

Control de alto nivel

❖ Waymo Operational Design Domain

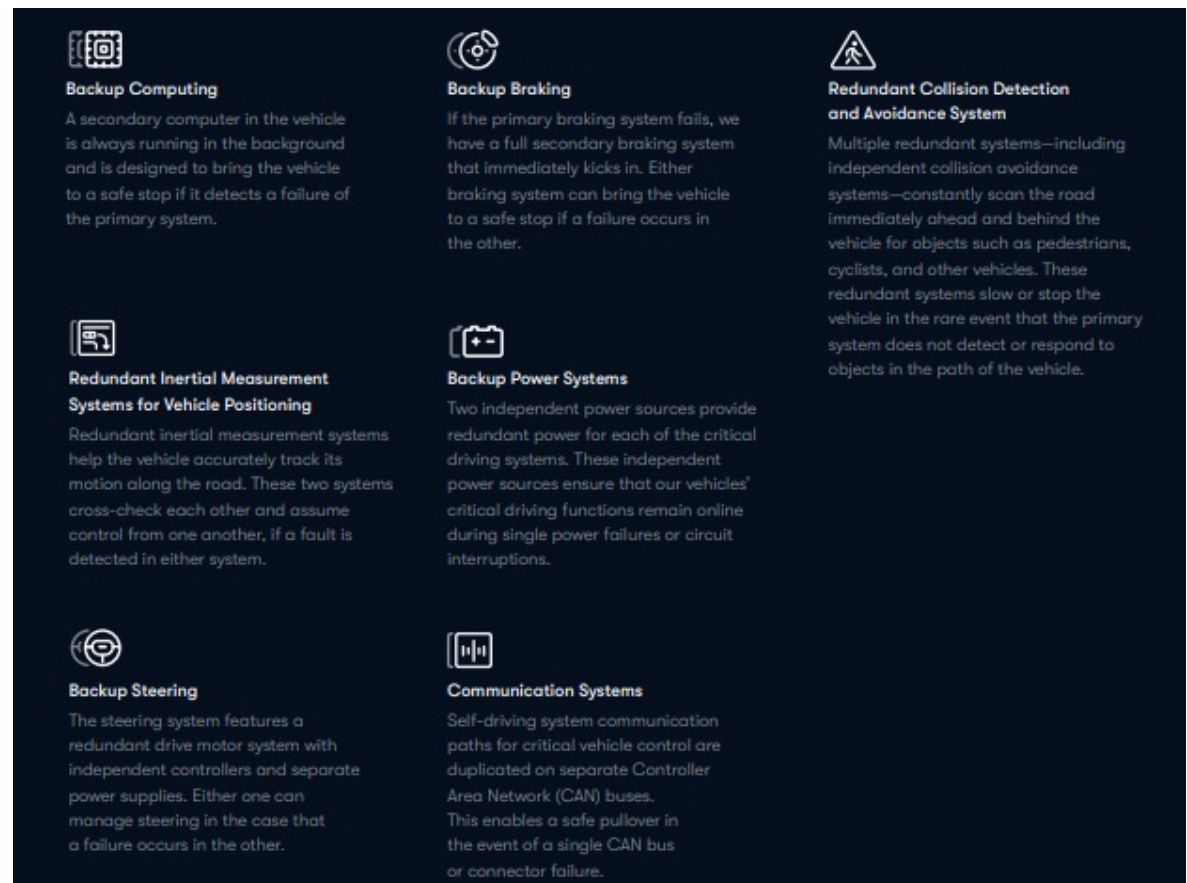
- ❖ El ODD se refiere a las condiciones en las que un sistema autónomo puede funcionar de forma segura. El dominio del diseño operativo de Waymo se define por elementos como geografías, tipos de carreteras, rango de velocidad, clima y hora del día.
- ❖ Un dominio de diseño operativo puede ser muy limitado: por ejemplo, una única ruta fija en calles públicas de baja velocidad o terrenos privados en condiciones climáticas templadas durante el día. Sin embargo, Waymo tiene como objetivo tener un amplio dominio de diseño operativo para cubrir la conducción diaria. Se está desarrollando tecnología de conducción autónoma que puede navegar por calles complejas de la ciudad en una variedad de condiciones climáticas y horas del día dentro de amplias áreas geográficas.



Control de alto nivel

Fallback: garantizar que el vehículo pueda realizar la transición a una parada segura.

Si el vehículo autónomo Waymo ya no puede continuar en un viaje planificado, debe ser capaz de realizar una parada segura, conocida como “Minimal Risk Condition” o retroceso. Esto puede incluir situaciones en las que el sistema de conducción autónoma experimenta un problema, cuando el vehículo está involucrado en una colisión o cuando las condiciones ambientales cambian de una manera que afectaría la conducción segura dentro de nuestro dominio de diseño operativo.



- Backup Computing**
A secondary computer in the vehicle is always running in the background and is designed to bring the vehicle to a safe stop if it detects a failure of the primary system.
- Backup Braking**
If the primary braking system fails, we have a full secondary braking system that immediately kicks in. Either braking system can bring the vehicle to a safe stop if a failure occurs in the other.
- Redundant Collision Detection and Avoidance System**
Multiple redundant systems—including independent collision avoidance systems—constantly scan the road immediately ahead and behind the vehicle for objects such as pedestrians, cyclists, and other vehicles. These redundant systems slow or stop the vehicle in the rare event that the primary system does not detect or respond to objects in the path of the vehicle.
- Redundant Inertial Measurement Systems for Vehicle Positioning**
Redundant inertial measurement systems help the vehicle accurately track its motion along the road. These two systems cross-check each other and assume control from one another, if a fault is detected in either system.
- Backup Power Systems**
Two independent power sources provide redundant power for each of the critical driving systems. These independent power sources ensure that our vehicles' critical driving functions remain online during single power failures or circuit interruptions.
- Backup Steering**
The steering system features a redundant drive motor system with independent controllers and separate power supplies. Either one can manage steering in the case that a failure occurs in the other.
- Communication Systems**
Self-driving system communication paths for critical vehicle control are duplicated on separate Controller Area Network (CAN) buses. This enables a safe pullover in the event of a single CAN bus or connector failure.

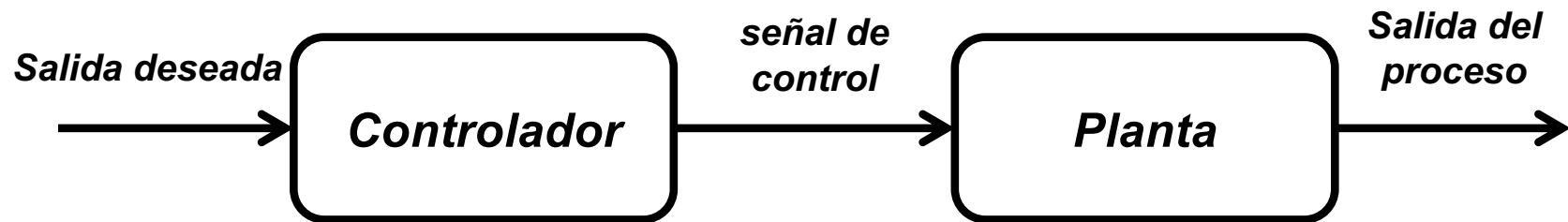
Control de bajo nivel

- ❖ Un controlador genera la salida de control para los diferentes actuadores a partir de la información procesada proveniente de los sensores.
- ❖ INPUT: Información sensorial procesada.
 - Localización del obstáculo
 - Distancia al objetivo
 - Nivel de CO2
 - Voltaje de la red eléctrica...
- ❖ OUTPUT: Órdenes para los actuadores.
 - Frenada de emergencia
 - Modificación de la inclinación de los timones
 - Aviso de incendios
 - Aumento del rendimiento de una central nuclear

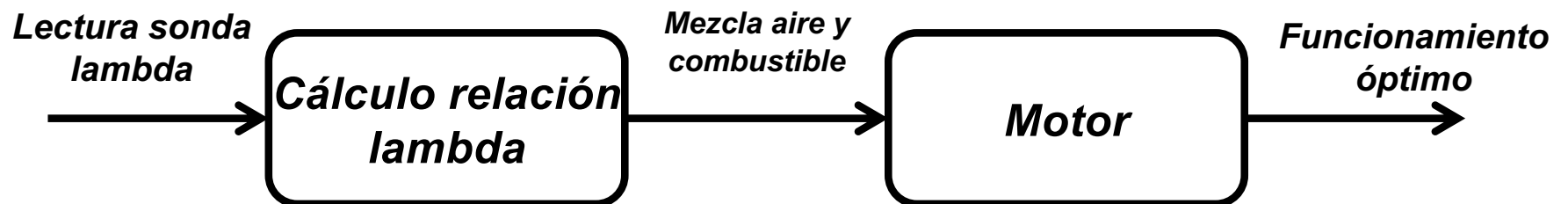
Control de bajo nivel

❖ Open Loop (Lazo de control abierto)

- En ellos la señal de salida no influye sobre la señal de entrada.

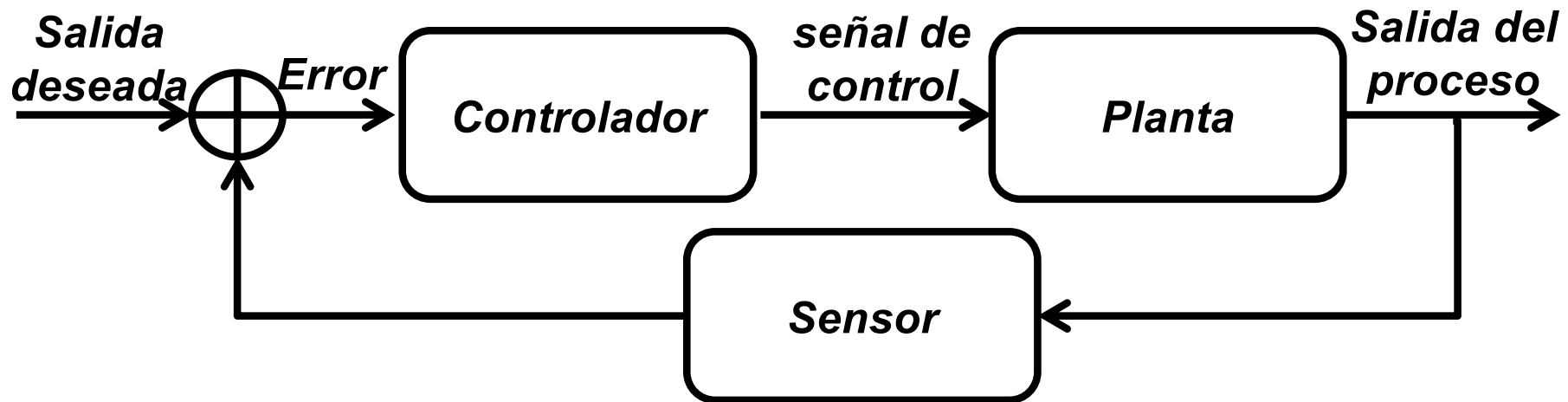


- Ejemplo: Inyección de un vehículo



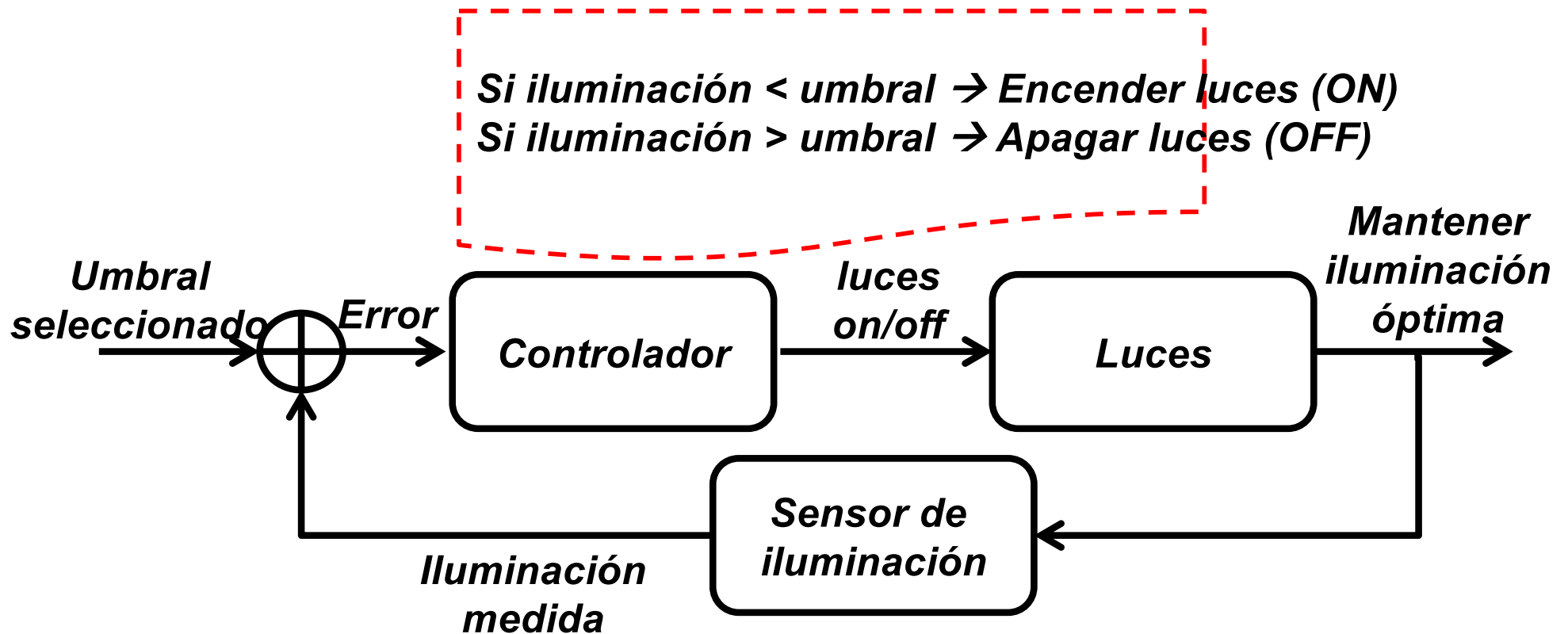
Control de bajo nivel

- ❖ Closed Loop (Lazo de control cerrado)
 - En ellos medimos cierta cantidad de la salida y luego la comparamos con un valor deseado, y el error resultante lo utilizamos para corregir la salida del sistema.



Control de bajo nivel

- ❖ Closed Loop (Lazo de control cerrado)
 - Ejemplo: Sistema automático de luces de in vehículo

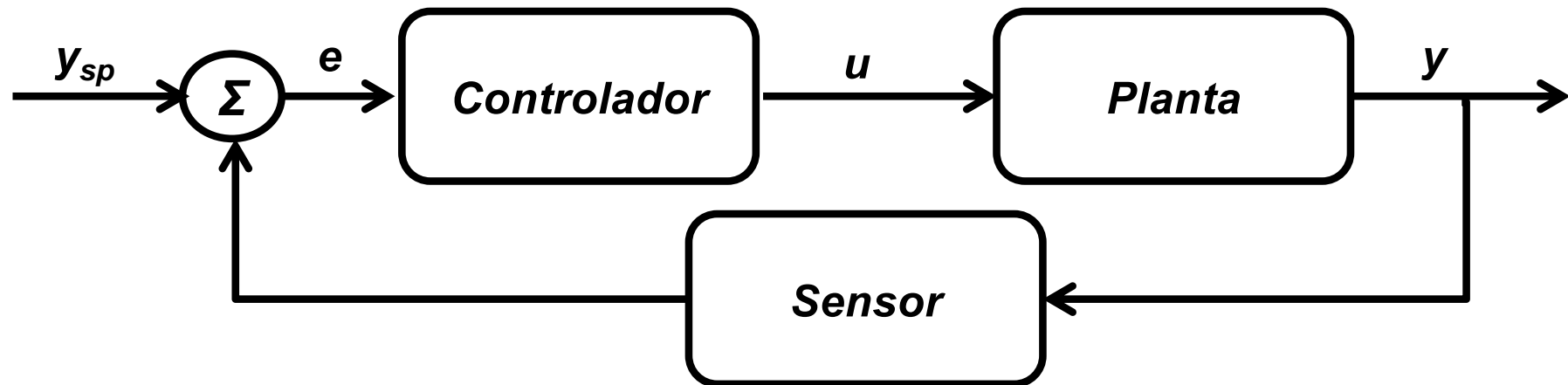


Control de bajo nivel

- ❖ SISO: Single Input / Multiple Output
- ❖ SIMO: Single Input / Multiple Output
- ❖ MISO: Multiple Input / Single Output
- ❖ MIMO: Multiple Input / Multiple Output

Control de bajo nivel

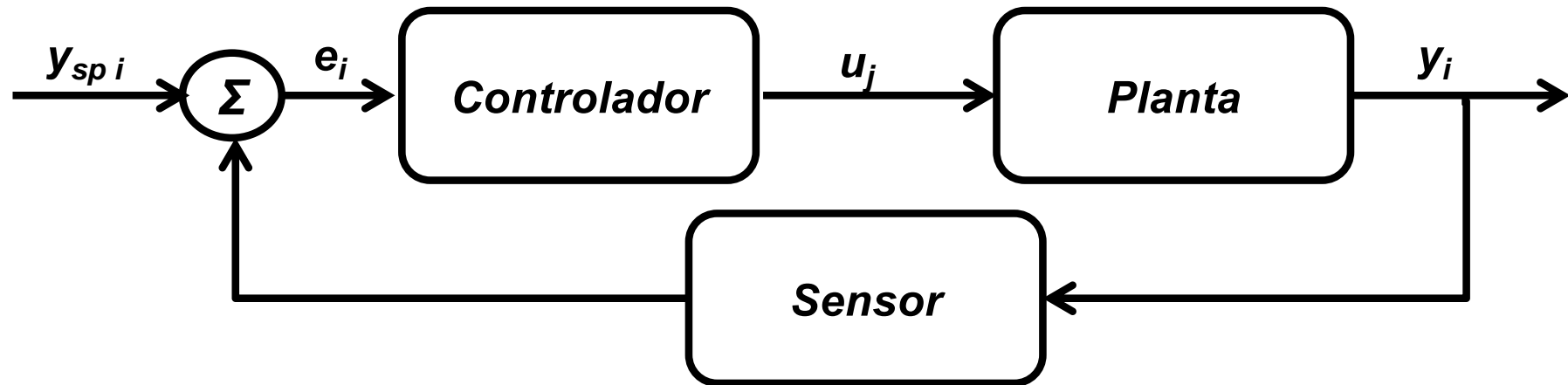
- ❖ SISO: Single Input / Single Output



- ❖ y = Salida de la planta
- ❖ u = Señal de control. Comando para planta
- ❖ y_{sp} = Valor deseado para la salida de la planta (setpoint)
- ❖ $e = y_{sp} - y$

Control de bajo nivel

- ❖ MIMO: Multiple Input / Multiple Output



- ❖ y_i = salida de la planta
- ❖ u_j = Señal de control. Comando para planta
- ❖ $y_{sp i}$ = Valor deseado para la salida de la planta (setpoint)
- ❖ $e_i = y_{sp i} - y_i$

Control de bajo nivel

- ❖ Ejemplo MIMO: Control de crucero de un coche
 - Variable de entrada: Velocidad deseada, aceleración deseada.
 - Variables de salida: Velocidad, Aceleración.
 - Señal de control: apertura de la mariposa

- ❖ Ejemplo 2: caldera

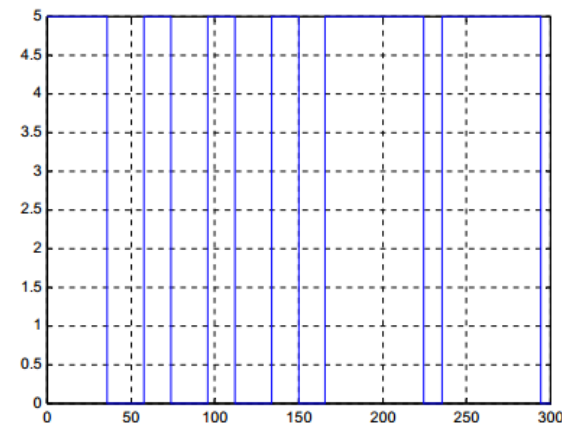
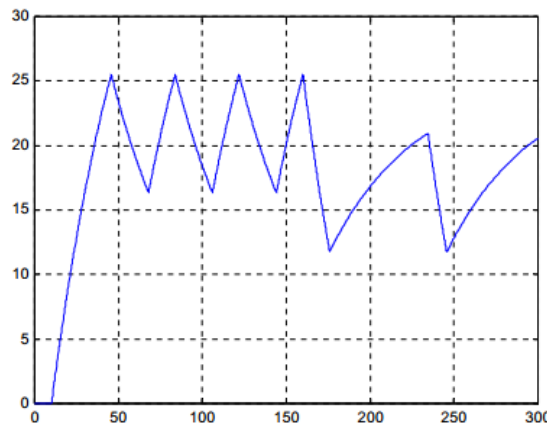
Control de bajo nivel

❖ Control ON – OFF (Todo o nada)

- 1. El control On-Off es la forma más simple de controlar.
- 2. Es comúnmente utilizado en la industria

$$u = \begin{cases} u_{\max} & \text{if } e > 0 \\ u_{\min} & \text{if } e < 0 \end{cases}$$

- Aire acondicionado. Setpoint = 20°. $u_{\max} = 5$ (on). $u_{\min} = 0$ (off)

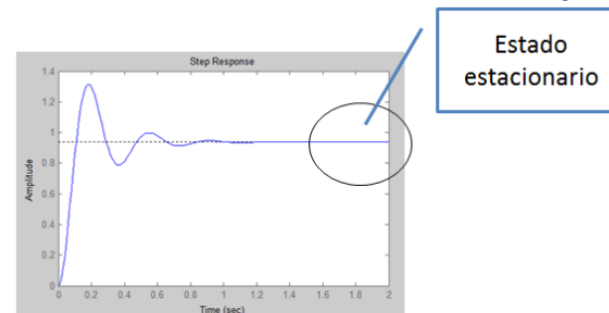


Control de bajo nivel

- ❖ Control Proporcional
- ❖ La acción de control proporcional, da una salida del controlador que es proporcional al error.
- ❖ El error están siempre normalizados entre el 0% y el 100% (habitualmente $[-1, 1]$), en función del tipo de entrada.

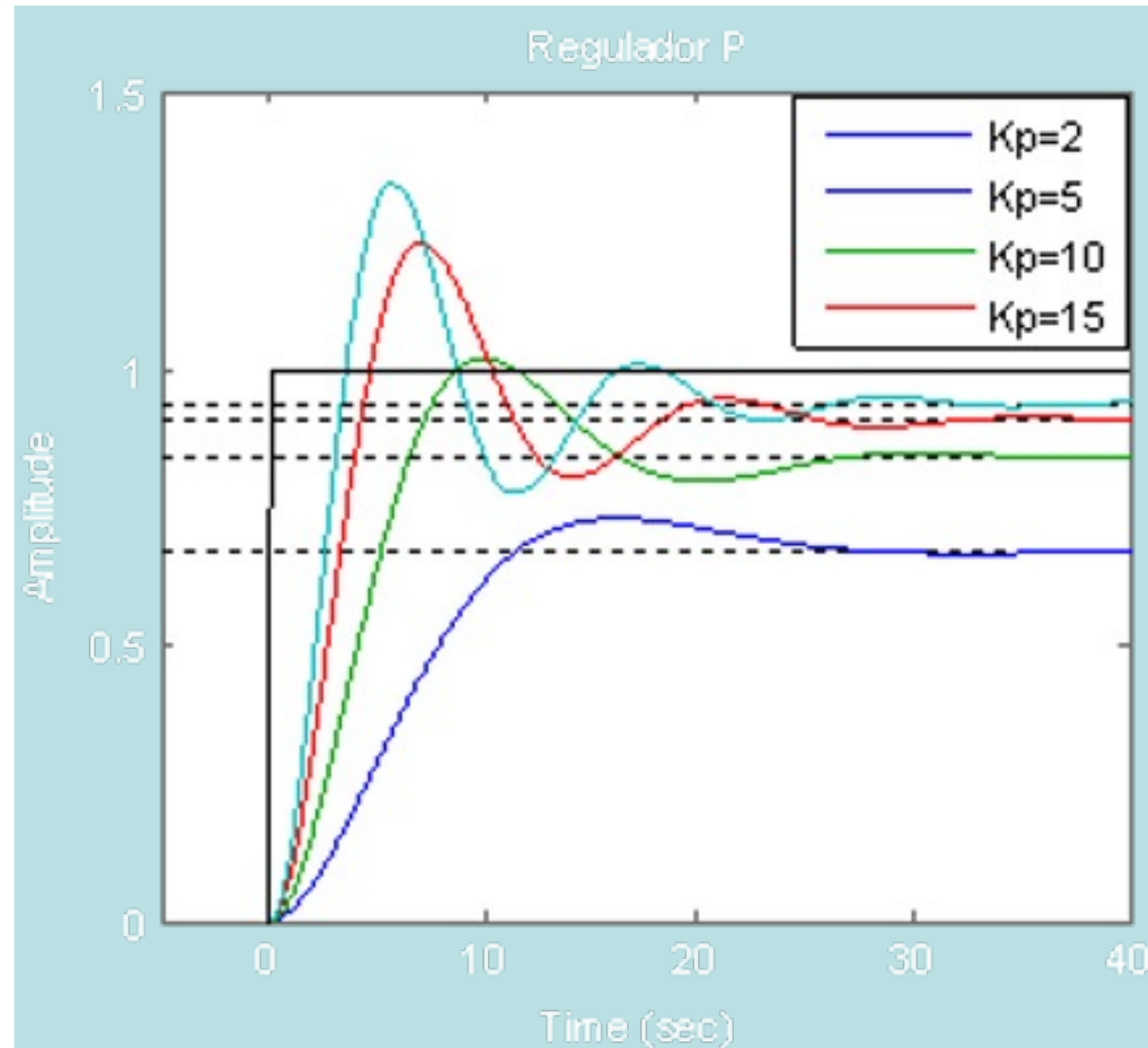
$$u(t) = K_p \cdot e(t) + u_b$$

- ❖ En caso necesario, se aplicará saturación al valor de salida.
- ❖ Donde $u(t)$ es la señal de control en un tiempo t , K_p es una ganancia proporcional ajustable y $e(t)$ es el error entre el setpoint y la medida de la salida de la planta en un tiempo t . u_b es el error estacionario.



Control de bajo nivel

❖ Ejemplo P



❖ Control Proporcional

- Cuanto mayor es la ganancia del control proporcional (K_p), mayor es la señal de control generada (u) para un mismo valor de señal de error (e).
- Ejemplo: Control de crucero de un vehículo.
 - Velocidad deseada = 100 km/h
 - Velocidad real = 80 km/h
 - Salida = apertura de la mariposa (0% - 100%)

Control de bajo nivel

- ❖ Control PID
- ❖ Un controlador PIO se caracteriza por combinar tres acciones (P, I y D) mediante el siguiente algoritmo de control:

$$u(t) = K_p \left[e(t) + \frac{1}{T_i} \int e(t) dt + T_d \frac{de(t)}{dt} \right] = P + I + D$$

- ❖ Donde la componente proporcional es P

$$P_{t+1} = K_p \cdot e(t) + u_b$$

Control de bajo nivel

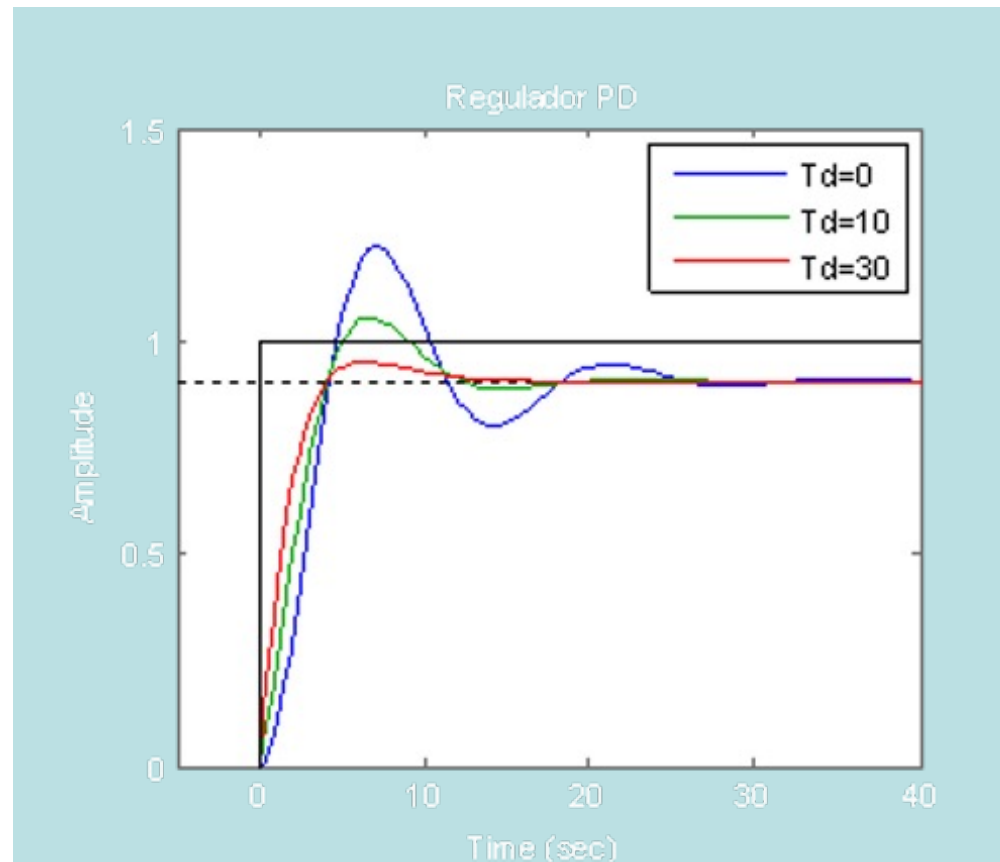
- ❖ Componente derivativa
- ❖ Permite eliminar la sobreactuación de los controladores P mediante una contraactuación, suavizando el valor de la señal de control (u).
- ❖ Se basa en introducir una acción de predicción sobre la señal de error.

$$D_{t+1} = \frac{T_d}{T_d + N \cdot h} D_t - \frac{K_p \cdot T_d \cdot N}{T_d + N \cdot h} (y_t - y_{t-1})$$

- ❖ Donde T_d es la constante de tiempo derivativa (es el intervalo de tiempo, generalmente expresado en segundos, en el que la acción derivativa adelanta a la acción proporcional), N es un filtrado de ruido cuyo valor oscila entre 8-20 y h es el periodo de muestreo. A medida que T_d aumenta la salida se va

Control de bajo nivel

❖ Ejemplo PD



❖ **Componente integral**

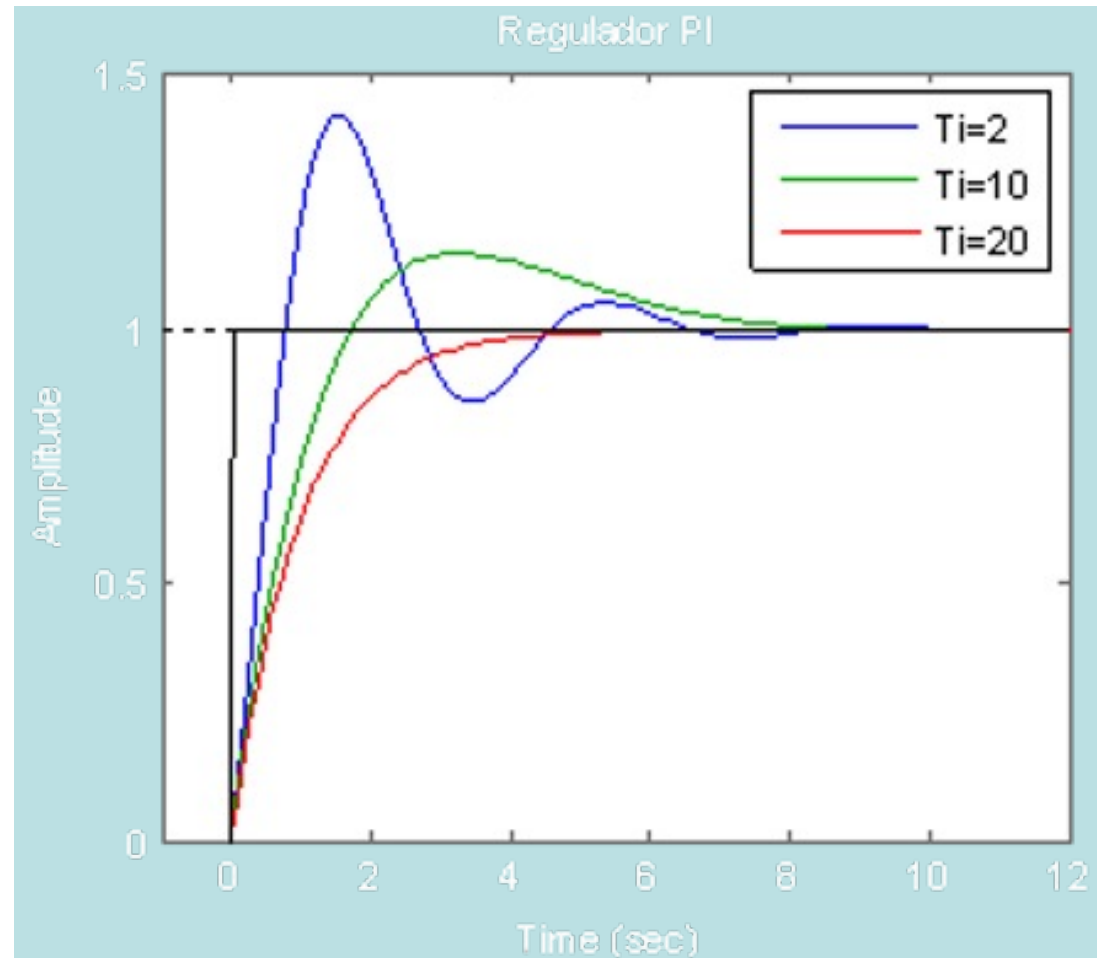
- ❖ Permite reducir los errores estacionarios de los controladores P mediante el incremento de la señal de control para pequeñas correcciones.

$$I_{t+1} = I_t + \frac{K_p \cdot h}{T_i} \cdot e(t)$$

- ❖ Donde T_i es la constante de tiempo integral (tiempo, generalmente expresado en segundos, que debe transcurrir para que la acción integral alcance (iguale o repita) a la acción proporcional) y h es el periodo de muestreo .

Control de bajo nivel

❖ Ejemplo PI



Control de bajo nivel

- ❖ Variación de la estabilidad, la velocidad y el error en estado estacionario cuando se modifican los parámetros del controlador.
 - Es necesario señalar que esta tabla contiene un conjunto de reglas heurísticas y, por tanto, hay excepciones.

	Kp aumenta	Ti disminuye	Td aumenta
Estabilidad	Disminuye	Disminuye	aumenta
Velocidad	aumenta	aumenta	Disminuye
Error est. estacionario	no eliminado	eliminado	no eliminado

❖ Sistemas lineales y no lineales

- Un sistema es lineal si la salida sigue fielmente los cambios producidos en la entrada.
- En caso contrario es no lineal.
- Los controladores PID están indicados fundamentalmente para sistemas lineales.
- Para sistemas no lineales, las técnicas de control clásico son muy complejas o bien se indica la utilización de técnicas de control inteligente.

Control de bajo nivel

- ❖ Técnicas de control inteligente: Soft Computing
- ❖ **Conjunto de técnicas dirigidas a resolver problemas reales imitando la forma en que lo hace el ser humano**
- ❖ El Soft Computing es una rama de la inteligencia artificial centrada en el diseño de sistemas inteligentes capaces de manejar adecuadamente la información incierta, imprecisa y/o incompleta. Esta cualidad permite abordar problemas reales obteniendo soluciones más robustas, manejables y de menor coste que las obtenidas mediante técnicas convencionales.

Control de bajo nivel

- ❖ Técnicas de control inteligente: Soft Computing
- ❖ Las principales técnicas que componen el Soft Computing son la lógica borrosa, las redes neuronales, la computación evolutiva y el razonamiento probabilístico. Desde que Lotfi Zadeh acuñó el término Soft Computing en 1991, esta área ha experimentado un rápido desarrollo tanto en sus aspectos teóricos como, sobretodo, en sus aplicaciones empresariales.
- ❖ Las técnicas de Soft Computing abordan problemas de gran diversidad tanto en tipología (modelado, optimización, planificación, control, predicción, minería de datos, etc.) como en el sector de aplicación (producción industrial, logística, energía, banca, agroalimentación, entre otros).

Control de bajo nivel

❖ Soft Computing



Control de bajo nivel: Lógica borrosa

- ❖ Generalización de la lógica multivaluada.
 - Utiliza grados de verdad de las fórmulas lógicas en lugar de una verdad estricta de cierto/falso.
- ❖ Permite representar y utilizar conceptos "imprecisos" y realizar razonamientos "aproximados".
- ❖ Un grado de verdad puede ser:
 - Un valor numérico del intervalo $[0,1]$. (0.2, 0.5, 0.75, ...)
 - Una etiqueta lingüística. (más o menos verdad, bastante ...)
- ❖ Los grados de verdad estarán asociados a conjuntos borrosos.
- ❖ OBJETIVO: Extender la inferencia lógica, aplicándola a los conjuntos borrosos de los grados de verdad.
- ❖ Combina métodos simbólicos (cualitativos) y numéricos (cuantitativos)"