# *Diseño y Arquitectura de Servicios Escalables*

## Unidad 5
## Gestión automatizada de servicios

# Índice

# Bibliografía


[CS13] Emiliano Casalicchio, Luca Silvestri: “Mechanisms for SLA provisioning in cloud-based service providers”. Computer Networks 57(3): 795-810 (2013)

[LML13] T. Lorido-Botrán, J. Miguel-Alonso, J.A. Lozano: “Comparison of auto-scaling techniques for cloud environments”, XXIV Jornadas de Paralelismo, pgs. 187-192, Universidad Complutense de Madrid, septiembre 2013.

[LML14] Tania Lorido-Botrán, José Miguel-Alonso, José Antonio Lozano: “A Review of Auto-scaling Techniques for Elastic Applications in Cloud Environments”. J. Grid Comput. 12(4): 559-592 (2014)

[LZG+84] Edward D. Lazowska, John Zahorjan, G. Scott Graham, Kenneth C. Sevcik: “Quantitative System Performance. Computer System Analysis Using Queueing Network Models”. Prentice-Hall, Inc, Englewood Cliffs, New Jersey, EEUU, 1984. ISBN: 0-13-746975-6.

[PC10] Tharindu Patikirikorala, Alan W. Colman: “Feedback controllers in the cloud”. APSEC 2010, Cloud Workshop (2010)

[SB98] Richard S. Sutton, Andrew G. Barto: “Reinforcement Learning. An Introduction”, The MIT Press, Cambridge, MA, EEUU, febrero 1998. ISBN: 9780262193986.

# Índice

# 1. Introducción

- ¿Para qué se necesita una gestión automatizada de un servicio?
  - Para garantizar que sea elástico.
    - Elasticidad:
      - Escalabilidad.
      - Adaptabilidad.
      - Autonomía en la gestión.
  - Con ello, el consumo de recursos será proporcional a la carga.

# 1. Introducción

- ¿Cómo garantizar su autonomía?
  - Mediante un ciclo de control MAPE-K.
    - Ya visto en el tema 1.
    - Cuatro fases:
      - Monitorización.
      - Análisis.
      - Planificación.
      - Ejecución.
        - Accediendo a una base de conocimiento común (“*Knowledge base*”).

# 1. Introducción

- ¿Qué debe tenerse en cuenta?
  - Análisis: SLA.
    - Es lo que se pretende respetar.
    - Hay que comprobar si el estado actual del servicio pone en riesgo ese SLA.
    - El resto de fases del ciclo de control dependen de esta comprobación.

# 1. Introducción

- ¿Qué debe tenerse en cuenta? (II)
  - Monitorización: Parámetros del SLA.
    - Debemos obtener los valores actuales en los parámetros relevantes.
    - Así se aporta la entrada necesaria para la fase de análisis.
  - Planificación:
    - La fase de análisis generará el tipo de acción a aplicar.
      - Escalado (incremento/decremento).
      - Migración.
    - En esta fase debe seleccionarse qué acción aplicar y en qué grado.

# 1. Introducción

- ¿Qué debe tenerse en cuenta? (III)
  - Ejecución:
    - Las acciones planificadas en la fase anterior, deben aplicarse.
    - Su resultado se comprobará en próximas monitorizaciones.
    - Habrá cierto plazo de espera:
      - Para garantizar que la acción aplicada en esta iteración surte efecto.

# 1. Introducción

- Estrategias posibles en una gestión autónoma:
  - Reactiva:
    - Fijar umbrales en los valores de los parámetros relevantes del SLA.
    - Reaccionar cuando esos umbrales sean sobrepasados.
  - Predictiva:
    - Generar un modelo del comportamiento del servicio o de la carga a soportar.
    - Observar el historial de comportamiento para refinar la predicción.
    - Predecir con el modelo el comportamiento futuro del servicio o de la carga.
    - Aplicar acciones de escalado cuando se prevea que ese comportamiento futuro pueda poner en peligro el SLA acordado.

# 1. Introducción

- Acciones de escalado a aplicar:
  - Horizontal:
    - El más sencillo de implantar: Iniciar o parar una o más instancias.
    - Dos mecanismos:
      - Máquinas virtuales (MV):
        - El tradicional.
        - Pesado.
        - Lento (entre 1 y 10 minutos, dependiendo del anfitrión y del tamaño de la imagen).
      - Contenedores:
        - Ligero.
        - Rápido (entre 0.5 y 5 segundos).
  - Vertical:
    - Variaciones en los recursos asignados a cada instancia.
    - Soportado por pocos gestores.
    - Rápido.

# Índice

# 2. Gestión reactiva

- Objetivo: Evitar que el SLA se incumpla.
- ¿Cómo conseguirlo?
  - Técnica: Establecimiento de límites para los valores de ciertas métricas y uso de reglas de escalado.
  - Por cada SLO:
    - Decidir qué parámetros de los recursos a monitorizar influyen directamente.
    - Decidir qué límites establecer para cada uno.
    - Si algún límite se supera, reconfigurar el servicio.

# 2. Gestión reactiva

- En la práctica, se crean reglas de escalado:
  - Una regla para cada acción:
    - Incremento en escalado horizontal.
    - Decremento en escalado horizontal.
  - Hay que decidir:
    - Qué parámetros evaluar en cada regla.
    - Qué límites fijar.

# 2. Gestión reactiva

- Ejemplo [LML13]:
  - SLO a gestionar: tiempo de respuesta.
  - Métrica a considerar: uso de procesador.
  - Regla de incremento:
    - Añadir 1 réplica si el uso de procesador supera el 70%.
  - Regla de decremento:
    - Eliminar 1 réplica (siempre que haya al menos 2) si el uso de procesador baja del 30%.

# 2. Gestión reactiva

- Ventajas:
  - Gestión muy sencilla.
  - Mecanismos fáciles de implantar.
- Inconvenientes:
  - Es difícil seleccionar las métricas a usar.
  - Resulta difícil establecer valores idóneos para esos límites superior e inferior.
  - Es casi imposible gestionar variaciones pronunciadas de la carga con esta técnica.
  - Su calidad depende en gran medida del tiempo necesario para aplicar sus decisiones de escalado.
    - Aceptable con contenedores.

# 2. Gestión reactiva

- [LML13] propone una segunda técnica de este tipo:
  - Límites dinámicos.
- ¿En qué consiste?
  - Se toman unos valores iniciales para los límites.
  - Si durante ciertos intervalos de evaluación ha habido:
    - Demasiados incumplimientos del SLA:
      - Los límites se acercan entre sí, reduciendo el superior y aumentando el inferior (o dejándolo estable).
        - Así se reaccionará antes.
    - Cero incumplimientos del SLA:
      - Los límites se separan algo más.
        - Así se tardará más en reaccionar.

# 2. Gestión reactiva

- Hay otras variantes en el uso de límites. Ejemplo [CS13]:
  - Emplear varios límites superiores y varios inferiores. Métrica: uso de procesador
    - > 70%: Añadir dos instancias.
    - > 62%: Añadir una instancia.
    - < 50%: Eliminar una instancia.
    - < 25%: Eliminar dos instancias.

# Índice

# 3. Gestión predictiva

- [LML14] distingue las siguientes técnicas con gestión potencialmente predictiva:
    1. Aprendizaje reforzado autónomo (“reinforcement learning”) [SB98].
    2. Teoría de colas / redes de colas [LZG+84].
    3. Teoría de control.
    4. Análisis de series temporales.

# 3.1. Aprendizaje reforzado

- En esta técnica:
  - Un agente (el gestor de elasticidad) interactúa con el entorno (aplicación a gestionar) para maximizar cierta función (cumplir cierto SLO). Para ello:
    - Cada interacción (decisión de escalado) reporta cierto resultado (beneficio).
    - El agente almacena los resultados de sus decisiones y los utiliza para mejorarlas.
    - Aprendizaje: Optimizar una función que acumula los resultados de cada decisión tomada hasta el momento.

# 3.1. Aprendizaje reforzado

- Ventajas:
  - Puede utilizarse sin disponer de ningún conocimiento inicial.
  - Su implantación es bastante sencilla.
- Inconvenientes:
  - Los resultados iniciales son malos, hasta que no se hayan explorado todas las alternativas para cada escenario posible.
  - El espacio necesario para mantener toda la información necesaria suele ser grande.
  - El intervalo de aprendizaje inicial, hasta obtener resultados fiables, suele ser demasiado largo.
  - Si cambia un entorno previamente estable, se necesita tiempo para identificar correctamente esa situación y modelarla.

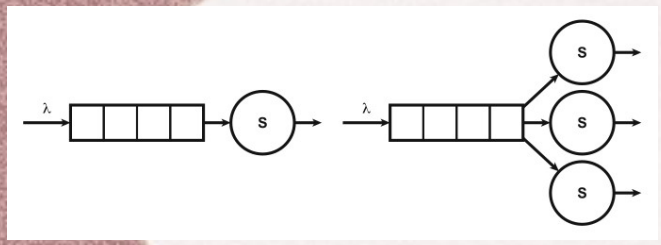
# 3.1. Aprendizaje reforzado

- Otros usos:
  - Puede utilizarse el aprendizaje reforzado para evaluar qué métricas convendrá utilizar en una gestión reactiva (límites+reglas de escalado) para cada SLO.
    - O para concretar los valores de los límites a usar en cada métrica.
  - Puede combinarse con teoría de control durante el intervalo de aprendizaje inicial.
    - La teoría de control toma las decisiones y el aprendizaje reforzado evalúa su calidad y recuerda las mejores acciones.
    - Posteriormente se utiliza únicamente el aprendizaje reforzado.

# 3.2. Teoría de colas

- **Objetivo**: Construir un modelo de comportamiento del servicio. Con él se podrá:
  - Predecir el rendimiento, los tiempos de espera y los tiempos de servicio en función de la carga recibida.
- Construcción:
  - Sencilla si solo hay un componente.
  - Redes de colas [LZG+84] cuando haya múltiples componentes relacionados.
- Uso:
  - Puede utilizarse de forma aislada como mecanismo predictivo.
  - Pero también como utilidad para fijar los límites en un mecanismo reactivo.

# 3.2. Teoría de colas



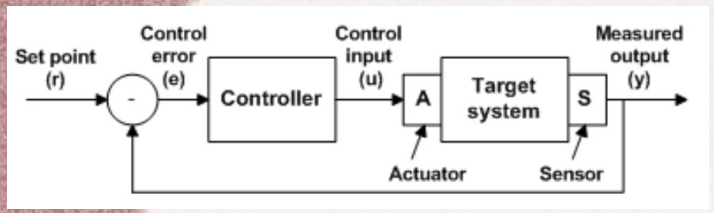

- Los modelos de colas están basados en:
  - Identificar cada componente servidor.
    - Tiempo de servicio.
  - Modelar una cola de llegada a cada servidor.
    - Evaluar el comportamiento de esa cola en función de la tasa de llegada de solicitudes.
- Con ello, para una determinada tasa de llegada:
  - Se puede calcular, analíticamente:
    - Utilización del servidor.
    - Tiempo de espera en cola.
    - Tiempo medio de servicio.
    - Tamaño de la cola.

# 3.2. Teoría de colas

- Ventajas:
  - Modelo teórico bien conocido.
  - Evaluación sencilla para servicios estacionarios.
    - Tasa de llegada, número de instancias y tiempo de servicio constantes.
- Inconvenientes:
  - Deben recalcularse todos los valores para cada variación de la carga o del número de instancias servidoras.
    - Es el supuesto que interesa en un servicio elástico.
    - La entrada la proporciona la fase de monitorización: tasa de llegada.
    - Resulta demasiado pesado.
  - Se puede recurrir a simulación en esos casos, en base al modelo de colas generado.

# 3.3. Teoría de control

- La teoría de control realiza esta gestión:
  - Un controlador debe mantener una *variable controlada* “y”…
  - ...lo más próxima posible a un *valor objetivo* “r”…
  - ...ajustando para ello otra *variable manipulada* “u” cuyo valor es directamente modificable.




- Por ejemplo, en nuestro caso:
  - “y”: Tiempo de respuesta.
  - “r”: <200ms
  - “u”: Número de instancias servidoras.

# 3.3. Teoría de control

- Para mejorar la reacción del controlador, se necesita algún *modelo de rendimiento* del sistema controlado. En ese modelo:
  - Se utilizarán algunos parámetros de sintonización.
    - Especifican las características del sistema.
      - Parámetros fijos, si el comportamiento sigue siempre el mismo patrón.
      - Parámetros variables, si el comportamiento de ese elemento/recurso depende de la carga.
- Los modelos de rendimiento se llaman “funciones de transferencia” en la teoría de control clásica.
  - Fuera del ámbito de la computación autónoma.

# 3.3. Teoría de control

- Según [PC10] en el ámbito de los servicios elásticos pueden distinguirse estos tipos de controladores:
  - Fijos: Los parámetros de sintonización que utiliza el controlador no pueden variar. Reactivo.
  - Modelo-predictivos (MPC): Utiliza algún modelo de comportamiento para predecir el estado futuro del sistema tras los cambios que haya en la carga, y ajusta la variable manipulada en consecuencia. Predictivo.
  - Adaptables: Son capaces de variar dinámicamente sus parámetros de sintonización. Reactivo.
  - Reconfigurables: Es un caso de controlador adaptativo en el que, además de modificar los parámetros de sintonización, se puede cambiar también el algoritmo/modelo de control. Reactivo.

# 3.3. Teoría de control

- Según [PC10], las características de estas cuatro clases son:

| Property | Fixed | MPC | Adaptive | Reconfiguring |
|---|---|---|---|---|
| Provide systematic/formal design tools | √ | √ | √ | √ |
| Less design complexity | √ | √ | x | x |
| Adapts at runtime to different conditions | x | x | √ | √ |
| Ability to handle fast varying condition | x | x | x | √ |
| Makes proactive decisions | x | √ | x | x |
| Provides MIMO control | √ | √ | √ | √ |

# 3.3. Teoría de control

- Los modelos de rendimiento utilizables en teoría de control para gestionar servicios elásticos son [LML14]:
  - ARMA / ARMAX: Series temporales.
    - Sección 3.4.
  - Filtros de Kalman: Series temporales.
  - Modelo Kriging.
    - También llamado “regresión de procesos de Gauss”.
  - Modelos *fuzzy*.
    - Lógica difusa: La pertenencia de un elemento a un conjunto se expresa con un valor real entre 0 y 1.
- Los dos primeros son los más sencillos.

# 3.3. Teoría de control

- Ventajas:
  - Puede considerarse una evolución de los sistemas reactivos, en la que un modelo proporciona los límites a emplear en las reglas de escalado.
  - Los de tipo MPC y reconfigurables llegan a manejar bien situaciones complejas.
- Inconvenientes:
  - Los dos tipos con mejor funcionalidad (MPC y reconfigurables) tienen un diseño e implantación muy exigentes.

# 3.4. Análisis de series temporales

- Serie temporal: Serie de “w” valores de alguna métrica obtenidos a intervalos regulares.
  - Durante la fase de monitorización del ciclo de control MAPE-K.
  - Intervalo de obtención = intervalo de monitorización.
    - Ejemplo: cada minuto.
- Objetivo: Predecir valores futuros en la serie, tras analizar los últimos “q” (q<w) valores en ella.
  - “q” = ventana de entrada (o ventana del histórico).

# 3.4. Análisis de series temporales

- Hay dos clases principales de técnicas de análisis de series temporales, según su objetivo [LML14]:
  - Predicción de valores.
    1) Media móvil (“moving average”: MA).
    2) Alisado exponencial (“exponential smoothing”, ES).
    3) Autorregresión (AR).
    4) Media móvil autorregresiva (ARMA).
    5) Regresión.
    6) Redes neuronales.
  - Identificación de patrones de repetición.
    7) “Pattern matching”.
    8) FFT.
    9) Autocorrelación.

# 3.4. Análisis de series temporales

- Para explicar las primeras técnicas utilizaremos dos ejemplos de series temporales, con:
  - “w”=12
  - “q”=5
  - Asumimos que la predicción se realiza tras obtener el décimo valor de la serie.
  - ST1: 2, 10, 3, 5, 2, 8, 5, 10, 1, 4, pred, 2, 9
  - ST2: 1, 3, 5, 3, 1, 3, 5, 3, 1, 3, pred, 5, 3
    - Esta segunda serie sigue un patrón repetitivo.

# 3.4.1. Media móvil (MA)

- El valor predicho se calcula como la media ponderada de los últimos “q” valores observados.
  - Se representa como MA(q).
  - Si asumimos pesos equitativos (1/q).
    - ST1: $s_{11}$= (8+5+10+1+4)/5=5.6
    - ST2: $s_{11}$= (3+5+3+1+3)/5=3
  - En esta técnica se asume que las variaciones entre los valores obtenidos se deben a “ruido” en la medición. Se pretende eliminar ese ruido.

# 3.4.2. Alisado exponencial

- Utiliza la siguiente fórmula para calcular el próximo valor ($s_n$),
  - en función del último valor observado ($x_{n-1}$)
  - y de la previsión anterior ($s_{n-1}$):
  - $s_n = ax_{n-1} + (1-a)s_{n-1}$
  - “a” representa el peso asignado a los valores reales frente a las predicciones.
    - Un valor entre 0 y 1.
- Esta técnica no puede limitarse a los últimos “q” valores.
  - Se aplica desde el principio.
  - En la primera aplicación, $s_{n-1}=x_{n-1}$
- Tiene un objetivo similar a MA: eliminar variabilidad.

# 3.4.2. Alisado exponencial

- Ejemplo: Aplicación sobre ST2, con “a”=0.5:

| n | $S_n$ | $X_n$ |
|---|---|---|
| 1 | -- | 1 |

# 3.4.2. Alisado exponencial

- Ejemplo: Aplicación sobre ST2, con “a”=0.5:

| n | $S_n$ | $X_n$ |
|---|---|---|
| 1 | -- | 1 |
| 2 | 1 | |

# 3.4.2. Alisado exponencial

- Ejemplo: Aplicación sobre ST2, con “a”=0.5:

| n | $S_n$ | $X_n$ |
|---|---|---|
| 1 | -- | 1 |
| 2 | 1 | 3 |
| 3 | 2 | |

# 3.4.2. Alisado exponencial

- Ejemplo: Aplicación sobre ST2, con “a”=0.5:

| n | $S_n$ | $X_n$ |
|---|---|---|
| 1 | -- | 1 |
| 2 | 1 | 3 |
| 3 | 2 | 5 |
| 4 | 3.5 | |

# 3.4.2. Alisado exponencial

- Ejemplo: Aplicación sobre ST2, con “a”=0.5:

| n | $S_n$ | $X_n$ |
|---|---|---|
| 1 | -- | 1 |
| 2 | 1 | 3 |
| 3 | 2 | 5 |
| 4 | 3.5 | 3 |
| 5 | 3.25 | |

# 3.4.2. Alisado exponencial

- Ejemplo: Aplicación sobre ST2, con “a”=0.5:

| n | $S_n$ | $X_n$ |
|---|---|---|
| 1 | -- | 1 |
| 2 | 1 | 3 |
| 3 | 2 | 5 |
| 4 | 3.5 | 3 |
| 5 | 3.25 | 1 |
| 6 | 2.125 | |

# 3.4.2. Alisado exponencial

- Ejemplo: Aplicación sobre ST2, con “a”=0.5:

| n | $S_n$ | $X_n$ |
|---|---|---|
| 1 | -- | 1 |
| 2 | 1 | 3 |
| 3 | 2 | 5 |
| 4 | 3.5 | 3 |
| 5 | 3.25 | 1 |
| 6 | 2.125 | 3 |
| 7 | 2.56 | ... |

# 3.4.3. Autorregresión (AR)

- La autorregresión de orden “p”, AR(p), selecciona los “p” términos más recientes de la serie ($x_i$) para aplicar esta fórmula:

  $s_{n+1}=b_1 \cdot x_n+b_2 \cdot x_{n-1}+...+b_p \cdot x_{n-p+1}+e_n$

- Donde:
  - $e_n$ es un término asociado al ruido.
  - Los $b_i$ son coeficientes que se suelen calcular (para $b_i \cdot x_j$) como la covarianza entre $x_j$ y $x_{j-1}$ dividida entre la varianza de los “x” en esa serie temporal, hasta ese momento.
- El cálculo es mucho más costoso que en MA.

# 3.4.4. Media móvil autorregresiva (ARMA)

- ARMA(p,q) combina las técnicas AR(p) y MA(q) para predecir el valor del próximo término en la serie temporal.
  - ARMA(0,q) = MA(q)
  - ARMA(p,0) = AR(p)
- Como ambas técnicas tomadas como base no gestionan bien las fluctuaciones en la serie, ARMA tampoco lo hará.
  - Proporciona buenas aproximaciones cuando la tendencia es uniforme.
    - Por ejemplo, lineal.

# 3.4.5. Regresión

- Modelo matemático que computa una función polinómica cuya imagen sea lo más cercana posible a los valores de la serie temporal.
  - En el caso de la regresión lineal, el polinomio es de primer grado.
  - El objetivo es idéntico al utilizado en AR(p).
    - Pero el número de puntos a considerar difiere.
      - Ahora son “w” o “q”, con w > q > p.
    - Y el método de cálculo, también.
      - Es más sencillo en la regresión lineal.

# 3.4.6. Redes neuronales

- Es un caso particular de “machine learning”.
- Una red neuronal es un grupo de elementos (“neuronas artificiales”) interconectados en niveles.
  - Un nivel de entrada con varias neuronas.
  - Un nivel de salida también con múltiples neuronas.
  - Uno o más niveles intermedios ocultos.
- En el caso del análisis de series temporales, el nivel de entrada tiene una neurona por cada valor en la ventana histórica y una neurona por cada valor predicho en el nivel de salida.
- Hay una etapa de aprendizaje en la que se van utilizando vectores de pesos con valores aleatorios.
- Esos pesos se van adaptando hasta que la precisión de las predicciones sea suficientemente buena.

# 3.4.7. “Pattern matching”

- Esta técnica y las dos siguientes tratan de identificar patrones repetitivos en la serie temporal.
- Esta búsqueda de patrones suele fijarse en cuatro características de la serie temporal:
  - Tendencia general: creciente, decreciente, o constante.
  - Estacionalidad: Diaria, semanal, mensual, anual…
  - Ciclo: Identificación de picos o simas periódicos.
  - Aleatoriedad: Existencia de intervalos que no respeten el patrón.
- En “pattern matching” se utilizan técnicas similares a la búsqueda de correspondencias entre cadenas.
  - Uso de un “corpus” de patrones.
  - Revisión de cada elemento del “corpus” sobre la serie a estudiar, hasta encontrar alguna coincidencia.

# 3.4.8. FFT

- FFT (“Fast Fourier Transform”) es una técnica utilizada tradicionalmente en el análisis de señales.
  - Descompone la serie temporal en componentes de diferentes frecuencias.
  - Las frecuencias dominantes (en caso de existir) proporcionarán el patrón repetitivo de la serie temporal.

# 3.4.9. Autocorrelación

- Se comparan dos copias de la serie temporal.
- Una de ellas se desplaza progresivamente hacia el pasado.
  - Tras cada deplazamiento se evalúa la correlación entre las dos series.
  - Si la correlación es alta, se habrá identificado un patrón repetitivo.

# Índice

# 4. Comparativa

- [LML13] compara estas técnicas:
  - Reactivas:
    - Límites + reglas de escalado (“Rules”)
    - Límites dinámicos (“Dynamic thresholds”).
    - Teoría de control. Controlador fijo. (“IController”).
  - Predictivas (análisis de series temporales):
    - Media móvil (“MA”).
    - Alisado exponencial (“ES”).
    - Regresión lineal (“LR”).

# 4. Comparativa

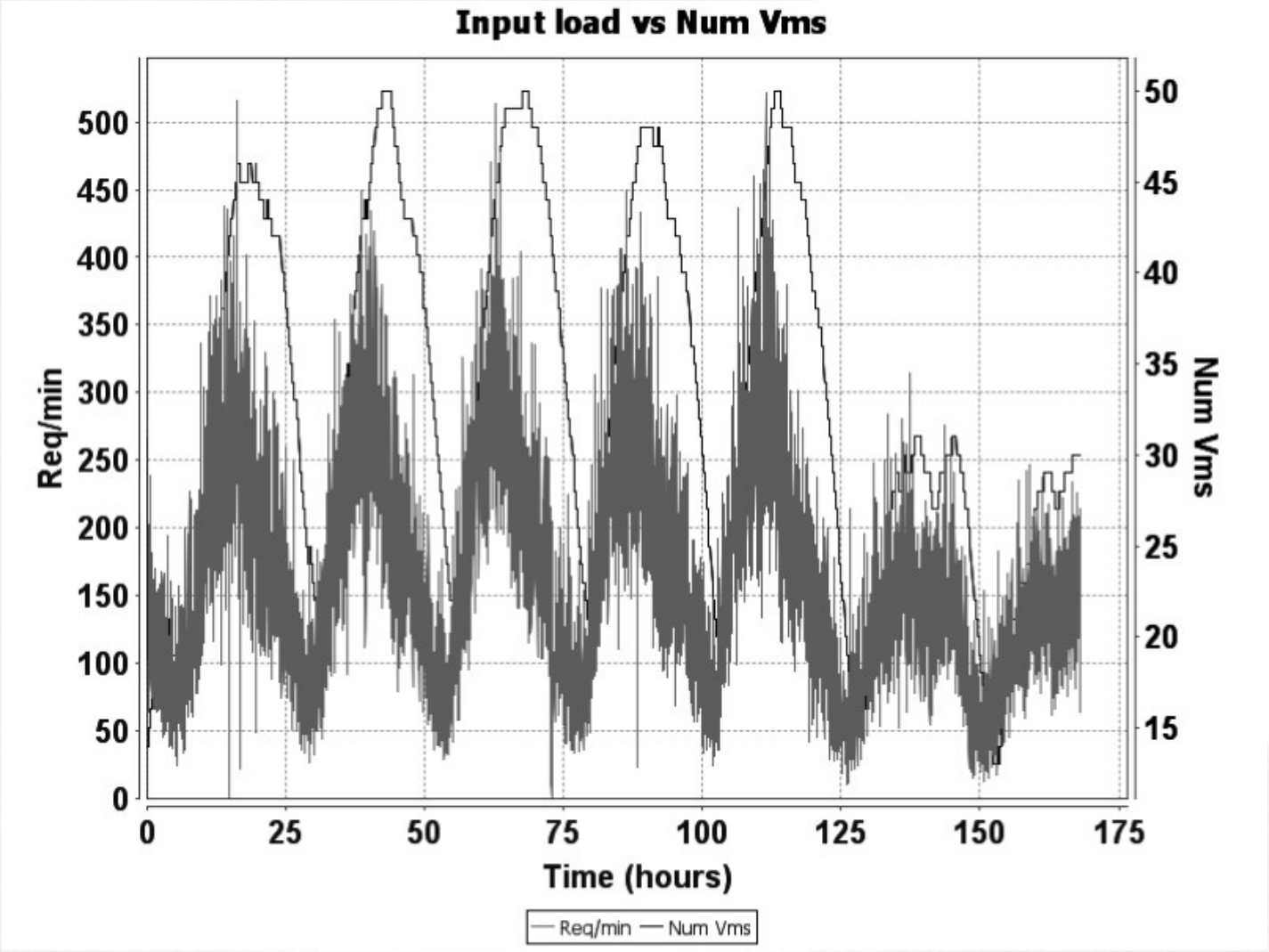
- Servicio a gestionar:
  - Servicio web interactivo.
    - Con un equilibrador de carga (proxy inverso).
    - Múltiples instancias servidoras.
    - Escalado horizontal.
    - Servidores independientes, sin compartición de estado.
  - Monitorización / evaluación:
    - Cada minuto.
    - Despliegue sobre máquinas virtuales.

# 4. Comparativa

- Gestión elástica:
  - Métrica: uso de procesador.
  - Las técnicas predictivas tratan de calcular el uso de procesador que habrá en el siguiente instante de monitorización.
    - Así pueden adelantar la decisión de escalado.
    - ...y compensar lo que se tarda en iniciar o para una máquina virtual.
- Análisis:
  - SLO: Tiempo de servicio.
  - VMcost: Coste económico de las VM utilizadas.
  - SLOv: Porcentaje de incumplimiento del SLO sobre el conjunto de peticiones realizadas por los clientes.

# 4. Comparativa

- Carga utilizada: Traza ClarkNet (una semana).



# 4.1. Resultados

- Obtenidos mediante simulación.
  - CloudSim.
  - Se analizan dos clases de VM, según el tiempo necesario para iniciarlas:
    - 0 min: Equivalente a un contenedor.
    - 10 min: VM.

# 4.1. Resultados

- Técnicas reactivas:

| Technique | Parameters | | 0 min | | 10 min | |
|---|---|---|---|---|---|---|
| | | | VMcost | SLOv | VMcost | SLOv |
| Rules | UT: 60 | DT: 20 | 5756.70 | 0.15 | 5575.80 | 0.33 |
| | UT: 70 | DT: 30 | 4677.00 | 0.86 | 4402.60 | 2.91 |
| | UT: 80 | DT: 30 | 4330.10 | 1.46 | 4155.80 | 4.92 |
| | UT: 80 | DT: 40 | 4037.20 | 3.24 | 3764.10 | 8.89 |
| | UT: 90 | DT: 30 | 4026.70 | 2.66 | 3914.90 | 7.13 |
| | UT: 90 | DT: 10 | 5172.80 | 0.49 | 5136.20 | 0.79 |
| Dynamic thresholds | UT: 60 | DT: 20 | 5175.50 | 0.50 | 5127.60 | 0.87 |
| | UT: 70 | DT: 30 | 5169.50 | 0.50 | 5127.50 | 0.87 |
| | UT: 80 | DT: 30 | 5168.90 | 0.51 | 5127.50 | 0.87 |
| | UT: 80 | DT: 40 | 5168.30 | 0.51 | 5126.80 | 0.90 |
| | UT: 90 | DT: 30 | 5168.70 | 0.51 | 5127.40 | 0.88 |
| | UT: 90 | DT: 10 | 5173.50 | 0.48 | 5136.30 | 0.79 |
| IController | K: -0.01 | Target: 60% | 6474.60 | 0.04 | 6538.50 | 0.29 |
| | K: -0.01 | Target: 65% | 5658.00 | 0.16 | 5492.60 | 0.71 |
| | K: -0.01 | Target: 70% | 5089.20 | 0.44 | 4906.00 | 1.81 |
| | K: -0.01 | Target: 75% | 4602.30 | 1.17 | 4467.20 | 4.07 |
| | K: -0.01 | Target: 80% | 4207.70 | 2.40 | 4098.40 | 7.55 |

# 4.1. Resultados

- Técnicas reactivas (valoración, contenedores):
  - Uno de los mejores resultados se da con límites fijos:
    - límite superior=70% uso,
    - límite inferior=30% uso.
    - Coste: 4677.
    - Incumplimiento SLO: 0.86%.
  - La gestión con límites variables parece tender a un límite superior=90% y un límite inferior=10%.
  - El controlador fijo requiere configuraciones más caras que la técnica con límites fijos para un porcentaje de incumplimiento similar.
  - Pregunta:
    - Si sus límites están más separados (y eso sugiere obtener más incumplimientos del SLO), ¿Por qué la configuración UT=90 + LT=10 genera menos incumplimientos que la UT=90 + LT=30?

# 4.1. Resultados

- Técnicas predictivas:

| T. | UT LT | Par. | 0 min VMcost | 0 min SLOv | 10 min VMcost | 10 min SLOv |
|---|---|---|---|---|---|---|
| MA | 60 20 | W:2 | 5699.70 | 0.15 | 5638.80 | 0.24 |
| | | W:3 | 5659.80 | 0.16 | 5600.90 | 0.31 |
| | | W:5 | 5572.30 | 0.18 | 5551.20 | 0.36 |
| | | W:10 | 5479.00 | 0.23 | 5393.70 | 0.41 |
| | 70 30 | W:2 | 4585.80 | 0.78 | 4414.00 | 2.59 |
| | | W:3 | 4535.70 | 0.90 | 4423.20 | 2.73 |
| | | W:5 | 4539.30 | 1.07 | 4502.20 | 2.27 |
| | | W:10 | 4505.20 | 1.43 | 4469.80 | 2.62 |
| | 90 10 | W:2 | 5063.00 | 0.88 | 5090.90 | 1.27 |
| | | W:3 | 5008.70 | 1.25 | 5066.20 | 1.57 |
| | | W:5 | 5079.30 | 1.57 | 5050.70 | 2.13 |
| | | W:10 | 4945.20 | 2.37 | 5003.40 | 3.41 |

| T. | UT LT | Par. | 0 min VMcost | 0 min SLOv | 10 min VMcost | 10 min SLOv |
|---|---|---|---|---|---|---|
| ES | 60 20 | $\alpha$:.1 | 5364.50 | 0.30 | 5371.20 | 0.45 |
| | | $\alpha$:.3 | 5544.50 | 0.18 | 5548.20 | 0.33 |
| | | $\alpha$:.6 | 5751.00 | 0.13 | 5715.10 | 0.23 |
| | | $\alpha$:.9 | 5735.70 | 0.15 | 5603.00 | 0.29 |
| | 70 30 | $\alpha$:.1 | 4453.30 | 1.85 | 4428.30 | 3.14 |
| | | $\alpha$:.3 | 4565.70 | 1.00 | 4514.40 | 2.26 |
| | | $\alpha$:.6 | 4596.20 | 0.71 | 4474.00 | 2.22 |
| | | $\alpha$:.9 | 4662.70 | 0.78 | 4436.80 | 2.72 |
| | 90 10 | $\alpha$:.1 | 4801.40 | 3.74 | 4844.60 | 4.65 |
| | | $\alpha$:.3 | 5092.10 | 1.68 | 5144.60 | 2.32 |
| | | $\alpha$:.6 | 4997.70 | 1.02 | 5039.30 | 1.40 |
| | | $\alpha$:.9 | 5099.00 | 0.58 | 5090.80 | 0.90 |

| T. | UT LT | Par. | 0 min VMcost | 0 min SLOv | 10 min VMcost | 10 min SLOv |
|---|---|---|---|---|---|---|
| LR | 60 20 | W:2 | 5602.30 | 0.71 | 4993.90 | 2.23 |
| | | W:3 | 5594.70 | 0.45 | 5057.50 | 1.76 |
| | | W:5 | 5604.70 | 0.26 | 5345.70 | 0.53 |
| | | W:10 | 5605.60 | 0.15 | 5471.60 | 0.26 |
| | 70 30 | W:2 | 4666.90 | 2.69 | 4144.70 | 9.70 |
| | | W:3 | 4691.00 | 1.88 | 4183.30 | 8.06 |
| | | W:5 | 4695.50 | 1.20 | 4261.50 | 5.87 |
| | | W:10 | 4525.30 | 0.97 | 4380.30 | 2.91 |
| | 90 10 | W:2 | 4734.60 | 1.42 | 4293.70 | 6.23 |
| | | W:3 | 4836.50 | 1.09 | 4581.60 | 3.52 |
| | | W:5 | 4901.80 | 0.73 | 4794.60 | 1.64 |
| | | W:10 | 4994.70 | 0.98 | 4944.00 | 1.51 |

- Como la carga tiene un comportamiento cíclico, difícilmente modelable mediante MA, ES y LR, interesa utilizar un histórico (“w” últimos valores) lo más breve posible.
- Para la configuración UT=70 y LT=30, MA(2) y ES(0.6) logran reducir el coste y la tasa de incumplimiento, comparadas con la mejor técnica reactiva.