por la red.

El control difuso diseñado es aplicado a ambos casos de estudio, utilizando el modelo difuso diseñado y el modelo de las imperfecciones, de diseñan las matrices de retroalimentación para cada submodelo discreto a través de un diseño LQR. Se cubren dos objetivos de control, el

seguimiento y la estabilización, donde el objetivo particular es mantener el desempeño del sistema en un nivel deseado ante el comportamiento dinámico e impredecible de la red, el desempeño del controlador es comparado con algunas investigaciones presentes en la literatura obteniendo un mejor desempeño en todos los casos.

Al diseñar el planificador difuso se completa el codiseño para una clase de NCSs, el planificador difuso obtiene información del desempeño del sistema y el comportamiento de la red con un período de planificación, y selecciona el período de muestreo del nodo sensor. Esto permite adaptar el período de muestreo en base al desempeño del sistema y la capacidad de transmisión de la red. Se compara el desempeño del codiseño con un control híbrido presente en la literatura obteniéndose mayor robustez y desempeño.

8. Conclusiones

Se establece el marco de referencia para definir los sistemas de control basados en red (NCS) que intercambiar información entre los diferentes componentes del control en lazo cerrado. Se presenta una clasificación de las imperfecciones inducidas por la red al incorporarla en el control a lazo cerrado y se define como afectan estas imperfecciones en el desempeño de control del sistema. Se muestra que Ethernet por su naturaleza no determinística presenta casi la totalidad de las imperfecciones dependiendo de su configuración, marcado la pauta para establecer un codiseño que puede ser aplicado a las distintas redes empleadas para control que presentan menos imperfecciones inducidas por la red. Se hace una revisión del modelo y control difusos a fin de establecer su aplicabilidad como aproximador y controlador universal, con la capacidad para modelar no linealidades o fenómenos que se presenten dentro del sistema NCS.

Se realiza una revisión de la literatura para NCS, donde los retardos de tiempo y la pérdida de paquetes son las principales imperfecciones de la red estudiadas. Para los retardos de tiempo se han considerado como constantes, aleatorios y estocásticos, presentando herramientas para modelado como funciones de probabilidad y cadenas de Markov. Para sistemas con pérdida de paquetes en la literatura han proliferados dos modelos de comportamiento, el modelo de Bernoulli y modelos como cadenas de Markov. Con cualquier modelo de pérdida de paquetes un objetivo común es diseñar un filtro para estimar los estados del sistema.

Referente al trabajo doctoral, se presentan tres modelos de imperfecciones inducidas por la red, con los elementos para llevar a cabo sus mediciones y el método probabilístico para su estimación, presentando una función de distribución exponencial generalizada. Se obtienen tres modelos, el modelo de retardos RTT que es una métrica de las imperfecciones de viaje completo (ida y vuelta), el modelo *one way* es una métrica que requiere sincronización de tiempos empleado para estimar retardos one way, pérdida de paquetes o intervalos de muestreo variantes y el modelo *one shot* que es estima las imperfecciones en intervalos discretos.

La parte fundamental del trabajo de doctorado, es el codiseño, cuyas tareas es diseñar el modelo, controlador y planificador difuso para NCS compensando las imperfecciones de tiempo inducidas por al red. El modelo difuso incorpora el comportamiento del sistema ante

imperfecciones de tiempo a través de la parte antecedente, empleando para esto cualquiera de los tres modelos de imperfecciones presentados. El control difuso diseñado aplica una ley de control retroalimentada paralela, obteniendo matrices de retroalimentación para cada submodelo discreto a través de un diseño LQR. El controlador difuso considera el desempeño del sistema y el comportamiento de la red cuyo objetivo de control es mantener el desempeño del sistema en un nivel deseado ante el comportamiento de la red. Como parte adicional el diseño del planificador difuso se completa el codiseño de NCS, el planificador difuso incorpora información del desempeño del sistema y el comportamiento de la red para adaptar el período de muestreo.

Así con el codiseño controlador – planificador difuso propuesto se contempla tanto el desempeño del sistema y de la red con el objetivo final de mejorar y mantener el desempeño del sistema. El controlador difuso puede compensar tanto las variaciones en el desempeño de la red como el cambio en el período de muestreo adaptado por el planificador. Mientras que el planificador difuso puede compensar las variaciones de carga de la red y mejorar el desempeño del sistema cuando el controlador por su diseño no tenga esta capacidad.

Cada una de las tareas de codiseño son verificadas a través de pruebas experimentales para validar el trabajo de doctorado, esto es que el modelo, controlador y planificador difuso para NCS compense las imperfecciones de tiempo inducidas por al red. Se presentan tres configuraciones de NCS, aplicados a un MAGLEV SISO y un simulador helicóptero 2-DOF MIMO, donde ambos son sistemas inestables a lazo abierto.

Se diseña el modelo difuso para ambos casos de estudio, incorporando en cada caso las métricas de imperfecciones propuestas. El diseño del modelo se realiza a partir del modelo lineal continuo del sistema y del modelo de imperfecciones, en todos los casos el desempeño de modelado cumple criterios para efectos de control. El controlador difuso cubre dos objetivos de control, el seguimiento y la estabilización, el desempeño del controlador difuso es comparado con un trabajo reconocido presente en la literatura obteniendo un mejor desempeño en todos los casos.

El diseño del planificador difuso se realiza a través de la información del desempeño del sistema y el comportamiento de la red con un período de planificación, y se selecciona el período de muestreo del nodo sensor en un intervalo válido para el sistema. El planificador difuso se aplica para el caso del simulador Helicóptero donde se adapta el período de muestreo en base al

desempeño del sistema y la capacidad de transmisión de la red. Se compara el desempeño del codiseño con un control híbrido presente en la literatura obteniéndose mayor robustez y desempeño ante la variación de tráfico en la red.

Como línea de investigación a futuro, el trabajo doctoral puede ser extendido a otro tipo de redes empleadas para control Ej. DeviceNet y ControlNet debido a que presentan menor complejidad en las imperfecciones de la red, por otro lado, también es extendible a otras configuraciones NCS que son combinaciones de las presentadas en este trabajo. Con la técnica de codiseño propuesta es posible agrupar múltiples lazos de control dentro de una misma red y poder sincronizarse a fin de evitar colisiones y obtener el mejor rendimiento de la red y el desempeño deseado para cada lazo de control.

9. Referencias

- [1] Y. Tipsuwan and M. Y. Chow; "Control methodologies in networked control systems"; *Control Engineering Practice*, Vol. 11, No. 10, pp. 1099–1111, febrero 2003.
- [2] K. Ji, W. J. Kim, and A. Srivastava; "Internet-based real-time control architectures with time-delay/packet-loss compensation"; *Asian Journal of Control*, Vol. 9, No.1, marzo 2006.
- [3] J. Eker and A. Cervin; "Distributed wireless control using Bluetooth"; *Proceeding of IFAC Conference on New Technologies for Computer Control*, Hong Kong, P.R. China, pp. 1– 6, noviembre 2001.
- [4] N. J. Ploplys, P. A. Kawka, and A. G. Alleyne; "Closed-loop control over wireless network"; *IEEE Control System Magazine*, Vol. 24, No. 3, pp. 58–71, junio 2004.
- [5] S. Lankes, M. Reke, and A. Jabs; "A time-triggered Ethernet protocol for real-time CORBA"; *Proceedings of the 5th IEEE International Symposium on Object-Oriented Real-Time Distributed Computing*, Washington DC, USA, pp. 215–222, Abril 2002.
- [6] C. Venkatramani; *Design, Implementation and Evaluation of RETHER: A Real-Time Ethernet Protocol*; Tesis Ph.D., State University of New York at Stony Brook, diciembre 1996.
- [7] J. Nilsson; *Real-Time Control Systems with Delays*; Tesis Ph.D, Dept. Automatic Control. Lund Institute of Technology, Lund, Suecia, enero 1998.
- [8] R. Luck and A. Ray; "An observer-based compensator for distributed delays"; *Automatica*, Vol. 26, No. 5, pp. 903–908, 1990.
- [9] Y. Halevi and A. Ray; "Integrated communication and control systems : Part I - Analysis"; *Journal of Dynamic Systems, Measurement, and Control*, Vol. 110, pp. 367- 373, 1988.
- [10] S. Shakkottai, A. Kumar, A. Karnik, and A. Anvekar; "TCP performance over end-to end rate control and stochastic available capacity"; *IEEE/ACM Transactions on Networking*, Vol. 9, No. 4, pp. 377-391, 2001.
- [11] J. Filipiak; *Modelling and Control of Dynamic Flows in Communication Networks*; Berlin, Alemania: Springer-Verlag, 1988.
- [12] Q. Li and D. L. Mills; "Jitter-based delay-boundary prediction of wide-area networks"; *IEEE/ACM Transactions on Networking*, Vol. 9, No. 5, pp. 578-590, 2001.
- [13] R. Krtolica, Ü. Özgüner, H. Chan, H. Göktas, J. Winkelman, and M. Liubakka; "Stability of linear feedback systems with random communication delays"; *International Journal of Control*, Vol. 59, No. 4, pp. 925-953, 1994.
- [14] S. H. Hong; "Scheduling Algorithm of Data Sampling Times in the Integrated Communication and Control Systems"; *IEEE Transactions of Control Systems Technology*, Vol.3 No.2, junio 1995.
- [15] C. G. Pegden, R. E. Shannon and R. P. Sadowski; "Introduction to simulation using SIMAN"; McGraw Hill, 1995.
- [16] O. C. Ibe y x. Cheng; "Stability conditions for multiqueue systems with cyclic service"; *Transaction on Automatic Control*, Vol. 33, pp.102-103, 1988.
- [17] K. J. Aström and B. Wittenmark; *Computer-controlled systems: Theory and Design*; 2 ed. Englewood Cliffs, NJ: Prentice–Hall, Inc., 1990.
- [18] L. W. Liou and A. Ray; "Integrated communication and control systems: Part III- Nonidentical sensor and controller sampling"; *Journal of Dynamics Systems, Measurement and Control*, Vol. 112, pp. 357-364, 1990.
- [19] H. Chan and U. Özgüner; "Closed-up control of systems over a communication network with queues"; *International Journal of Control*, Vol.62, pp. 493-510, 1995.
- [20] J. Nilsson B. Bernhardsson and B. Wittenmark; "Stochastic analysis and control of real-time systems with random time delays"; *Automatica*, Vol. 34, pp. 57-74, 1998.
- [21] G. C. Walsh, O. Beldiman, L. Bushnell; "Asymptotic Behavior of Networked Control Systems"; *Proceedings 1999 IEEE International Conference on Control Applications*, Vol. 2, pp. 1448-1453, Kohala, Hawaii, EU, agosto 1999.
- [22] G. C. Walsh, H. Ye and L. Bushnell; "Stability analysis of networked control systems"; *American Control Conference*, Vol. 4, pp. 2876 - 2880, junio 1999.
- [23] G. C. Walsh, O. Beldiman and L. Bushnell; "Error encoding algorithms for networked control systems"; *Proceedings of the 38th IEEE conference on decision and control*, Vol. 5, pp. 4933 - 3938, diciembre 1999.
- [24] H. K. Khalil; *Nonlinear systems*; 2 ed. Upper Saddle River, NJ: Prentice Hall, 1995.
- [25] L. Sha, T. Abdelzaher, K. E. Arzén, T. Baker, A. Burns, G. buttazzo, M. Caccamo, A. Cervin, J. Lehoczky, A. Mok; "Real-Time scheduling theory: A historial perspective"; *Real-time Systems*, Vol. 28, pp. 101-155, 2004.
- [26] C. Lu, J. A. Stankovic, G. Tao, and S. H. Son; "Feedback control real-time scheduling: framework, modeling and algorithms"; *Real-Time Systems*, Vol. 23, pp. 85-126, 2002.
- [27] K. E. Arzen, B. Bernhardsson, J. Eker, A. Cervin, K. Nilsson, P. Persson, and L. Sha; "Integrated control and scheduling"; Technical Report ISRN LUTFD2/TFRT7586SE. Lund Institute of Technology, Suecia, 1999.
- [28] M. S. Branicky, S. M. Phillips and W. Zhang; "Scheduling and feedback co-design for networked control systems"; *Proceedings of IEEE Conference on Decision and Control*, Vol. 2 pp. 1211-1217, diciembre 2002.
- [29] J. W. S. Liu, *Real-time Systems*, Prentice Hall, 2000.
- [30] T. J. Tarn and N. Xi; "Planning and control of internet-based teleoperation"; *Proceedings of SPIE: Telem manipulator and telepresence technologies V*, Vol. 35, No.24, pp. 189 - 193. Boston, MA, 1998.
- [31] N. B. Almutairi, M. Y. Chow, Y. Tipsuwan; "Network-Based Controlled DC motor with Fuzzy Compensation"; *Proceedings The 27th Annual Conference of the IEEE Industrial Electronics Society*, Vol. 3, pp. 1844-1849, Denver, EU, 2001.
- [32] M. S. Branicky, S. M. Phillips, W. Zhang; "Stability of Networked Control Systems: Explicit Analysis of Delay"; *Proceedings of American Control Conference*, Chicago, EU, pp. 2352-2357, junio 2000.

- [33] Y. Y. Cao, Y. X. Sun, C. Cheng “Delay-Dependent Robust Stabilization of Uncertain Systems with Multiple State Delays”; *IEEE Transactions on Automatic Control*, Vol. 43, No. 11, pp. 1608-1612, noviembre 1998.
- [34] Open DeviceNet Vendors Association; *DeviceNet Specification*, 2.0 Edición, 1997.
- [35] V. Gupta, D. Spanos, B. Hassibi, R. M. Murria; “On LQG Control Across a Stochastic Packet-Dropping Link”; *Proceedings 2005 American Control Conference*, pp. 360-365, Portland, EU, junio 2005.
- [36] J. Eidson and W. Cole; Ethernet Rules Closed-Loop Systems, in *Tech*, 39-42, junio 1998.
- [37] F. Göktas, J. M. Smith, R. Bajcsy; “ μ -Synthesis For Distributed Control Systems with Network-Induced Delays”; *Proceedings of 35th Conference on Decision and Control*, pp. 813-814, Kobe, Japón, diciembre 1996.
- [38] F. Halsall; *Data communications, computer networks and open systems*; 3rd ed., Addison- Wesley Publishing Company, 1992.
- [39] D. Hristu-Varsakelis, P. R. Kumar; “Interrupt-based feedback control over a shared communication medium” ; *Proceedings of the 41st IEEE Conference on Decision and Control*, pp. 3223-3228, Las Vegas, EU, diciembre 2002.
- [40] O. C. Imer, S. Yüksel, T. Basar; “Optimal Control of LTI Systems over Unreliable Communication Links” ; *Automatica*, Vol. 42, No. 9, pp. 1429-1439, septiembre 2006.
- [41] V.K. Khanna and S. Singh; “An improved Pigback Ethernet protocol and its analysis”; *Computer Networks and ISDN Systems*, Vol. 26, No. 11, pp. 1437-1446, agosto 1994.
- [42] G. Kaplan; Ethernet’s winning ways; *IEEE Spectrum*, Vol. 38, No. 1, pp. 113-114, junio 2001.
- [43] H. Koivo, P. Virtanen, M. Puseenius; “A Self-Tuning Controller for Processes with Time-Varying Delay: Application to a Paper Machine Head Box” ; *Preprint of IFAC International Symposium of Adaptive Control of Chemical Processes*, Copenhagen, Dinamarca, agosto 1988.
- [44] S.A. Koubias and G.D. Papadopoulos; “Modern fieldbus communication architectures for real/time industrial applications”; *Computing in Industry*, Vol. 26, No. 3, pp. 243-252, agosto 1995.
- [45] G. Paula; “Building a better fieldbus” ; *Mechanical Engineering*, pp. 90 – 92, 1997.
- [46] X. Li, C. E. de Souza; “Criteria for Robust Stability and Stabilization of Uncertain Linear Systems with State Delay”; *Automatica*, Vol. 33, No. 9, pp. 1657-1662, 1997.
- [47] F.L. Lian, J.R. Moyne and D.M. Tilbury; “Performance evaluation of control networks: Ethernet, ControlNet, and DeviceNet”; *IEEE Control Systems Magazine*, Vol. 21, pp. 66-83, febrero 2001.
- [48] B. Lincoln; *Dynamic programming and Time-Varying Delay Systems*; Tesis Ph.D., Lund Institute of Technology, 2003.
- [49] B. Lincoln, B. Bernhardsson; “Optimal control over networks with long random delays”; *Proceedings International Symposium on Mathematical Theory of Networks and Systems*, enero 2000.
- [50] L. A. Montestruque, P. Antsaklis; “Stability of Model-Based Networked Control Systems with Time-Varying Transmission Times” ; *IEEE Transactions on Automatic Control*, Vol. 49, No. 9, pp. 1562-1572, septiembre 2004.
- [51] P. Naghshtabrizi, J. P. Hespanha; “Designing an observer-based controller for a network control system” ; *Proceedings of the 44th IEEE Conference on Decision and Control, and the European Control Conference 2005*, Sevilla, España, diciembre 2005.
- [52] X. Nian; “Stability of Linear Systems with Time-Varying Delays: An Lyapunov Functional Approach”; *Proceedings of the American Control Conference*, Denver, EU, junio 2003.
- [53] G. Nikolakopoulos, A. Panousopoulou, A. Tzes, J. Lygeros; “Multi-hopping Induced Gain Scheduling for Wireless Networked Controlled Systems”; *Proceedings of the 44th IEEE Conference on Decision and Control, and the European Control Conference 2005*, Sevilla, España, diciembre 2005.
- [54] R. Oboe, P. Fiorini; “Issues on Internet-Based Teleoperation”; *Proceedings of the 5th IFAC Symposium on Robot Control*, pp. 611-617, Nantes, Francia, 1997.
- [55] Y. J. Pan, H. J. Marquez, T. Chen; “Stabilization of remote control systems with unknown time varying delays by LMI techniques”; *International Journal of Control*, Vol. 79, No. 7, pp. 752-763, julio 2006.
- [56] M. Pohjola; *PID controller design in networked control systems*; Tesis de maestría, Helsinki University of Technology, enero 2006.
- [57] B. Sinopoli, L. Schenato, M. Franceschetti, K. Poolla, S. Sastry; “An LQG Optimal Linear Controller for Control Systems with Packet Losses”; *Proceedings of the 44th IEEE Conference on Decision and Control, and the European Control Conference 2005*, Sevilla, España, diciembre 2005.
- [58] Y. Tan; “Time-Varying Time-Delay Estimation for Nonlinear Systems Using Neural Networks”; *International Journal of Applied Mathematics and Computer Science*, Vol. 14, No. 1, pp. 63-68, 2004.
- [59] A.S. Tanenbaum; *Computer Networks*; Prentice-Hall Inc., 3rd. Edición, 1996.
- [60] B. Tavassoli, P. J. Maralani; “Robust Design of Networked Control Systems with Randomly Varying Delays and Packet Losses”; *Proceedings of the 44th IEEE Conference on Decision and Control, and the European Control Conference 2005*, Sevilla, España, diciembre 2005.
- [61] Y. Tipsuwan, M. Y. Chow; “Network-Based Controller Adaptation Based on QoS Negotiation and Deterioration”; *Proceedings of the 27th Annual Conference of the IEEE Industrial Electronics Society*, Vol. 3, pp. 1794-1799, Denver, EU, 2001.
- [62] Y. Tipsuwan, M. Y. Chow; “On the Gain Scheduling for Networked PI Controller Over IP Network”; *IEEE/ASME Transactions on Mechatronics*, Vol. 9, No. 3, septiembre 2004.
- [63] E. Verriest, M. Egerstedt; “Control with Delayed and Limited Information: A First Look”; *Proceedings of the 41st IEEE Conference on Decision and Control*, pp. 1231-1236, Las Vegas, EU, diciembre 2002.
- [64] J.D. Wheelis; “Process Control Communication: Token Bus, CSMA/CD, or Token Ring”; *ISA Transactions*, Vol. 32, No. 2, pp. 193-198, julio 1993.
- [65] E. Witrant, P. G. Park, M. Johansson, C. Fischione, K. H. Johansson; “Predictive control over wireless multi-hop networks”; *Proceedings of the IEEE Conference on Control Applications*, Singapore, octubre 2007.
- [66] J. K. Yook, D. M. Tilbury, H. S. Wong, N. R. Soparkar; “Trading computation for bandwidth: state estimators for reduced communication in distributed control systems”; *Proceedings of 2000 Japan-USA Symposium on Flexible Automation*, Ann Arbor, EU, julio 2000.
- [67] W. Zhang; *Stability Analysis of Networked Control Systems*, Tesis Ph.D., Department of Electrical Engineering and Computer Science, Case Western Reserve University, agosto 2001.

- [68] L. Zhang, D. Hristu-Varvakelis; "LQG Control under Limited Communication"; *Proceedings of the 44th IEEE Conference on Decision and Control, and the European Control Conference 2005*, Sevilla, España, diciembre 2005.
- [69] T. Zhang, Y. C. Li; "A Fuzzy Smith Control of Time-Varying Delay Systems based on Time Delay Identification"; *Proceedings of the Second International Conference on Machine Learning and Cybernetics*, noviembre 2003.
- [70] X. M. Zhang, M. Wu, J. H. She, Y. He; "Delay-dependent stabilization of linear systems with time-varying state and input delays"; *Automatica*, Vol. 41, pp. 1405-1412, 2005.
- [71] O. Senane, D. Simon, D. Robert; "Feedback scheduling for real-time control of systems with communication delays"; *Proceedings of the IEEE Confer. On Emerging Technologies and Factory Automation*, 2003.
- [72] Truetime software: www.control.lth.se/truetime/
- [73] H. Benítez-Pérez, P. Quiñones-Reyes, P. E. Méndez-Monroy, F. García-Nocetti, And F. Cardenas-Flores; "Reconfigurable Fault Tolerant PID Networked Control for Magnetic Levitation Case Study "; *IFAC Symposium on Safeprocess*, Beijing, P.R. China. agosto 2006.
- [74] T. Takagi and M. Sugeno; "Fuzzy Identification of Systems and Its Applications to Modelling and Control"; *IEEE Transactions on Systems, Man and Cybernetics*, Vol. 15, No. 1, junio 1985.
- [75] M. Sugeno and G. T. Kang; "Structure Identification of Fuzzy Model"; *Fuzzy Sets and Systems*, Vol. 28, No. 1, pp. 15-33, octubre 1988.
- [76] K. Tanaka and M. Sugeno; "Stability Analysis and Design of Fuzzy Control systems"; *Fuzzy Sets and Systems*, Vol. 45. No. 2, pp. 135-156, junio 1992.
- [77] K. Tanaka, T. Ikeda and H. Wang; "Design of fuzzy Control Systems based on Relaxed LMI Stability Conditions"; *Proceedings of the 35th IEEE Decision and Control*, Vol. 1, pp. 598-603, diciembre 1996.
- [78] K. Tanaka and H. O. Wang; *Fuzzy Control Systems Design and Analysis: A Linear Matrix Inequality Approach*; New York: Wiley, 2001.
- [79] K. Takaba; "Robust Preview Tracking Control for Polytopic Uncertain Systems"; *Proceedings of the 37th Conference on Decision and Control*, pp. 1765-1770, diciembre 1998.
- [80] M. Heemels, A. N. Van de Wouw, D. Nęsić; "Networked Control Systems With Communication Constraints Tradeoffs Between Transmission Intervals, Delays and Performance"; *IEEE Transaction on Automatic Control*, Vol. 55, No. 8, agosto 2010.
- [81] F. Lian, J. Moyne, D. Tilbury; "Network Design Consideration for Distributed Control Systems"; *IEEE Transactions on Control Systems Technology*, Vol. 10, No.2, marzo 2002.
- [82] Betts, J. T; *Practical Methods for Optimal Control Using Nonlinear Programming*, SIAM Press, Philadelphia, Pennsylvania, 2001.
- [83] R. Brockett, "Stabilization of motor networks," in *Proc. 34th IEEE Conf. Decision Control*, 1995, vol. 2, pp. 1484–1488.
- [84] R. Brockett and D. Liberzon, "Quantized feedback stabilization of linear systems," *IEEE Trans. Autom. Control*, vol. 45, no. 7, pp. 1279–1289, Jul. 2000.
- [85] D. Carnevale, A. Teel, and D. Nęsić, "ALyapunov proof of improved maximum allowable transfer interval for networked control systems," *IEEE Trans. Autom. Control*, vol. 52, no. 5, pp. 892–897, May 2007.
- [86] A. Chaillet and A. Bicchi, "Delay compensation in packet-switching networked controlled systems," in *Proc. IEEE Conf. Decision Control*, 2008, pp. 3620–3625.
- [87] M. Cloosterman, L. Hetel, N. van de Wouw, W. Heemels, J. Daafouz, and H. Nijmeijer, "Controller Synthesis for Networked Control Systems," *Automatica*, vol. 46, no. 10, Oct. 2010.
- [88] M. Cloosterman, N. van de Wouw, W. Heemels, and H. Nijmeijer, "Stabilization of networked control systems with large delays and packet dropouts," in *Proc. Amer. Control Conf.*, 2008, pp. 4991–4996.
- [89] M. Cloosterman, N. van de Wouw, W. Heemels, and H. Nijmeijer, "Stability of networked control systems with uncertain time-varying delays," *IEEE Trans. Autom. Control*, vol. 54, no. 7, pp. 1575–1580, Jul. 2009.
- [90] D. B. Dăcić and D. Nęsić, "Quadratic stabilization of linear networked control systems via simultaneous protocol and controller design," *Automatica*, vol. 43, pp. 1145–1155, 2007.
- [91] D. F. Delchamps, "Stabilizing a linear system with quantized state feedback," *IEEE Trans. Autom. Control*, vol. 35, no. 8, pp. 916–924, Aug. 1990.
- [92] M. C. F. Donkers, L. Hetel, W. P. M. H. Heemels, N. van de Wouw, and M. Steinbuch, "Stability analysis of networked control systems using a switched linear systems approach," in *Hybrid Systems: Computation and Control, Lecture Notes in Computer Science*. New York: Springer Verlag, 2009, pp. 150–164.
- [93] H. Fujioka, "Stability analysis for a class of networked/embedded control systems: A discrete-time approach," in *Proc. Amer. Control Conf.*, 2008, pp. 4997–5002.
- [94] H. Gao, T. Chen, and J. Lam, "A new delay system approach to network-based control," *Automatica*, vol. 44, no. 1, pp. 39–52, 2008.
- [95] R. H. Gielen, S. Oлару, M. Lazar, W. P. M. H. Heemels, N. van de Wouw, and S.-I. Niculescu, "On polytopic inclusions as a modeling framework for systems with time-varying delays," *Automatica*, vol. 46, no. 3, pp. 615–619, Mar. 2010.
- [96] V. Gupta, T. H. Chung, B. Hassibi, and R. M. Murray, "On a stochastic sensor selection algorithm with applications in sensor scheduling and sensor coverage," *Automatica*, vol. 42, no. 2, pp. 251–260, 2006.
- [97] W. P. M. H. Heemels, R. J. A. Gorter, A. van Zijl, P. P. J. v. d. Bosch, S. Weiland, W. H. A. Hendrix, and M. R. Vonder, "Asynchronous measurement and control: A case study on motor synchronisation," *Control Eng. Prac.*, vol. 7, no. 12, pp. 1467–1482, 1999.
- [98] W. P. M. H. Heemels, D. Nęsić, A. R. Teel, and N. van de Wouw, "Networked and quantized control systems with communication delays," in *Proc. Joint IEEE Conf. Decision Control (CDC) 28th Chinese Control Conf.*, 2009, pp. 7929–7935.
- [99] W. P. M. H. Heemels, H. Siahhan, A. Juloski, and S. Weiland, "Control of quantized linear systems: An \square -optimal control approach," in *Proc. Amer. Control Conf.*, Denver, CO, 2003, pp. 3502–3507.
- [100] L. Hetel, J. Daafouz, and C. Lung, "Stabilization of arbitrary switched linear systems with unknown time-varying delays," *IEEE Trans. Autom. Control*, vol. 51, no. 10, pp. 1668–1674, Oct. 2006.
- [101] L. Hetel, J. Daafouz, and C. Lung, "Analysis and control of LTI and switched systems in digital loops via an event-based modeling," *Int. J. Control*, vol. 81, no. 7, pp. 1125–1138, 2008.

- [102] D. Hristu and K. Morgansen, "Limited communication control," *Syst. Control Lett.*, vol. 37, no. 4, pp. 193–205, Jul. 1999.
- [103] C.-Y. Kao and B. Lincoln, "Simple stability criteria for systems with time-varying delays," *Automatica*, vol. 40, pp. 1429–1434, 2004.
- [104] D. Liberzon, "On stabilization of linear systems with limited information," *IEEE Trans. Autom. Control*, vol. 48, no. 2, pp. 304–307, Feb. 2003.
- [105] D. Liberzon, "Quantization, time delays, and nonlinear stabilization," *IEEE Trans. Autom. Control*, vol. 51, no. 7, pp. 1190–1195, Jul. 2006.
- [106] A. S. Matveev and A. V. Savkin, "The problem of state estimation via asynchronous communication channels with irregular transmission times," *IEEE Trans. Autom. Control*, vol. 48, no. 4, pp. 670–676, Apr. 2003.
- [107] L. A. Montestruque and P. Antsaklis, "Stability of model-based networked control systems with time-varying transmission times," *IEEE Trans. Autom. Control*, vol. 49, no. 9, pp. 1562–1572, Sep. 2004.
- [108] P. Naghshtabrizi and J. P. Hespanha, "Designing an observer-based controller for a network control system," in *Proc. Conf. Decision Control and Eur. Control Conf.*, Seville, Spain, 2005, pp. 848–853.
- [109] P. Naghshtabrizi and J. P. Hespanha, "Stability of network control systems with variable sampling and delays," in *Proc. Annu. Allerton Conf. Commun., Control, Comp.*, 2006, [CD ROM].
- [110] P. Naghshtabrizi, J. P. Hespanha, and A. R. Teel, "Stability of delay impulsive systems with application to networked control systems," in *Proc. Amer. Control Conf.*, New York, 2007, pp. 4899–4904.
- [111] G. N. Nair and R. J. Evans, "Stabilizability of stochastic linear systems with finite feedback data rates," *SIAM J. Control Optim.*, vol. 43, pp. 413–436, 2004.
- [112] D. Nęsić and D. Liberzon, "A unified framework for design and analysis of networked and quantized control systems," *IEEE Trans. Autom. Control*, vol. 54, no. 4, pp. 732–747, Apr. 2009.
- [113] D. Nęsić and A. R. Teel, "Input-output stability properties of networked control systems," *IEEE Trans. Autom. Control*, vol. 49, no. 10, pp. 1650–1667, Oct. 2004.
- [114] D. Nęsić and A. R. Teel, "Input-to-state stability of networked control systems," *Automatica*, vol. 40, pp. 2121–2128, 2004.
- [115] H. Rehlinger and M. Sanfridson, "Scheduling of a limited communication channel for optimal control," *Automatica*, vol. 40, no. 3, pp. 491–500, 2004.
- [116] B. Sinopoli, L. Schenato, M. Franceschetti, K. Poolla, M. I. Jordan, and S. S. Sastry, "Kalman filtering with intermittent observations," *IEEE Trans. Autom. Control*, vol. 49, no. 9, pp. 1453–1464, Sep. 2004.
- [117] S. C. Smith and P. Seiler, "Estimation with lossy measurements: Jump estimators for jump systems," *IEEE Trans. Autom. Control*, vol. 48, no. 12, pp. 2163–2171, Dec. 2003.
- [118] S. Tatikonda and S. K. Mitter, "Control under communication constraints," *IEEE Trans. Autom. Control*, vol. 49, no. 7, pp. 1056–1068, Jul. 2004.
- [119] Y. Tipsuwan M. Chow, "Gain scheduler middleware: a methodology to enable existing controllers for networked control and teleoperation-part II: teleoperation" *IEEE Transaction on Industrial Electronics*, Vol. 51, pp. 1228 – 1237, 2004.
- [120] N. van de Wouw, P. Naghshtabrizi, M. Cloosterman, and J. P. Hespanha, "Tracking control for sampled-data systems with uncertain time-varying sampling intervals and delays," *Int. J. Robot. Nonlin. Control*, vol. 20, no. 4, pp. 387–411, 2010.
- [121] G. C. Walsh, O. Belidman, and L. G. Bushnell, "Asymptotic behavior of nonlinear networked control systems," *IEEE Trans. Autom. Control*, vol. 46, no. 7, pp. 1093–1097, Jul. 2001.
- [122] G. C. Walsh, O. Belidman, and L. G. Bushnell, "Stability analysis of networked control systems," *IEEE Trans. Control Syst. Technol.*, vol. 10, no. 3, pp. 438–446, May 2002.
- [123] L. Zhang, Y. Shi, T. Chen, and B. Huang, "A new method for stabilization of networked control systems with random delays," *IEEE Trans. Autom. Control*, vol. 50, no. 8, pp. 1177–1181, Aug. 2005.
- [124] H.J. Gao, T.W. Chen, H1 estimation for uncertain systems with limited communication capacity, *IEEE Trans. Automat. Contr.*, vol. 52, no. 11, pp. 2070 – 2080, 2007.
- [125] Z.P. Jin, V. Gupta, R.M. Murray, State estimation over packet dropping networks using multiple description coding, *Automatica*, vol. 42, no. 9, pp. 1441–1452, 2006.
- [126] M. Sahebsara, T.W. Chen, S.L. Shah, "Optimal H_∞ filtering in networked control systems with multiple packet dropouts", *Syst. Contr. Lett.*, vol. 57, no. 9, pp. 696 – 702, 2008.
- [127] S. Smith, P. Seiler, "Estimation with lossy measurements: jump estimators for jump systems," *IEEE Trans. Automat. Contr.*, vol. 48, no. 12, pp. 2163–2171, 2003.
- [128] B. Sinopoli, L. Schenato, M. Franceschetti, K. Poolla, M.I. Jordan, S.S. Sastry, "Kalman filtering with intermittent observations," *IEEE Trans. Automat. Contr.*, vol. 49, no. 9, pp. 1453 – 1464, 2004.
- [129] R. Vanijirattikhan, M. Chow y Y. Tipsuwan, "Feedback preprocessor for Unmanned Ground Vehicle Network-Based Controller", *Proc. Of the 30th Annual Coference of the IEEE Industrial electronics Society*, 2004.
- [130] Z.D. Wang, D.W.C. Ho, X.H. Liu, "Variance-constrained filtering for uncertain stochastic systems with missing measurements," *IEEE Trans. Automat. Contr.*, vol. 48, no. 7, pp. 1254–1258, 2003.
- [131] D. Yue, Q.L. Han, "Network-based robust H_∞ filtering for uncertain linear systems," *IEEE Trans. Signal Process.*, vol. 54, no. 11, pp. 4293 – 4301, 2006.
- [132] S.S. Zhou, G. Feng, " H_∞ filtering for discrete-time systems with randomly varying sensor delays," *Automatica*, vol. 44, no. 7, pp. 1918 – 1922, 2008.
- [133] Y. Zhang y N. Duffield, "On the constancy of internet path properties" *Proc. of the 1st ACM SIGCOMM Workshop on Internet Measurement*, pp. 197 – 211, 2001
- [134] P. Martí and M. Velasco, "Toward Flexible Scheduling of Real-Time Control Tasks: Reviewing Basic Control Models," *10th International Conference on Hybrid Systems, Computation and Control*, LNCS, 2007.
- [135] C. Lozoya, P.Martí, M. Velasco, and J.M. Fuertes. "Control Performance Evaluation of Selected Methods of Feedback Scheduling of Real-time Control Tasks," in 17th IFAC World Congress, July, 2008.