Trabajo Fin de Grado
Ingeniería de Organización Industrial

Modelado y control de un centro de proceso de datos en Anylogic

Autor: Miguel García Ruiz

Tutores: Daniel Limón Marruedo y Alfonso Daniel Carnerero Panduro

Dpto. Ingeniería de Sistemas y Automática
Escuela Técnica Superior de Ingeniería
Universidad de Sevilla

Sevilla, 2020

Trabajo Fin de Grado
Ingeniería de Organización Industrial

Modelado y control de un centro de proceso de datos en Anylogic

Autor:
Miguel García Ruiz

Tutores:
Daniel Limón Marruedo
Catedrático

Alfonso Daniel Carnerero Panduro
Personal Investigador en Formación

Dpto. Ingeniería de Sistemas y Automática
Escuela Técnica Superior de Ingeniería
Universidad de Sevilla
Sevilla, 2020

Trabajo Fin de Grado: Modelado y control de un centro de proceso de datos en Anylogic

Autor: Miguel García Ruiz

Tutores: Daniel Limón Marruedo y Alfonso Daniel Carnerero Panduro

El tribunal nombrado para juzgar el Proyecto arriba indicado, compuesto por los siguientes miembros:

Presidente:

Vocales:

Secretario:

Acuerdan otorgarle la calificación de:

Sevilla, 2020

El Secretario del Tribunal

A mi familia

A mis maestros

Agradecimientos

Quiero utilizar estas líneas para dar gracias a todo aquel que de alguna manera me ha formado para llegar hasta aquí hoy.

A mi familia, por educarme desde pequeño en el esfuerzo y la humildad, por enseñarme a valorar las alegrías y a no venirme abajo ante las dificultades. Por aguantarme más que nadie.

Gracias a amigos, conocidos y desconocidos que durante este tiempo se han preocupado por este TFG motivándome a hacerlo.

Gracias a Daniel Limón por las muchas veces que me ha guiado a lo largo de mi vida académica, si no fuera por él nunca habría sido ingeniero. Gracias también a Daniel Carnerero, siempre amable y dedicado, valoro de corazón su implicación en este trabajo.

Miguel García Ruiz

Sevilla, 2020

Resumen

Los Centros de Procesamiento de Datos (CPD) son las infraestructuras físicas encargadas de albergar los recursos necesarios para almacenar y procesar información a gran escala en internet.

En los últimos años se está produciendo un incremento exponencial de los servicios de red, la computación en la nube o el internet de las cosas, lo que conlleva lógicamente un crecimiento de estas infraestructuras.

Satisfacer las necesidades de operación de los servidores y a la vez mantener una refrigeración de los equipos que combata el sobrecalentamiento de los mismos implica un enorme consumo de energía por parte de los centros de datos.

Es por ello que existen multitud de estudios y modelos diferentes para predecir y optimizar el consumo energético de los centros de datos.

En este trabajo analizaremos el diseño de un modelo que controle tanto el consumo de los equipos de procesamiento como la refrigeración de estos.

Abstract

Data Centers are the physical infrastuctures in charge of hosting the necessary resources to store and process large-scale data on the internet.

In recent years, there has been an exponential increase in network services, cloud computing or the Internet of Things, which logically leads to a growth in these infrastructures.

Meet the operational needs of the servers and at the same time keep cooling the equipment to struggle against his overheating involves an enormous consumption of power by the data centers.

That is why there are many different studies and models to predict and optimize the energy consumption of data centers.

In this work we will analyze the design of a model that controls both the consumption of the processing equipment and its cooling.

Índice

ÍNDICE DE FIGURAS

1. INTRODUCCIÓN

Los centros de procesamiento de datos son entidades físicas cuya labor es almacenar, procesar y distribuir datos en red entre diferentes usuarios.

En los últimos años se está produciendo un crecimiento exponencial de los servicios informáticos a los que la sociedad demanda acceso, lo que conlleva un aumento proporcional del volumen de datos que estos centros deben tratar. De esta manera, es necesario realizar diseños cada vez más potentes y eficientes para evitar así un consumo excesivo de energía y de espacio, garantizando además una cierta velocidad de procesado de las tareas.

El objeto de este trabajo es presentar el concepto del centro de procesamiento de datos, sus características y contexto actual, y basándonos en estudios recientes diseñar un modelo teórico que defina el sistema en términos térmicos y de procesamiento de las tareas, aplicándole leyes de control que nos permitan satisfacer ciertas condiciones de funcionamiento.

Posteriormente, abordamos la implementación del modelo expuesto en la herramienta de simulación Anylogic©, explicando paso a paso el proceso y cómo los controladores actúan sobre este alterando ciertas variables para obtener determinados valores de otras.

Finalmente, se presentan los resultados obtenidos mediante la simulación acompañados de algunas interpretaciones de los mismos y sugerencias para próximos estudios.

2. CENTROS DE DATOS

2.1 Contexto histórico

Desde el surgimiento de internet, los teléfonos inteligentes y otras nuevas tecnologías, estamos constantemente conectados a una red y demandando intercambiar información mediante nuestros ordenadores, teléfonos, videoconsolas o televisores. Aunque hoy en día todavía existen documentos en papel, los hemos sustituido en la mayoría de los trámites por correos electrónicos, páginas web, archivos PDF y otros archivos digitalizados generados a través de software y reproducidos en pantallas de ordenador. Incluso los libros han pasado del papel a ser imágenes en nuestros lectores electrónicos.

Hoy en día es necesario intercambiar datos electrónicos para casi cualquier tipo de transacción ya sea personal o profesional.

Debido a esta demanda masiva de entrega prácticamente instantánea de información digital surgió la necesidad de concentrar equipos informáticos y de redes que puedan gestionar las solicitudes y entregarlas a sus destinatarios. De esta manera nacieron los centros de datos modernos.

2.2 Qué es es un centro de datos

Los centros de datos existen de alguna manera desde la aparición de las computadoras. Son el lugar donde los equipos de computación y redes se concentran con el propósito de recopilar, almacenar, procesar, distribuir o permitir el acceso a grandes volúmenes de datos.

Con el paso de los años los equipos se han ido haciendo cada vez más pequeños y baratos y las necesidades de procesamiento de datos empezaron a crecer exponencialmente, se empezaron a conectar múltiples servidores en red para aumentar a capacidad de procesamiento. Se conectan a redes de comunicación de manera que la gente pueda acceder a ellos o a su información de forma remota.

Gran cantidad de estos servidores agrupados se pueden encontrar albergados en una habitación, un edificio entero o incluso un conjunto de edificios.

Debido a esta alta concentración de servidores, normalmente apilados en estantes y colocados en fila, estos centros de datos son llamados granjas de servidores a menudo. Estos centros proporcionan servicios importantes como son el almacenamiento de datos, copias de seguridad y recuperación, gestión de datos y redes.

A día de hoy casi todas las empresas y entidades gubernamentales necesitan tener su propio centro de datos o al menos acceso a uno ajeno. Algunos los construyen y mantienen internamente, otros alquilan servidores en instalaciones de uso compartido (también llamados colos) y otros usan servicios públicos basados en la nube en servidores como Amazon, Microsoft, Sony o Google.

Los colos y el resto de grandes centros de datos comenzaron a surgir a finales de la década de 1990 y principios de la década de 2000, algún tiempo después de que se generalizara el uso de Internet. Los centros de datos de algunas grandes empresas están espaciados en todo el planeta para satisfacer la necesidad constante de acceso a grandes cantidades de información. Según los informes, alrededor del mundo hay más de 3 millones de centros de datos de diversas formas y tamaños.

2.3 Por qué son necesarios los centros de datos

A pesar de que el hardware se vuelve cada vez más pequeño, rápido y potente, el aumento de los servicios informáticos de los que hacemos uso en nuestra sociedad, hace cada vez más necesario el acceso a una gran cantidad de datos, y la demanda de capacidad de procesamiento, espacio de almacenamiento e información en general está creciendo y constantemente amenazando con superar la capacidad de entrega de las empresas.

Cualquier entidad que genere o utilice datos necesita centros de datos en algún nivel, incluidos organismos gubernamentales, organismos educativos, empresas de telecomunicaciones, instituciones financieras, minoristas de todos los tamaños y proveedores de información en línea y servicios de redes sociales como Google o Facebook. La falta de acceso rápido y de confianza a los datos puede significar la incapacidad de proporcionar servicios fundamentales o la pérdida de satisfacción e ingresos del cliente.

Toda esta información debe almacenarse en algún lugar. Y en estos días, cada vez más servicios se están moviendo también hacia la nube, lo que significa que en lugar de ejecutarlas o almacenarlas en nuestros propios ordenadores de casa o del trabajo, estamos accediendo a ellas a través de los servidores de los proveedores de la nube. Muchas empresas también están trasladando sus aplicaciones profesionales a servicios en la nube para reducir el coste de ejecutar sus propias redes y servidores informáticos centralizados.

La nube no significa que las aplicaciones y los datos no se alojen en un hardware informático. Simplemente significa que alguien externo al usuario mantiene ese hardware y el software en ubicaciones remotas donde sus clientes pueden acceder a ellos a través de Internet. Esas ubicaciones son centros de datos. [1]

2.4 Escala y diseño de los centros de datos

Cuando se piensa en los centros de datos, mucha gente imagina enormes almacenes llenos de estantes de servidores, parpadeando y pitando, cables de un lado a otro. Y en algunos casos estaríamos en lo cierto. Pero existen de todas las formas, tamaños y configuraciones. Van desde unos pocos servidores en una habitación hasta enormes estructuras independientes que miden cientos de miles de metros cuadrados con decenas de miles de servidores y un hardware que lo acompaña. Sus tamaños y los tipos de equipos que contienen varían según las necesidades de la entidad o entidades a las que dan soporte.

Existen diferentes tipos, incluidos proveedores de nube privada como los colos, proveedores de nube pública como Amazon y Google, centros de datos privados de empresas y centros de datos gubernamentales como los de la NSA o algunas instalaciones de investigación científica.

Estas instalaciones no cuentan con personal como oficinas con una persona por computadora, sino con un número menor de personas que monitorizan una gran cantidad de computadoras y dispositivos de red, así como la energía, la refrigeración y otras instalaciones necesarias del edificio. Algunos son tan grandes que los empleados se desplazan en scooters o bicicletas. Los suelos tienen que soportar más peso que un edificio de oficinas típico debido a que el equipo puede llegar a ser muy pesado. También deben tener techos altos para poder acomodar cosas como estantes altos, suelos elevados y cableado suspendido en el techo, entre otras cosas.

Las grandes empresas con una fuerte presencia en internet tienen grandes centros de datos ubicados por todo el mundo, incluidos Google, Facebook, Microsoft, AOL y Amazon.

Google tiene trece grandes centros de datos, entre los que se incluyen ubicaciones en el condado de Douglas, Ga .; Lenoir, N.C .; Condado de Berkeley, S.C .; Council Bluffs, Iowa; Condado de Mayes, Okla .; The Dalles, Ore .; Quilicura, Chile; Hamina, Finlandia; St. Ghislain, Bélgica; Dublín, Irlanda; Hong Kong, Singapur y Taiwán; así como muchos pequeños centros de datos, algunos incluso en lugares de ubicación conjunta. El gigante tecnológico también es propenso a experimentar con el diseño. Por ejemplo, en 2005, Google utilizó contenedores de envío que contenían equipos de servidor en sus centros de datos, y desde entonces ha pasado a otros diseños personalizados.

La configuración de los servidores, la topología de la red y el equipo de soporte pueden variar mucho según el tipo de empresa, el propósito, la ubicación, la tasa de crecimiento y el concepto de diseño inicial del centro de datos. Su diseño puede afectar en gran medida a la eficiencia del flujo de datos y las condiciones ambientales dentro del centro. Algunos casos pueden dividir sus servidores en grupos por funciones, como la separación de

servidores web, servidores de aplicaciones y servidores de bases de datos, y algunos pueden hacer que cada uno de sus servidores realice múltiples tareas. No hay reglas estrictas ni estándares oficiales.

Algunos grupos están tratando de crear ciertas pautas. La Asociación de la Industria de Telecomunicaciones desarrolló un estándar de clasificación de nivel de centro de datos en 2005 llamado proyecto TIA-942, que identificó cuatro categorías diferentes de centros de datos, clasificadas por métricas como redundancia y nivel de tolerancia a fallas. Éstas incluyen:

- Nivel 1: infraestructura básica del sitio con una única ruta de distribución que no tiene redundancia incorporada.
- Nivel 2: infraestructura de sitio redundante con una única ruta de distribución que incluye componentes redundantes.
- Nivel 3: infraestructura de sitio que se puede mantener de manera concurrente que tiene múltiples rutas, solo una de las cuales está activa a la vez.
- Nivel 4: infraestructura de sitio tolerante a fallas que tiene múltiples rutas de distribución activas para mucha redundancia.

En teoría, los sitios que se identifican con las categorías de nivel 1 y 2 deben cerrarse ocasionalmente por mantenimiento, mientras que los sitios de nivel 3 y 4 deben poder mantenerse activos durante el mantenimiento y otras interrupciones. Un nivel mayor se traduce tanto en más fiabilidad (lo que significa menos tiempo de inactividad potencial) como en un mayor coste.

No todos los centros de datos siguen estos estándares. Los centros de datos actuales son un fenómeno tan novedoso que no hay códigos de construcción específicos para ellos en la mayoría de las áreas en este momento. Generalmente se agrupan en algún otro tipo genérico.

Sus diseños, equipos y necesidades evolucionan constantemente, pero hay algunos elementos comunes que se encontraran en muchos centros de datos.

2.5 Hardware

Una característica física común de los centros de datos son los conjuntos de servidores interconectados (*clusters*). Pueden ser todos muy similares, apilados de forma ordenada en estantes abiertos o armarios cerrados de igual altura, ancho y profundidad, o podría haber un montón de diferentes tipos, tamaños y antigüedad de máquinas coexistiendo, como pequeños servidores modernos planos junto a viejos y voluminosos.

Cada servidor es una computadora de alto rendimiento con memoria, espacio de almacenamiento, un procesador o procesadores y capacidad de entrada/salida, algo así como una versión mejorada de un ordenador personal, pero con un procesador más rápido y potente y mucha más memoria y, por lo general, sin un monitor, teclado u otros periféricos. Los monitores pueden existir en una ubicación centralizada, cercana o en una sala de control separada, para monitorizar grupos de servidores y equipos relacionados.

Los servidores pueden estar dedicados a una sola tarea o ejecutar muchas aplicaciones diferentes. Algunos servidores en centros de datos de ubicación conjunta están dedicados a clientes particulares. Algunos incluso son virtuales en lugar de físicos, una nueva tendencia que reduce la cantidad necesaria de servidores físicos. También es posible, cuando solicita algo a través de internet, que varios servidores estén trabajando juntos para entregarle el contenido.

2.6 Redes, software y control ambiental

Los equipos de redes y comunicación son totalmente necesarios en un centro de datos para poder mantener una red de gran ancho de banda para la comunicación con el mundo exterior, y entre los servidores y otros equipos dentro del centro de datos. Esto incluye algunos componentes como enrutadores, conmutadores, controladores de interfaz de red (NIC) de los servidores y, en algunos casos, kilómetros y kilómetros de cableado. El cableado puede ser de varias formas, incluyendo par trenzado (cobre), coaxial (también cobre) y fibra óptica (vidrio o plástico). Los tipos de cable y sus diversos subtipos afectarán la velocidad a la que fluye la información a través

del centro de datos.

Todo ese cableado debe estar además organizado. Recorre la superficie por encima de bandejas colgadas del techo o unidas a la parte superior de los estantes, o se ejecuta debajo de un suelo elevado, a veces en bandejas debajo del suelo. Se utilizan una codificación de color y un etiquetado meticuloso para identificar las diversas líneas de cableado. Los suelos elevados de los centros de datos tienen normalmente paneles que se pueden levantar para acceder al cableado y otros equipos. Las unidades de refrigeración y los equipos de energía a veces se encuentran también debajo del suelo.

Otros equipos importantes del centro de datos son los dispositivos de almacenamiento (como unidades de disco duro, unidades de estado sólido y unidades de cinta robótica), fuentes de alimentación ininterrumpida (UPS), baterías de reserva, generadores de reserva y otros equipos relacionados con la energía.

Los centros de datos también disponen de equipos para controlar la temperatura y la calidad del aire, aunque los métodos y tipos de equipos varían de un caso a otro. Pueden incluir ventiladores, controladores de aire, filtros, sensores, aires acondicionados de sala de computadoras (CRAC), enfriadores, tuberías de agua y tanques de agua. Algunos sitios también colocan barreras de plástico o metal o usan elementos como gabinetes de servidores de chimenea para controlar el flujo de aire caliente y frío para evitar el sobrecalentamiento de los equipos informáticos.

Por supuesto, se necesita también software para dar uso a todo este hardware, incluidos los diversos sistemas operativos y aplicaciones que se ejecutan en los servidores, software de agrupación en cluster como MapReduce o Hadoop de Google para permitir que el trabajo se distribuya en cientos o más máquinas, programas de conexión a Internet para controlar las conexiones de red, las aplicaciones de monitorización del sistema y el software de virtualización como VMware para ayudar a reducir la cantidad de servidores físicos.

2.7 Problemas a los que se enfrentan los centros de datos

Los centros de datos se esfuerzan en proporcionar un servicio rápido e ininterrumpido. Los fallos de los equipos, las interrupciones de la comunicación o el suministro eléctrico, la congestión de la red y otros problemas que impiden que las personas accedan a sus datos y aplicaciones deben abordarse de forma inmediata. Debido a la constante demanda de acceso instantáneo, se espera que los centros de datos funcionen 24 horas al día los 365 días del año, lo que genera una gran cantidad de problemas.

Las necesidades de red de un centro de datos son muy diferentes de las de, por ejemplo, un edificio de oficinas lleno de trabajadores. Las redes de centros de datos son enormemente potentes. Las redes de fibra óptica de Google envían datos hasta 200,000 veces más rápido que el servicio de internet ofrecido para uso doméstico. Sin embargo, Google tiene que manejar diariamente más de 3 mil millones de solicitudes de motores de búsqueda, indexar miles de millones de páginas web y manejar y almacenar correo electrónico para cientos de millones de usuarios, entre sus muchos otros servicios.

Casi nadie tiene tanto tráfico como Google, pero todos los centros de datos probablemente verán crecer más y más su uso. Necesitan tener capacidad de escalar sus redes para aumentar el ancho de banda y mantener la fiabilidad. Lo mismo ocurre con los servidores, que se pueden ampliar para aumentar la capacidad del centro de datos. La red existente necesita poder manejar la congestión controlando el flujo adecuadamente. Una red solo será tan rápida como su componente más lento. Los acuerdos de nivel de servicio (SLA) con los clientes también deben cumplirse, y a menudo incluyen elementos como el rendimiento y el tiempo de respuesta.

Hay varios puntos de posible fallo. Los servidores o equipos de red pueden apagarse, los cables pueden fallar o los servicios que ingresan desde el exterior, como la alimentación y la comunicación, pueden verse interrumpidos. Los sistemas deben estar en su lugar para monitorizar, responder y notificar al personal sobre cualquier problema que surja. La planificación de la recuperación ante desastres es de vital importancia en caso de fallos importantes, pero también se deben manejar los problemas menores.

2.8 Enfriamiento y consumo energético

Los centros de datos deben tener controles ambientales estrictos y ser capaces de absorber o generar grandes cantidades de energía para mantener los procesos en funcionamiento. Y esto conlleva un elevado coste.

Debido a que los servidores y otros equipos no funcionan adecuadamente en temperaturas extremas, la mayoría de los centros de datos tienen enormes sistemas de enfriamiento que consumen grandes cantidades de energía, siendo a veces necesario recurrir a técnicas como la refrigeración líquida para aliviar grandes cantidades de calor. Los sensores deben estar en su lugar para monitorizar las condiciones ambientales para que se puedan hacer ajustes.

No solo la temperatura es un problema. Otros factores como la humedad deben mantenerse bajo control. En 2011, en uno de los centros de datos de Facebook, la lluvia produjo goteras en el edificio que provocaron que algunos servidores se reiniciaran y se cortaran las fuentes de alimentación. A consecuencia de estos hechos modificaron su sistema de administración de edificios e hicieron que los servidores fueran más resistentes a la intemperie.

Los racks de servidores a menudo se organizan en filas que crean pasillos donde los servidores están uno frente al otro para controlar el flujo de aire y la temperatura de manera más eficiente. El pasillo donde se enfrentan es el pasillo frío (Figura 1), y el aire en el pasillo caliente se canaliza a conveniencia.

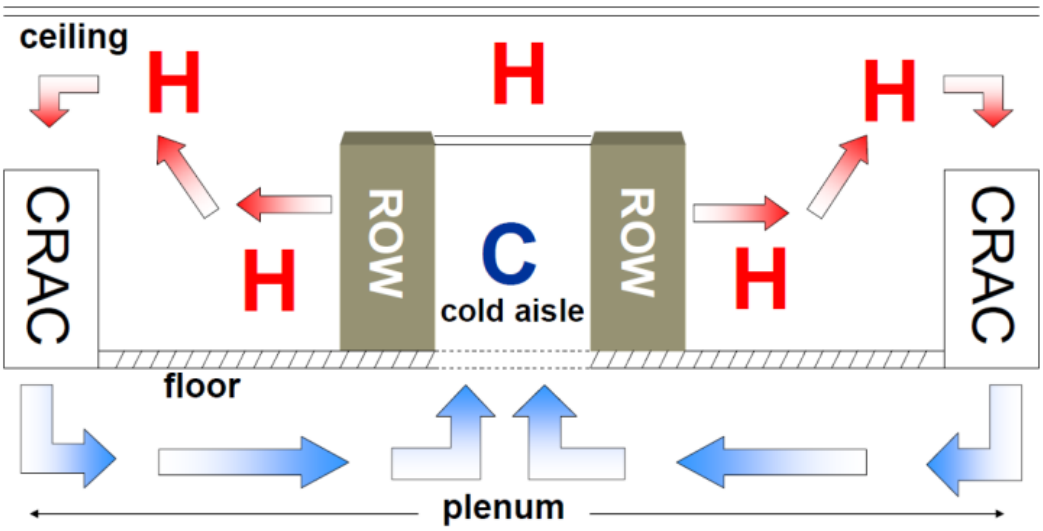

Figura 1. Unidad asilada en frío.

El consumo de energía es otra cuestión importante de la que preocuparse. Es absolutamente necesario que estas instalaciones tengan acceso constante a la energía adecuada, algunas incluso tienen sus propias subestaciones de energía. Una medida utilizada para juzgar la eficiencia energética del centro de datos es la eficacia del uso de energía (PUE). Se trata de dividir el uso de energía únicamente para fines de cálculo entre el uso total de energía (servidores y sistema de refrigeración). La puntuación PUE de Yahoo, Google y Facebook es de alrededor de 1.1 o 1.2 para algunos de sus grandes centros de datos, mientras que la mayoría de la industria se sitúa en torno a un 2.0. Eso significa que la mitad de la energía se destina a la informática y la otra mitad a otras tareas o se desperdicia.

Se están tomando multitud de medidas para reducir el consumo energético de los centros de datos y otras necesidades de recursos. Las salas de servidores solían mantenerse a unos 16ºC, pero la tendencia en los centros de datos con mayor eficiencia energética es mantenerlos a unos 27ºC, al menos en el pasillo frío, aunque no todos han adoptado esto práctica. Aparentemente, los servidores funcionan bien a esta temperatura, y requieren un menor gasto energético para la refrigeración.

En los últimos años existe una tendencia creciente a usar refrigeración al aire libre, que extrae aire del exterior en lugar de utilizar muchas unidades de aire acondicionado y enfriadores que requieren mucha energía. Otra

tendencia es ubicar los centros de datos cerca de fuentes de agua que estén preparadas para poder reciclarse para su uso en refrigeración, como el centro de datos de Google en Finlandia, que utiliza agua de mar. Otra es ubicar centros de datos en climas fríos.

Realizar cambios en el equipo informático real también puede ser de gran utilidad. Muchos componentes en los centros de datos pierden energía, lo que significa que parte de la energía que consumen nunca llega al procesamiento real: se desperdicia. Reemplazar servidores antiguos por modelos más nuevos y eficientes energéticamente es obviamente una mejora. Pero el equipo también se puede rediseñar para requerir menos consumo energético. La mayoría de los centros de datos usan servidores tradicionales y otros equipos, pero Google y Facebook usan servidores personalizados. El modelo de Google fue diseñado para eliminar componentes innecesarios como tarjetas gráficas y minimizar la pérdida de energía en la fuente de alimentación y el regulador de voltaje. Los paneles que contienen el logotipo del fabricante se omiten para permitir un mejor flujo de aire hacia y desde los componentes, y la compañía fabrica algunos de sus propios equipos de red.

Además, los procesadores y los ventiladores también pueden reducir la velocidad cuando no son necesarios. Los servidores más eficientes también tienden a generar menos calor, reduciendo todavía más el consumo de energía necesario para la refrigeración. Los servidores ARM de baja potencia, creados originalmente para dispositivos móviles pero rediseñados para usos del servidor, están llegando también a los centros de datos.

El uso de aplicaciones fluctúa dependiendo de lo que se está haciendo y en qué momento en varios software y aplicaciones web, cualquiera de los cuales tiene diferentes necesidades de recursos. La gestión de recursos de aplicaciones es importante para aumentar la eficiencia y reducir el consumo. El software se puede desarrollar a medida para que funcione de manera más eficiente con la arquitectura del sistema. La virtualización del servidor también puede reducir el consumo de energía al reducir el número de servidores en ejecución.

2.9 Sistemas de refrigeración de centros de datos

El consumo energético destinado a la refrigeración de los centros de datos puede superar el 40% del consumo total de energía, por lo que la optimización de los sistemas de refrigeración en grandes centros de datos es esencial para reducir los costes de operación.

2.9.1 Sistemas refrigerados por aire

El equipo de refrigeración debe proporcionar aire con la capacidad de enfriamiento y la distribución adecuada. Hay varios parámetros que pueden influir en la eficiencia de la refrigeración, como la altura del techo, donde puede producirse la estratificación del aire caliente, la altura del suelo elevado/caída del techo, que es importante para lograr una distribución correcta del aire entre el equipo de TI y dirección del flujo de aire en la habitación.

Se identifican dos problemas principales de la distribución de aire en los centros de datos, el aire de derivación y el aire de recirculación.

El aire de recirculación ocurre cuando el flujo de aire hacia el equipo no es suficiente y parte del aire caliente se recircula, lo que resulta en una diferencia considerable entre la temperatura de entrada en la parte inferior y la parte superior del estante. La derivación del aire frío ocurre debido a un alto caudal o fugas a través de la ruta del aire frío. En este caso, parte de la corriente de aire frío salta directamente del suministro de aire frío al aire de escape sin contribuir al proceso de enfriamiento. Esta mala gestión del aire resulta en una baja eficiencia de enfriamiento.

La contención de los pasillos calientes (o fríos) es una de las estrategias más efectivas y menos costosas para mejorar la eficiencia energética de un centro de datos. La contención permite la separación física de las corrientes de aire, evitando así problemas de recirculación o derivación. De esta forma se puede suministrar aire a una temperatura más alta, aumentando así la eficiencia de enfriamiento.

2.9.2 Sistemas refrigerados por líquido

Cuando los centros de datos tienen un equipo de alta densidad de potencia, los sistemas refrigerados por aire pueden no ser la mejor solución en términos de eficiencia y fiabilidad. Por lo tanto, se deben emplear tecnologías diferentes de enfriamiento en tales casos, como los sistemas refrigerados por líquido, que son capaces de soportar potencia de alta densidad y ofrecen una amplia gama de ventajas. La principal ventaja es la mayor capacidad de transferencia de calor por unidad, que permite trabajar con una menor diferencia de temperatura entre el servidor y el refrigerante. Además, esta solución elimina dos pasos de baja eficiencia de los sistemas refrigerados por aire, la transferencia de calor del disipador de calor al aire y del aire al refrigerante. Por lo tanto, se puede obtener una disminución en la resistencia térmica del sistema y un aumento en la eficiencia energética. Las temperaturas de entrada más altas pueden eliminar potencialmente la necesidad de equipos activos para el rechazo de calor, y también abren la posibilidad de reutilización de calor. [2]

Figura 2. Sistema refrigerado por líquido de IBM.

2.9.3 Sistemas refrigerados por inmersión

La carcasa del servidor está sellada y contiene un refrigerante dieléctrico fluoro-orgánico en contacto directo con la electrónica, que se utiliza para transferir calor a una camisa de agua por convección natural. El calor puede transferirse directamente desde el gabinete a un circuito externo y eventualmente liberarse o reutilizarse.

Una ventaja respecto a los sistemas por aire es el considerable ahorro de energía del ventilador y un menor nivel de ruido. Por otro lado, el principal inconveniente de los sistemas refrigerados por líquido es la introducción de líquido dentro del centro de datos y el daño potencial que puede causar una falla.

Figura 3. Módulo de inmersión líquida.

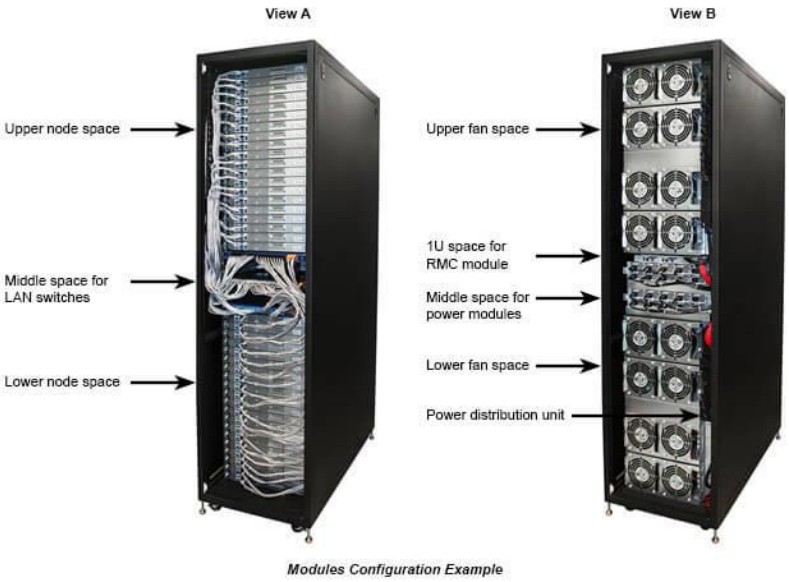

Figura 4. Racks.

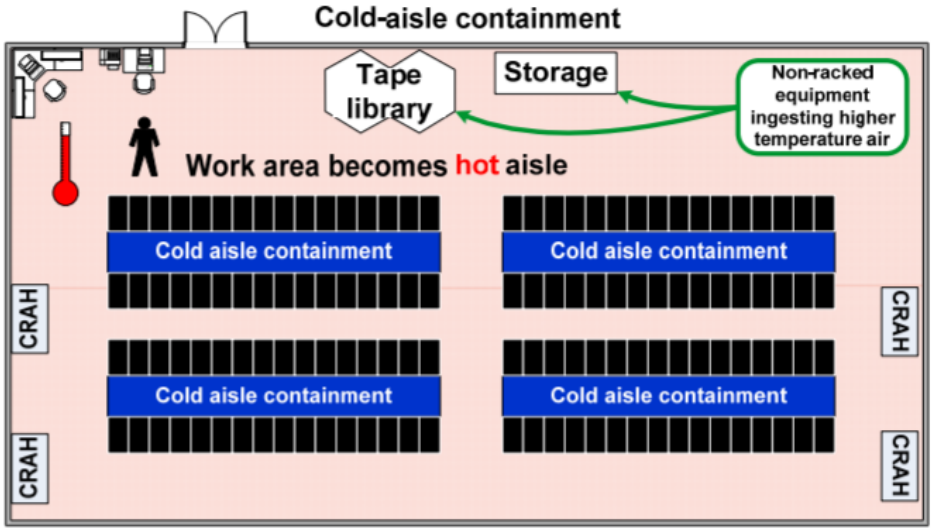

Figura 5. Cold aisle containment.

3. MODELADO DEL CENTRO DE DATOS

3.1 Conceptos básicos del modelado

El proceso de gestión del consumo de energía de un centro de datos consta generalmente de cuatro pasos principales: extracción de características, construcción del modelo, validación del modelo y aplicación del modelo.

- Extracción de características: en esta fase es necesario identificar en qué componentes se produce un mayor consumo de energía.
- Construcción del modelo: en segundo lugar, se utilizan como inputs las características identificadas en la fase anterior para construir un modelo de consumo de energía utilizando técnicas de análisis como regresión, machine learning, etc.
- Validación del modelo: a continuación, el modelo debe ser validado para adecuarse a su propósito.
- Uso del modelo: finalmente, se puede usar el modelo identificado como base para predecir el consumo de energía del centro de datos y así llegar a mejorar su eficiencia energética. [3]

Un modelo es una abstracción formal de un sistema real. Los modelos para sistemas informáticos se pueden representar como ecuaciones, modelos gráficos, reglas, árboles de decisión, conjuntos de ejemplos representativos, redes neuronales, etc. La elección de la representación afecta la precisión de los modelos, así como a su interpretabilidad. Diseñar modelos precisos de consumo de energía es muy importante para muchos esquemas de eficiencia energética empleados en equipos informáticos.

Para nuestro caso contemplamos un modelo típico de aislamiento en frío (Cold Aisle). El centro de datos está compuesto por unidades aisladas térmicamente en las que se disponen los racks de servidores de forma perimetral. Entra el aire frío del CRAC por el suelo, circula por un pasillo aislado de donde toman el aire los racks gracias a unos ventiladores que hacen pasar el aire frío por los servidores y salen al exterior, donde se unen con el aire de otras unidades que circulan en retorno al CRAC (Figura 1).

Las tareas en un servidor se gestionan mediante un sistema que manipula el nivel de tensión y la escala de frecuencia (DVFS) del servidor con el fin de mejorar su eficiencia energética en función de la carga de trabajo de los servicios que se gestionan por máquinas virtuales (VM).

El centro de datos da servicio a unos usuarios (*Cloud Users*) que alquilan un número concreto de servidores virtuales a un proveedor en la nube (*Cloud Provider*).

El proveedor de servicios se compromete a una calidad de servicio (*QoS*) en forma de un tiempo de respuesta máximo y una fiabilidad del servicio en forma de restricciones de temperatura.

3.2 Modelado del consumo de un servidor

En este estudio no vamos a tener en cuenta infraestructuras fundamentales des sistema. Podemos aproximar el consumo total de energía como la suma de energía consumida por el equipo informático y por los subsistemas de enfriamiento.

El centro de datos está formado por un número de servidores M que los J usuarios existentes pueden alquilar.

Cada usuario j tiene por consiguiente reservado M_j servidores como total para que lleven a cabo las tareas que debe gestionar. Se asume que para el usuario j, el número de tareas que debe de gestionar por unidad de tiempo es $L_j(t)$.

Cada uno de los servidores puede encontrarse en 3 estados diferentes: ocupado, ocioso o apagado. En el estado apagado, el servidor no tiene capacidad de hacer nada. Para poder estar operativo (ya sea ocioso u ocupado) se debe arrancar, lo cual conlleva un tiempo de arranque (T_{arr}), durante el cual el servidor consume energía pero no realiza tareas.

Cada usuario tiene un número de servidores que se encuentran operativos, m_j. En consecuencia, el número de servidores apagados será $M_j - m_j$. El número de servidores operativos es una variable manipulable del sistema con el fin de mejorar la eficiencia y está limitado a los contratados.

$$0 \leq m_j \leq M_j$$

El sistema reparte las tareas por los servidores dependiendo de este número de servidores operativos m_j y de una cierta cantidad de peticiones de servicio L_j (se trata de una perturbación, por lo que no es manipulable en nuestro modelo).

Este sistema realiza dicho reparto de tareas con el objetivo de mejorar la eficiencia regulando la temperatura de servicio, lo que se consigue ajustando la frecuencia y el voltaje del servidor. De esta manera, cuanto mayor sea la carga por servidor, es decir el número de tareas entre el número de servidores en uso (L_j/m_j), menor será la frecuencia de funcionamiento y en consecuencia al aumento del tiempo de servicio, por lo que empeorará la calidad (*QoS*).

Así, podríamos afirmar que cuanto mayor sea el número de servidores mejor será la calidad del servicio. Sin embargo, debemos valorar también que esto aumentaría el consumo energético.

El consumo de energía de un conjunto de servidores se puede representar de la siguiente manera [4]:

$$p_j(t) = a_1 ma_j(t) + a_2 m_j(t)$$

Donde a_1 identifica el consumo marginal del servidor, que depende de su frecuencia s, ma_j el número de servidores ocupados en tareas para el usuario j en el instante t, a_2 es el consumo un servidor en estado ocioso o en arranque, el cual se genera por componentes externos como la fuente de alimentación o los dispositivos de almacenamiento y m_j es el número de servidores activos para j.

$$a_1 = C_f s_{ij}$$

El consumo de un servidor i asociado al usuario j vendrá dado por

$$p_{ij} = \begin{cases} 0 & si\ está\ apagado \\ a_2 & si\ está\ ocioso \\ a_1 + a_2 & si\ está\ ocupado \end{cases}$$

Por tanto, el consumo total de energía de los servidores en el centro de datos será:

$$P(t) = \sum_{j=1}^{j=J} p_j(t)$$

3.3 Modelado de la cola

Se asume que el sistema de gestión de tareas mantiene una cola única para cada usuario j en la que las tareas aún sin realizar esperan a que se le asigne un servidor ocioso disponible entre los m_j operativos. [4]

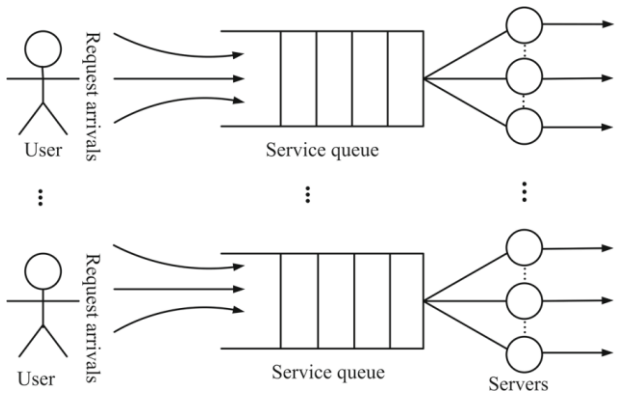

Figura 6. Cola M/M/m_j.

Se asume además que el tiempo entre llegadas (T_{lleg}) de las solicitudes de tareas viene descrito por una distribución exponencial, cuya función de densidad de probabilidad viene dada por:

$$f(t) = \frac{1}{T_{lleg}} e^{-\frac{t}{T_{lleg}}}$$

También es posible modelar la llegada de tareas mediante el número de peticiones que llegan en un intervalo de tiempo ΔT_{lleg}. En este caso sigue una distribución de probabilidad de Poisson cuya función de densidad viene dada por:

$$f(n) = \frac{(\frac{\Delta T_{lleg}}{T_{lleg}})^n e^{-\frac{\Delta T_{lleg}}{T_{lleg}}}}{n!}$$

El tiempo de servicio del servidor i será otra variable aleatoria, $T_{ser,ij}$, la cual sigue una distribución de probabilidad exponencial cuya función de densidad se define:

$$f(t) = \frac{1}{T_{ser,ij}} e^{-\frac{t}{T_{ser,ij}}}$$

Cada servidor i tendrá una velocidad de procesado determinada por su frecuencia s_{ij}, medida en número de instrucciones por segundo. De esta manera, el tiempo medio de servicio será:

$$T_{ser,ij} = \frac{K_j}{s_{ij}}$$

Donde K_j representa la complejidad media del servicio que solicita el usuario j en número de instrucciones.

En cuanto a la disciplina de la cola se podría optar por diversos criterios para establecer la prioridad de tareas a procesar. Utilizaremos una cola FIFO (First In, First Out) en la cuál se procesan las tareas por orden de llegada.

Si definimos λ como la media de llegadas previstas y μ la media de peticiones que se procesas, si $\lambda > \mu$ la cola tendería a crecer al infinito. Esto inestabilizaría la cola y por consiguiente el sistema, por lo que es necesario imponer una condición de estacionariedad:

$$M\frac{1}{T_{ser}} > \frac{1}{T_{lleg}}$$

3.4 Modelado térmico

Se asume que el modelo térmico de cada servidor sigue una dinámica lineal de primer orden basada en la siguiente ecuación de balance térmico:

$$K_t \frac{dT_{ij}(t)}{dt} = c_p q_a(t)(T_c(t) - T_{ij}(t)) + p_{ij}(t)$$

Donde T_c y q_a son la temperatura de la unidad aislada en frío (CRAC) y el caudal de aire. T_{ij} y p_{ij} son la temperatura y el consumo de potencia del servidor i para el usuario j, K_t es la capacidad térmica del servidor y c_p la capacidad térmica del aire.

3.5 Modelado de la refrigeración

Para las máquinas de refrigeración se suele diseñar un sistema de control que manipule la velocidad del compresor refrigerante para regular la temperatura del aire de salida con un caudal de aire constante. El controlador podrá garantizar que la temperatura de salida del aire sea la de consigna siempre que el salto térmico que deba aportar la máquina sea menor que el máximo de diseño. Esta situación suele darse en las máquinas de frío, ya que suelen estar sobredimensionadas.

De esta manera se podría modelar la temperatura de la máquina como un sistema de primer orden de ganancia estática la unidad. Teniendo la temperatura de consigna del aire T_r, como la variable manipulable.

$$\tau \frac{dT_c}{dt} = T_r - T_c$$

El consumo de un CRAC se puede definir de la siguiente manera [5]:

$$P_{CRAC} = \frac{P}{CoP(T_c(t))}$$

Donde *P* es la potencia que demanda el centro de datos, Y *CoP* representa la eficiencia de la refrigeración de la máquina. Se trata de una función parabólica tal que [5]:

$$CoP(T) = bT^2 + cT + d$$

Siendo *b*, *c* y *d* coeficientes que dependen de la máquina y que nos proveerá el fabricante.

3.6 Restricciones del modelo

Podemos calcular el tiempo de respuesta del sistema como:

$$T_{res,j} = \frac{1}{\sum_i^{m_j} \mu_{ij} - \lambda_j}$$

Como indicamos anteriormente, el número medio de servicios (μ_{ij}) se puede calcular como el ratio entre la frecuencia s y la complejidad en número de instrucciones K_j:

$$\mu_{ij} = \frac{K_j}{s_{ij}}$$

Existe un tiempo medio de servicio máximo impuesto por cada usuario denominado D_j.

$$T_{res,j} \leq D_j$$

Otra restricción es el número de servidores activos, m_j, que debe ser siempre menor que el número de contratados.

$$0 \leq m_j \leq M_j$$

3.7 Índice de desempeño

Utilizaremos como índice de desempeño el Performance Usage Efficiency (PUE), el cual se define como la razón entre la energía consumida por los servidores entre la energía total consumida por el sistema (servidores y refrigeración).

$$PUE(t) = \frac{E_c(t)}{E_c(t) + E_{CRAC}(t)}$$

Donde:

$$E_c(t) = \int_0^t P(\tau) d\tau$$

$$E_{CRAC}(t) = \int_0^t \frac{P(\tau)}{CoP(T_c(\tau))}\, d\tau$$

3.8 Definición de las variables del modelo

Empezaremos definiendo las variables manipulables, que son aquellas cuyo valor podemos modificar para producir cambios en el sistema y poder estudiar su comportamiento. Estas variables son:

- Número de servidores activos m_j.
- Frecuencia de trabajo de los servidores s_{ij}.
- Temperatura de consigna del aire T_r.

Variables internas del sistema:

- On: si el servidor está apagado o encendido.

$$On = \begin{cases} 0 & si\ el\ servidor\ está\ apagado \\ 1 & si\ el\ servidor\ está\ encendido\ o\ arrancando \end{cases}$$

- Temperatura de refrigeración T_c.

El objeto de estudio es el cambio que producen las variables manipulables en las siguientes, las variables de salida:

- Tareas en cola.
- Temperatura de los servidores.
- Potencia de los servidores p_{ij}.
- Potencia del CRAC P_{CRAC}.

3.9 Dimensionamiento de los parámetros del modelo

Utilizamos los datos experimentales del centro de datos del trabajo de Fu [4] para asumir valores factibles para los parámetros de nuestro modelo.

En el citado artículo se contemplan 1700 servidores y 4 aplicaciones que solicitan tareas que varían entre 1000 y 150000 tareas por segundo cada aplicación. La media se estima en 100000. Escalando estos valores a nuestro modelo, con una simple regla de tres obtenemos que la media de llegadas para nuestro modelo será 580, por lo que consideraremos 600.

Como explicamos anteriormente, la media de llegadas debe ser menor que la media de procesadas para evitar el problema de la cola infinita:

$$M\frac{1}{T_{ser}} > \frac{1}{T_{lleg}}$$

Contamos con M=10 servidores y $\frac{1}{T_{lleg}}$= 600, de manera que el tiempo de servicio T_{ser} se estimaría 0,01 y el T_{lleg} 0,0016. Partiendo de estos datos, hemos multiplicado ambos tiempos por diez para un mejor desarrollo de la simulación en Anylogic, resultando T_{ser}= 0,1 y T_{lleg}= 0,016. Sería necesario un mínimo de al menos 7 servidores activos para cubrir esta demanda de tareas. Sin embargo, esta restricción puede violarse durante un período determinado de tiempo.

No se va a aplicar escalado de frecuencia, por lo que el parámetro s toma un valor unidad. De esta manera, teniendo en cuenta que el tiempo de servicio es la media de la complejidad de las instrucciones K dividida por la frecuencia, obtenemos K=0,01.

En el artículo se tiene en cuenta un consumo energético $a_1 = a_2 = 40\text{W}$. Por consiguiente, considerando $a_1 = C_f s_{ij}$, se obtiene C_f=40. Del trabajo de Fu asumimos también que $C_p q_a$=1,6

La temperatura de refrigeración será siempre:

$$15 \leq T_c \leq 25\ ºC$$

En nuestro caso tendrá un valor inicial T_c=20ºC. La constante de tiempo τ se estima del orden de 5 milésimas de segundo.

Los coeficientes de la máquina los consideramos como b= c=d=1 de forma arbitraria.

Tomaremos como valor inicial de la temperatura de los servidores T_{ij}=20ºC y la temperatura de consigna del aire T_r=20ºC.

El tiempo de arranque de los servidores será $T_{arr} = 1$ segundo, que es equivalente al tiempo que tardarán los servidores en apagarse.

4. CONTROLADORES DEL SISTEMA

Se define el control como el uso de algoritmos y realimentación en sistemas de ingeniería.

Por lo tanto, el control incluye ejemplos tales como bucles de realimentación en amplificadores electrónicos, controladores de punto de ajuste en el procesamiento químico y de materiales e incluso protocolos de enrutadores que controlan el flujo de tráfico en Internet. Las aplicaciones emergentes incluyen sistemas de software de alta confianza, vehículos autónomos y robots, sistemas de gestión de recursos en tiempo real y sistemas de ingeniería biológica. En esencia, el control es una ciencia de la información e incluye el uso de información tanto en representaciones analógicas como digitales.

Un controlador moderno detecta el funcionamiento de un sistema, lo compara con el comportamiento deseado, calcula las acciones correctivas basadas en un modelo del sistema respuesta a entradas externas y activa el sistema para efectuar el cambio deseado.

Este ciclo de realimentación de detección, cálculo y actuación es el concepto básico en el control. Los problemas clave a resolver en el diseño de la lógica de control son garantizar que la dinámica del sistema de circuito cerrado sea estable (las perturbaciones limitadas dan errores limitados) y que tengan un comportamiento adicional deseado (buena atenuación de perturbaciones, rápida respuesta a los cambios en el punto de operación, etc.). Estas propiedades se establecen utilizando una variedad de técnicas de modelado y análisis que capturan la dinámica esencial del sistema y permiten la exploración de posibles comportamientos en presencia de incertidumbre, ruido y fallas de componentes.

4.1 Controlador PID

Con un control proporcional, la característica del controlador es proporcional al error de control para pequeños errores. Esto se puede lograr con la ley de control

$$u = \begin{cases} u_{max} & si\ e \geq e_{max} \\ k_p e & si \quad e_{min} < e < e_{max} \\ u_{min} & si\ e \leq e_{min} \end{cases}$$

Donde k_p es la ganancia del controlador, $e_{min} = u_{min}/k_p$ y $e_{max} = u_{max}/k_p$. El intervalo (e_{min}, e_{max}) se denomina banda proporcional ya que el comportamiento del controlador es lineal cuando el error está en este intervalo:

$$u = k_p(r - y) = k_p e \quad si \quad e_{min} < e < e_{max}$$

El control proporcional tiene el inconveniente de que la variable de proceso a menudo se desvía de su valor de referencia. En particular, si se requiere algún nivel de señal de control para que el sistema mantenga un valor deseado, entonces debemos tener $e \neq 0$ para generar la entrada requerida.

Esto se puede evitar haciendo que la acción de control sea proporcional a la integral del error:

$$u(t) = k_i \int_0^t e(\tau)d\tau$$

Esta forma de control se llama control integral (PI), y k_i es la ganancia integral. Se puede demostrar a través de argumentos simples que un controlador con acción integral tiene cero errores de estado estacionario. El problema es que no siempre puede haber un estado estable porque el sistema puede estar oscilando. [7]

Un refinamiento adicional es proporcionar al controlador una capacidad de anticipación mediante el uso de una predicción del error. Una predicción simple viene dada por la extrapolación lineal

$$e(t + T_d) \approx e(t) + T_d \frac{de(t)}{dt}$$

que predice el error T_d unidades de tiempo previas. Combinando control proporcional, integral y derivativo, obtenemos un controlador que se puede expresar matemáticamente como

$$u(t) = k_p e(t) + k_i \int_0^t e(\tau)d\tau + k_d \frac{de(t)}{dt}$$

La acción de control es en este caso una suma de tres términos: el pasado, representado por la integral del error, el presente, representado por el término proporcional y el futuro representado por la extrapolación lineal del error (el término derivado).

Esta forma de realimentación se denomina controlador proporcional integral derivativo (PID) y es muy útil y capaz de resolver una amplia gama de problemas de control. Más del 95% de todos los problemas de control industrial se resuelven mediante control PID, aunque muchos de estos controladores son en realidad controladores proporcionales integrales (PI) debido a que la acción derivada no se incluye a causa de la dificultad de su estimación. [7]

Es lo que ocurre en nuestro caso, en el que programaremos un PI en tiempo discreto, para el que no hace falta más que calcular el error en cada instante y la integral del error y aplicar la siguiente fórmula:

$$e_{int}(k) = e(k) + e_{int}(k-1)$$

$$u(k) = Ke(k) + \frac{e_{int}(k)}{K_i}$$

Utilizaremos este método para controlar la temperatura de refrigeración del CRAC en función de la temperatura de los servidores y la cantidad de servidores encendidos en función de la cantidad de tareas en cola.

4.2 Controlador de la temperatura de consigna

El objetivo del CRAC es enfriar los servidores para evitar su sobrecalentamiento manteniéndolos por debajo de una temperatura determinada. En nuestro caso, queremos evitar que estos superen los 70ºC.

Como disponemos de un único controlador para manejar 10 servidores, es necesario averiguar cuál es el servidor con mayor temperatura y calcular sobre este la ley de control. Si mantenemos por debajo del umbral indicado de 70ºC al servidor cuya temperatura sea la más elevada todos los demás cumplirán también la condición.

El error en cada instante será por lo tanto:

$$e(k) = 70ºC - T_{max}(k)$$

El siguiente paso es acumular el error del instante al error integral del control.

$$e_{int}(k) = e(k) + e_{int}(k-1)$$

Aplicamos la fórmula del controlador PI descrita anteriormente sumando el valor de la temperatura inicial de refrigeración (20ºC) para centrar la acción de control.

$$u(k) = Ke(k) + \frac{e_{int}(k)}{K_i} + u_0$$

Donde K y K_i toman valores de 0,1 y 1000 respectivamente (establecidos experimentalmente) y u_0 es la temperatura inicial de refrigeración.

Además, queremos que la temperatura de refrigeración cumpla siempre la siguiente condición:

$$15ºC \leq u(k) \leq 25ºC$$

4.3 Controlador de la cola de tareas

Para controlar la cola de tareas el proceso es análogo al anterior.

En este caso tenemos una única cola sobre la que influir, por lo que no es necesario hacer un cálculo de máximos como para la temperatura de los servidores. Para la cola queremos que el número de tareas que albergue se mantenga alrededor de las 50, de manera que:

$$e(k) = q(k) - 50$$

Donde $q(k)$ es el tamaño de la cola en el instante k.

Para el caso del error integral el cálculo es exactamente el mismo que en el caso de la temperatura:

$$e_{int}(k) = e(k) + e_{int}(k-1)$$

Y para implementar la ley de control utilizamos también la misma fórmula que en la refrigeración, aunque cambiado los parámetros:

$$u(k) = Ke(k) + \frac{e_{int}(k)}{K_i} + u_0$$

Siendo en este caso K y K_i 0,1 y 10000 respectivamente (establecidos también experimentalmente) y u_0 es el número inicial de servidores encendidos, en nuestro caso 5.

Como es lógico, el número de servidores encendidos debe ser mínimo 1 para que el sistema funcione y máximo 10 ya que son los servidores de los que disponemos:

$$1 \leq u(k) \leq 10$$

5. SOFTWARE DE SIMULACIÓN ANYLOGIC©

5.1 Conceptos generales

La incertidumbre en los tiempos de operación y los resultados se pueden representar fácilmente en los modelos de simulación, lo que permite la cuantificación del riesgo y encontrar soluciones más sólidas.

Para simular el funcionamiento del centro de datos se empleará el software de modelado Anylogic©. Anylogic© es un software que permite generar modelos en cualquiera de los tres métodos modernos de simulación e incluso combinarlos para representar sistemas de cualquier nivel de complejidad.

5.2 Características de Anylogic©

El modelado con simulación resuelve problemas del mundo real de manera segura y eficiente. Proporciona un método de análisis que se verifica, comunica y comprende fácilmente. Siendo aplicable a cualquier tipo de industria y disciplina, el modelado de simulación proporciona soluciones valiosas al proporcionar información clara sobre sistemas complejos.

El software de simulación proporciona un entorno dinámico para el análisis de modelos de computadora mientras se ejecutan. Los modelos de simulación se pueden animar en 2D/3D, lo que permite verificar, comunicar y comprender conceptos e ideas con mayor facilidad. Los analistas del modelo ganan confianza al verlo en acción y pueden demostrar claramente los hallazgos.

La capacidad de analizar el modelo mientras se ejecuta la simulación marca una diferencia entre Anylogic© y otros métodos, como los que usan Excel o la programación lineal. Al poder inspeccionar los procesos e interactuar con el modelo de simulación en acción, tanto su comprensión como su confianza aumentan considerablemente.

La creación de AnyLogic© se inspiró mucho en Java, creyendo que es el lenguaje ideal para los modeladores. Por un lado, Java es un lenguaje de nivel suficientemente alto en el que no es necesario preocuparse por la asignación de memoria, distinguir entre objetos y referencias, etc. Por otro lado, es un lenguaje de programación orientado a objetos, potente y con alto rendimiento. En Java es posible definir y manipular estructuras de datos de cualquier complejidad deseada, desarrollar algoritmos eficientes, usar numerosos paquetes disponibles de Sun ™/Oracle ™ y otros proveedores. Java está respaldado por los líderes de la industria y a medida que Java mejora, los modeladores de AnyLogic© se benefician automáticamente de él.

Un modelo desarrollado en AnyLogic© está completamente escrito en código Java y, después de haber sido vinculado con el motor de simulación AnyLogic© (también escrito en Java) y, opcionalmente, con un optimizador de Java, se convierte en una aplicación Java independiente completamente autosuficiente. Esto hace que los modelos AnyLogic© sean multiplataforma: pueden ejecutarse en cualquier entorno habilitado para Java. [6]

5.3 Entorno del programa

AnyLogic© es un programa que además de ser tan potente y versátil como acabamos de ver, es muy intuitivo y visual, lo que produce que sea más sencillo y ameno modelar.

La ventana de trabajo de la aplicación tiene el siguiente aspecto, con la división de zonas señalada (Figura 7):

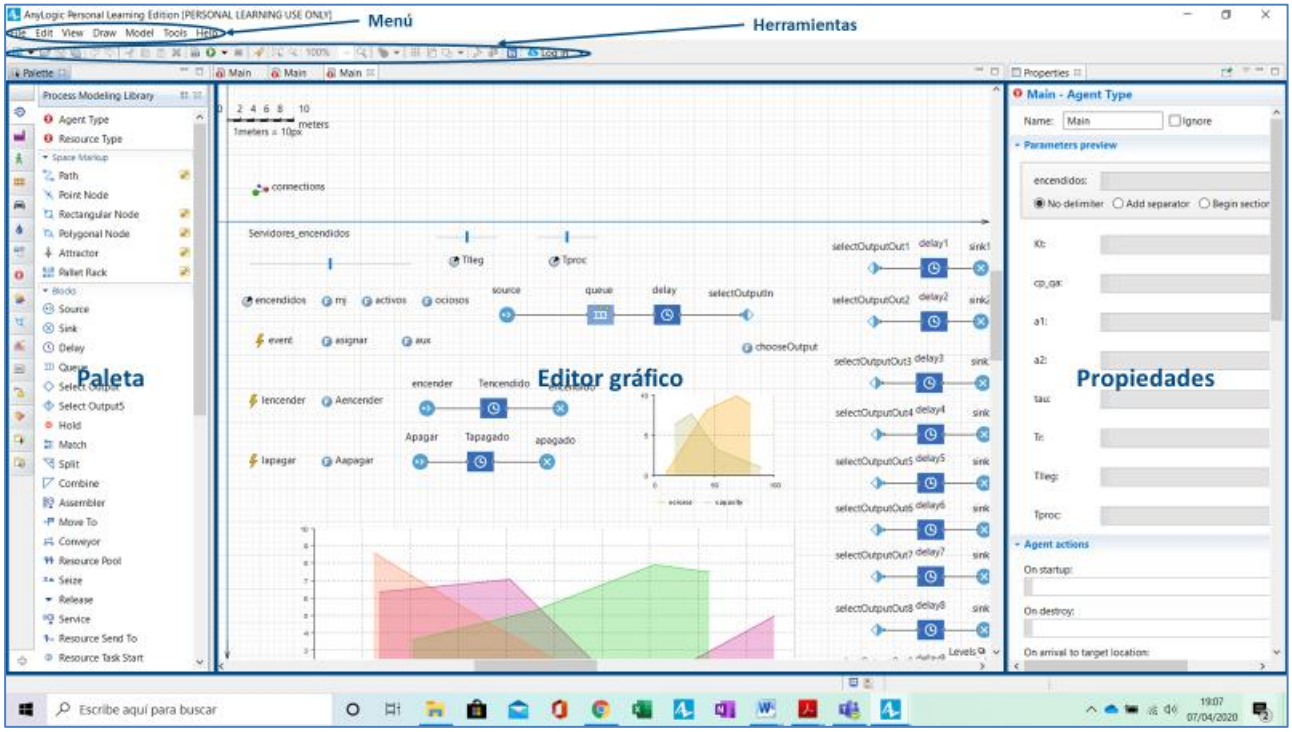

Figura 7. Entorno Anylogicrno Anylogic©.

- Editor gráfico: espacio donde poder editar el diagrama de agentes y eventos.
- Paleta: proporciona la lista de elementos con los que crear el modelo agrupados por categorías.
- Propiedades: permite ver y modificar las propiedades del elemento del modelo que seleccionemos.

5.4 Librería de modelado de procesos

Para la simulación de eventos discretos Anylogic© incorpora una librería de modelado de procesos que sirve como herramienta para modelar a un nivel de detalle las operaciones de cualquier tipo de sistema ya sea logístico, industrial, de banca, etc. La librería incluye la posibilidad de realizar simulaciones del flujo del proceso permitiendo al usuario comprender su dinámica y las interdependecias de los componentes obteniendo así una valiosa información para la toma de decisiones.

Al utilizar esta librería el usuario tiene la posibilidad de modelar procesos del mundo real: secuencias de operaciones que involucran colas, demoras y recursos. [6]

Figura 8. Bloques de modelado de procesos en Anylogic©.

Los bloques mostrados en la Figura 8 son todos los que incorpora la librería de modelado de procesos, que como se puede apreciar, es bastante extensa y versátil, lo que permite multitud de opciones de modelado al usuario.

A continuación, se detallan los bloques que utilizaremos en este trabajo a lo largo del modelo:

- Source: Genera agentes que entran en el flujo de modelo.
- Sink. Sumidero de agentes entrantes.
- Delay. Retrasa el flujo de los agentes durante un determinado tiempo.
- Queue. Cola que almacena agentes en un orden específico.
- SelectOutputIn. Puerto lanzador del agente.
- SelectOutputOut. Puerto receptor de agente procedente del bloque anterior en función de una condición dada.

5.5 Componentes de Agentes

Los componentes de los Agentes en Anylogic© son los elementos que permiten analizar la información referente a estos a lo largo de su flujo por el modelo. Con ellos se controlan las variables del sistema.

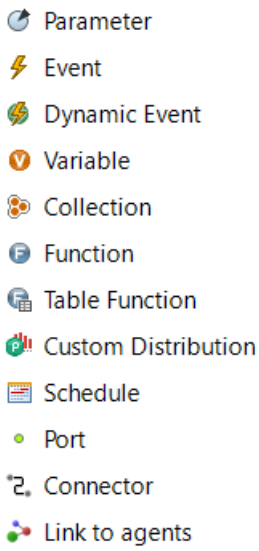

Figura 9. Componentes de agentes en Anylogic©.

Al igual que con los bloques de modelado de procesos, después de ver las opciones disponibles (Figura 9), estos son los componentes de agentes que utilizaremos en nuestro modelo:

- Parameter. Se utiliza para describir objetos de forma estática. En nuestro caso lo usaremos para definir constantes.
- Variable. Sirve para representar valores que variarán a lo largo de la simulación del modelo.
- Function. Permite desarrollar nuestras propias funciones que nos devuelvan un valor o ejecuten una operación. Resulta de gran utilidad para funciones que se repitan en distintos puntos del modelo.
- Event. Es el método más sencillo para programar acciones en la simulación. Con él se ejecutan cálculos o alteraciones en el modelo cada cierto tiempo que determinemos o al cumplirse una condición. [6]

5.6 Otras librerías en Anylogic

Como hemos comentado anteriormente, Anylogic© incorpora una gran cantidad de opciones para todo tipo de industrias y sistemas que representar. A continuación, se exponen algunas librerías que podemos utilizar más allá de las que hemos necesitado en nuestro modelo.

5.6.1 Librería Peatonal

Es una herramienta de simulación de peatones y análisis de multitudes que permite a los usuarios modelar, visualizar y analizar con precisión el comportamiento de los flujos de multitudes en un entorno físico y eliminar sus posibles ineficiencias.

Un peatón en un modelo Anylogic© se mueve de acuerdo con leyes físicas simuladas. Interactúa con los objetos circundantes, como paredes y escaleras mecánicas, y evita posibles colisiones. Los usuarios pueden configurar a los peatones con propiedades, preferencias y estados individuales. El conjunto de herramientas de la biblioteca incluye un mapa de densidad de flujo, contadores de peatones y elementos para calcular los tiempos de espera y servicio.

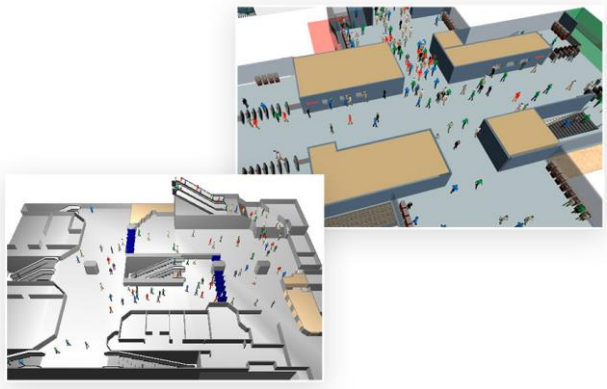

Figura 10. Simulación con Librería Peatonal en Anylogic©.

5.6.2 Librería de Tráfico

Permite a los usuarios planificar, diseñar y simular flujos de tráfico de un nivel físico detallado. La librería es ideal para el modelado explícito del comportamiento de cada conductor y para representar la dinámica del flujo de transporte.

Los algoritmos predefinidos de la librería tienen en cuenta las normas de circulación típicas, como el control de velocidad o la prevención de colisiones. Al mismo tiempo, en los modelos de tráfico por carretera, cada vehículo representa un agente que puede tener sus propios parámetros físicos y patrones de comportamiento. Esto, asociado a la posibilidad de crear modelos 2D y 3D de cada vehículo y sus alrededores, hace que los modelos de tráfico sean flexibles y visuales.

Figura 11 Simulación con Librería de Tráfico en Anylogic©.

5.6.3 Librería Ferroviaria

Esta librería permite a los usuarios modelar, simular y visualizar eficientemente las operaciones de lolas estaciones y el transporte ferroviario de cualquier complejidad y escala. Con esta librería se pueden modelar estaciones de ferrocarril, instalaciones de reparación de vagones, estaciones de metro, trenes de enlace con el aeropuerto e incluso redes de tranvías. También ayuda a los usuarios con la planificación de operaciones, la gestión de flotas y la programación de trenes y su mantenimiento.

En los modelos ferroviarios, los trenes se mueven de acuerdo con la lógica establecida en un diagrama de flujo, mientras que cada vagón y locomotora en un modelo son agentes con sus propios estados y propiedades. Esto,

junto con la interoperabilidad de otras bibliotecas, proporciona capacidades para simulaciones precisas de sistemas ferroviarios complejos.

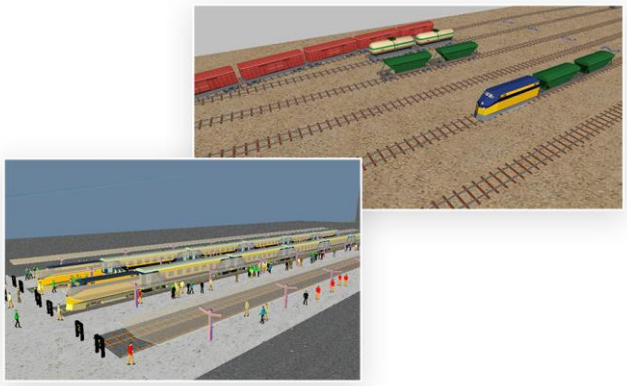

Figura 12. Simulación con Librería Ferroviaria en Anylogic©.

5.6.4 Librería de Fluidos

En este caso los usuarios pueden simular la logística de materiales a granel, fluidos y flujos de gas, y modelar operaciones de tuberías, procesos de minería y transmisiones de gas y energía, entre otras cosas. Con los componentes de la librería, los usuarios pueden crear representaciones precisas de tanques, tuberías, transportadores y sus redes, y realizar el seguimiento por lotes de los flujos. Captura fácilmente diversas características de los flujos, como la velocidad y el rendimiento, para encontrar posibles cuellos de botella y tiempos de inactividad, y optimizar los procesos operativos.

Para simular con precisión los comportamientos, la librería utiliza el enfoque de simulación de velocidad discreta. Esto hace que el proceso de modelado sea más transparente y permite a los usuarios realizar un seguimiento de los cambios de flujo cuando se producen.

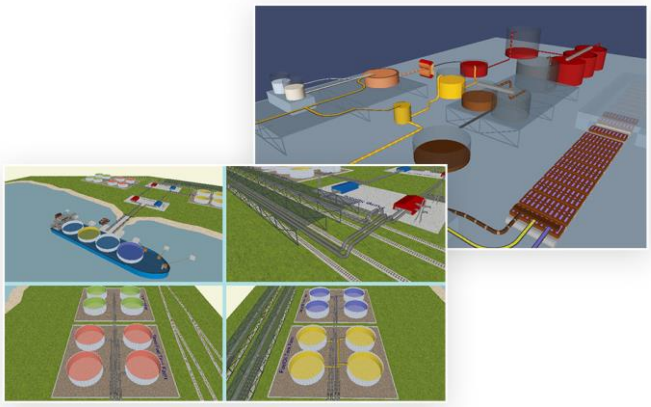

Figura 13. Simulación con Librería de Fluidos en Anylogic©.

5.6.5 Librería de Gestión de Materiales

Simplifica la simulación de sistemas y operaciones de fabricación complejos. Se puede usar para diseñar modelos detallados de instalaciones de producción y almacenamiento y administrar flujos de trabajo de materiales dentro de cuatro paredes. El modelo de fábrica digital, creado con el kit de herramientas de simulación de manejo de materiales, puede ayudar a probar y optimizar las políticas de producción, transporte e inventario, así como reducir posibles errores y demoras en el flujo de materiales en la fábrica.

En los modelos de red de transportadores, los usuarios pueden usar estrategias de enrutamiento predeterminadas

o personalizadas para unidades de material, robots industriales, máquinas de fabricación y operadores. Los transportadores evitan automáticamente colisiones, detectan posibles puntos muertos y los resuelven.

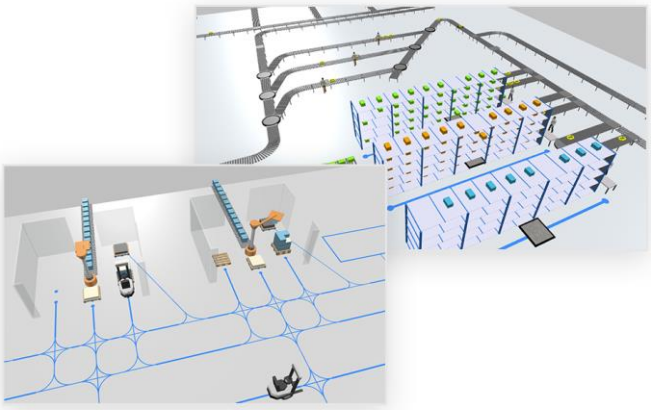

Figura 14. Simulación con Librería de Gestión de Materiales en Anylogic©.

6. DESARROLLO DEL MODELO EN ANYLOGIC©

Llegados a este punto, en el que conocemos el modelo que queremos desarrollar y el funcionamiento de la herramienta Anylogic©, el siguiente paso es implementarlo.

En nuestro caso interpretaremos un sistema para un único usuario con diez servidores contratados.

6.1 Flujo de las tareas

El flujo de las tareas desde su petición hasta que son terminadas de realizar por algún servidor se puede ver claramente representado en la Figura 15.

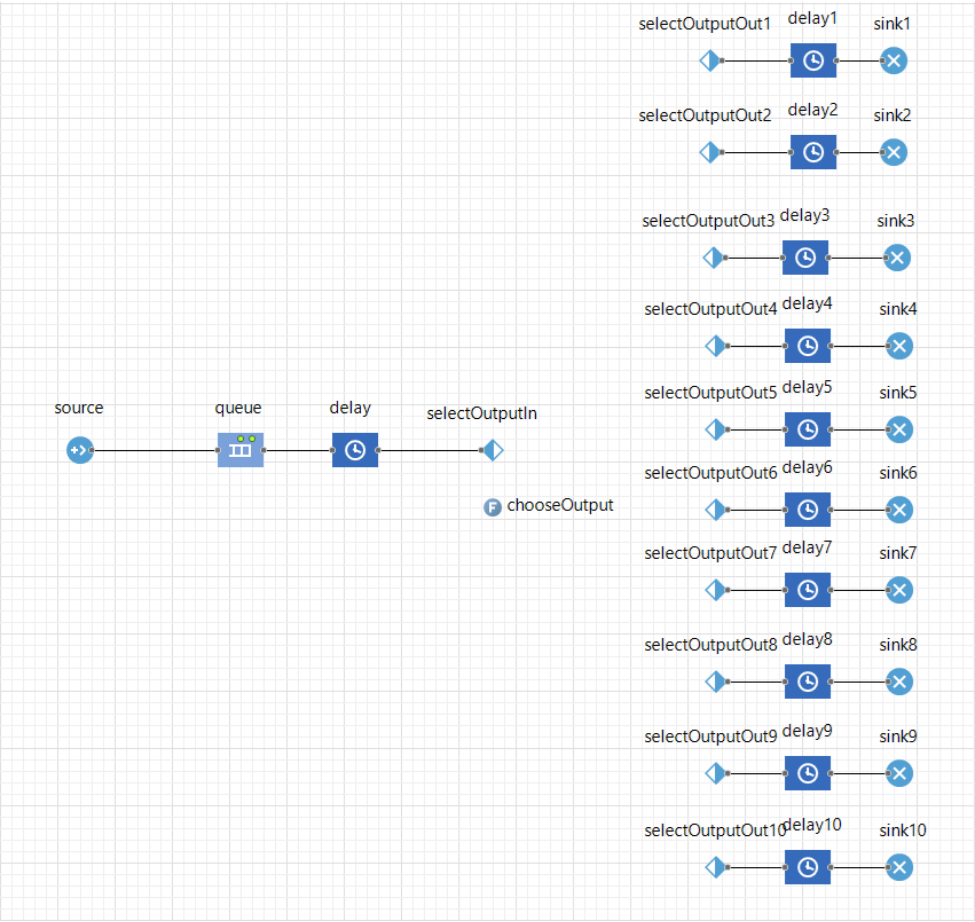

Figura 15. Flujo de tareas en Anylogic©.

Las tareas nacen en el modelo en el bloque *source* siguiendo una distribución exponencial cuyo tiempo medio entre llegadas es T_{lleg}. La primera llegada al sistema está programada en el instante 0 de simulación, entendiendo este momento como el inicio del modelo.

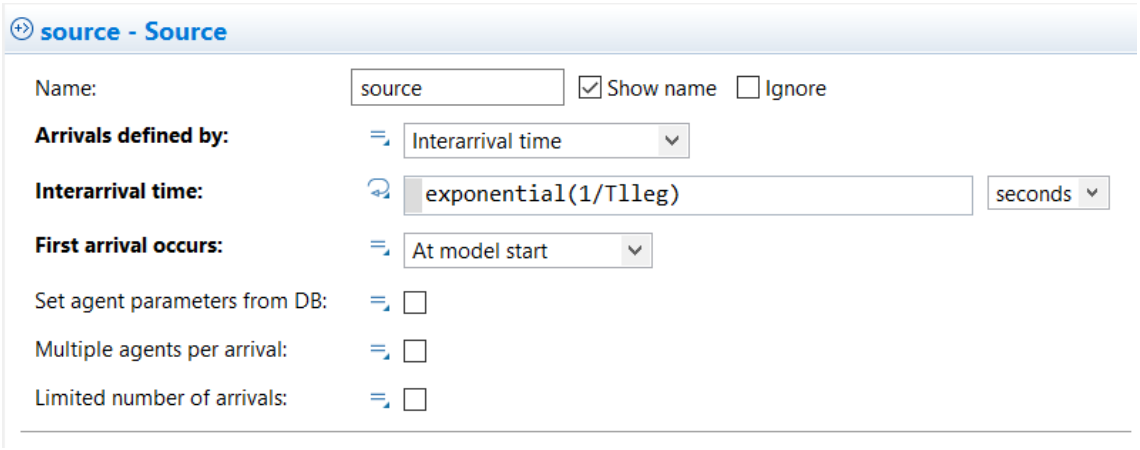

Figura 16. Llegada de tareas.

Una vez llegan las tareas al sistema, pasan a una cola a la espera de que haya algún servidor encendido y ocioso que se encargue de procesarlas. Esta cola está diseñada para albergar hasta 1000 tareas, y prioriza la salida con un orden FIFO, como hemos explicado anteriormente.

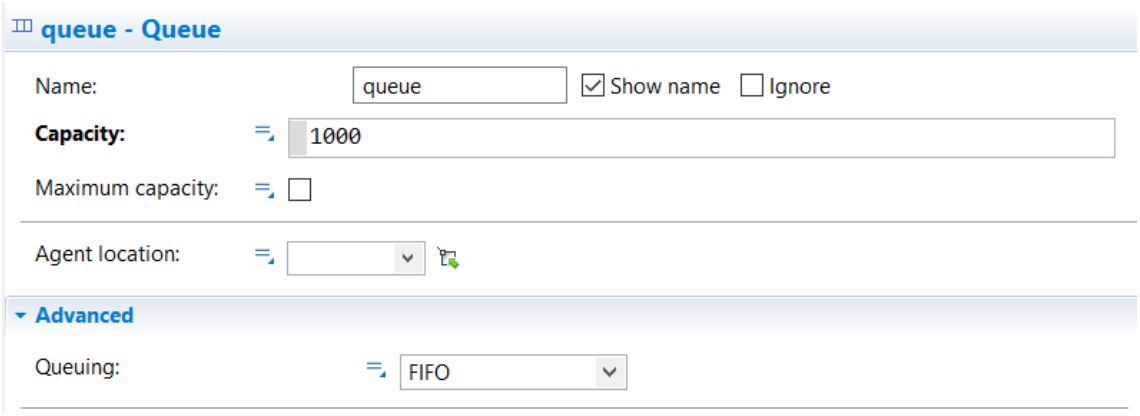

Figura 17. Cola de tareas.

Para hacer avanzar las tareas desde esta cola, es necesario que haya algún servidor ocioso, es decir, encendido y no procesando ya otra tarea.

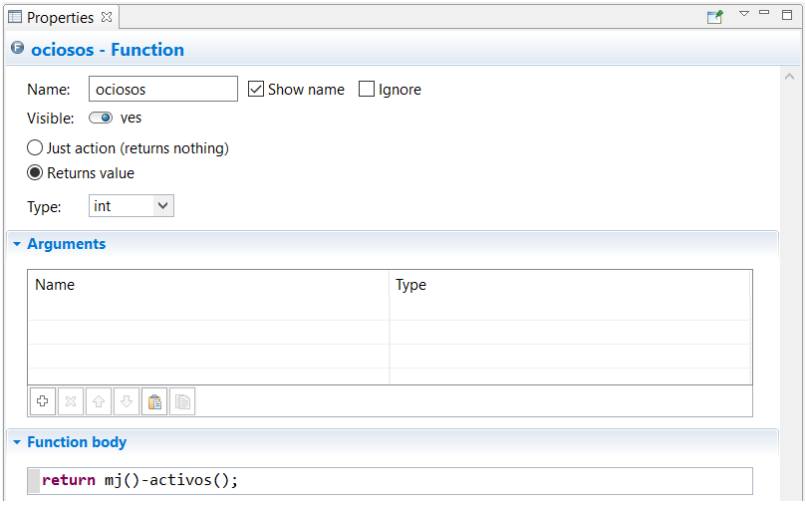

Figura 18. Cálculo de servidores ociosos.

Sabiendo si hay o no servidores ociosos, la función *aux* define el tiempo de espera que deben tener las tareas entre la cola y su servidor asignado, siendo cero en caso de haber ociosos e infinito en caso contrario.

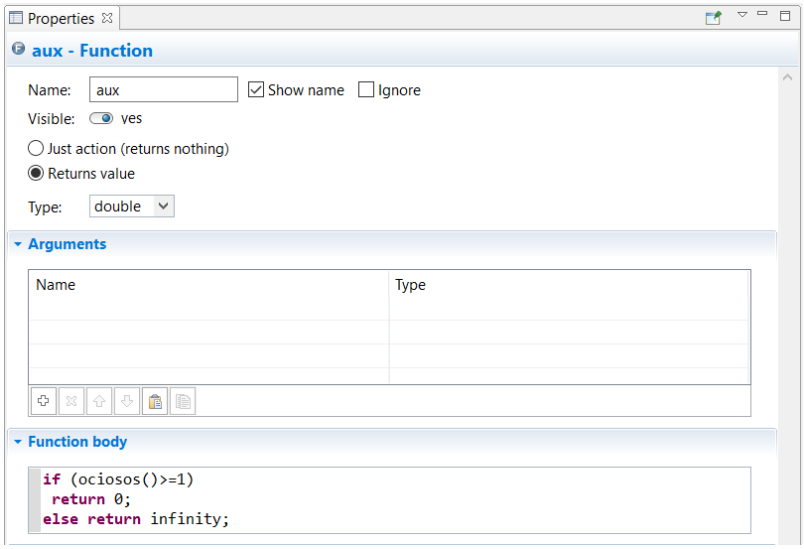

Figura 19. Función aux.

Esta espera se hace mediante un proceso artificial (*delay*) cuya duración es el resultado de la función *aux*.

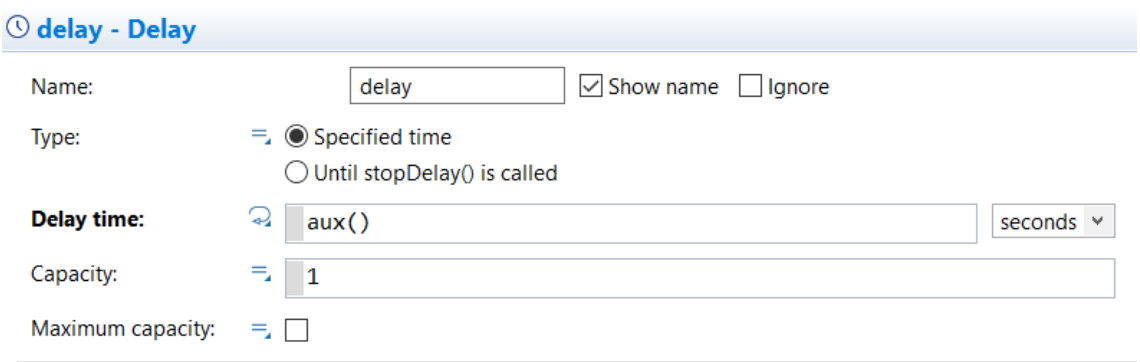

Figura 20. Delay artificial.

Después de este proceso artificial, las tareas llegan a un selector (*selectOutputin*), el cual las dirige al servidor que corresponda según la función *ChooseOutput*.

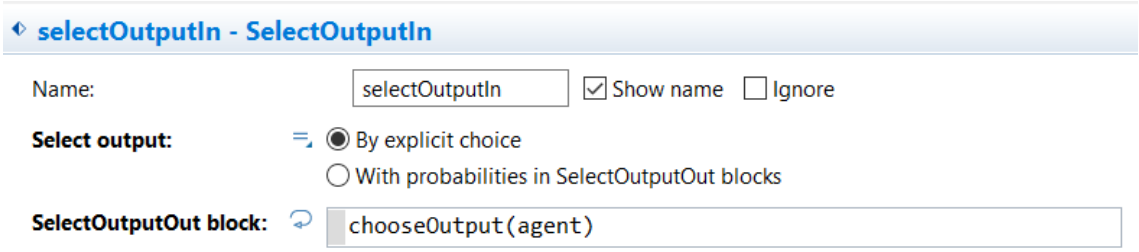

Figura 21. Asignador de servidores para tareas.

La función *ChooseOutput* devuelve como resultado el servidor al que debe lanzarse la tarea. Analiza desde el primer servidor hasta el último si está encendido y sin ninguna tarea asignada, devolviendo como resultado el primero que encuentre.

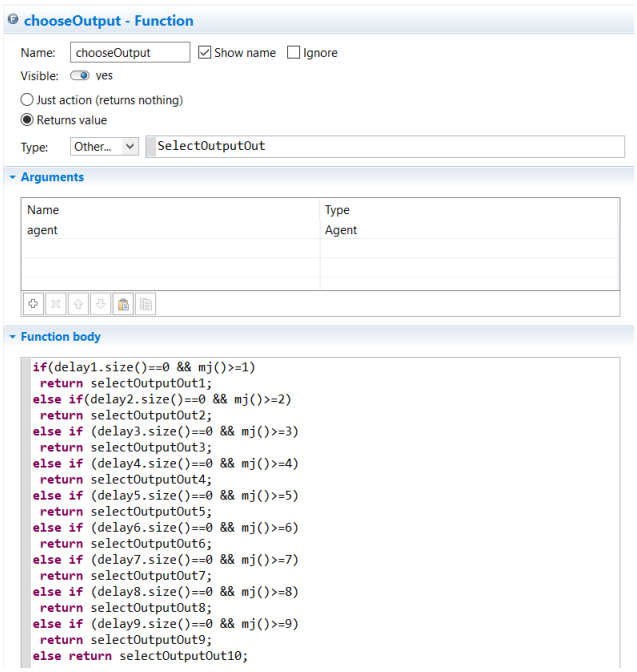

Figura 22. Función ChooseOutput.

Cada servidor realiza las tareas que se le asignan siguiendo una distribución exponencial de tiempo medio de proceso T_{proc}. Un servidor solo puede procesar una única tarea a la vez.

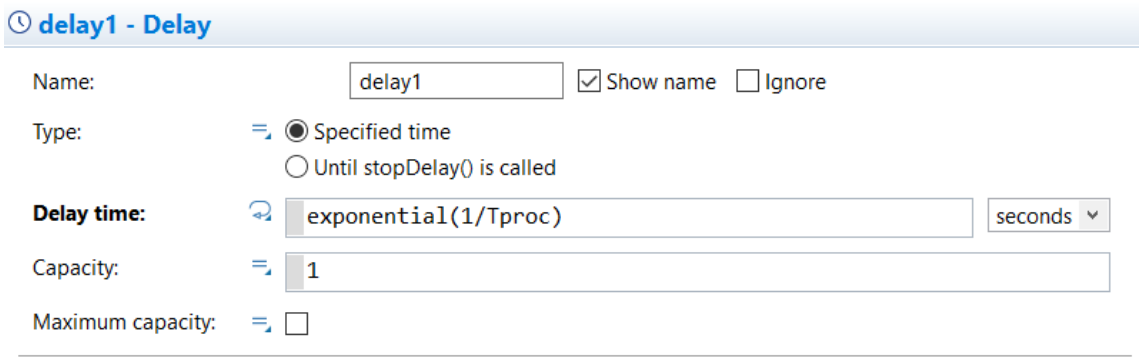

Figura 23. Ejemplo de procesado de las tareas en los servidores.

6.2 Proceso de encendido y apagado de los servidores

Una vez explicado el proceso principal, el flujo de las tareas desde que se generan hasta que son procesadas, es el turno del proceso de encendido y apagado de servidores.

Como ya avanzamos en el capítulo 3, los servidores necesitan un proceso temporal tanto para encenderse como para apagarse.

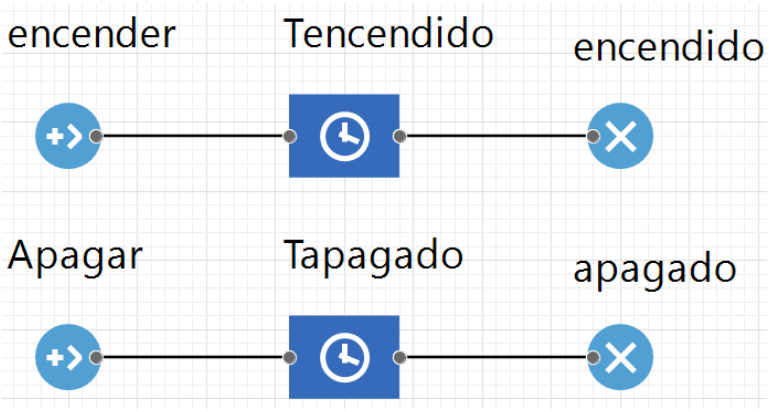

Figura 24. Procesos de encendido y apagado de servidores.

Estos procesos se inician cuando desde algún evento del modelo se inyecta un número de servidores a encender (o a apagar, según corresponda).

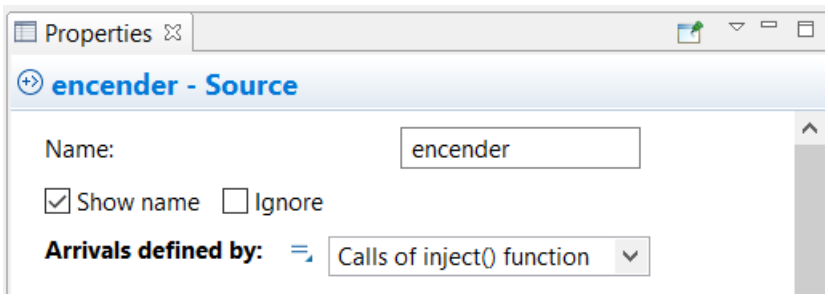

Figura 25. Llegada de servidores al proceso de encendido.

Una vez llegan al proceso, los servidores se encuentran en estado de arranque (o apagado) durante 1 segundo. Como es lógico, al tener 10 servidores en total en el sistema, esa es la capacidad máxima de servidores que se pueden someter al proceso de encendido (o apagado) a la vez.

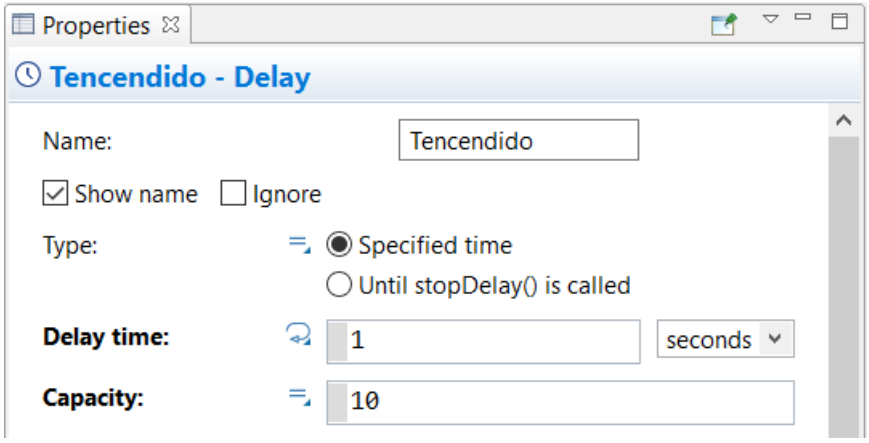

Figura 26. Tiempo de arranque de servidores.

Una vez que se ha cumplido este proceso, el número de servidores arrancados (o apagados) se suma (o resta) al que había anteriormente para calcular los servidores encendidos *mj*. En nuestra simulación partimos inicialmente con 5.

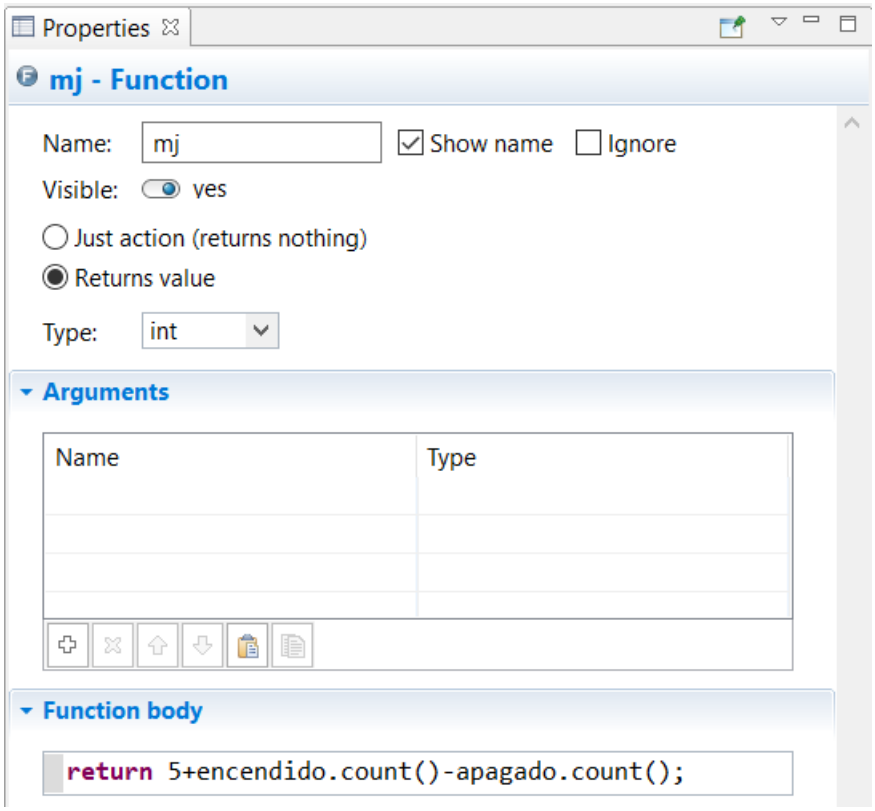

Figura 27. Servidores encendidos.

6.3 Implementación del modelado térmico

Llegados a este punto, es el momento de añadir los aspectos térmicos al modelo que queremos simular.

Para poder analizar térmicamente el modelo lo primero que debemos saber es qué servidores se encuentran encendidos y ocupados en cada momento. En la siguiente figura se puede observar el ejemplo para el servidor número 1, extrapolable a todos los demás.

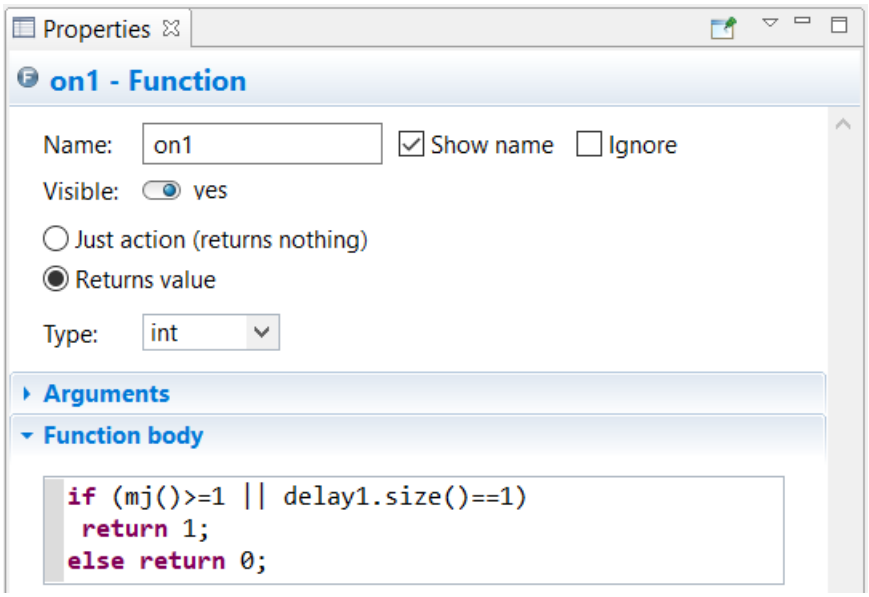

Figura 28. Evaluación de la ocupación de un servidor.

Conociendo si los servidores están encendidos, y, en ese caso, si están además ocupados, podemos saber la potencia de estos.

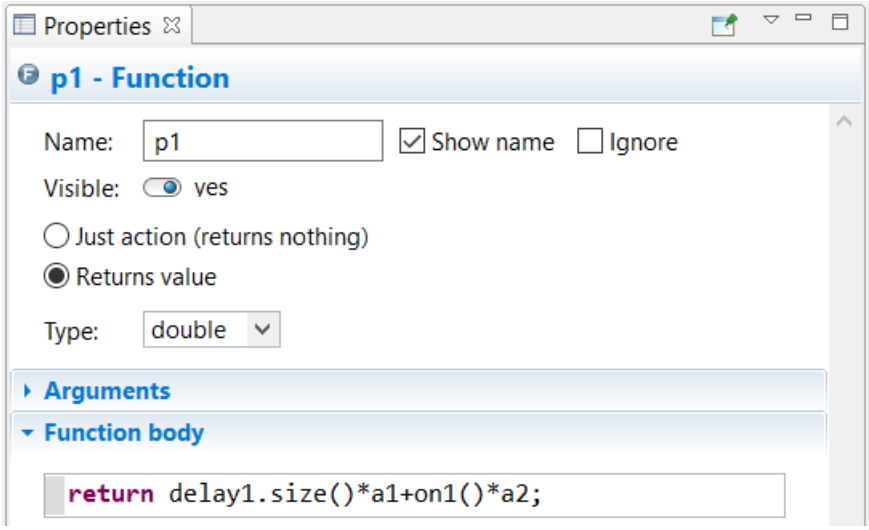

Figura 29. Evaluación de la ocupación de un servidor.

Con estos datos estamos en disposición de calcular las temperaturas del CRAC y de los servidores.

De la función de temperatura del CRAC:

$$\tau \frac{dT_c}{dt} = T_r - T_c$$

Podemos obtener su valor mediante tiempos discretos como:

$$T_{c,2} = T_{c,1} + \frac{\Delta t}{\tau}(T_r - T_c)$$

Donde $T_{c,2}$ es la temperatura del CRAC en el instante *k* y $T_{c,1}$ su valor en el instante *k-1*.

Partiendo del balance térmico de los servidores:

$$K_t \frac{dT_{ij}(t)}{dt} = c_p q_a(t)(T_c(t) - T_{ij}(t)) + p_{ij}(t, u_{ij}, \alpha_{ij})$$

Al considerar tiempos discretos, podemos definir la temperatura de cada servidor como:

$$T_{ij,2} = T_{ij,1} + \Delta t(\frac{c_p q_a}{K_t}(T_c - T_{ij,1}) + p_{ij}\frac{1}{K_t})$$

Siendo $T_{ij,2}$ la temperatura del servidor en el instante *k* y $T_{ij,1}$ su temperatura en el instante *k-1*.

Controlando todos estos datos, somos capaces de averiguar la temperatura del CRAC y de cada servidor en cada instante. En nuestra simulación se calcula en el siguiente evento, el cual se ejecuta cada milisegundo.

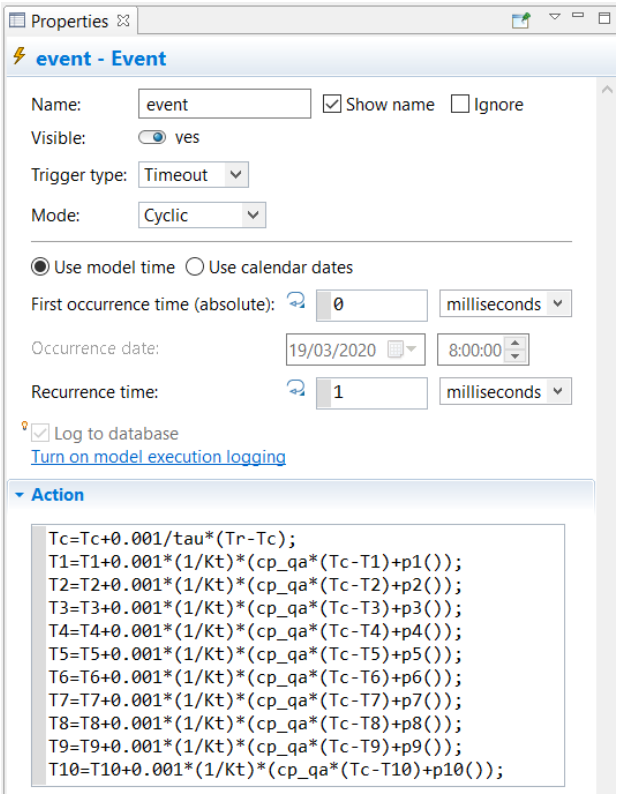

Figura 30. Evento de cálculo de temperaturas.

Como se puede observar, en este evento primero se calcula la temperatura del CRAC, y con ese valor actualizado, se procede a obtener el valor de la temperatura de cada uno de los 10 servidores.

Los parámetros, variables y funciones utilizados para el modelado térmico se encuentran todos definidos en Anylogic© con los valores establecidos en el capítulo 3 y los bloques descritos en el 5.

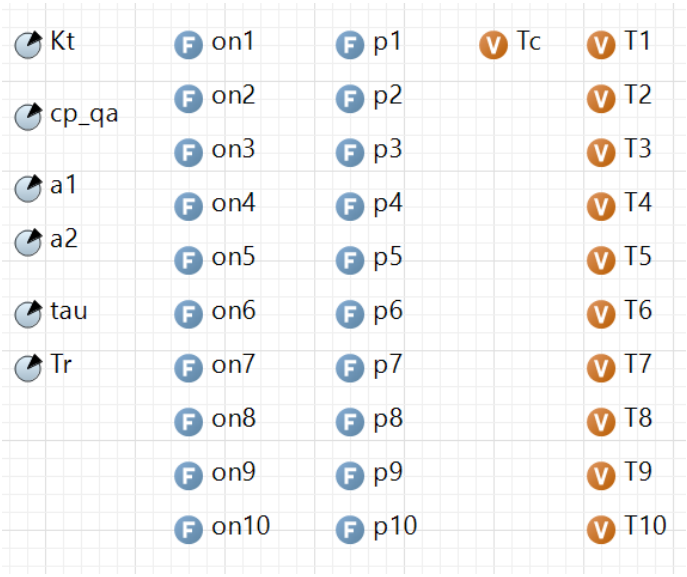

Figura 31. Parámetros, variables y funciones térmicos.

6.4 Implementación de los controladores

Ahora que tenemos representado el sistema por completo, necesitamos ser capaces de manipular las variables de entrada mediante controladores para asegurar unas condiciones idóneas de funcionamiento.

Las variables que controlaremos son el número de servidores encendidos (m_j) y la temperatura de consigna del aire del CRAC (T_r).

6.4.1 Implementación del controlador de la temperatura de consigna

Para implementar la ley de control de modere la temperatura de consigna, lo primero que debemos calcular es el error de la temperatura en cada instante k, como indicamos en el capítulo 4:

$$e(k) = 70ºC - T_{max}(k)$$

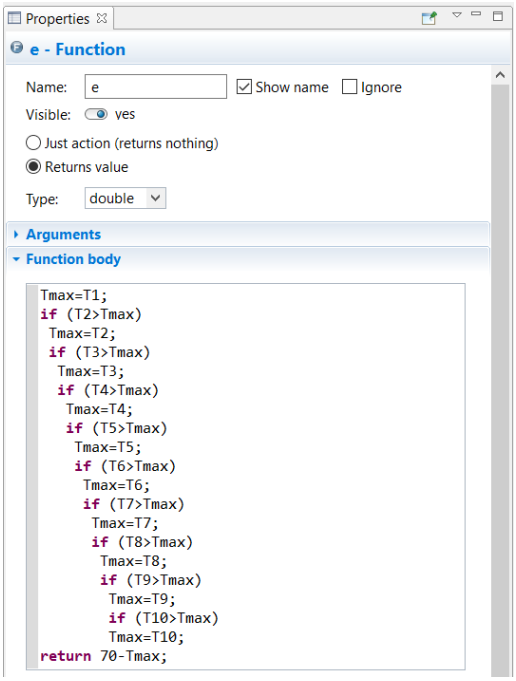

Figura 32. Cálculo del error de temperatura.

Como se puede apreciar en la Figura 32, calculamos $T_{max}(k)$ suponiendo que corresponde a la temperatura del primer servidor y comparándola una a una con la del resto hasta llegar al número 10. Una vez obtenido $T_{max}(k)$ calculamos $e(k)$.

El resto de cálculos para llegar al valor deseado de la temperatura de refrigeración (T_r) se realizan en el evento *regular_Tr* expuesto a continuación.

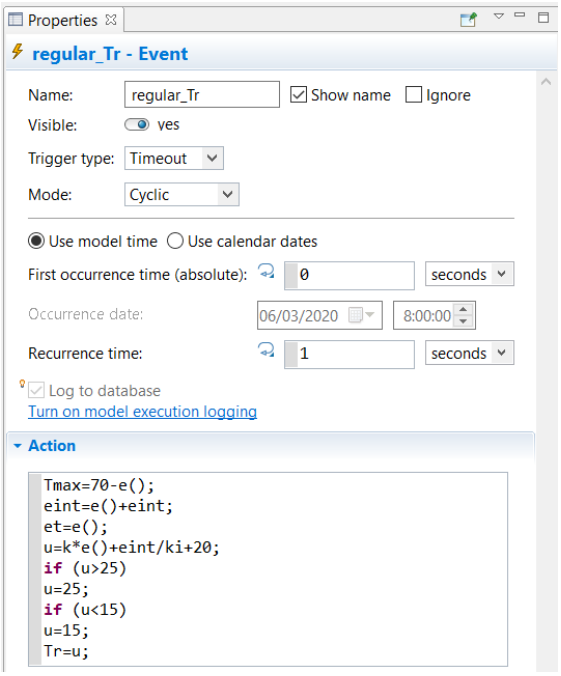

Figura 33. Evento de control de la temperatura de consigna.

En la Figura 33 se aprecia como se calculan las funciones definidas en el capítulo 4 para la aplicación del PI de la temperatura de consigna.

$$e_{int}(k) = e(k) + e_{int}(k-1)$$

$$u(k) = Ke(k) + \frac{e_{int}(k)}{K_i} + u_0$$

$$15ºC \leq u(k) \leq 25ºC$$

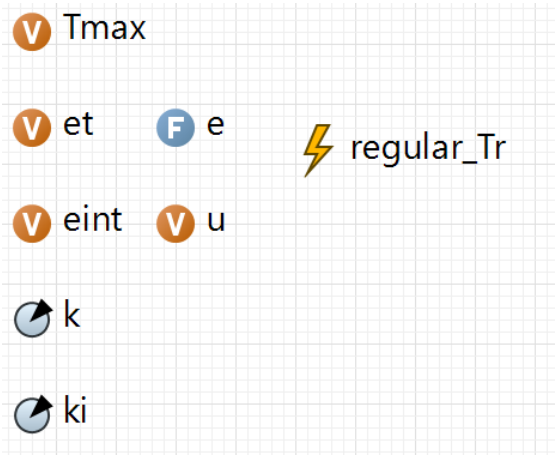

Figura 34. Bloques para el control de la temperatura de consigna.

6.4.2 Implementación del controlador de la cola de tareas

Para el caso del control de la cola, calculamos el error de la siguiente manera en Anylogic© (Figura 35) para que cumpla:

$$e(k) = q(k) - 50$$

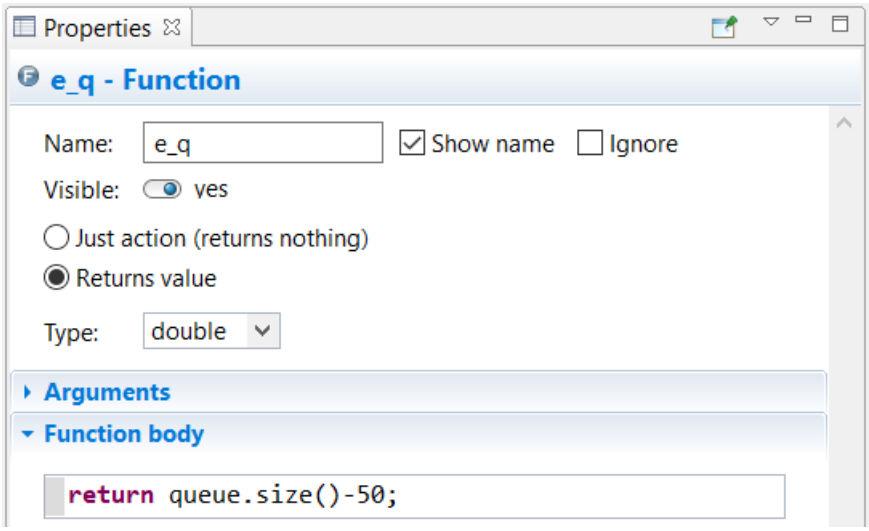

Figura 35. Cálculo del error de cola.

El evento de cálculo del número de servidores que deben estar encendidos es *regular_mj (Figura 36)*. En él se ejecutan los cálculos que explicamos en el capítulo 4:

$$e_{int}(k) = e(k) + e_{int}(k-1)$$

$$u(k) = Ke(k) + \frac{e_{int}(k)}{K_i} + u_0$$

$$1 \leq u(k) \leq 10$$

Como el resultado puede ser decimal y debemos tener un número entero de servidores encendidos, creamos la variable auxiliar *servs* como entero de la ley de control.

Una vez calculado, se compara con el número de servidores encendidos y la diferencia se manda al proceso de encendido si es positiva o al de apagado si es negativa.

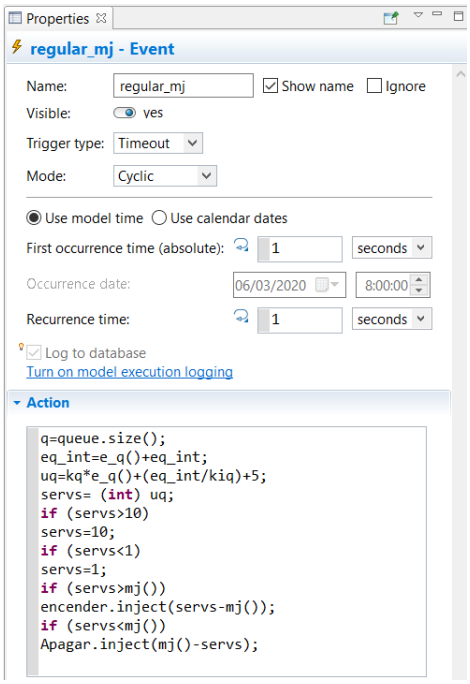

Figura 36. Evento de control de la cola.

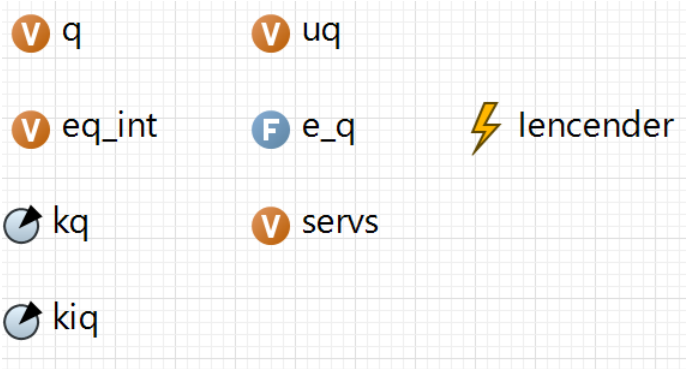

Figura 37. Bloques para el control de la cola.

7. SIMULACIÓN Y RESULTADOS

Procedemos a simular el modelo que hemos diseñado del centro de datos con los valores anteriormente comentados para los parámetros y durante un periodo de 100 segundos.

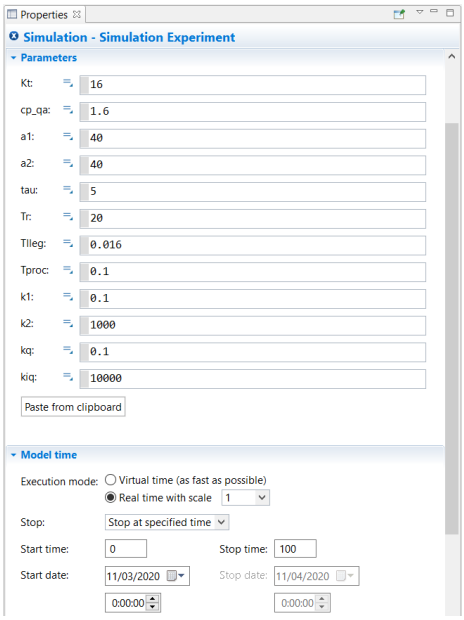

Figura 38. Configuración de la simulación.

7.1 Resultados de la simulación

Al ejecutar la simulación con la configuración descrita, obtenemos los siguientes resultados:

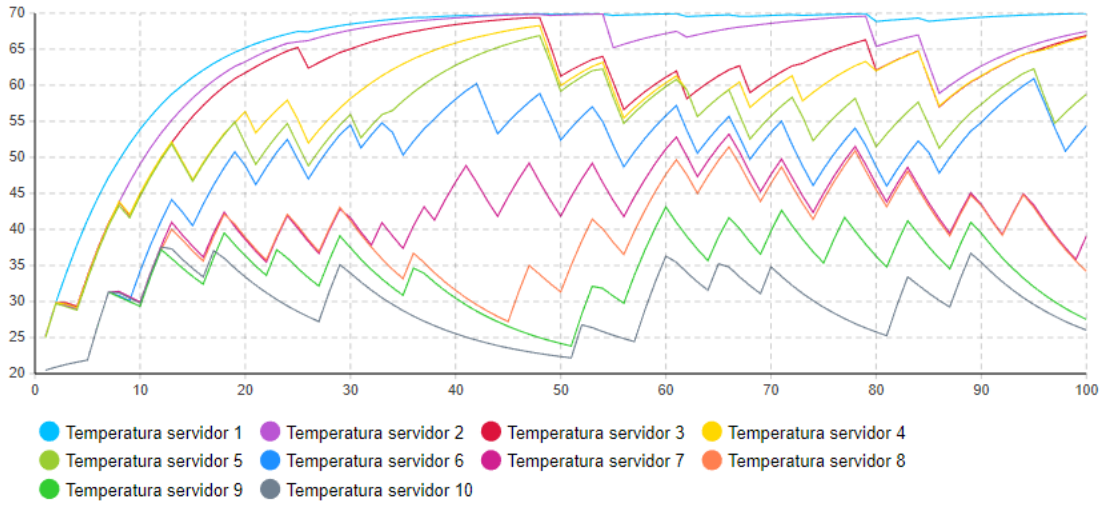

Figura 39. Resultado de las temperaturas de los servidores.

En la anterior gráfica se muestra la evolución de la temperatura de cada uno de los servidores respecto al tiempo de simulación. Como se puede observar, los 10 servidores presentan durante toda la simulación una temperatura inferior al umbral de 70ºC marcado por el controlador, por lo que se demuestra su efectividad.

Se aprecia también claramente un gran descenso de temperatura entre los servidores conforme mayor es su índice ordinal, debido a que utilizamos ese orden tanto para encenderlos como para asignarles las tareas. Por ejemplo, el servidor 6 no se encenderá si no están encendidos los 5 anteriores ni se le asignará una tarea si estos no están todos ocupados.

En la siguiente figura (40) se puede contrastar cómo los primeros servidores pasan mas tiempo encendidos (gráfica verde) y ocupados (gráfica azul) que los últimos.

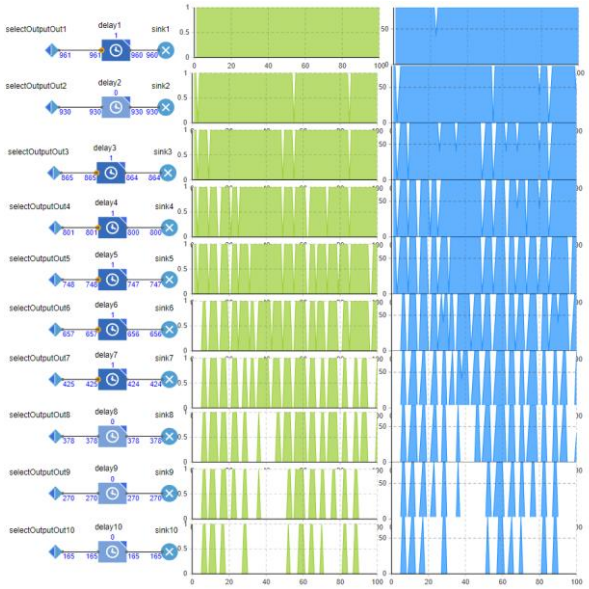

Figura 40. Resultado de los estados de los servidores.

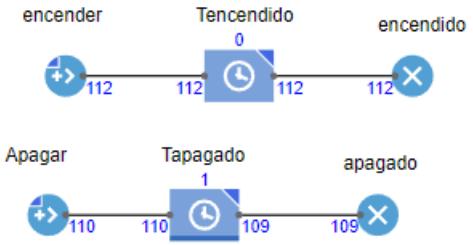

Figura 41. Resultado de los procesos de encendido y apagado de servidores.

Como se muestra en la Figura 41, el número de veces que se ejecutan los procesos de encendido (112) y apagado (110) es muy elevado para una simulación de tan solo 100 segundos.

En cuanto a la cola, nuestro objetivo era que el controlador la mantuviera próxima a 50 tareas durante el proceso. Como se puede ver a continuación (Figura 42), la cola se encuentra prácticamente en todo momento entre las 30 y 150 tareas, y se observan grandes pendientes en la curva debido a la elevada cifra de encendidos y apagados de servidores que acabamos de comentar.

Podemos atribuir estos resultados al hecho de que el tiempo de respuesta del controlador y de los procesos de encendido y apagado de servidores, todos de duración 1 segundo, es muy lento en comparación con la evolución de la cola, ya que se generan nuevas tareas con un tiempo medio entre llegadas de 0,016 segundos, lo que implica que el controlador manipule los servidores con un desfase de tiempo durante el cuál las desviaciones de la cola siguen aumentando, y cuando vuelve a calcularse la ley de control, el número de servidores que altera es también

elevado para contrarrestar dicha desviación, lo cuál produce en la gráfica esas pendientes pronunciadas.

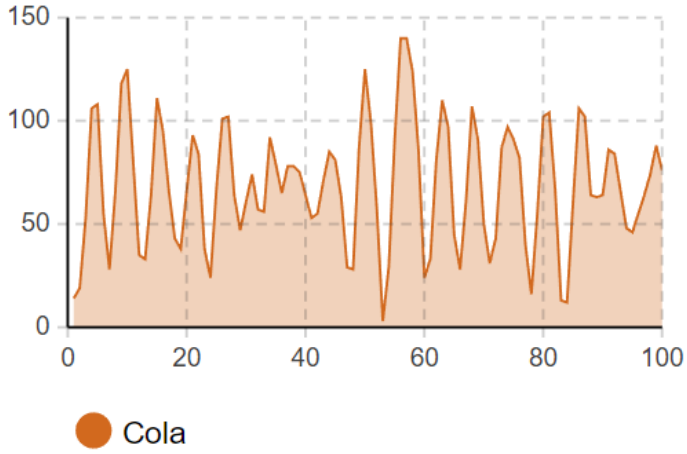

Figura 42. Resultado de la cola de tareas.

7.2 Conclusiones

Los controladores que hemos implementado consiguen que tanto la temperatura de los servidores como la cola de tareas se mantengan en unos niveles adecuados.

Sin embargo, creemos que se pueden afinar todavía más los resultados si se tienen en cuenta más detalles del sistema.

En cuanto a las temperaturas, sería interesante estudiar un criterio de encendido de los servidores y de reparto de las tareas que tenga en cuenta la temperatura, dando prioridad a aquellos servidores cuya temperatura sea menor, lo que produciría una menor diferencia térmica entre unos y otros y permitiría reducir la temperatura máxima.

Respecto a la ley de control implementada para la cola, consigue que esta no permanezca vacía ni se sature, manteniendo un nivel relativamente cercano a las 50 tareas establecidas a pesar de los inconvenientes mencionados. Esto se evitaría reduciendo el intervalo de cálculo del controlador o la duración de los procesos de encendido y apagado de los servidores.

Por otro lado, encender y apagar repetidamente los servidores (Figura 41) hace que estos se desgasten, lo que reduce su vida útil y aumenta su probabilidad de fallo, además de suponer un coste adicional, por lo que se podría valorar la opción de añadir estos factores y tenerlos en cuenta reduciendo así la alteración del estado de los servidores a cambio de tener un menor control del volumen de la cola.

REFERENCIAS

[1] B. Johnson, "How Data Centers Work", <https://computer.howstuffworks.com/data-centers.htm>.

[2] A. Capazzoli et al, "Cooling Systems in Data Centers: State of Art and Emerging Technologies", Energy Procedia 83, p. 484-493, 2015.

[3] Y. S. M. I. a. R. F. Miyuru Dayarathna, "Data Center Energy Consumption", IEEE COMMUNICATIONS SURVEYS & TUTORIALS, 2016.

[4] Fu et al, "Dynamic thermal and IT resource management strategies for data center energy minimization", Journal of Cloud Computing:Advances, Systems and Applications, 2017.

[5] G. S. V. G. Tang Q, "Energy efficient thermal aware task scheduling for homogeneous high performance computing data centers: A cyber physical approach.", IEEE Trans Parallel and distributed systems, p. 1458 1472, 2008.

[6] Anylogic Simulation Software, [En línea]. Available: https://www.anylogic.com/.

[7] K.J. Astrom, R.M. Murray, "Feedback Sistems", Princeton University Press, 2008.