Modelado matemático del rendimiento de un sistema informático

Félix Armando Fermín Pérez, Jorge Leoncio Guerra Guerra

fferminp@unmsm.edu.pe, jguerrag@unmsm.edu.pe

Universidad Nacional Mayor de San Marcos
Facultad de Ingeniería de Sistemas e Informática
Lima, Perú

Resumen: *Generalmente, la administración del rendimiento de sistemas informáticos es manual. La computación autonómica mediante la autoadministración trata de minimizar la intervención humana utilizando controladores autonómicos. Así, primero se modela matemáticamente el sistema en estudio y luego se diseña un controlador que gobierne su comportamiento. La determinación del modelo matemático de un servidor web se basa en un modelo de caja negra mediante la identificación de sistemas, que utiliza los datos recolectados durante el funcionamiento del sistema informático en estudio, y se almacenan en su propio log. En este caso, el modelo paramétrico ARX estimado se aproxima en un 87% a los datos medidos.*

Palabras clave: Computación autonómica, auto-administración, identificación de sistemas.

Abstract: *Performance management of a computer system is a manual work, generally. Autonomic computing through the self-administration tries to minimize human intervention using autonomics controllers. So, first the system under study is modeled mathematically and then a controller is designed to govern its behavior. The estimation of the mathematical model of a web server is based on a black box model by system identification, using data collected during operation of the computer system under study and stored in your own log. In this case, the estimated parametric ARX model approximates 87% of the measured data.*

Keywords: Autonomic computing, self-management, system identification.

1 Introducción

La tecnología influye en cada aspecto de la vida cotidiana. Las aplicaciones informáticas, por ejemplo, son cada vez más complejas, heterogéneas y dinámicas, pero también la infraestructura de información, como Internet que incorpora grandes cantidades de recursos informáticos y de comunicación, almacenamiento de datos y redes de sensores, con el riesgo de que se tornen frágiles, inmanejables e inseguras. Así, se hace necesario contar con administradores de servicios y de servidores informáticos, experimentados y dedicados, además de herramientas software de monitoreo y supervisión, para asegurar los niveles de calidad de servicio pactados. [Fermín12]

[Diao05] promueve la utilización de la computación autonómica mediante sistemas de control automático en lazo cerrado, reduciendo la intervención humana, que [Fox03] también ha identificado como parte principal del problema, debido al uso de procedimientos ad hoc donde la gestión de recursos está aun fuertemente dependiente del control y administración manual. Sobre la computación autonómica, [Kephart03] indica que ésta hace que un sistema informático funcione igual que el sistema nervioso autonómico humano cuando regula nuestra temperatura, respiración, ritmo cardíaco y otros sin que uno se halle siempre consciente de ello. Esto es, promueve la menor intervención humana en la administración del rendimiento de los sistemas informáticos tendiendo hacia la auto administración de los mismos. Tal auto administración se caracteriza por las propiedades de auto-configuración (self-configuration), auto curación (self-healing), auto optimización (self-optimization) y auto protección (self-protection).

En la visión de la computación autonómica, [Lalanda13], a los administradores humanos, simplemente especifican los objetivos de alto nivel del negocio, los que sirven como guía para los procesos autonómicos subyacentes. Así, los administradores humanos se concentran más fácilmente en definir las políticas del negocio, a alto nivel, y se liberan de tratar permanentemente con los detalles técnicos de bajo nivel, necesarios para alcanzar los objetivos, ya que estas tareas son ahora realizadas por el sistema autonómico mediante un controlador autonómico en lazo cerrado, que monitorea el sistema permanentemente utilizando los datos recolectados del propio sistema en funcionamiento y los compara con los propuestos por el administrador humano, para luego decidir a acción a realizar de acuerdo con lo programado previamente por el administrador humano. En [Kurian13], se mencionan proyectos desarrollados desde el 2003, tales como AUTONOMIA, FOCALE, PAWS, SASSY, e IPAutomata, entre otros.

De acuerdo con la teoría de control, el diseño de un controlador depende de un buen modelo matemático del sistema en estudio. En el caso de los sistemas informáticos, primero debe tenerse un modelo matemático para luego diseñar un controlador en lazo cerrado o realimentado, pero sucede que los sistemas informáticos son bastante complicados de modelar, ya que su comportamiento es altamente estocástico. [Hellerstein04] detalla que se ha utilizado la teoría de colas para modelar sistemas informáticos, tratándolos como redes de colas y de servidores, bastante bien, pero principalmente en el modelado del comportamiento estacionario y no cuando se trata de modelar el comportamiento muchas veces altamente dinámico de la respuesta temporal de un sistema informático en la zona transitoria, donde la tarea se complica.

De manera que en el presente artículo se trata el modelado matemático de un sistema informático mediante la identificación de sistemas, enfoque empírico donde según [Lung87] debe identificarse los parámetros de entrada y salida del sistema en estudio, basándose en los datos recolectados del mismo sistema en funcionamiento, para luego construir un modelo paramétrico, como el ARX por ejemplo, con las técnicas estadísticas de autoregresión. La sección 2 describe la teoría sobre el modelado matemático de un sistema informático. En la sección 3, se trata la metodología empleada para identificar un sistema informático, y, finalmente, en la sección 4, se describen las conclusiones y trabajos futuros.

2 Monitoreo del rendimiento

En la computación autonómica los datos obtenidos del monitoreo del rendimiento del sistema en estudio contribuye fundamentalmente en la representación del estado o del comportamiento del sistema, esto es, en el modelo matemático del mismo. De manera similar, [Lalanda13] menciona que conocer el estado del sistema desde las perspectivas funcionales y no funcionales es vital para llevar a cabo las operaciones necesarias que permitan lograr los objetivos en el nivel deseado y que tal monitoreo permite saber cuán bien lo está logrando. Generalmente, los datos del rendimiento se consiguen vía el log del sistema en estudio, con herramientas de análisis utilizando técnicas de análisis estadísticas, principalmente.

Entre las métricas del rendimiento inicialmente se encontraba la velocidad de procesamiento, pero al agregarse más componentes a la infraestructura informática, surgieron nuevas métricas, siendo las principales las que proporcionan una idea del trabajo realizado durante un periodo, la utilización de un componente, o el tiempo en realizar una tarea en particular, como por ejemplo, el tiempo de respuesta. [Lalanda13] indica que las métricas más populares son las siguientes:

- Número de operaciones en punto flotante por segundo (FLOPS), representa una idea del rendimiento del procesamiento, realiza comparaciones entre máquinas que procesan complejos algoritmos matemáticos con punto flotante en aplicaciones científicas.
- Tiempo de respuesta, representa la duración en tiempo que le toma a un sistema llevar a cabo una unidad de procesamiento funcional. Se le considera como una medición de la duración en tiempo de la reacción a una entrada determinada y es utilizada principalmente en sistemas interactivos. La sensibilidad es también una métrica utilizada especialmente en la medición de sistemas en tiempo real. Consiste en el tiempo transcurrido entre el inicio y fin de la ejecución de una tarea o hilo.
- Latencia, medida del retardo experimentado en un sistema, generalmente se le utiliza en la descripción de los elementos de comunicación de datos, para tener una idea del rendimiento de la red. Toma en cuenta no solo el tiempo de procesamiento de la CPU, sino también los retardos de las colas durante el transporte de un paquete de datos, por ejemplo.
- Utilización y carga, métricas entrelazadas y utilizadas para comprender la función de administración de recursos y proporcionan una medida de cuan bien se están utilizando los componentes de un sistema y se describe como un porcentaje de utilidad. La carga mide el trabajo realizado por el sistema y usualmente es representado como una carga promedio en un periodo de tiempo.

Existen muchas otras métricas de rendimiento:

- Número de transacciones por unidad de coste.
- Función de confiabilidad, tiempo en el que un sistema ha estado funcionando sin fallar.
- Función de disponibilidad, indica que el sistema está listo para ser utilizado cuando sea necesario.
- Tamaño o peso del sistema, indica la portabilidad.
- Rendimiento por vatio, representa la tasa de cómputo por vatio consumido.
- Calor generado por los componentes, ya que en sistemas grandes es costoso un sistema de refrigeración.

Todas ellas, entre otras más, permiten conocer mejor el estado no funcional de un sistema o proporcionar un medio para detectar un evento que ha ocurrido y ha modificado el comportamiento no funcional. Así, en el presente caso se ha elegido como métrica al tiempo de respuesta, ya que el sistema informático en estudio es un servidor web, por esencia de comportamiento interactivo, de manera que lo que se mide es la duración de la reacción a una entrada determinada.

3 Modelado matemático

En la computación autonómica, basado en la teoría de control realimentado, para diseñar un controlador autonómico, primero debe hallarse el modelo matemático del sistema en estudio, tal como se muestra en la Figura 1. El modelo matemático de un servidor informático se puede hallar mediante dos enfoques: uno basado en las leyes básicas, y otro en un enfoque empírico denominado identificación de sistemas. [Parekh02] indica que en trabajos previos se ha tratado de utilizar principios, axiomas, postulados, leyes o teorías básicas para determinar el modelo matemático de sistemas informáticos, pero sin éxito, ya que es difícil construir un modelo debido a su naturaleza compleja y estocástica. Además es necesario tener un conocimiento detallado del sistema en estudio, más aún, cuando cada cierto tiempo se va actualizando las versiones del software, y, finalmente, que en este enfoque no se considera la validación del modelo.

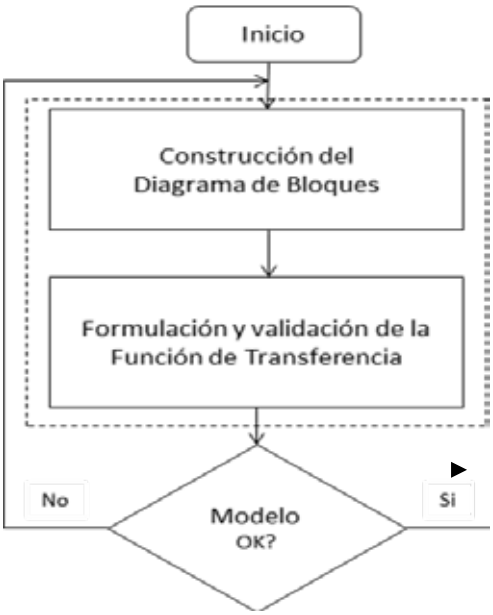

Figura 1. Modelado matemático basado en la teoría de control. Adaptado de [Hellerstein04].

En contraste, según Ljung (1987) la identificación de sistemas es un enfoque empírico donde debe identificarse los parámetros de entrada y salida del sistema en estudio, para luego construir un modelo paramétrico, como el ARX por ejemplo, mediante técnicas estadísticas de autoregresión. Este modelo o ecuación paramétrica relaciona los parámetros de entrada y de salida del sistema de acuerdo con la siguiente ecuación:

$$y(k+1) = Ay(k) + Bu(k) \qquad (1)$$

donde $y(k)$: variable de salida

$u(k)$: variable de entrada

A, B : parámetros de autoregresión

k : muestra k-ésima.

Este enfoque empírico trata al sistema en estudio como una caja negra, de manera que no afecta la complejidad del sistema o la falta de conocimiento experto, incluso cuando se actualicen las versiones del software bastaría con estimar nuevamente los parámetros del modelo. Así, para un servidor web Apache, la ecuación paramétrica relaciona el parámetro entrada, Max Clients (MC), un parámetro de configuración del servidor web Apache que determina el número máximo de conexiones simultáneas de clientes que pueden ser servidos; y el parámetro de salida Tiempo de respuesta (TR), que indica lo rápido que se responde a las solicitudes de servicio de los clientes del servidor, ver Figura 2.

Figura 2. Entrada y salida del sistema a modelar. [Elaboración propia]

En [Hellerstein04], se propone realizar la identificación de sistemas informáticos, como los servidores web, de la siguiente manera:

1. Especificar el alcance de lo que se va a modelar en base a las entradas y salidas consideradas.

2. Diseñar experimentos y recopilar datos que sean suficientes para estimar los parámetros de la ecuación diferencia lineal del orden deseado.

3. Estimar los parámetros del modelo utilizando las técnicas de mínimos cuadrados.

4. Evaluar la calidad de ajuste del modelo y si la calidad del modelo debe mejorarse, entonces debe revisarse uno o más de los pasos anteriores.

4 Experimento

La arquitectura del experimento implementado, se muestra en la Figura 3.

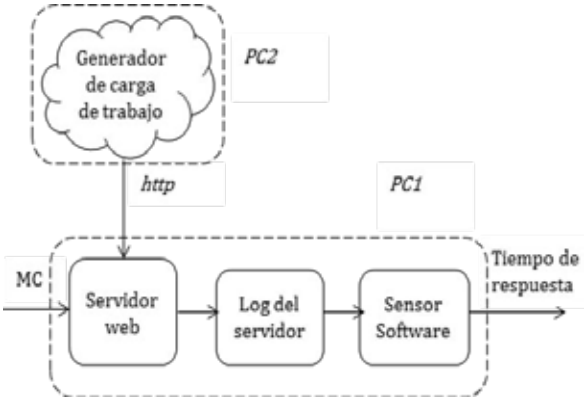

Figura 3. Arquitectura para la identificación del modelo de un servidor web. [Elaboración propia]

La computadora personal PC1 es el servidor web Apache. Asimismo, contiene el sensor software que recoge los tiempos de inicio y fin de cada http que ingresa al servidor web, datos que se almacenan en el log del mismo servidor, y luego se realiza el cálculo del tiempo de respuesta de cada http completado en una unidad de tiempo y el tiempo de respuesta promedio del servidor. El servidor web, por sí mismo no hace nada, por ello, según [Liu01], en la computadora personal PC2, un generador de carga de trabajo simula la actividad de los usuarios que desean acceder al servidor web; en este caso se utilizó el JMeter una aplicación generadora de carga de trabajo y que forma parte del proyecto Apache.

En [Gandhi02], se establece que la operación del servidor web no solo depende de la actividad de los usuarios, sino también de la señal de entrada MaxClients (MC), y que en el modelado matemático del comportamiento, MC debe tomar la forma de una sinusoide discreta variable para que, de acuerdo con [Ljung87], posea componentes de frecuencia altas y bajas, que excite toda la dinámica del sistema dentro del tiempo en el que el servidor funciona, mientras también se estimula al servidor web con solicitudes http del generador de carga de trabajo (PC2), simulando a los usuarios que tratan de acceder al servidor. Un valor alto de MC permite que el servidor web procese más solicitudes http de los usuarios, pero si es demasiado

grande utiliza recursos en exceso que finalmente degrada el rendimiento para todos los clientes. En este caso, de acuerdo con [Gandhi02], y con la finalidad de obtener un modelo cercano al lineal, MC se situó entre valores mínimo y máximo, fuera de los límites de corte y saturación del servicio, como se aprecia en la Figura 4. Luego, con la actividad del servidor web almacenada en su log, un sensor software calcula los valores de la señal de salida Tiempo de Respuesta (TR).

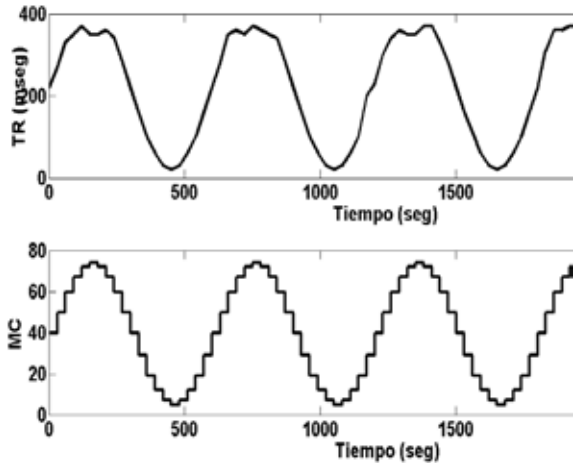

Figura 4. Señal de entrada MaxClients MC y señal de salida Tiempo de respuesta TR.

Con los datos de MC y TR obtenidos se estiman los parámetros de regresión A y B de la ecuación paramétrica ARX, haciendo uso de las técnicas de mínimos cuadrados, implementadas en el ToolBox Identificación de Sistemas del Matlab [Ljung87], logrando la siguiente ecuación paramétrica:

$$TR(k+1) = 0.06545TR(k) + 4.984MC(k+1)$$

Se puede observar que el Tiempo de respuesta actual depende del Tiempo de respuesta anterior y del parámetro de entrada MaxClients. El modelo hallado es evaluado utilizando la métrica r2, calidad de ajuste, del ToolBox utilizado, que indica el porcentaje de variación respecto a la señal original. En el caso de estudio, el modelo hallado tiene una calidad de ajuste del 87%, lo que se puede considerar como un modelo aceptable. En la Figura 5, puede observarse la gran similitud entre la señal de salida medida y la señal de salida estimada.

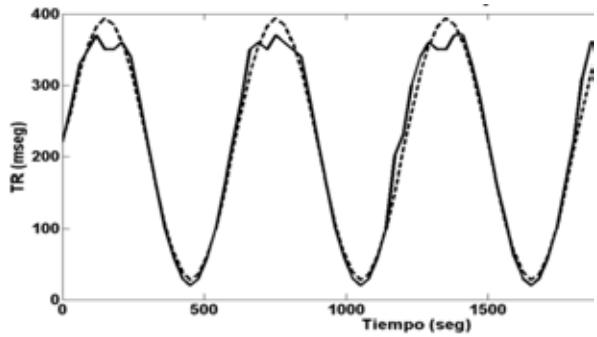

Figura 5. TR medida y TR estimada.

5 Conclusiones

La identificación del comportamiento de un sistema informático tratado como una caja negra es posible, para ello debe simularse el funcionamiento del mismo con hardware y herramientas software como el Jmeter, por ejemplo.

El modelo matemático determinado en base a los datos recopilados de su propio log, se aproxima bastante al modelo real, en este caso se obtuvo un 87% de calidad de ajuste.

El sensor software implementado ha permitido calcular el tiempo de respuesta en base a los datos almacenados en el log del mismo servidor en estudio.

De los datos tomados, se observa que los sistemas informáticos poseen un comportamiento cercano al lineal solo en tramos, por lo que se sugiere experimentar con modelos matemáticos no lineales para comparar la calidad de ajuste de ambos.

Como trabajo futuro, se plantea diseñar un controlador autonómico basado en el modelo lineal ARX, aunque más adelante se planteará el diseño de controladores autonómicos no lineales para servidores informáticos en general, ya que el comportamiento temporal no lineal, hace adecuada la utilización de técnicas de inteligencia artificial como la lógica difusa y las redes neuronales.

Referencias bibliográficas

[Diao05] Diao, Y. Hellerstein, J. Parekh, S. Griffith, R. Kaiser, G. Phung, D. *A Control Theory Foundation for Self-Managing Computing Systems*. IEEE. DOI:10.1109/JSAC.2005.857206

[Fermín12] Fermín, F. *La Teoría de Control y la Gestión Autónoma de Servidores Web*. Memorias del IV Congreso Internacional de Computación y Telecomunicaciones. ISBN 978-612-4050-57-2. 2012. Lima.

[Fox03] Fox, A. Patterson, D. *Self-Repairing Computers*. Scientific American, Vol. 288, Issue 6

[Gandhi02] Gandhi, N. Tilbury, M. Diao, J. Hellerstein, J. Parekh, S. *MIMO control of an Apache web server: modeling and controller design*. American Automatic Control Council. DOI:10.1109/ACC.2002.1025440.

[Hellerstein04] Hellerstein, J. Diao, Y. Parekh, S. Tilbury, D. Feedback Control of Computing Systems. Hoboken, New Jersey: John Wiley & Sons, Inc.

[Kephart03] Kephart, J. Chess, D. *The vision of autonomic computing.* IEEE Computer Society.

[Kurian13] Kurian, D. Raj, P. *Autonomic Computing for Business Applications*. International Journal of Advanced Computer Science and Applications, Vol. 4 No. 8.

[Lalanda13] Lalanda, P. McCann, J. Diaconescu, A. A*utonomic Computing. Principles, Design and Implementation.* London: Springer-Verlag.

[Liu01] Liu, Z. Niclausse, N. Jalpa-Villanueva, C. *Traffic model and performance evaluation of web servers*.

Performance Evaluation, an International Journal. Elsevier Science B.V.

[Ljung87] Ljung, L. System Identification: Theory for the User. Englewood Cliffs, New Jersey: PTR Prentice Hall.

[Parekh02] Parekh, S. Gandhi, N. Hellerstein, J. Tilbury, D. Jayram, T. Bigus, J. *Using control theory to achieve service level objectives in performance management*. Real-Time Systems.