

# Experiments for Decentralized and Networked Dynamic Resource Allocation, Scheduling, and Control

Nicanor Quijano, Alvaro E. Gil, and Kevin M. Passino

The Ohio State University,  
Department of Electrical Engineering,  
2015 Neil Avenue, Columbus, Ohio 43210



# Agenda

- ✓ Introducción
- ✓ Balls-in-Tubes
- ✓ Electromechanical Arcade
- ✓ Multizone Temperature Control
- ✓ Conclusiones

# Introducción

- ★ En la actualidad existen en la industria numerosos sistemas de control distribuido (DCS).
- ★ Dichos DCSs pueden utilizar sistemas como *network embedded computing* para el control de sus plantas.
- ★ Los avances más significativos en esta última área se deben a:
  - Microcontroladores de alto rendimiento y bajo costo.
  - La interconexión de dichos microcontroladores por intermedio de redes inalámbricas u otros protocolos (e.g., *controller area network* CAN).

- ★ Las universidades tienen el compromiso de ajustar sus currículos a las necesidades de la industria.
- ★ Existen tres importantes áreas de investigación que hacen frente a dicho compromiso:
  1. *Cooperative robotics.*
  2. *Wireless Sensor Networks* utilizando *motes*.
  3. *Distributed networked feedback control based.*
- ★ Se ha implementado un laboratorio cuyo énfasis es:
  - Experimentos de bajo costo que sean lo suficientemente interesantes tanto para educación como para investigación.
  - Problemas completamente diferentes a los que

estudian las otras áreas.

- Nuevos retos en *information technology-enabled feedback control*.

## Balls-in-Tubes

El objetivo de este experimento es estudiar el comportamiento de las estrategias de asignación de recursos (i.e., *resource allocation*) por intermedio de dos estrategias de control.

## Descripción Física

- La Figura 1 muestra el experimento.
- Cada tubo contiene un ventilador de corriente directa (dc), una pelota, y un sensor ultrasónico.
- Los *manifolds* fuerzan la distribución de aire a la entrada, o restringen el flujo de aire a la salida.

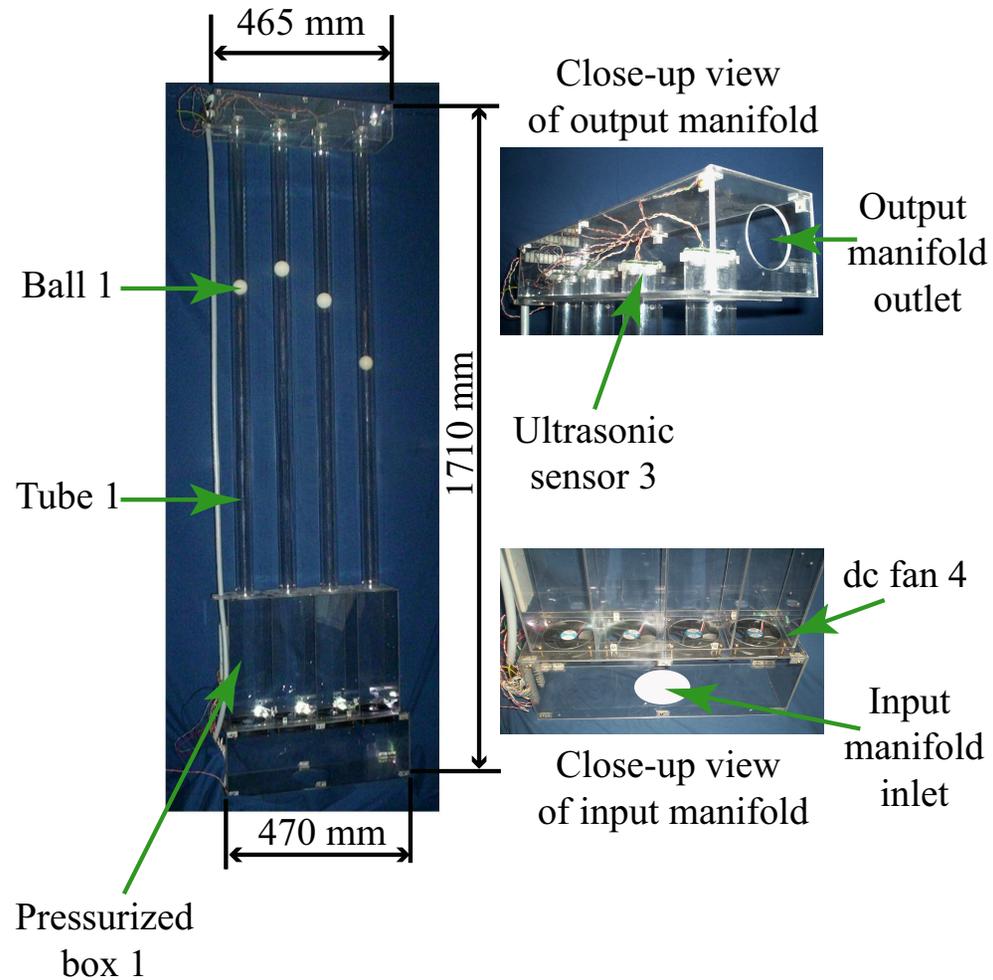


Figure 1: Balls-in-tubes. Los tubos están numerados de izquierda a derecha.

## Descripción Física

- Se tienen en total ocho salidas digitales y cuatro entradas digitales conectadas a la tarjeta de dSPACE.
- El sensor tiene ciertas restricciones:
  - Cambios de la velocidad del sonido con respecto a la temperatura ambiente.
  - Ruido del sensor.
  - La resolución del sensor (esta depende del tiempo de muestreo que se haya seleccionado. Para nuestro caso, el tiempo de muestreo nos da una resolución de  $\pm 1$  centímetro).

## Descripción Física, continuación

- Hay dos tipos de perturbaciones externas:
  - Las entradas (salidas) de aire de los *manifolds* localizados en la parte inferior (superior) del experimento se pueden limitar por intermedio de objetos externos.
  - En el *manifold* de la parte inferior, se pueden colocar objetos que limiten la cantidad de aire que viene siendo común a todos los tubos.

# Descripción del experimento

1. **Balancear** las pelotas que se encuentran dentro de los tubos a una cierta altura, tratando de asignar la presión del aire de forma óptima para mantener las pelotas a una máxima altura promedio.
2. **Re-asignar** la presión del aire cuando alguna perturbación es introducida.
3. Tratar de **alcanzar un valor deseado**, el cual es fijado por la pelota que se encuentra a la mayor altura utilizando recursos limitados, y en presencia de perturbaciones.

# ESTRATEGIAS DE ASIGNACIÓN DE RECURSOS Y RESULTADOS

# Estrategia de Asignación de Recursos: *Juggler*

El objetivo de esta estrategia es mantener las pelotas en la vecindad de una altura común, de acuerdo con el aire que se tiene disponible en el *manifold* de entrada.

## *Juggler, continuación*

- ★ Existen cuatro características importantes en esta estrategia:
  1. El tiempo que toma elevar una pelota una cierta altura es mayor que el tiempo que le toma a ella caer de esa misma altura.
  2. El flujo de aire de los tubos 2 y 3 es menos restringido que el de los tubos 1 y 4. Por ende, los tiempos de ascenso y descenso de las pelotas en 2 y 3 es menor que en los otros dos.
  3. Aparentemente debido a diferencias en la construcción de los módulos, el tiempo que toma

“levantar” la pelota en el cuarto tubo es mayor que en el primero.

4. El porcentaje mínimo del ciclo de trabajo del PWM destinado a elevar las pelotas es alrededor de 60 %.
- ★ Definamos a  $D_u^i(t)$  y  $D_d^i(t)$  como los ciclos de trabajo necesarios para que la pelota  $i$  suba o baje en un determinado tiempo  $t$ .

## *Juggler, continuación*

- ★ La estrategia puede verse de la siguiente manera:
  1. Definamos  $i^*$  como la pelota que tiene la menor altura en el tiempo  $t$ , y  $h_i^*(t)$  su respectiva altura.
  2. Para todo tiempo  $t' \in [t, t + T]$ , se le asigna  $D_u^{i^*}(t')$  al ventilador de la pelota  $i^*$ , y para  $j \neq i^*$  se asigna  $D_d^j(t')$  pero su entrada se modifica de acuerdo a:
    - *Truncation rule 1*: Si  $h_i^*(t') > 0.6$  metros en un tiempo  $t'$ ,  $t' \in [t, t + T]$ , se aborta y retorna al primer paso. La idea es evitar empujar la pelota muy alto.
    - *Truncation rule 2*: Si  $h_i(t') > 0.6$  metros para

todo  $i$  en un tiempo  $t'$ ,  $t' \in [t, t + T]$ , entonces se aborta el algoritmo y se asigna  $D_d^i(t) = 60\%$  a todos los ventiladores durante 0.6 segundos. Esto evita que las pelotas se queden pegadas en el tope.

- ★ La Figura 2 muestra los resultados de este experimento cuando  $T = 3$  segundos.

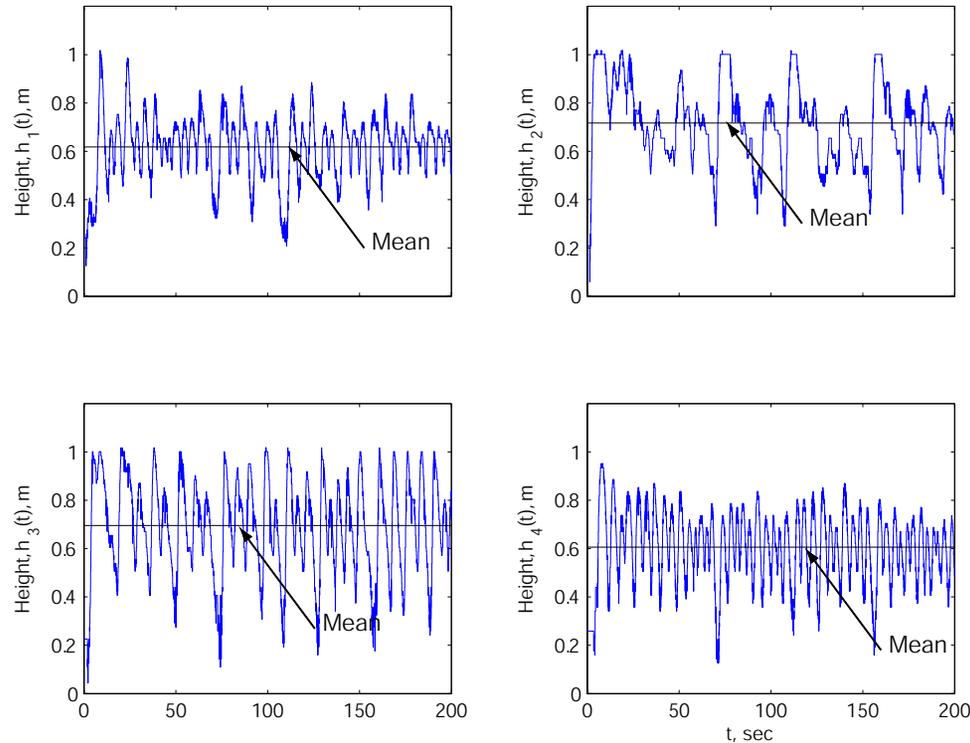


Figure 2: Altura de las pelotas en la estrategia *juggler*. El promedio de las alturas es cercano, y tiene una desviación estándar para las alturas de 0.14, 0.17, 0.20, and 0.15 m para los tubos 1 al 4, respectivamente.

- ★ Se coloca una pequeña perturbación dentro del *manifold* inferior debajo del ventilador del cuarto tubo.
- ★ La Tabla 1 muestra cómo se distribuyó el porcentaje de tiempo en que cada uno de los tubos fue “seleccionado.”

	Porcentaje de tiempo que cada tubo es "atendido"	Promedio de altura(m)
Tubo # 1	23.22	0.6182
Tubo # 2	10.95	0.7174
Tubo # 3	23.69	0.6957
Tubo # 4	43.13	0.6063
Tubo # 1	24.41	0.5863
Tubo # 2	6.98	0.6441
Tubo # 3	13.61	0.7085
Tubo # 4	54.99	0.5057

Table 1: Resultados con y sin perturbaciones.

# Estrategia de Asignación de Recursos: *Dynamic Proportioning*

Esta estrategia consiste en tener un controlador proporcional interno para cada tubo, y por intermedio de un controlador externo que selecciona cada 10 segundos la altura deseada  $h_d(t)$ , se llevan las pelotas a dicho punto.

## *Dynamic Proportioning, continuación*

- En primera instancia se diseñó un controlador proporcional interno para cada tubo, el cual tiene una ganancia no lineal de acuerdo al error:

Error de la pelota

$i$  en el tiempo  $t$

$$\underbrace{e_i(t)}$$

=

$$\underbrace{h_d(t)}_{\text{Altura deseada}}$$

–

Altura de la pelota

$i$  en el tiempo  $t$

$$\underbrace{h_i(t)}$$

→ Cada diez segundos se selecciona  $h_d(t)$  de acuerdo a:

$$h_d(t) = \max\{h_i(t) : i = 1, 2, 3, 4\}$$

→ La diferencia entre esta estrategia y la anterior es que en *juggler* se asignan pulsos de aire con respecto al tiempo, mientras que acá la asignación de los recursos es tanto espacial como temporal, ya que  $h_d(t)$  cambia dinámicamente.

→ Las Figuras 3 y 4 muestran los resultados obtenidos para dos tipos diferentes de perturbaciones.

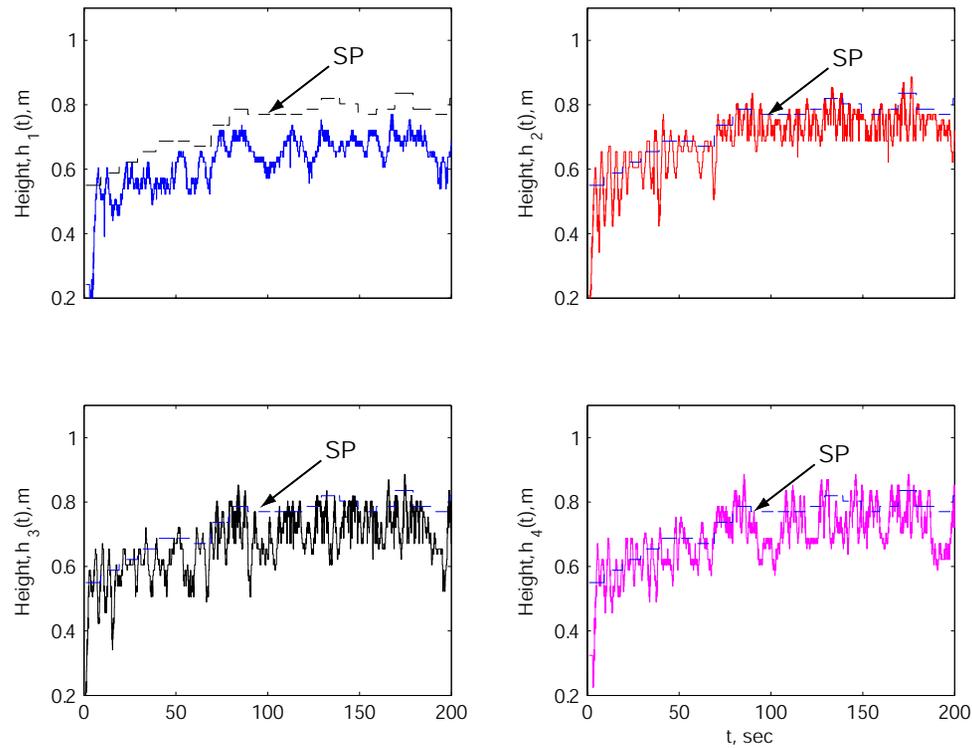


Figure 3: Resultados de la estrategia *Dynamic proportioning*.

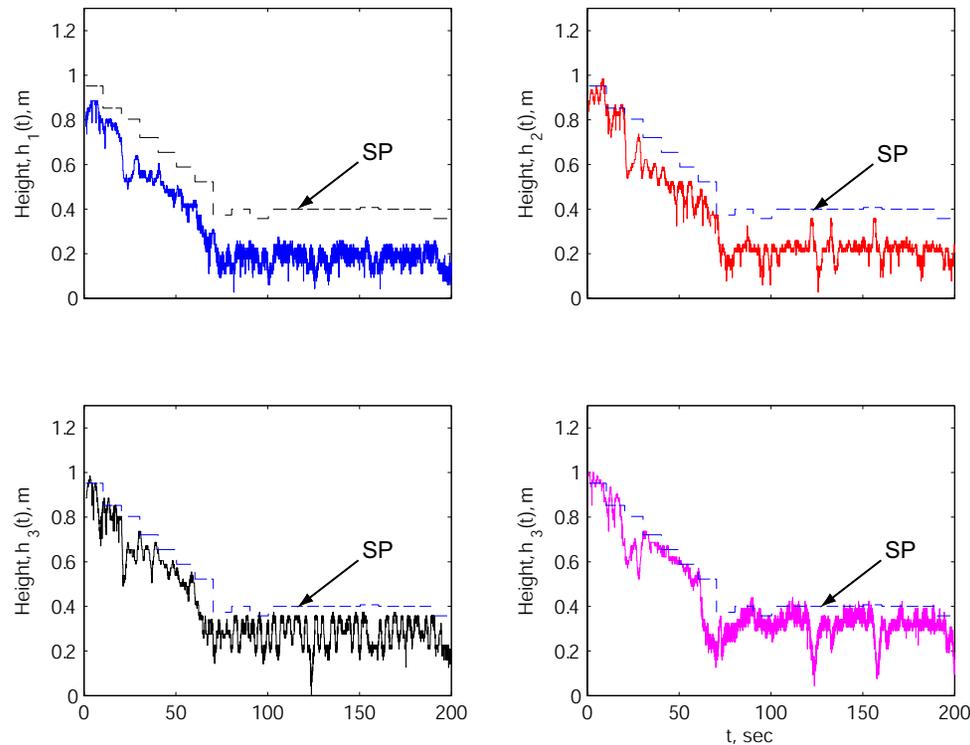


Figure 4: Resultados de la estrategia *Dynamic proportioning*. En este caso se introduce un orificio más pequeño en el tiempo  $t = 20$  seg, lo que demuestra la asignación dinámica de los recursos.

# ELECTROMECHANICAL ARCADE

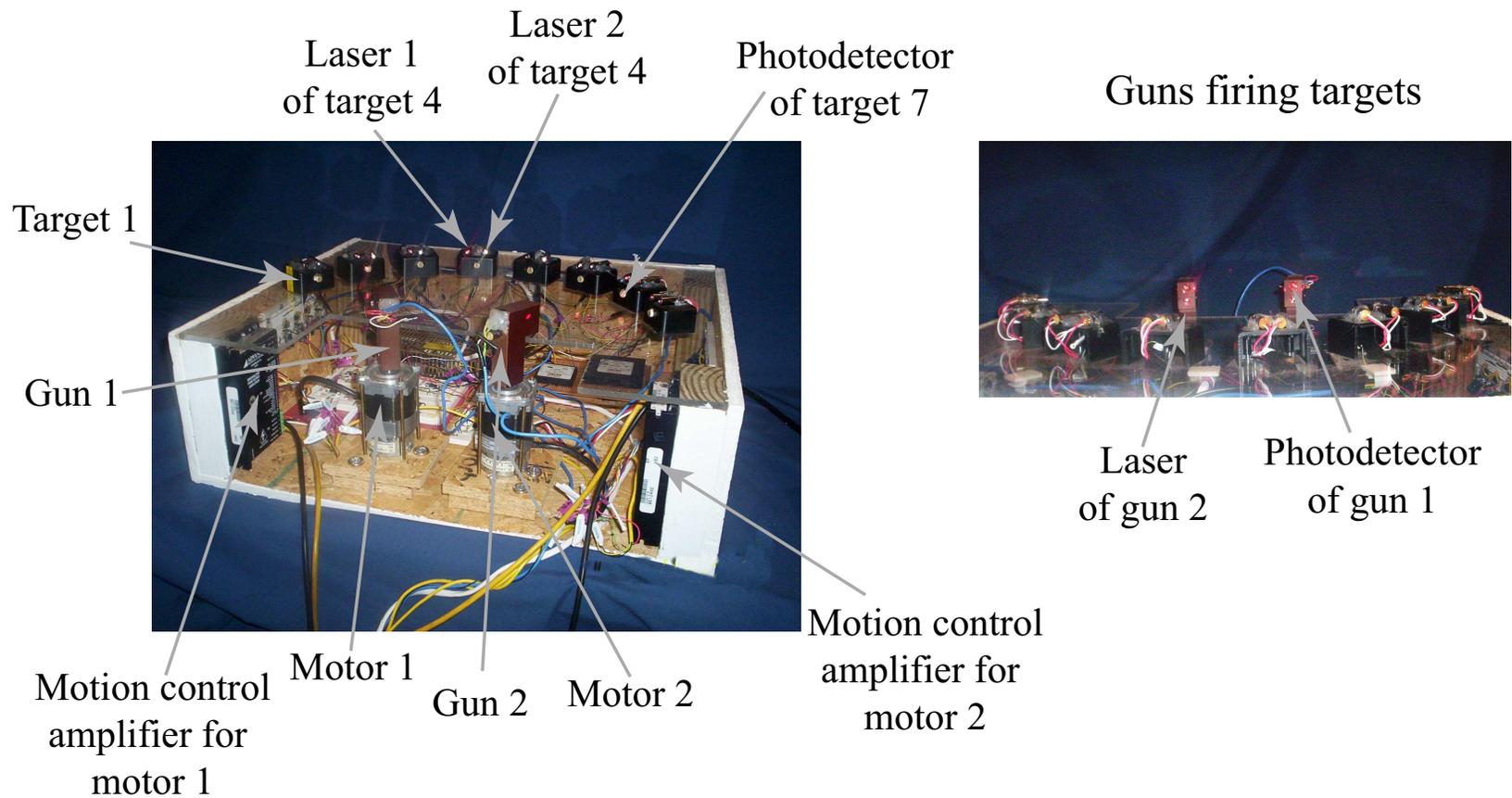


Figure 5: Electromechanical arcade. El experimento tiene dos “pistolas” y ocho “blancos.”

## Descripción del experimento

- ★ Se asume que las pistolas no tienen información acerca de la frecuencia a la cual aparecen los blancos (sin embargo, se pueden inventar estrategias para estimar dicha secuencia).
- ★ Las pistolas saben la posición de los blancos, y ellas pueden comunicarse las decisiones de procesar o seguir un determinado blanco.

## Descripción del experimento, cont'd

- ★ Los objetivos son los siguientes:
  1. Programar (*schedule*) una serie de disparos en tiempo real para maximizar el número de puntos que se reciben por el grupo que constituyen las dos pistolas.
  2. Programar de forma cooperativa el disparo de las pistolas en presencia de una comunicación imperfecta entre ellas.

# Estrategias de Programación y Resultados

- El conjunto de blancos se denota  $P = \{1, 2, \dots, 8\}$ .
- El conjunto de pistolas se denota  $Q = \{1, 2\}$ .
- $p_i, i \in P$ , es la prioridad de procesar el blanco  $i$ . Procesar significa mover a y disparar al blanco cuando este aparece.
- Sea  $T_i(t) = p_i t_i$ , donde  $i \in P, t \geq 0$ , y  $t_i$  la cantidad de tiempo que pasó desde que se disparó al blanco  $i$ .  $T_i(t)$  es lo que se denomina como el *prioritized time*. Inicialmente  $T_i(0) = 0$  para todo  $i \in P$ .

- Existen tres tipos de retardos:
1. Retardo entre apariciones de cada blanco.
  2. “Tiempo de viaje” para mover la pistola de un blanco a otro.
  3. Retardo de comunicación entre las pistolas. Este retardo, simulado, es aleatorio pero limitado.
- Para definir el controlador, primero definamos lo siguiente:
- ★  $U(t) \subset P$  es el conjunto de blancos desatendidos.
  - ★  $U_j^a(t) = \{i_j^*(t)\} \in U$  es el conjunto de blancos que la pistola  $j$  puede considerar después de haber procesado el blanco  $i_j^*(t)$ .

- El conjunto  $U(t)$  se distribuye por la red con ciertos retardos que hacen que las pistolas sigan disparando al blanco actual (eso si, no se dan puntos por dicho efecto).

# Process-the-Target-Ignored-the-Longest Strategy

- Definamos  $D_j$  como el tiempo de decisión para la pistola  $j$ .
- Una estrategia de programación es seleccionar el blanco si este ha sido ignorado durante más tiempo que el promedio de todos los demás tiempos, es decir

$$T_{i_j^*}(D_j) \geq \frac{1}{N - M + 1} \sum_{i_j \in U_j^a(D_j)} T_{i_j}(D_j).$$

- Donde  $N = 8$  (número de blancos), y  $M = 2$  (número de pistolas). Los resultados de dicha estrategia se pueden observar en la Figura 6.

# Process-the-Target-Ignored-the-Longest Strategy

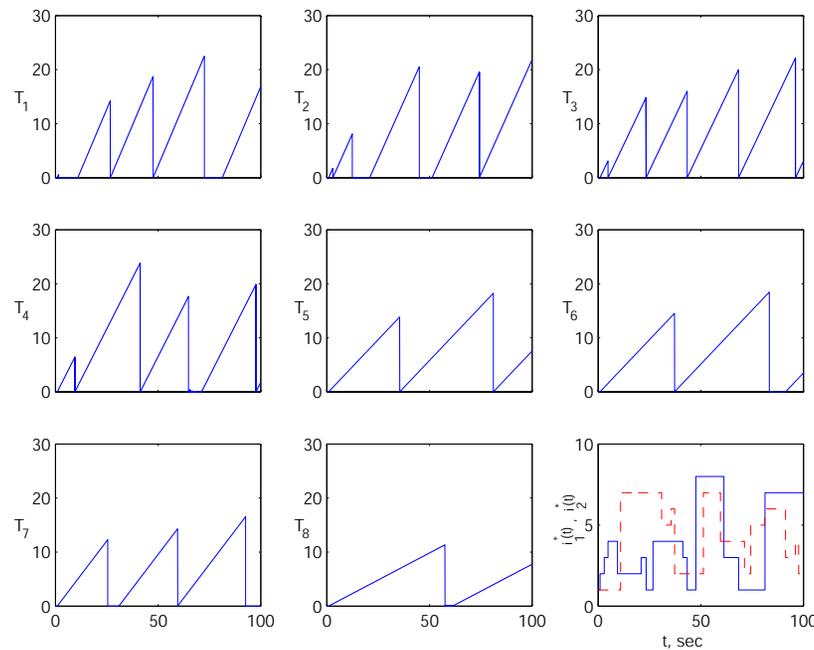


Figure 6: En el tiempo  $t = 0$ , ambas pistolas estan apuntando al blanco 1. Existe un retardo de comunicacion fijo de 10 segundos. Los valores de  $T_i$  que fueron graficados contra el tiempo  $t$  muestran el *prioritized time* en el cual el blanco  $i \in \{1, \dots, 8\}$  fue disparado. La grafica que se muestra en el recuadro inferior derecho corresponde a  $i_1^*(t)$  (linea solida), y a  $i_2^*(t)$  (linea punteada).

## Process-the-Closest-Highest-Priority-Targets Strategy

★ Esta estrategia programa el siguiente blanco para evitar que los blancos con alta prioridad esperen mucho tiempo, y a la vez, minimiza el “tiempo de viaje” para procesar los blancos.

★ La pistola procesa el blanco  $i_j^*(D_j)$  en el tiempo  $D_j$  si

$$T_{i_j^*}(D_j) - \delta_{ji_j^*}(D_j) \geq \frac{1}{N - M + 1} \sum_{i_j \in U_j^a(D_j)} [T_{i_j}(D_j) - \delta_{ji_j}(D_j)] \quad (1)$$

★  $\delta_{ji_j}$  es la cantidad de tiempo requerida por la pistola para moverse del blanco  $j$  al blanco  $i_j$ .

- ★ La escogencia del máximo valor del término en la parte izquierda de la Ecuación (1) resulta en procesar el blanco con la más alta prioridad, y que a su vez sea el más cercano.
- ★ La Figura 7 muestra los resultados de dicha estrategia.

# Process-the-Closest-Highest-Priority-Targets Strategy

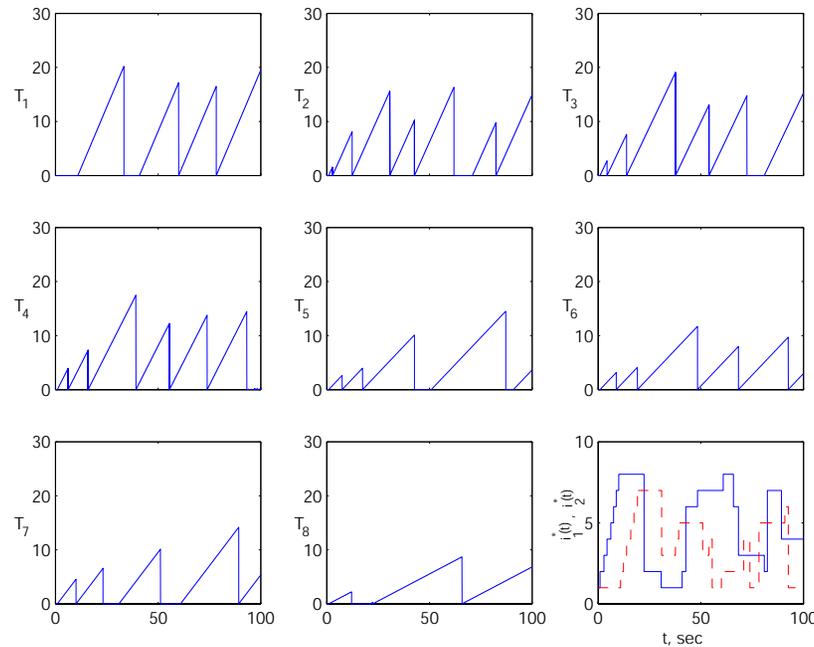


Figure 7: En el tiempo  $t = 0$ , ambas pistolas estan apuntando al blanco 1. Existe un retardo de comunicacion fijo de 10 segundos. Los valores de  $T_i$  que fueron graficados contra el tiempo  $t$  muestran el *prioritized time* en el cual el blanco  $i \in \{1, \dots, 8\}$  fue disparado. La grafica que se muestra en el recuadro inferior derecho corresponde a  $i_1^*(t)$  (linea solida), y a  $i_2^*(t)$  (linea punteada). El resultado es mejor que el de la anterior estrategia.

Table 2: Desempeño del experimento. La estrategia *process-the-closest-highest-priority-targets* es mejor para cada elemento, debido a que está buscando permanentemente el blanco más cercano.

	Ignored-the-Longest	Closest-Highest-Priority
Average of average	7.28	5.22
Maximum of average	11.15	8.69
Average of maximum	14.46	11.56
Max of max	23.87	20.21
Number of points	25	35

# MULTIZONE TEMPERATURE CONTROL

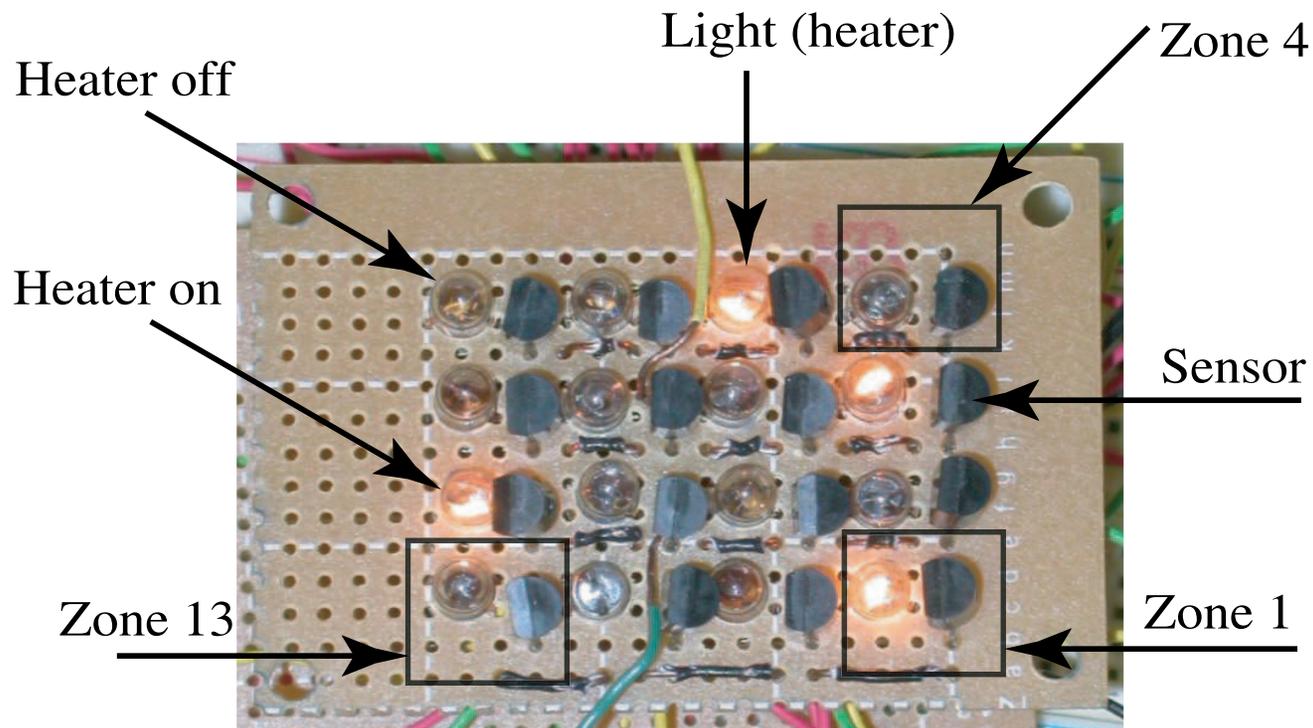


Figure 8: Planar temperature grid control. Este experimento tiene 16 zonas individualmente controladas, cada una de las cuales consiste de una lámpara y un sensor de temperatura.

## Descripción Física

- Este experimento consiste en dieciséis zonas, cada una de las cuales se compone de una lámpara y un sensor de temperatura.
- Ocho de dichos sensores y actuadores están conectados a un computador, y los otros ocho a otro. La información entre los dos computadores se envía por medio del protocolo RS232.
- Existen un sinnúmero de elementos que afectan el experimento (e.g., temperatura ambiente, calibración de los sensores, influencia de otras zonas, etc.).

# Descripción del experimento

Los principales retos de este experimento son:

1. *Temperature tracking* en la cual un conjunto de zonas tiene como referencia la temperatura de otra zona.
2. **Regulación** de la temperatura en toda la cuadrícula, pero con un valor máximo final.
3. **Control distribuido** con diferentes controladores por zona, de acuerdo a una topología determinada que puede simular redes de comunicación, la cual puede tener retardos aleatorios pero limitados.

## Superzone Temperature Tracking

- ★ Este experimento se denomina “*superzone*” *temperature tracking* porque las 16 zonas se agrupan en 4 “superzonas.” La idea en este caso es que cada una de las superzonas siga una temperatura deseada  $T_i^{ds}$  ( $i = 1, 2, 3, 4$ ) ajustada por el usuario.
- ★ Una modificación a esta estrategia es que las superzonas 1 y 2 fijen la temperatura que las superzonas 3 y 4 desean. El usuario fija la temperatura deseada para 1 y 2.

## Estrategias de Asignación de Recursos

- ★ El objetivo es alcanzar la máxima temperatura de manera uniforme en toda la cuadrícula, pero con la limitante de que sólo puede tener una lámpara prendida a la vez.
- ★ Sea  $T_i$  la temperatura de la zona  $i$ ,  $i = 1, 2, \dots, 16$ , y  $u_i$  la entrada del controlador de dicha zona. La idea es aplicar cada 100 ms un pulso de la misma duración a la zona que tenga la temperatura más baja. Es decir, a cada instante de tiempo  $k$ , y para

cada zona  $i$  se tiene que

$$u_i(k) = \begin{cases} 1, & \text{if } T_i(k) \leq T_j(k) \text{ for all } j \in \{1, 2, \dots, 16\}, \\ 0, & \text{otherwise.} \end{cases}$$

- ★ Los resultados de este experimento se presentan en la Figura 9.

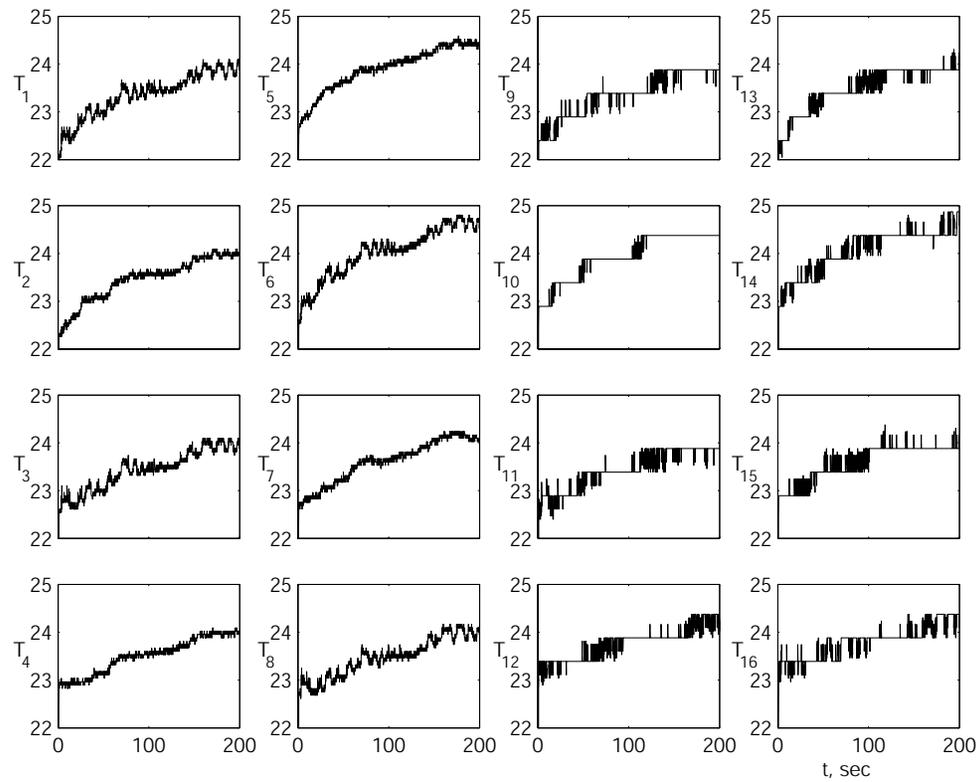


Figure 9: Valores de temperatura para el experimento de asignación de recursos de forma centralizada. La temperatura promedio final es de 24.1 grados Centígrados. La temperatura ambiente era de 22.8 grados Centígrados al comenzar el experimento.

- ★ Otra estrategia que se puede desarrollar, es una versión distribuida de la estrategia anterior. Para ello se tiene un controlador por cada zona, la cual sensa a sus vecinos, de acuerdo a una topología determinada.
- ★ Sea  $N(i)$  el conjunto de los vecinos que la zona  $i$  puede sensar para tomar diversas decisiones.
- ★ Para todo  $i$  definamos pues

$$T_i^{min}(k) = \min \{T_j(k) : j \in N(i)\}$$

y

$$u_i(k) = \begin{cases} 1, & T_i(k) = T_i^{min}(k), \\ 0, & \text{otherwise.} \end{cases}$$

- ★ En este caso más de una lámpara puede estar prendida a la vez, y por ende más de una zona puede incrementar su temperatura como se puede ver en la Figura 10.

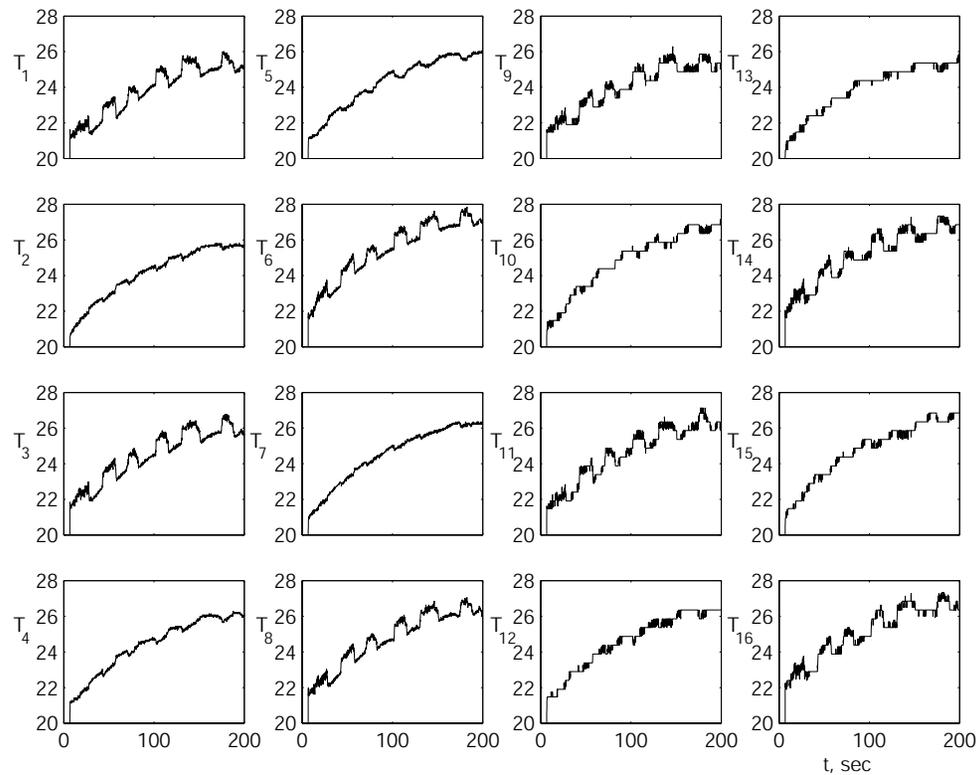


Figure 10: Valores de temperatura para el experimento de asignación de recursos de forma descentralizada de acuerdo a una cierta topología. Las temperaturas finales oscilan entre 26 y 27 grados Centígrados. La temperatura ambiente era de 21.8 grados Centígrados al comenzar el experimento.

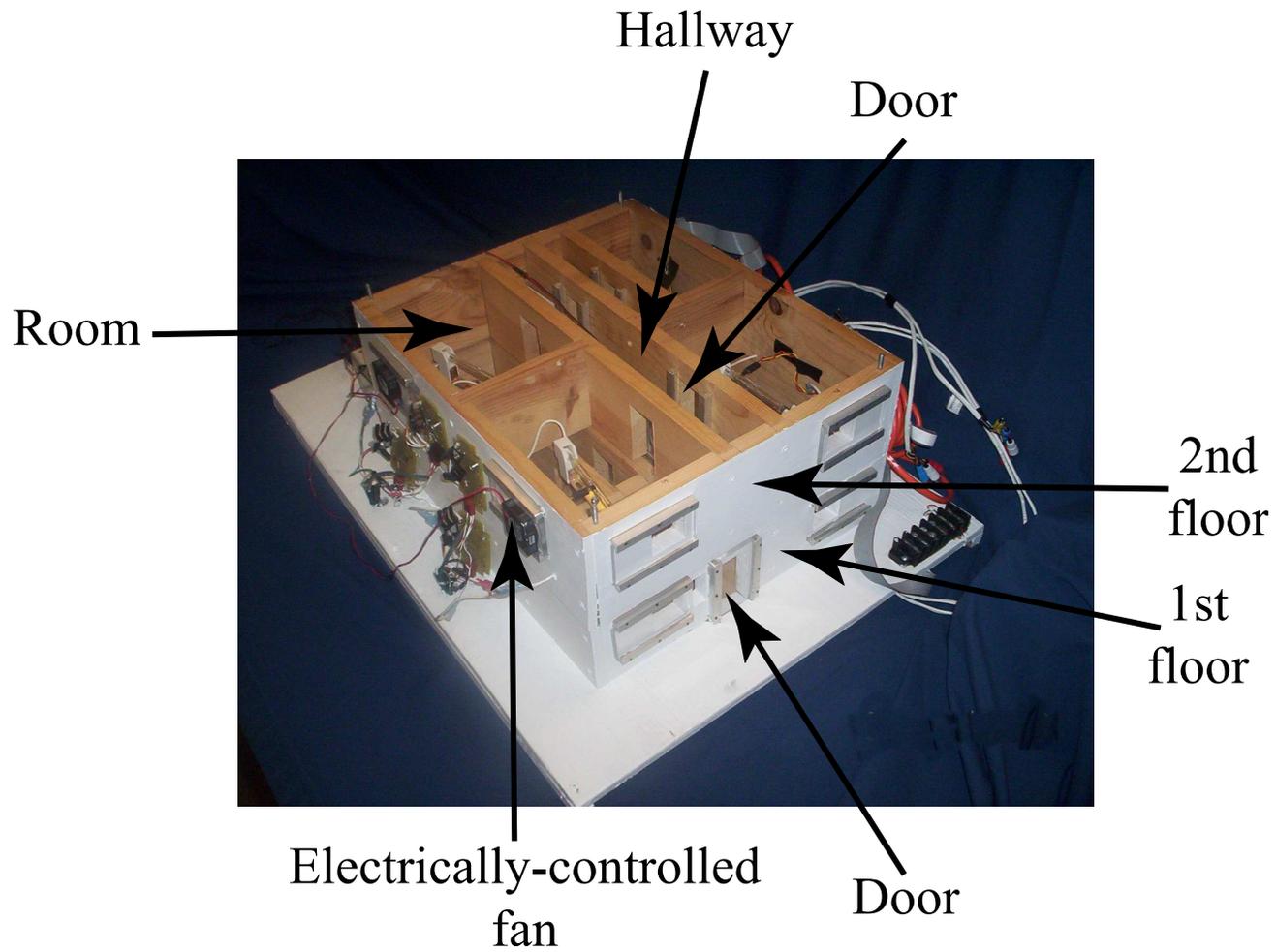


Figure 11: Building temperature control.

# Conclusiones

- Se presentaron algunos experimentos de bajo costo para el estudio e implementación de estrategias de control, programación, y asignación dinámica de recursos de forma distribuida y en red.
- Cada uno de los experimentos se puede expandir fácilmente debido a sus características modulares.
- La documentación acerca de cada uno de estos experimentos se puede encontrar en <http://www.ece.osu.edu/~passino/distdynamicsyslab.html>