



Universidad de Jaén

Escuela de Doctorado

FOLLOW THE RENEWABLE-BASED VIRTUAL MACHINE MIGRATION AMONG GEO-DISPERSED MODULAR DATA CENTERS WITH EXPERT SYSTEMS

Autor: Doraid Seddiki Ben Friha

Directores de la tesis:

Dr. Sebastián García Galán

Dra. Rocío Pérez de Prado

Fecha: JUNIO 2024

RUJJA

Presidente D.

Vocal D.....

Secretario D.....

Realizado el acto de defensa y lectura de la Tesis el día

De de 2024.

En

Calificación:

EL PRESIDENTE

EL VOCAL

EL SECRETARIO

DEDICATORIA

بِسْمِ اللَّهِ الرَّحْمَنِ الرَّحِيمِ

En el nombre de Allah, el Compasivo, el Misericordioso.

Allah, a Ti dedico esta obra, fruto de Tu guía y misericordia. Tú dijiste a Tu Profeta "اقْرَأْ بِاسْمِ رَبِّكَ الَّذِي خَلَقَ" (Lee en el nombre de tu Señor, quien creó), encendiendo en nosotros la llama del conocimiento. Con Tu promesa "فَإِنَّ مَعَ الْعُسْرِ يُسْرًا" (Ciertamente, con la dificultad viene la facilidad), encontré consuelo y fortaleza en cada desafío. Mi súplica constante siempre fue "رَبِّ زِدْنِي عِلْمًا" (Señor, concédeme conocimiento), y en Tu generosidad, me has permitido aprender y crecer. En Tu promesa "وَلَسَوْفَ يُعْطِيكَ رَبُّكَ فَتَرْضَى" (Y tu Señor te dará hasta que quedes satisfecho), encuentro esperanza y determinación. Al recordar Tus palabras "فَاذْكُرُونِي أَذْكَرُكُمْ" (Recordadme y yo os recordaré), siento Tu presencia y bendición. "وَمَا تَوْفِيقِي إِلَّا بِاللَّهِ" (Y mi éxito no puede venir sino de Allah), en Ti confío y a Ti regreso en cada momento de necesidad. Alabado seas, por haberme guiado hasta aquí. "الحمد لله الذي هدانا لهذا وما كنا لنهتدي لولا أن هدانا الله". En Tu luz encuentro mi camino, y en Tu amor, mi propósito.

A mi querido abuelo, cuya visión y fe en mí desde mi infancia fueron faros que iluminaron mi camino. Me llamabas "Doctor Doraid" cuando apenas era un niño, un nombre que se convirtió en mi meta y motivación. Si hoy presento esta memoria es para honrarte, gracias querido abuelo, que tu alma repose en paz.

A mi querida abuela, mi "Mani" te fuiste demasiado pronto, me hubiese encantado que hoy asistieras a este logro, tu logro. Gracias por quererme, cuidarme y hacerme sentir importante. Descansa en paz.

A mi querida madre, cuyo sacrificio y dedicación hicieron posible todos mis logros. Su lucha incansable por nuestra educación es la verdadera razón de este éxito. Madre, este logro es tan tuyo como mío; tu esfuerzo y dedicación están presentes en cada página de esta tesis. Tu valentía y determinación son un ejemplo para mí y para todos aquellos que te conocen.

A mi querido padre, a quien me parezco tanto y de quien he heredado la pasión por el conocimiento y la verdad. Tus cualidades de inconformista e intelectual empedernido han sido una fuente constante de inspiración para mí. Tu espíritu indomable y tu búsqueda incansable del conocimiento han dejado una huella profunda en mí.

A mi querido hermano, hombre sabio, honorable, amable y de profunda fe donde los haya. Tu generosidad, estima y respeto hacia mí han sido un pilar fundamental. Tu aliento constante y tu fe inquebrantable en mí han sido una fuente de fuerza inagotable. Este logro es también tuyo, por estar siempre a mi lado, apoyándome en cada paso del camino.

A mi querida esposa, cuyo amor y sacrificio han sido fundamentales para que pudiera alcanzar este logro. Tu carácter virtuoso, tu paciencia y tu fortaleza han sido mi refugio y mi motivación en los momentos más difíciles. Has soportado conmigo las noches largas y los días complicados, brindándome siempre tu apoyo incondicional. Tu presencia en mi vida es una bendición incomparable. Este logro no hubiera sido posible sin ti.

"El trabajo es el amor hecho visible." – Jabran, Kahlil Jabran

Al final, esta obra es una manifestación del amor y el apoyo que he recibido de todos ustedes. Cada página de esta tesis está impregnada de vuestro amor, paciencia y fe en mí. Vuestras acciones y palabras me han dado la fortaleza para continuar, y vuestro amor se ha convertido en la fuerza motriz de este trabajo. Esta dedicatoria es para ustedes, que habéis hecho que este trabajo sea una realidad visible y tangible. Gracias.

Doraid Seddiki.

AGRADECIMIENTOS

En la filosofía estoica, se dice que la gratitud es la madre de todas las virtudes, y no hay mayor virtud que reconocer el impacto positivo de quienes han sido fundamentales en nuestro camino. Es por eso que, de manera premeditada, comienzo estos agradecimientos con la persona que ha sido el pilar fundamental de este proyecto: mi director de tesis, Sebastián García Galán.

Dirigir una tesis es un desafío que pocos pueden asumir con la excelencia y humanidad que tú, Sebastián, has demostrado. Durante este viaje, marcado por desafíos personales y académicos, tu guía ha sido un faro de esperanza y motivación. No solo me brindaste una orientación académica de primer nivel, sino que también estuviste a mi lado como amigo en los momentos más oscuros. Tu apoyo constante, tu paciencia y tu amistad han sido esenciales para la realización de esta tesis. Gracias por todo, Sebastián.

A Rocío Pérez Prado, mi codirectora de tesis, quiero expresarte mi sincero agradecimiento. Tu papel fue fundamental en la redacción de los objetivos y el plan de esta tesis. Además, tu meticuloso trabajo de revisión y los valiosos consejos que me brindaste fueron esenciales para mejorar la calidad de este documento. Tu apoyo y orientación han sido invaluable, y por ello te estoy profundamente agradecido.

También agradezco profundamente al Departamento de Ingeniería de Telecomunicación en general, y de manera especial, a mi increíble grupo de investigación, por su apoyo y contribuciones esenciales durante esta travesía.

José Enrique Muñoz Expósito, gracias por tus orientaciones y consejos técnicos, tu paciencia, y por ser un profesor que no solo me enseñó, sino que también me inspiró y motivó a seguir adelante.

Pedro Aguilar, nuestro técnico de laboratorio, siempre al rescate cuando SERBOT1 y 2 decidían que era hora de hacer un show de luces al estilo árbol de Navidad. Gracias, Pedro, por evitar que esos artilugios salieran ardiendo y por estar siempre dispuesto a echar una mano con una sonrisa.

Fran Francisco Javier Maldonado y Antonio Jiménez, a ustedes, por su invaluable ayuda y aportes en la redacción de los Papers de investigación. Su dedicación y disposición para colaborar han sido invaluable para el éxito de este proyecto.

En el transcurso de esta travesía, hubo tres personas cuyo apoyo y generosidad fueron esenciales para llegar a esta meta.

El Arbi El Harti, siempre creyendo en mí, incluso cuando las cosas se ponían difíciles. Tus consejos para la redacción de esta tesis fueron tan valiosos como tus obras literarias. Gracias por tu inspiración y apoyo incondicional.

Hicham Ait Meskine, mi compañero de carrera, mi amigo y hermano. Eres probablemente la única persona que se alegra más que yo de este logro. Tu generosidad es legendaria y tu abnegación, un ejemplo a seguir. Gracias por siempre estar ahí y por desear mi éxito con tanta pasión.

Mi tío Hassan, tu apoyo durante mis estudios y en los momentos difíciles ha sido fundamental. Eres alguien en quien siempre puedo confiar, y tu generosidad y orgullo por mis logros me han dado la fuerza necesaria para seguir adelante.

Y finalmente, después de casi 15 años en el mundo corporativo, me animé a seguir mi sueño de ser emprendedor, y no lo habría logrado sin el apoyo de mis socios y amigos, Fayçal y Mehdi. Desde el primer momento, compartimos una visión y aspiraciones comunes que nos impulsaron a formar nuestra propia empresa.

Fayçal y Mehdi, durante la etapa final de esta tesis, ustedes fueron un valioso soporte. Me dieron el espacio necesario, me protegieron para que pudiera enfocarme en terminar la redacción y siempre me animaron, haciéndome sentir valorado y respetado. Este logro no habría sido posible sin su incondicional apoyo. Gracias.

Contenido

1	Introducción	14
2	Estado del arte	23
2.1	Introducción al Estado del Arte.....	24
2.2	Historia y evolución del cloud computing	28
2.3	Modelos de Servicio en el cloud y su Aplicación en CDCs Sostenibles.....	33
2.4	Desafíos en Gestión de Energía en el cloud computing	36
2.4.1	Importancia de la Eficiencia Energética.....	37
2.4.2	Desafíos de Sostenibilidad	39
2.4.3	Estrategias Actuales y Limitaciones.....	40
2.5	Sistemas expertos y Fuzzy logic en el cloud computing.....	49
2.5.1	introducción.....	49
2.5.2	Ventajas Distintivas de la Lógica Difusa en el Cloud.....	50
2.5.3	Sistemas Expertos Basados en Reglas Difusas, clasificación y aplicación	50
2.5.4	Clasificación de los sistemas basados en reglas difusas.....	51
2.6	Machine learning en el cloud computing	60
2.6.1	Introducción	60
2.6.2	Asignación y Ubicación Eficiente de Máquinas Virtuales.....	61
2.6.3	Gestión de Máquinas Virtuales en Tiempo de Ejecución para la Optimización Energética.....	74
2.6.4	Predicción de Carga de Trabajo en Cloud computing	90
2.6.5	Gestión Térmica con ML	94
2.7	Conclusión.....	97
3	Contribuciones y aspectos metodológicos	99
3.1	Introducción	100
3.2	Contribución 1 - Sistema experto basado en reglas borrosas para la optimización de energía renovable en CDCs geográficamente distantes	101
3.3	Contribución 2 - Sistema de adquisición del conocimiento para la mejora del comportamiento del sistema borroso.....	103
3.4	Aspectos metodológicos.....	104
3.4.1	Escenarios de simulación	104
3.4.2	Comparación de Algoritmos	106
3.4.3	Evaluación del Rendimiento de Algoritmos mediante Técnicas Paramétricas y No Paramétricas	110
3.4.4	Simulación con CloudSim.....	112
3.4.5	Emulación de Hardware real para las simulaciones	116
3.4.6	Desarrollo de un sistema experto basado en FRBS para implementar la Políticas de Migración	118

3.4.7	Integración de Swarm intelligence	122
3.4.8	Configuración de la Simulación en Matlab	141
3.4.9	Consideraciones Éticas y Limitaciones	142
3.5	Conclusión.....	144
4	Resultados y análisis	146
4.1	Introducción	147
4.2	Resultados Previos a la contribución 1 - Comparativa de eficiencia renovable para los algoritmos IQR, LR, LRR, THR y RODI	148
4.2.1	Introducción	148
4.2.2	Descripción de los escenarios de simulación	151
4.2.3	Resultados experimentales	152
4.3	Resultados contribución 1 - Comparativa de eficiencia renovable para los algoritmos IQR, LR, LRR, THR, RODI y FRBS con conocimiento de experto	190
4.3.1	Introducción	190
4.3.2	Descripción de los escenarios de simulación	191
4.3.3	Resultados experimentales	193
4.4	Resultados contribución 2 - Comparativa de eficiencia renovable FRBS con conocimiento de experto, FRBS con aprendizaje KASIA y FRBS con aprendizaje KAGWO 199	
4.4.1	Introducción	199
4.4.2	Descripción de los escenarios de simulación	200
4.4.3	Resultados experimentales	200
5	Conclusiones, limitaciones y líneas futuras	207
5.1	Conclusiones y limitaciones.....	208
5.2	Líneas futuras	209
6	Publicaciones.....	212
6.1	Derivadas de la tesis.....	213
6.1.1	Sustainable expert virtual machine migration in dynamic clouds.....	213
6.1.2	Enhanced Virtual Machine Migration for Energy Sustainability Optimization in Cloud computing through Knowledge Acquisition.....	214
6.1.3	Sustainability-based Framework for Virtual Machines Migration Among Cloud data centers.....	215
6.2	Relacionadas con la tesis.....	216
6.2.1	Multi-Objective Optimization of Virtual Machine Migration among Cloud data centers	216
6.2.2	Knowledge acquisition approaches for virtual machine migration in cloud computing.....	217
6.2.3	Comparison between Fuzzy and Neuro-Fuzzy Inference Systems in Cloud computing Scheduling.....	218

6.2.4	Optimized interpretability for Expert Virtual Machine Migrations among Data Centers using Fingrams	221
7	Bibliografía	224
8	Tabla de Tablas	254
9	Tabla de Ilustraciones	257
10	Tabla de Abreviaciones	262

RESUMEN

Esta tesis doctoral aborda el creciente problema del consumo energético en la industria del cloud computing, un sector que ha experimentado un crecimiento exponencial en las últimas dos décadas, impulsado por gigantes tecnológicos como Google, Amazon y Microsoft. A medida que los Cloud data centers (CDC) se han expandido para satisfacer la demanda de almacenamiento y procesamiento de datos, también ha aumentado el impacto ambiental derivado del uso intensivo de electricidad, que a menudo proviene de fuentes no renovables [Shuja et al., 2016; Srinivasan, 2014]. Se estima que los CDC consumen más del 2,5% de la electricidad mundial, con un impacto económico cercano a los 30 billones de dólares anuales, contribuyendo al 2% de las emisiones globales de carbono [Meijer, 2010; Brown, 2007].

El objetivo central de esta tesis es desarrollar estrategias para hacer que los CDC sean más eficientes y sostenibles, mediante la integración de fuentes de energía renovable y técnicas de planificación inter-cloud. Los Centros de Datos Modulares (MDC) permiten la migración de máquinas virtuales (VM) entre CDC distribuidos geográficamente, ajustando la carga de trabajo según la disponibilidad de energía renovable [Google, 2012]. Sin embargo, este enfoque requiere mecanismos de planificación eficientes para gestionar la variabilidad y minimizar el impacto ambiental [Shuja et al., 2016].

Los sistemas expertos borrosos son especialmente útiles en este contexto, ya que permiten gestionar la incertidumbre mediante lógica aproximada y reglas "IF-THEN" [Cordón, 2001]. Este enfoque es crucial para abordar la variabilidad de los recursos en los CDC y la incertidumbre asociada a la disponibilidad de energías renovables, que depende de factores ambientales y temporales [Srinivasan, 2014]. Los objetivos principales de esta tesis son:

1. Estudiar y adaptar sistemas expertos borrosos para el diseño de planificadores inter-cloud "follow the renewables". Esta estrategia busca una migración eficiente de máquinas virtuales entre CDCs, maximizando el uso de energías renovables. La lógica borrosa permite procesar información imprecisa y tomar decisiones fundamentadas para optimizar el consumo energético [Shuja et al., 2016].
2. Aplicar mecanismos de adquisición de conocimiento para mejorar la eficiencia de los planificadores inter-cloud expertos. El rendimiento de un sistema experto depende de la calidad de sus bases de conocimiento y de la estrategia de aprendizaje. Se propone el uso

de técnicas Inteligencia Artificial (IA) de aprendizaje, en sus vertientes evolutivas y bioinspiradas para mejorar la eficiencia de los sistemas expertos, permitiendo una planificación más precisa y sostenible en entornos cloud [Cordón, 2001].

La tesis evalúa el sistema propuesto mediante simulaciones con WorkflowSim-DVFS, una extensión del simulador WorkflowSim que permite estimar el consumo energético y gestionar la migración de VM [Cotes-Ruiz et al., 2017]. Los resultados obtenidos servirán para validar la eficacia del sistema experto y su impacto en términos de eficiencia energética y sostenibilidad.

En resumen, esta tesis busca soluciones innovadoras para abordar el problema del alto consumo energético en el cloud computing, mediante el uso de sistemas expertos borrosos y técnicas de machine learning bioinspiradas para la planificación inter-cloud. El enfoque propuesto tiene el potencial de contribuir a la creación de CDC más sostenibles, reduciendo el impacto ambiental y promoviendo una industria del cloud computing más ecológica.

ABSTRACT

This doctoral thesis addresses the growing issue of energy consumption in the cloud computing industry, a sector that has experienced exponential growth over the past two decades, driven by technology giants such as Google, Amazon, and Microsoft. As Cloud data centers (CDCs) have expanded to meet the demand for data storage and processing, the environmental impact resulting from the intensive use of electricity, often derived from non-renewable sources, has also increased [Shuja et al., 2016; Srinivasan, 2014]. It is estimated that CDCs consume more than 2.5% of the world's electricity, with an economic impact close to \$30 trillion annually, contributing to 2% of global carbon emissions [Meijer, 2010; Brown, 2007].

The central objective of this thesis is to develop strategies to make CDCs more efficient and sustainable through the integration of renewable energy sources and inter-cloud planning techniques. Modular Data Centers (MDCs) allow the migration of virtual machines (VMs) between geographically distributed CDCs, adjusting the workload according to the availability of renewable energy [Google, 2012]. However, this approach requires efficient planning mechanisms to manage variability and minimize environmental impact [Shuja et al., 2016].

Fuzzy expert systems are particularly useful in this context, as they allow uncertainty to be managed through approximate logic and "IF-THEN" rules [Cordón, 2001]. This approach is crucial to addressing the variability of resources in CDCs and the uncertainty associated with the availability of renewable energies, which depends on environmental and temporal factors [Srinivasan, 2014]. The primary objectives of this thesis are:

1. To study and adapt fuzzy expert systems for the design of inter-cloud "follow the renewables" schedulers. This strategy aims for efficient migration of virtual machines among CDCs, maximizing the use of renewable energies. Fuzzy logic allows the processing of imprecise information and informed decision-making to optimize energy consumption [Shuja et al., 2016].
2. To apply knowledge acquisition mechanisms to improve the efficiency of inter-cloud expert schedulers. The performance of an expert system depends on the quality of its knowledge bases and the learning strategy employed. This thesis proposes the use of

evolutionary and bio-inspired learning techniques to improve the efficiency of expert systems, allowing for more precise and sustainable cloud scheduling [Cordón, 2001].

The thesis evaluates the proposed system through simulations with WorkflowSim-DVFS, an extension of the WorkflowSim simulator that allows estimating energy consumption and managing VM migration [Cotes-Ruiz et al., 2017]. The results will validate the effectiveness of the fuzzy expert system and assess its impact in terms of energy efficiency and sustainability.

In summary, this thesis seeks innovative solutions to address the high energy consumption in cloud computing by using fuzzy expert systems and bio-inspired machine learning techniques for inter-cloud planning. The proposed approach has the potential to contribute to creating more sustainable CDCs, reducing environmental impact, and promoting a greener cloud computing industry.

1 Introducción

El crecimiento exponencial de las actividades digitales, como la interacción en redes sociales, la búsqueda en internet, el intercambio de archivos y los servicios de streaming, está generando cantidades masivas de datos que requieren ser procesados, almacenados y transmitidos [Shuja et al., 2016]. Este incremento en la demanda de servicios digitales está directamente relacionado con el aumento del consumo energético de los CDC. Los modelos de negocio y los acuerdos de nivel de servicio (SLA) exigen que las instalaciones de almacenamiento y procesamiento se repliquen de forma redundante para proporcionar tolerancia a fallos y minimizar los retrasos en el servicio [Srinivasan, 2014]. En consecuencia, los proveedores de servicios de TI operan CDC las 24 horas del día, los 7 días de la semana, con miles de servidores y dispositivos de almacenamiento y servicios de red para garantizar una disponibilidad del 99.99% de los servicios en la nube [Ali et al., 2016; Bilal et al., 2014; Shuja et al., 2012].

Este alto nivel de disponibilidad y redundancia lleva a un consumo energético considerable y a un impacto ambiental significativo en forma de emisiones contaminantes [Romero et al., 2014]. La viabilidad de un sistema económico sostenible es uno de los principales desafíos a los que se enfrenta el mundo hoy en día, y los CDC representan uno de los mayores consumidores de electricidad. Se estima que los CDC consumen más del 2.5% de la electricidad de todo el mundo, con un impacto económico global de 30 billones de dólares anuales [Meijer, 2010]. Además, se estima que los centros de datos han sido responsables de la emisión del 2% de las emisiones contaminantes a nivel mundial en las últimas décadas, y que esta cifra ha aumentado en la actualidad [Brown, 2007; Shuja et al., 2016].

Estas cifras impulsan la necesidad de aplicar elementos innovadores y medidas disruptivas en los CDC para conseguir una mayor eficiencia en el uso de la energía y reducir las emisiones contaminantes. Entre los posibles ámbitos de trabajo en este sentido, los CDC sostenibles y verdes requieren la aplicación de múltiples técnicas y tecnologías. Dos de las técnicas fundamentales son, en primer lugar, el empleo de centros de procesamiento modulares que se ubiquen de forma estratégica para permitir el uso de energías renovables y, en segundo lugar, las técnicas de migración de carga dinámica que permitan la ejecución de la carga en el centro de datos CDC que tenga la mayor posibilidad de consumir energías renovables, conocidas como estrategias "follow the renewables" [Shuja et al., 2016].

Proveedores de servicios de TI como Google y Facebook están apostando por técnicas de migración que permitan el uso de energías renovables en centros de datos distribuidos geográficamente de forma estratégica, acuñando el término de CDC sostenibles para aquellos que permiten la aplicación de estrategias y tecnologías orientadas a la reducción del consumo energético y de emisiones [Google, 2012; Wang, 2014]. Estas estrategias incluyen la integración de energía eólica y fotovoltaica en la operación diaria de los CDC, lo cual reduce significativamente la dependencia de fuentes de energía no renovables.

Los planificadores (schedulers) son los elementos software encargados de la asignación de tareas a los diferentes nodos de los CDC y MDC bajo la consideración de múltiples criterios y configuraciones. Constituyen uno de los componentes principales de la estructura de procesamiento en la nube, interactuando con otros componentes como los sistemas de información de los sistemas de gestión de recursos locales y los sistemas de mantenimiento de red para realizar su función. Específicamente, los planificadores inter-cloud se encargan de la coordinación de los planificadores de CDC para conseguir una planificación global eficiente de toda la nube [Shuja et al., 2016]. La interacción y coordinación de estos planificadores es crítica para la eficiente ejecución de las tareas.

En particular, es necesario el diseño de estrategias de planificación capaces de trabajar con la incertidumbre inherente y el dinamismo de las redes cloud con el fin de reducir el consumo de energía del CDC y realizar el procesamiento en aquellos nodos o se permita un consumo de energía de origen renovable. Así, se buscan nuevas propuestas de planificación capaces de tolerar y gestionar la incertidumbre respecto al estado de la red, de forma que se pueda ofrecer un procesamiento sostenible basado en una descripción precisa de las condiciones reales del cloud [Liu et al., 2011; Li et al., 2015; Shojafar et al., 2016].

Las técnicas de inteligencia artificial Soft-Computing (SC) [Cordón et al., 2001] están destinadas a operar en entornos sujetos a la incertidumbre e imprecisión. Su aplicación puede ser beneficiosa en entornos descritos como inherentemente dinámicos y llenos de incertidumbre, como las redes cloud, donde no se puede migrar la carga a centros de procesamiento sin tener en cuenta un estado "conocido" del sistema. En esta tesis se propone la aplicación de diversas técnicas de SC para la mejora de los planificadores inter-cloud o CDC schedulers. En concreto, se centra en la aplicación de la lógica borrosa y la computación evolutiva (EC) para el diseño de meta-planificadores

borrosos basados en reglas que incorporan estrategias de adquisición de conocimiento para su aprendizaje automático.

Los sistemas borrosos basados en reglas (FRBS) son sistemas expertos derivados de la Lógica Borrosa y de sistemas basados en reglas que se proponen en esta tesis como una alternativa para el desarrollo de planificadores inter-cloud. Los FRBS usan FL para la caracterización de las variables del sistema y basan sus decisiones en reglas "IF-THEN" que representan afirmaciones borrosas, relacionando estas variables. De esta forma, son capaces de proporcionar decisiones en problemas complejos donde existe incertidumbre mediante un proceso de razonamiento aproximado con características similares al humano [Cordón et al., 2001]. El objetivo del planificador inter-cloud es seleccionar, en cada etapa de la planificación, aquel CDC que permita el uso de energías renovables con un consumo de energía reducido y realizar las operaciones de migración de carga necesarias para maximizar el uso de dichas energías renovables [Shuja et al., 2016].

El objetivo fundamental de esta tesis es el diseño e implementación de "follow the renewables" schedulers en los CDCs. En esta investigación se propone diseñar estos sistemas teniendo en cuenta que la información sobre el estado de los CDC provista por los sistemas de información posee un grado de imprecisión debido al alto dinamismo de la red y a la variabilidad en la disponibilidad de energía renovable. Se propone que el planificador inter-cloud siga la estructura general de los FRBS. De esta forma, se deben estudiar las variables que describen el estado de cada dominio de recursos del cloud y diseñar los componentes fundamentales que proporcionan la capacidad de razonamiento borroso: el sistema fuzzificador, el motor de inferencia, el sistema defuzzificador y las bases de conocimiento.

El funcionamiento de los FRBS está fuertemente relacionado con la calidad de sus bases de conocimiento, que describen las reglas y la representación de las variables y, por tanto, con sus procesos de aprendizaje [Cordón et al., 2001]. Debido a la naturaleza cambiante y los complejos espacios de búsqueda típicos de los sistemas de computación cloud "follow the renewables", la obtención automática de una base de conocimiento de alta calidad es extremadamente relevante. En esta tesis se propondrá también la aplicación de estrategias de aprendizaje para la obtención de bases de reglas (RB) para sistemas de planificación inter-cloud basados en FRBS, derivadas de la computación evolutiva, que permitan mejorar el funcionamiento de dichos sistemas expertos.

Esta tesis doctoral se enmarca en la línea de investigación “Aplicación de Técnicas de Inteligencia Artificial en Computación Grid/Cloud” que se desarrolla en la Escuela Politécnica Superior de Linares dentro del grupo de investigación "Tratamiento de Señales en Sistemas de Telecomunicación" del Plan Andaluz de Investigación, Desarrollo e Innovación (PAIDI), con código TIC-188.

En cuanto a los resultados previos del equipo investigador, los más relevantes se centran en diversas técnicas de adquisición de conocimiento o de aprendizaje automático, así como en meta-planificadores grid/cloud. La difusión científica de estos resultados incluye la publicación de una treintena de artículos en revistas y congresos internacionales de gran relevancia [Garcia-Galan - e et al., 2009] [Garcia-Galan -c et al., 2012; Garcia-Galan -b et al., 2014; Garcia-Galan -a et al., 2015; Muñoz et al., 2015; Prado -d et al., 2010; Prado -e et al., 2010; Prado -c et al., 2011; Prado -b et al., 2012; Prado et al., 2014] y la obtención de proyectos de investigación, como “Optimización de la sostenibilidad energética en centros de computación cloud mediante planificación experta con análisis de la interpretabilidad”, código P18-RT-4046, financiado por el Plan andaluz de investigación, de la junta de Andalucía siendo el investigador principal el Dr. Sebastián García Galán.

La investigación de esta tesis se fundamenta en dos hipótesis principales:

HIPÓTESIS 1:

Si la aplicación de sistemas borrosos basados en reglas al diseño de sistemas planificadores inter-cloud "follow the renewables" consigue una gestión eficiente de la información sobre el estado de procesamiento inherentemente incierto de los centros de datos sostenibles del cloud y de su accesibilidad a energías renovables, se espera maximizar el uso de energía de origen renovable en un entorno altamente dinámico.

HIPÓTESIS 2:

Si el funcionamiento de los sistemas planificadores inter-cloud "follow the renewables" borrosos basados en reglas en redes cloud sostenibles depende fuertemente de la calidad de sus bases expertas y de la velocidad de la obtención de éstas, se espera incrementar la eficiencia de su planificación mediante procesos de aprendizaje automático que les permitan mejorar tanto el nivel como la forma de adquisición de su conocimiento.

Para alcanzar el objetivo general de la investigación, se plantean los siguientes objetivos específicos:

1. Estudio y adaptación de sistemas expertos borrosos para el diseño de planificadores inter-cloud "follow the renewables" que permitan una migración de máquinas virtuales de forma eficiente. Las redes cloud, por definición, son redes de gran dinamismo y de naturaleza cambiante, donde el estado de los recursos presenta un alto grado de incertidumbre [Srinivasan, 2014]. En las redes cloud sostenibles, esta incertidumbre se agrava debido a la variabilidad en la disponibilidad de energías renovables, que depende de condiciones ambientales y temporales.

2. Aplicación de mecanismos de adquisición de conocimiento a los planificadores inter-cloud "follow the renewables" expertos. La eficiencia de un sistema experto puede verse comprometida por la calidad de sus bases de conocimiento y la estrategia de aprendizaje considerada [Cordón et al., 2001]. Se propone el estudio y aplicación de estrategias clásicas de aprendizaje en sistemas borrosos basados en reglas para mejorar su funcionamiento.

3. Extensión de la plataforma de simulación inter-cloud no sostenible a plataforma de simulación inter-cloud sostenible basada en CDC. Esto incluye la implementación de la plataforma CDC, el estudio e implementación de la modificación de energía renovable en nodos CDC sostenibles, la adaptación y aplicación de trazas reales a la plataforma y la implementación de capacidades para la migración de máquinas virtuales.

4. Adaptación de sistemas borrosos a planificadores inter-cloud "follow the renewables". Esto implica la caracterización del estado de carga y acceso a energía renovable de cada nodo o CDC en la red cloud sostenible, y la especificación de las características del sistema experto borroso basado en reglas Mamdani, adaptado a la caracterización del estado del cloud sostenible.

5. Adquisición de conocimiento en sistemas de scheduling inter-cloud borrosos basados en reglas para clouds sostenibles basados en MDC. Se propone la aplicación de técnicas de aprendizaje evolutivas y bioinspiradas a los sistemas expertos de scheduling.

Esta tesis propone una combinación de áreas diferenciadas: planificación en redes cloud sostenibles y Soft-Computing, que requiere una revisión bibliográfica exhaustiva de ambos campos, analizando aspectos generales como los tipos de redes cloud

sostenibles, la planificación en otros sistemas distribuidos, y conceptos esenciales sobre la carga, los recursos y las máquinas virtuales en dichas redes. También se estudian las técnicas y tecnologías que permiten la creación de centros de datos sostenibles, como el acceso y gestión de energías renovables, diseños modulares de centros de datos, y la migración de máquinas virtuales basada en la disponibilidad de fuentes de energía renovables accesibles [Shuja et al., 2016].

Además, se estudian proyectos relevantes en el área para extraer escenarios y trazas de carga reales que permitan obtener resultados representativos de casos reales. En cuanto al área de Soft-Computing, se deben tratar los fundamentos de la lógica borrosa y de los sistemas borrosos basados en reglas, así como las técnicas evolutivas y bioinspiradas para la adquisición de conocimiento de forma eficiente en términos de velocidad y precisión.

Para analizar el comportamiento de los planificadores, se han implementado escenarios y condiciones de redes cloud sostenibles mediante simulaciones. En particular, se ha trabajado con simuladores que permiten la simulación de planificadores en redes cloud, la integración de recursos heterogéneos, y la ejecución de flujos de carga intensivos obtenidos de instalaciones relevantes.

Para la implementación de las estrategias de planificación, se han utilizado librerías avanzadas de Java y el software Matlab. En particular, en el aprendizaje de los planificadores expertos de forma distribuida se utiliza el toolkit Matlab Distributed Computing Environment (MDCE). Esto ha permitido acelerar los procesos de entrenamiento y optimizar las bases de conocimiento mediante diversas estrategias evolutivas y bioinspiradas.

Para llevar a cabo esta investigación, se ha contado con de los siguientes recursos:

- Recursos propios del grupo de investigación TIC-188 "Tratamiento de señales y sistemas de telecomunicación" de la Escuela Politécnica de Linares en el Campus Científico Tecnológico de Linares.
- Instalaciones y recursos del laboratorio de investigación L-132, ubicado en la Escuela Politécnica de Linares en el Campus Científico Tecnológico de Linares.
- Software para el desarrollo de aplicaciones, incluyendo herramientas especializadas para simulaciones y análisis de datos.

- Bibliografía del Departamento de Ingeniería de Telecomunicación, que proporciona acceso a una amplia gama de publicaciones científicas relevantes.
- Recursos de Internet, que han permitido acceder a información actualizada y proyectos en curso en el ámbito de las redes cloud sostenibles.

En cuanto al impacto científico-técnico, internacional, social y económico de los resultados esperados, cabe destacar que la gestión eficiente de infraestructuras en la nube es una de las tareas más importantes en los sistemas cloud utilizados extensivamente hoy en día. Muchas organizaciones y empresas tecnológicas han instalado centros de datos a gran escala con miles de servidores para proporcionar servicios de computación en la nube, como Google, IBM, Amazon y Microsoft [Hussain, 2013]. El increíble crecimiento en la cantidad y tamaño de los centros de datos conlleva un considerable consumo de energía. Según [Kirti, 2023], los centros de datos consumen actualmente alrededor del 8% del consumo energético global. Dentro de los centros de datos, el 40% de la energía se consume en las infraestructuras de computación, el 45% en las máquinas de refrigeración, y el 15% se pierde en las unidades de generación de energía. Así, el 70% del consumo de energía en centros de datos puede ahorrarse aplicando métodos de eficiencia de última generación en el enfriamiento, las unidades de potencia y la infraestructura informática. Así, la consideración de técnicas de planificación cada vez más eficientes en los centros de datos, como se plantean en esta tesis doctoral, tiene un impacto científico-económico muy importante. Además, la consideración de energía de origen renovable para el funcionamiento de los centros de datos, ubicados estratégicamente para el acceso a estas fuentes, supone una mejora significativa en cuanto al impacto medioambiental de estos centros de datos.

Se espera que las estrategias propuestas en esta tesis doctoral contribuyan de manera significativa a la sostenibilidad en entornos cloud. A través de la aplicación de sistemas expertos borrosos y técnicas de machine learning bioinspiradas, se busca optimizar la migración de máquinas virtuales, maximizando el uso de energía renovable y minimizando el impacto ambiental. Esto no solo tiene el potencial de reducir el consumo energético global, sino también de promover la adopción de prácticas más sostenibles en la industria de la tecnología de la información.

La estructura detallada de esta tesis se presenta a continuación:

1. **Introducción:** Se establece el contexto general del Cloud Computing, destacando los problemas de sostenibilidad y las estrategias para abordar el alto consumo de energía y la contaminación asociada.

2. **Estado del Arte:** Se revisan investigaciones previas sobre sostenibilidad en CDC, resaltando la aplicación de inteligencia artificial y lógica difusa para optimizar la migración de VM.

3. **Contribuciones y Metodología:** Se detallan las contribuciones clave de la tesis y se describe la metodología utilizada para abordar el problema de sostenibilidad energética en entornos cloud, con énfasis en el uso de sistemas expertos borrosos y algoritmos de optimización para mejorar la eficiencia del sistema de planificación.

4. **Resultados y Análisis:** Se presentan los resultados obtenidos a partir de simulaciones, evaluando el rendimiento del sistema de planificación inter-cloud y el sistema experto borroso. También se comparan diferentes enfoques, como KAGWO, KASIA y Pittsburgh, para evaluar mejoras en eficiencia energética y sostenibilidad.

5. **Conclusiones, Limitaciones y Líneas Futuras:** Se resumen las conclusiones clave, identificando las limitaciones del estudio y sugiriendo futuras direcciones de investigación para avanzar en la sostenibilidad en entornos cloud.

6. **Publicaciones:** Se compilan las principales publicaciones relacionadas con la tesis, que exploran enfoques para optimizar el uso de energía renovable y eficiencia del sistema.

En resumen, esta tesis doctoral propone estrategias innovadoras para mejorar la sostenibilidad en entornos de Cloud Computing mediante la aplicación de sistemas expertos borrosos y técnicas de machine learning bioinspiradas a la planificación inter-cloud. Se espera que los resultados obtenidos contribuyan a la creación de centros de datos más sostenibles y a la reducción del consumo energético global, impactando positivamente en el medio ambiente y promoviendo prácticas más ecológicas en la industria tecnológica.

2 Estado del arte

2.1 Introducción al Estado del Arte

El cloud computing ha evolucionado desde un concepto emergente en 2006 hasta convertirse en la tecnología dominante en el sector de las tecnologías de la información en la actualidad [Katal, 2023]. Este paradigma ha transformado las operaciones comerciales, permitiendo una escalabilidad y eficiencia sin precedentes a través de los Cloud data centers (CDCs) que son infraestructuras críticas para el almacenamiento de datos, el procesamiento y la provisión de servicios en línea. Estos centros soportan desde comercio electrónico hasta el procesamiento de Big-data, lo que resalta su importancia central en la economía digital. Sin embargo, esta centralidad trae consigo desafíos significativos, especialmente en términos de la gestión energética de los data centers que sustentan estos servicios. Estos centros, fundamentales para la operatividad del cloud computing, son grandes consumidores de energía, lo cual ha suscitado preocupaciones no solo económicas sino también ambientales debido al uso intensivo de recursos energéticos no renovables. Ante esta situación, la integración de fuentes de energía renovable emerge como una solución esencial para mitigar los impactos negativos asociados con los altos consumos energéticos.

En este contexto, las fluctuaciones en la producción de energía renovable debido a su dependencia de condiciones ambientales variables, como son el viento y el sol, introducen un nivel de incertidumbre que los sistemas de cloud computing tradicionales no están diseñados para manejar eficazmente. Aquí es donde el combinado de Fuzzy logic, machine learning y la migración de máquinas virtuales (VMM) ofrecen una oportunidad interesante. La capacidad de migrar VMs entre data centers dispersos geográficamente, en función de la disponibilidad de energía renovable, puede facilitar una gestión más sostenible y eficiente del consumo energético. Sin embargo, esta migración requiere mecanismos avanzados de decisión que puedan adaptarse dinámicamente a la disponibilidad fluctuante de la energía renovable y equilibrar la carga de trabajo de manera eficiente.

Estos mecanismos se hacen todavía más críticos al observar la evolución de los Cloud data centers que han pasado de ser simples instalaciones de almacenamiento de datos hasta convertirse en complejos sistemas que integran potencia de cómputo, redes y almacenamiento que tienden a estar geográficamente dispersos para servir mejor a sus usuarios globales. Los data centers de hiperescala, que son más grandes y eficientes que

los data centers corporativos tradicionales se benefician de economías de escala y de ingeniería especializada, definidos por la Corporación Internacional de Datos (IDC) como instalaciones con al menos 5000 servidores y 10,000 pies cuadrados, o por el Grupo de Investigación Synergy que enfatiza más las capacidades operativas que en el tamaño físico [Fiona, 2021].

En todo problema de optimización, entender el modelo a optimizar es primordial; El consumo energético en los CDCs se distribuye entre varios sistemas clave, con una porción considerable destinada a la refrigeración para mantener la funcionalidad óptima de los servidores y prevenir el sobrecalentamiento. La energía utilizada directamente por los servidores es crucial no solo para procesar y almacenar datos, sino también para mantener sistemas de respaldo y redundancia operativos. Sistemas de soporte como iluminación, seguridad e infraestructura de red, aunque menos demandantes, también contribuyen al consumo total de energía, resaltando la complejidad y la necesidad de estrategias integrales de eficiencia energética [Reis, 2023].

Como lo que no se puede medir, no se puede mejorar, para evaluar y mejorar la eficiencia energética en los CDCs, se utiliza el indicador PUE (Power Usage Effectiveness), introducido por [Belady & Malone, 2007]. El PUE compara el consumo total de un data center con el consumo de los equipos de TI específicos. Reducir el PUE mediante soluciones de enfriamiento más eficientes o una mejor distribución de energía permite utilizar más energía directamente en equipos de TI, posponiendo la necesidad de construir nuevos centros. Este indicador es crucial para comprender y mejorar la eficiencia energética en los CDCs [Zoie, 2017].

Sin embargo, el desafío va más allá de la mejora interna de la eficiencia. El alto consumo energético de los CDCs, ejemplificado por los servidores de Google que consumen 26 millones de vatios mensuales, plantea preocupaciones ambientales significativas. Para contextualizar, esto es más de tres veces la capacidad de generación de la central nuclear más grande del mundo, Kashiwazaki-Kariwa en Japón [Y. Zhang, 2023]. Además, en 2010, los CDCs consumían aproximadamente el 1.5% de la electricidad mundial, cifra que aumentó al 8% en 2020. Se proyecta que para 2030, podrían alcanzar hasta el 13% de la electricidad global, consumiendo hasta 8.000 TWh [Kirti, 2023]. Estas estadísticas alarmantes subrayan la urgente necesidad de implementar estrategias sostenibles en los CDCs [S. Ghazanfari-Rad and S. Ebneyousef, 2023].

A nivel financiero, y a pesar de los desafíos presentados por la pandemia de COVID-19, se espera que el mercado global de CDCs crezca de 59.3 mil millones de dólares en 2020 a 143.4 mil millones de dólares para el 2027, con una tasa compuesta de crecimiento anual del 13.4% entre 2020 y 2027, llegando a casi 311 mil millones de dólares en 2030 [Analysts, 2021]. En los Estados Unidos solo, el negocio de los CDCs fue valorado en 16 mil millones de dólares para 2020, mientras que se espera que China, la segunda industria más grande del mundo, alcance una industria de data centers de 32 mil millones de dólares para 2027, con una tasa compuesta anual del 17.5% entre 2020 y 2027 [Analysts, 2021]. Estas cifras destacan no solo la relevancia económica de los CDCs sino también la importancia estratégica de optimizar su gestión energética.

Ante este panorama, la transición hacia energías renovables y la mejora en la eficiencia del uso de energía emergen como soluciones clave. La promoción de CDCs “verdes” y la integración de tecnologías de gestión que promuevan la eficiencia manteniendo la calidad del servicio se vuelven imperativas para una operación de servicios IT más respetuosa con el medio ambiente. En este entorno, la inteligencia artificial surge como una herramienta fundamental para impulsar la eficiencia energética en los CDCs. A través de la optimización de la carga de trabajo y la distribución de recursos, la IA no solo permite predecir patrones de demanda sino también asignar recursos de manera eficiente, lo que se traduce en un ahorro energético significativo [M. Yekta and H. S. Shahhoseini, 2023].

Además, entre otras cosas, la IA desempeña un papel crucial en la gestión inteligente de enfriamiento, donde monitoriza y ajusta los sistemas de climatización para reducir el consumo energético. El mantenimiento predictivo automatizado, basado en el análisis de datos de sensores y tendencias históricas, anticipa fallos o necesidades de mantenimiento, manteniendo una eficiencia energética constante y reduciendo los costes operativos [B. V. Vivek and S. M. Kusuma, 2023]. Además, la IA facilita la integración de energías renovables, como la solar o la eólica, optimizando su uso y reduciendo la dependencia de energías basadas en combustibles fósiles [X. Peng and X. Qin, 2020].

La asignación dinámica de máquinas virtuales, que es una técnica clave en la gestión energética de los CDCs, permite una operación más eficiente, equilibrando el rendimiento y la adherencia a los SLA. Este proceso es fundamental para optimizar la utilización de recursos y mejorar la sostenibilidad de los data centers. En este contexto, la integración del swarm intelligence (SI) y el fuzzy logic se presenta como una opción a

tener muy en cuenta en la aplicación del machine learning (ML) para la gestión de data centers. Este enfoque no solo aborda los desafíos energéticos actuales, sino que también contribuye al desarrollo de un futuro sostenible y económicamente viable en la era de la digitalización.

El propósito de esta sección es revisar exhaustivamente la literatura existente relacionada con la gestión energética de data centers, la aplicación de energías renovables en estos sistemas, y la utilización de tecnologías avanzadas, como la inteligencia artificial y los algoritmos metaheurísticos, en la optimización de la migración de máquinas virtuales. Esta revisión permitirá identificar tanto las tendencias actuales como las brechas en la investigación que justifiquen la necesidad de abordar estos problemas mediante nuevos enfoques.

Además, se discutirán las tecnologías y metodologías actuales, subrayando las limitaciones y desafíos que enfrentan. Esto establecerá el marco necesario para argumentar a favor de la investigación propuesta, que busca integrar sistemas inteligentes en la gestión de carga entre data centers dispersos geográficamente, con un enfoque especial en el uso eficiente de energías renovables. Al final de esta sección, se destacará cómo el enfoque de esta tesis no solo aborda las brechas identificadas, sino que también propone soluciones viables y sostenibles, estableciendo un precedente significativo para futuras investigaciones y prácticas en el campo.

Para llevar a cabo esta investigación, se han seleccionado artículos y estudios de diversas fuentes reconocidas como IEEE, Springer y Elsevier, revisados por pares y publicados en gran parte entre el 2018 y el 2023. La selección se centró en trabajos que abordan la lógica difusa y el machine learning, con un enfoque especial en los algoritmos que componen el Swarm intelligence, priorizando aquellos que ofrecen detalles sobre innovaciones de software para la optimización del consumo energético en los CDCs. Se ha dado preferencia a los estudios que ofrecen una perspectiva útil en la aplicación de tecnologías para la gestión energética sostenible.

2.2 Historia y evolución del cloud computing

Los orígenes del cloud computing se remontan a la era del mainframe en los años 60, cuando estas potentes máquinas ofrecían recursos informáticos centralizados, accesibles mediante terminales de teletipo [Flynn, 1966]. Estas configuraciones iniciales sentaron las bases para la arquitectura cliente-servidor [Singh et al., 2017]. Con el tiempo, el alto coste y la exclusividad de los mainframes impulsaron la exploración de alternativas más accesibles, como los clústeres de computación, que utilizaban componentes estándar y ofrecían una opción económicamente viable frente a los supercomputadores [Compton, 2002].

A medida que avanzaba la tecnología, la descentralización tomó mayor protagonismo con el desarrollo de la conmutación de paquetes y la estandarización de protocolos de red como TCP/IP. Este avance fue crucial para la transición de mainframes centralizados a una red más distribuida de ordenadores. Durante este periodo, también se creó ARPANET [Yux & Xuey, 2016], que fue el precursor del Internet moderno, permitiendo el intercambio global de mensajes y el hospedaje de servicios remotos.

La introducción de HTTP y HTML a finales de los 80 y principios de los 90 revolucionó la manera de compartir información, lo que llevó a la creación de los primeros navegadores web, sitios web y servidores. La estandarización de TCP/IP estableció la infraestructura para lo que ahora conocemos como la World Wide Web, un hito crucial en la historia de la computación distribuida que allanó el camino para el desarrollo del cloud computing.

Entre 1994 y 2000, un periodo definido por la aparición de tecnologías como las aplicaciones Peer-to-Peer (P2P), los Servicios Web [Peltz, 2003] y el Grid computing [Foster et al., 2001], se demostró el potencial de las redes descentralizadas para el procesamiento y almacenamiento a gran escala. Estas innovaciones facilitaron el acceso abierto a recursos informáticos y almacenamiento a través de protocolos abiertos y middleware, marcando la transición hacia la provisión de recursos informáticos como un servicio [Gill et al., 2020].

Con la entrada del nuevo milenio, el cloud computing comenzó a integrarse significativamente con el Internet de las Cosas (IoT), marcando un punto de inflexión en la forma en que interactuamos con la tecnología cotidiana. La adición de sensores,

actuadores y capacidades de red a objetos comunes expandió enormemente el alcance y las capacidades del cloud computing, creando una red globalmente conectada de 'cosas' [Whitmore et al., 2015]. Esta sinergia entre el cloud computing y el IoT facilitó nuevas aplicaciones y servicios que aprovechan la recolección de datos en tiempo real y la capacidad de procesamiento remoto, configurando así un entorno más inteligente y conectado.

Además, la maduración del cloud computing trajo consigo la evolución de los modelos de servicio, que se ajustaron para proporcionar distintos niveles de control y abstracción que satisficieran diversas necesidades y requerimientos operativos. Los tres modelos principales que surgieron fueron Infraestructura como Servicio (IaaS), Plataforma como Servicio (PaaS) y Software como Servicio (SaaS), cada uno ofreciendo un espectro único de gestión y escalabilidad [Gill et al., 2020].

IaaS proporciona a los usuarios recursos de hardware virtualizados, como almacenamiento y servidores, dándoles control sobre el sistema operativo y las aplicaciones implementadas, pero sin la carga de gestionar la infraestructura física subyacente. PaaS ofrece un entorno de desarrollo completo, incluidos sistemas operativos, entornos de ejecución de lenguajes de programación, bases de datos y servidores web, donde los usuarios pueden desarrollar, ejecutar y gestionar aplicaciones sin preocuparse por la infraestructura subyacente. SaaS, por otro lado, entrega aplicaciones de software a través de Internet en un modelo de suscripción, liberando a los usuarios de la instalación, mantenimiento y gestión de la infraestructura, pues todo es manejado por el proveedor del servicio.

Después de 2010, la proliferación de dispositivos móviles y plataformas de IoT generó un volumen masivo de datos que necesitaba ser procesado de manera más eficiente para reducir la latencia y mejorar la experiencia del usuario. Esto llevó al desarrollo de paradigmas como la computación Edge y Fog. La computación Edge [Shi et al., 2016] se refiere a la colocación de infraestructura de procesamiento entre los dispositivos del consumidor y los data centers tradicionales, buscando procesar los datos más cerca del punto de origen. Por su parte, la computación Fog [Brogi et al., 2019] extiende el cloud a la periferia de la red, permitiendo que aplicaciones y servicios operen directamente en dispositivos de borde o cerca de ellos, lo que mejora significativamente los tiempos de respuesta y el ancho de banda agregado, además de permitir un aprovisionamiento de

recursos que tiene en cuenta la geolocalización. La **¡Error! No se encuentra el origen de la referencia.** representa la evolución del cloud computing hasta nuestros días.

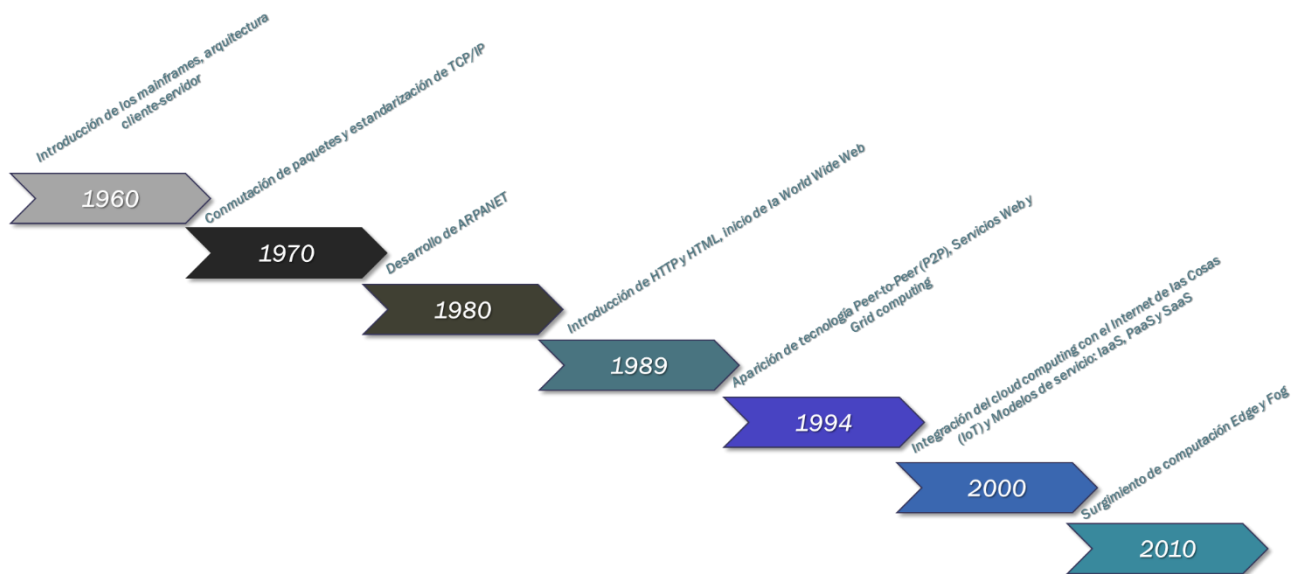


Figura 2-1 Evolución del Cloud Computing

Hay que decir que a pesar de las numerosas ventajas que ofrece, el cloud computing enfrenta desafíos significativos, especialmente en términos de seguridad, privacidad, y soberanía de datos. Además, el aumento en la demanda de servicios en el cloud ha llevado a un crecimiento exponencial en el consumo de energía y un impacto ambiental considerable. Las aplicaciones de Big Data, Machine Learning e IoT han aumentado el uso global de energía, contribuyendo a la contaminación ambiental [Alqahtani et al., 2019]. La necesidad de sistemas de Machine Learning que utilicen clústeres de GPUs para aplicaciones de Deep Learning en ambientes distribuidos ha justificado la investigación y la inversión en algoritmos de gestión de tareas y programación en data centers que sean eficientes en el uso de la energía [Lix et al., 2018].

En respuesta a los desafíos ambientales y de eficiencia energética, la industria del cloud computing ha comenzado a adoptar estrategias más sostenibles. Los data centers modernos están implementando soluciones innovadoras para mejorar la eficiencia energética. Uno de los enfoques más prometedores es la adopción de fuentes de energía renovable, como la solar y la eólica, que ayudan a mitigar el impacto ambiental de las operaciones masivas de datos [Gill & Tuli, 2020]. Además, la gestión inteligente de la

energía mediante software avanzado permite a estos centros optimizar el consumo energético y reducir el desperdicio de energía en componentes “Zombies”.

La virtualización ha sido otro avance significativo en el cloud computing, permitiendo a los proveedores de servicios maximizar el uso de sus recursos físicos al alojar múltiples entornos virtuales en una única plataforma física. Esta tecnología no solo mejora la eficiencia operativa, sino que también reduce la cantidad de hardware necesario, disminuyendo así el consumo de energía y los costes asociados [Gill et al., 2020].

A medida que el cloud computing se expande, también lo hacen las preocupaciones sobre la seguridad y la privacidad de los datos. Las infracciones de datos y los ataques cibernéticos son riesgos constantes en un entorno en línea, lo que obliga a los proveedores de servicios en el cloud a fortalecer sus políticas de seguridad. Las estrategias para abordar estos desafíos incluyen el uso de encriptación avanzada, autenticación multifactorial y técnicas de segmentación de red para proteger la infraestructura y los datos de los usuarios [Katal, 2023].

Además, las regulaciones como el GDPR en Europa han impulsado a las empresas a adoptar prácticas más rigurosas en la gestión de datos personales, garantizando que la privacidad del usuario se mantenga al más alto nivel. Este marco legal ha motivado a los proveedores del cloud a ser más transparentes en sus operaciones y a ofrecer mejores controles a los usuarios sobre sus propios datos.

Mirando hacia el futuro, se espera que la integración de tecnologías emergentes como la inteligencia artificial y el machine learning transformen aún más el paisaje del cloud computing. Estas tecnologías no solo automatizan los procesos y mejoran la eficiencia, sino que también permiten analizar grandes conjuntos de datos para extraer insights valiosos, ofreciendo así servicios más personalizados y efectivos. Además, la potencial incorporación de la computación cuántica podría llevar a avances significativos en la velocidad de procesamiento y en la capacidad para resolver problemas que actualmente son inabordables para los sistemas convencionales.

El Cloud computing ha recorrido un largo camino desde sus humildes inicios con los mainframes y continúa evolucionando a un ritmo acelerado. Mientras que enfrenta desafíos significativos, las oportunidades para la innovación y el mejoramiento son vastas. Con su capacidad para adaptarse y transformarse, El Cloud seguirá desempeñando un papel crucial en la configuración del futuro tecnológico, impulsando nuevos modelos

de negocio, mejorando la accesibilidad a la tecnología y fomentando un enfoque más sostenible y seguro en el manejo de los recursos computacionales y de datos.

2.3 Modelos de Servicio en el cloud y su Aplicación en CDCs Sostenibles

Tras haber establecido la importancia de los CDCs y su impacto energético, es esencial comprender los modelos de servicio en el cloud que forman la base de estas infraestructuras. Los modelos de servicio en el cloud, como IaaS, PaaS y SaaS, ofrecen distintos niveles de control, gestión y flexibilidad, y son fundamentales para la operación y el despliegue de aplicaciones y servicios. Estos modelos no solo optimizan la gestión de recursos y reducen costes operativos, sino que también juegan un papel crucial en la sostenibilidad energética. Al permitir una mayor eficiencia y flexibilidad, facilitan la implementación de estrategias de energía renovable y la adaptación a la demanda fluctuante, alineándose con los objetivos de eficiencia energética y reducción de emisiones de carbono.

En el contexto de esta tesis, donde se explora la migración de máquinas virtuales basada en energías renovables entre CDCs modulares geo-dispersos, estos modelos de servicio proporcionan el marco necesario para una gestión energética más eficiente y sostenible. La integración de sistemas expertos y la del Swarm intelligence en este marco puede potenciar aún más la optimización del consumo de energía renovable en los CDCs, contribuyendo a un futuro más sostenible en la industria del cloud computing.

Los modelos de servicio en el cloud emergen como pilares fundamentales en la arquitectura de los Cloud data centers modernos permitiendo el acceso a datos y la comunicación entre redes desde cualquier ubicación y ofreciendo un modelo de pago "pay per use" [Weinman, 2018; Nowotny et al., 2017], brindando a los usuarios la flexibilidad de consumir solo lo que necesitan. Este modelo de negocio ofrece flexibilidad y adaptación dinámica a las necesidades de los usuarios, lo cual ha sido fundamental para su adopción en el mundo empresarial [Rittinghouse & Ransome, 2016; Vaquero et al., 2008]. Estos modelos, clasificados primordialmente como Infraestructura, Plataforma y Software como Servicios se [Parast et al., 2022; Armbrust et al., 2010; Liu et al., 2011; Mather et al., 2009]., no solo representan la diversificación de la oferta en el cloud, sino que también reflejan la evolución de las necesidades empresariales en un ecosistema tecnológico cambiante.

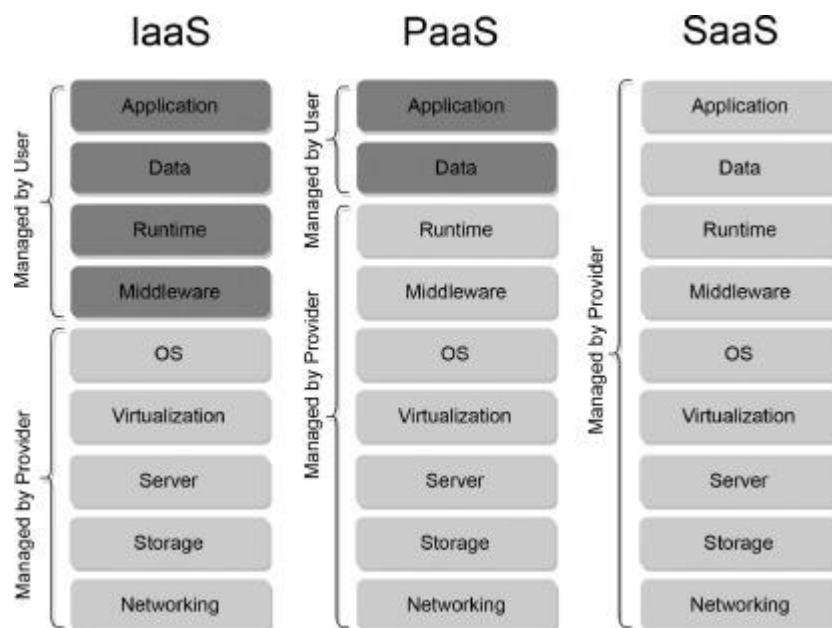


Figura 2-2 Perímetros de control por usuario final y proveedor de servicios para los tres principales modelos del cloud computing [Parast et al., 2022]

IaaS, es la capa “base”, la más fundamental, ofrece recursos computacionales virtualizados, incluyendo almacenamiento, redes y poder de procesamiento, sobre los cuales las organizaciones pueden construir y escalar sus operaciones sin la carga financiera y operativa de mantener hardware físico [Mell & Grance, 2011]. Este modelo es particularmente relevante para los CDCs que buscan optimizar su consumo energético, ya que permite una gestión más dinámica y eficiente de los recursos, adaptándose a las fluctuaciones en la demanda y la oferta de energía renovable [Srivastava et al., 2023; Shaikh & Meshram, 2021].

PaaS, por otro lado, proporciona un entorno de desarrollo y hospedaje para aplicaciones, donde los desarrolladores pueden crear, ejecutar y gestionar aplicaciones sin la complejidad de construir y mantener la infraestructura típicamente asociada con el proceso [Bach-Nutman, 2020; Pham et al., 2017]. En la Figura 2-2 se puede observar como la suma de la capa IaaS, más un middleware y un Runtime constituyen el concepto de PaaS; AmazonWS y Windows Azure son dos ejemplos de PaaS [Ajana & Singh, 2019; Toraskar & Borse, 2018].

SaaS, el modelo más abstracto, entrega software listo para usar a través de Internet [Bhardwaj, 2010], liberando a los usuarios finales de la instalación, mantenimiento y actualización de aplicaciones. Como se puede ver en Figura 2-2 el SaaS es la suma del

IaaS, el PaaS y una capa de aplicación y datos. En este modelo, todo el conjunto de software y la infraestructura de hardware son proporcionados y gestionados por el proveedor de servicios en el cloud (CSP), mientras que los usuarios acceden a aplicaciones listas para usar a través de una tarifa de suscripción mensual o anual. Este modelo de suscripción es beneficioso tanto para los proveedores como para los usuarios. Como ejemplos de SaaS podemos encontrar a Google Drive, Microsoft 365 y Amazon AWS [Loukis et al., 2019; Cook, 2018].

La intersección de estos modelos de servicio con la gestión de energía renovable en los CDCs es de particular interés. La capacidad de los CDCs para integrar fuentes de energía renovable depende en gran medida de su flexibilidad y adaptabilidad, características que son intrínsecas a los servicios en el cloud. Al adoptar un enfoque modular y distribuido, los CDCs pueden aprovechar la variabilidad de las fuentes de energía renovable, como la solar y eólica, para alimentar sus operaciones de manera más eficiente y sostenible.

La migración de máquinas virtuales entre CDCs geo-dispersos, guiada por la disponibilidad de energía renovable y optimizada por sistemas expertos y algoritmos de swarm intelligence, representa un avance significativo en la búsqueda de eficiencia energética. Este enfoque no solo mejora la sostenibilidad de los CDCs, sino que también asegura la continuidad y la calidad del servicio en el cloud.

La transición hacia un paradigma de CDCs impulsados por energía renovable es, por tanto, una convergencia de innovación tecnológica y responsabilidad ambiental. Los modelos de servicio en el cloud actúan como catalizadores en este proceso, permitiendo una gestión más inteligente y adaptable de los recursos computacionales y energéticos, y abriendo nuevas avenidas para la investigación y el desarrollo en el campo de la sostenibilidad de TI.

2.4 Desafíos en Gestión de Energía en el cloud computing

En la evolución de los Cloud data centers, la eficiencia energética y la sostenibilidad han son preocupaciones críticas, especialmente considerando la demanda creciente de servicios en el cloud y el impacto ambiental asociado. A medida que las empresas y consumidores se vuelven más conscientes del estrés en recursos energéticos que los servicios que consumen demandan, los CDCs enfrentan el desafío de equilibrar el rendimiento con la responsabilidad del uso de recursos.

Como se ha mencionado, los Cloud data centers son grandes consumidores de energía, debido principalmente a la operación continua de servidores, sistemas de refrigeración y redes. A medida que se expanden para satisfacer la creciente demanda de servicios en línea, también lo hacen sus necesidades energéticas, lo que lleva a un aumento en el uso de recursos no renovables.

En esta sección, exploraremos los desafíos clave que enfrentan los CDCs en términos de gestión energética y sostenibilidad. Comenzaremos discutiendo la importancia de la eficiencia energética y cómo esta se relaciona con la viabilidad económica y ambiental de los data centers. Luego, analizaremos los principales desafíos para alcanzar una gestión sostenible, incluyendo la variabilidad en la producción de energía renovable, la presión para reducir el uso de recursos no renovables, y las dificultades para integrar tecnologías innovadoras en infraestructuras existentes. Finalmente, examinaremos las estrategias actuales para abordar estos desafíos y sus limitaciones, destacando las oportunidades para nuevas soluciones basadas en inteligencia artificial y algoritmos de swarm intelligence.

Esta revisión establece un marco para comprender por qué la gestión eficiente de la energía es crítica para el futuro de los CDCs y cómo los avances tecnológicos, como los sistemas expertos y los algoritmos de swarm intelligence, pueden ofrecer soluciones novedosas para estos desafíos. Al abordar estos problemas, podemos movernos hacia un modelo más sostenible y ecológico para la industria del cloud computing.

2.4.1 Importancia de la Eficiencia Energética

La eficiencia energética se ha convertido en un aspecto crucial en la operación y desarrollo de Cloud data centers. Dado el aumento exponencial en el uso de energía por parte de estos centros y su impacto ambiental, la búsqueda de soluciones sostenibles y más eficientes es una prioridad para la industria de TI. La importancia de la eficiencia energética se destaca por varios factores, entre ellos, el crecimiento del consumo de energía y el coste asociado al uso de recursos energéticos no renovables.

Desde 2006, el consumo energético de los data centers ha experimentado un crecimiento significativo, pasando de 200 TWh a una proyección de cerca de 2967 TWh para 2030 [Katal, 2023]. Este aumento exponencial en el consumo energético implica mayores costes operativos para las empresas que gestionan los CDCs. Además, el uso intensivo de recursos no renovables para la producción de energía eléctrica plantea un problema ambiental, dado que muchas regiones del mundo generan electricidad mediante la quema de carbón, aumentando así la contaminación y el impacto ambiental.

El indicador Power Usage Effectiveness (PUE) es ampliamente utilizado para medir la eficiencia energética en los CDCs. Este indicador, introducido por [Belady & Malone, 2007], compara el consumo total de energía de un data center con el consumo de equipos de TI específicos, proporcionando así una métrica para evaluar y mejorar la eficiencia energética [Zoie, 2017]. Una reducción en el PUE sugiere un uso más eficiente de la energía, permitiendo dedicar más recursos energéticos a los equipos de TI y menos a los sistemas de soporte, como la refrigeración. Una mejora en la eficiencia energética puede retrasar la necesidad de construir nuevos centros y reducir el coste asociado a la energía. En la figura vemos como [Bharany et al., 2022] desglosan la anatomía del consumo por data center, servidor y estado de este.

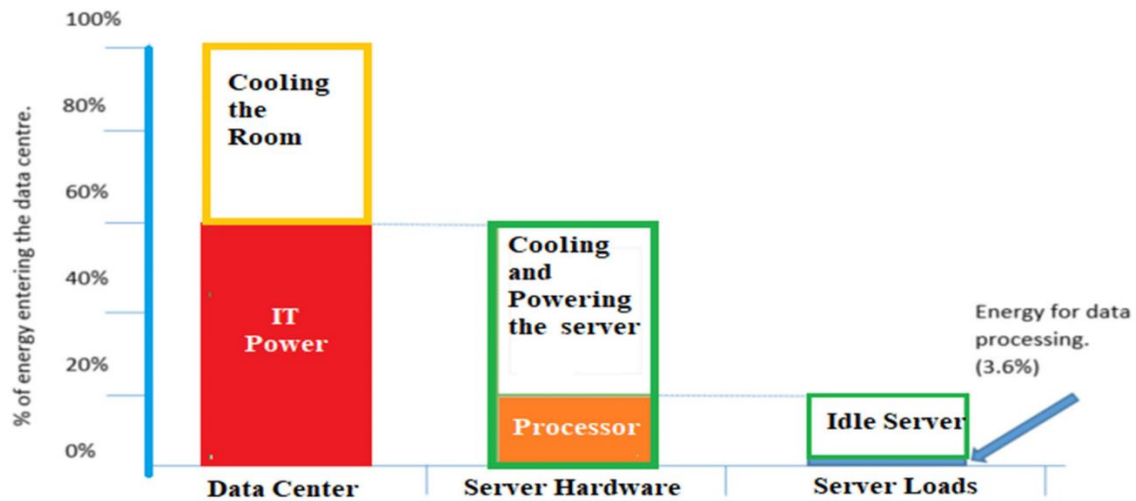


Figura 2-3 Anatomía del consumo de energía en los data centers y en los servidores

La presión para reducir las emisiones de CO₂ está impulsando a la industria a adoptar fuentes de energía renovable, como la solar y la eólica. Estas fuentes de energía limpias ofrecen un gran potencial para disminuir el impacto ambiental de los CDCs y reducir la dependencia de combustibles fósiles [X. Peng and X. Qin, 2020]. Sin embargo, la variabilidad en la producción de energía renovable plantea desafíos para los CDCs, que deben ser abordados con estrategias innovadoras para garantizar la continuidad del servicio y la eficiencia energética.

En este contexto, la inteligencia artificial y el machine learning están desempeñando un papel fundamental en la optimización de la eficiencia energética en los CDCs. Estas tecnologías permiten predecir patrones de demanda, ajustar la carga de trabajo de manera dinámica y mejorar la distribución de recursos, todo lo cual contribuye a un ahorro energético significativo [M. Yekta and H. S. Shahhoseini, 2023]. La integración de la IA con la gestión energética permite una operación más eficiente y sostenible de los CDCs, abordando al mismo tiempo los desafíos relacionados con el uso de recursos no renovables y la reducción de emisiones de carbono.

Con esta visión, queda claro que la eficiencia energética es un componente crítico para el futuro sostenible de los Cloud data centers. Los desafíos actuales requieren soluciones innovadoras que aprovechen la tecnología y enfoques sostenibles para garantizar la operación eficiente y responsable de estos centros.

2.4.2 Desafíos de Sostenibilidad

Los desafíos de sostenibilidad en los Cloud data centers son numerosos y de gran importancia para la industria de TI. A medida que el consumo energético de los CDCs sigue aumentando, surgen preocupaciones ambientales y económicas que requieren soluciones innovadoras y sostenibles. En esta sección, exploraremos algunos de los principales desafíos de sostenibilidad y las estrategias para abordarlos, resaltando la necesidad de un enfoque integral para la gestión energética en los CDCs.

El consumo energético elevado en los CDCs plantea serios problemas ambientales. En muchas regiones, la producción de energía para los data centers proviene de fuentes no renovables, como el carbón y el gas natural. Por ejemplo, en 2018, los data centers en Estados Unidos generaron aproximadamente 31.5 millones de toneladas de CO₂-equivalente, lo que representó el 0.5% de las emisiones totales del país [Siddik et al., 2021]. Esto evidencia la necesidad de estrategias sostenibles en los CDCs.

Otro desafío es la variabilidad en la producción de energía renovable. Si bien fuentes como la solar y la eólica tienen potencial para reducir el impacto ambiental de los CDCs, su producción es intermitente y depende de condiciones climáticas cambiantes. Esta inestabilidad plantea problemas para los data centers, que requieren un suministro constante de energía para mantener sus operaciones. La adopción de energías renovables implica soluciones de almacenamiento y distribución para compensar esta variabilidad, lo cual puede aumentar los costes operativos y la complejidad de gestión [X. Peng and X. Qin, 2020].

El uso intensivo de recursos no renovables también incrementa la demanda de agua y otros recursos para la refrigeración de los CDCs. Los sistemas de enfriamiento consumen grandes cantidades de agua, especialmente en climas cálidos o áridos, creando un problema ambiental significativo. La refrigeración eficiente y el uso de tecnologías que reduzcan el consumo de agua son esenciales para abordar este desafío.

El problema de los residuos electrónicos es otro aspecto clave de la sostenibilidad en los CDCs. El crecimiento rápido y la obsolescencia tecnológica conducen a grandes cantidades de desechos electrónicos, un problema ambiental significativo. La gestión de estos residuos requiere esfuerzos coordinados para la reutilización y el reciclaje, así como la implementación de prácticas sostenibles por parte de fabricantes, empresas y

gobiernos. Según [Panneerselvam, 2017], muchos CDCs tienen servidores "zombis", es decir, servidores que no están en uso, pero aún consumen energía. Esto genera más desechos electrónicos y aumenta la huella de carbono de los data centers.

En el ámbito económico, el alto consumo energético puede ser costoso para las empresas que gestionan los CDCs, especialmente cuando se requieren infraestructuras adicionales para mantener la continuidad operativa. Las estrategias para abordar la sostenibilidad deben equilibrar el impacto ambiental y los costes operativos, buscando soluciones que sean económicamente viables y respetuosas con el medio ambiente.

Para abordar estos desafíos, la eficiencia energética y el uso de fuentes de energía renovable se consideran fundamentales. La migración de máquinas virtuales, la gestión dinámica de cargas de trabajo y la inteligencia artificial pueden contribuir a reducir la huella de carbono y mejorar la eficiencia operativa, permitiendo a los CDCs operar de manera más sostenible y económica. El uso de energías renovables, como la solar y la eólica, y la implementación de estrategias para reducir el uso de recursos no renovables son esenciales para hacer frente a estos desafíos y garantizar un futuro más sostenible para los Cloud data centers.

2.4.3 Estrategias Actuales y Limitaciones

2.4.3.1 Modelo energético de un data center

La búsqueda de eficiencia energética dentro de los data centers representa un pilar fundamental del cloud computing que ha surgido como una preocupación primordial en

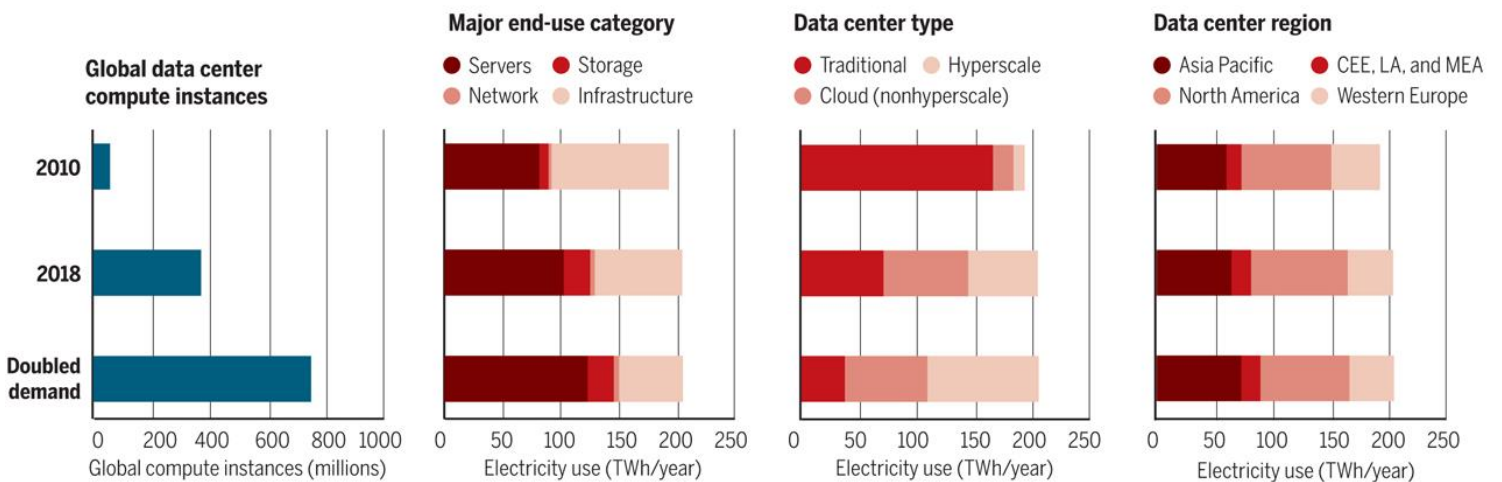


Figura 2-4 Evolución del consumo de energía en diferentes tipologías de data centers

el sector de TI. El crecimiento exponencial en las demandas de procesamiento de datos ha llevado a un aumento en el consumo de energía, con los data centers que pasaron de consumir 200 TWh en el 2006 a proyectárseles un consumo cercano a los 2967 TWh para 2030 [Katal, 2023], ósea más de un orden de magnitud en menos de 25 años. Este creciente requerimiento de energía no solo exacerba las emisiones de CO₂, sino que también amplifica los costes operativos, obligando a la industria a buscar soluciones sostenibles y más económicas. En la figura Figura 2-4 vemos la evolución constatada y estimada del consumo de energía en diferentes tipos de data centers de [Masanet et al., 2020].

En los últimos años, se ha notado un impacto medioambiental sustantivo de los data centers en el cloud, donde se ha constatado un aumento exponencial en número de estos que se han creado por varias empresas con el fin de ofrecer servicios de computación en el cloud. Además del alto consumo de electricidad, el uso creciente de energía por los data centers conduce a efectos ambientales perjudiciales. En numerosas regiones del mundo, la electricidad se genera quemando carbón, lo que resulta en resultados adversos como el aumento de las emisiones de dióxido de carbono y el aumento de la contaminación.

Las recientes plantas de energía renovable ofrecen un gran potencial para una gestión más eficiente de los data centers distribuidos. La asignación y migración dinámica de cargas de trabajo a través de los data centers podría ayudar al ahorro de costes al transferir la carga de trabajo a áreas donde la energía es más limpia o los costes de enfriamiento sean más bajos. Al recibir las peticiones de los usuarios, el proveedor de servicios en el cloud puede elegir la región objetivo en función de varios factores. [Laganà et al., 2018] han utilizado la técnica EcoMultiCloud que es una estrategia multiobjetivo de gestión de carga que han adaptado a escenarios de presencia intermitente de energía renovable en data centers (DC) distribuidos diferentes zonas geográficas y donde existen plantas productoras de energía renovable ubicadas junto a los data centers para disminuir la cantidad de electricidad comprada de la red eléctrica pública. Como las fuentes de energía renovable son esporádicas, las soluciones de control de carga de infraestructura deben adaptarse a la naturaleza intermitente de las fuentes. Además, [Khalil et al., 2019] han abordado el problema de reducir los costes de energía para los data centers geográficamente distantes mientras se garantiza la calidad del servicio en contextos en los que el consumo de los sistemas que varía con el tiempo. Propusieron una solución de

balanceo de carga geográfica “verde” que llamaron “GreenGLB” para la distribución de trabajo en tiempo real y para tareas indivisibles basada en un enfoque de diseño de “algoritmo codicioso”. Hay que recordar que un trabajo indivisible es aquel que debe asignarse a un solo data center sin posibilidad de división en subtareas más pequeñas.

También en relación con el impacto medioambiental, encontramos el problema de los residuos electrónicos, donde el rápido aumento de su uso, junto con la expansión de la computación a escala empresarial, han contribuido al problema de su acumulación y gestión. Minimizar estos residuos, reutilizarlos y reciclarlos es una responsabilidad colectiva de todos los actores ya sean consumidores, fabricantes, empresas o gobiernos.

Al estar los data centers compuestos de un gran número de servidores además de los sistemas de red, almacenamiento, suministro de energía, junto con enormes unidades de HVAC (heating, ventilation and air conditioning) que sirven para su refrigeración, esto consume grandes cantidades de energía que contribuyen a una producción importante de CO₂. En 2018 y solo en estados unidos, la producción de estos gases alcanzo los 3.15 x 10⁷ toneladas de CO₂-eq representando a ellos solos el 0.5% del total de emisiones de este país [Siddik et al., 2021].

Queda patente que uno de los vectores de ataque para paliar el efecto de producción de CO₂ de estas compañías de servicio en internet es el de optimizar la utilización de sus recursos con el fin de reducir el consumo energético de sus data centers y por ende reducir su huella de carbono. Se estima que muchas de estas empresas no realizan más de entre el 10 al 15% del uso de sus servidores y que más del 25% de estos se pueden calificar como “Zombis” en el sentido en que no se utilizan, pero aun así consumen energía y esto significa que el margen de mejora es considerable para los data centers en el cloud [Panneerselvam, 2017]. Existen muchas iniciativas que tienen como objetivo alcanzar las “Cero emisiones” de CO₂ y que combinan el uso de fuentes de energía renovables como la energía sola, que según el trabajo de [Ctrils, 2020] alcanza a reducir hasta el 70% del coste energético de los data centers, junto con estrategias de optimización a diferentes niveles de uso de estos que se explicaran a continuación.

En este contexto y en términos de consumo de energía y su impacto, varios estudios han abordado diferentes aspectos de este problema. A continuación, se presenta un resumen de algunos de estos estudios:

Los autores de [Baliga et al., 2011] presentaron un análisis sobre el uso de energía en el cloud computing. La investigación consideró tanto los clouds públicos como privados, así como la energía consumida en la conmutación, comunicación, cálculo de información y almacenamiento. Demostraron que el uso de energía en tránsito y conmutación puede representar una parte considerable de la demanda total de energía en el cloud computing. Su método propuesto considera el cloud computing (CC) como un equivalente de un problema de logística y cadena de suministro, teniendo en cuenta el uso de energía y el gasto de computación, conservación y transporte de bienes físicos.

En [Beloglazov et al., 2011] se presentó una taxonomía de la arquitectura del sistema eficiente en energía a nivel de SO, hardware, virtualización y data center. Destacaron las razones y dificultades asociadas con el uso excesivo de energía y evaluaron contribuciones importantes en el área para guiar futuras iniciativas de desarrollo e investigación. Investigaron y categorizaron numerosas formas de controlar el uso de energía de un sistema desde el nivel del SO utilizando DVFS y otras estrategias y algoritmos de ahorro de energía.

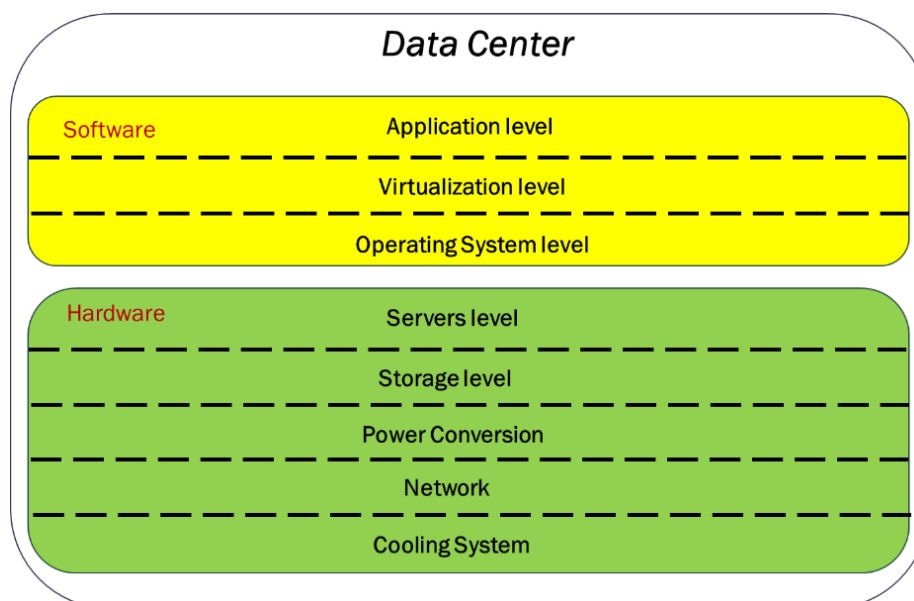


Figura 2-5 Modelo simplificado que presenta los niveles donde se utiliza la energía en los data centers

Además, los autores en [Shuja et al., 2016] abordaron la idea de crear un controlador de data center eficiente en energía capaz de combinar las capacidades de la data center mientras se reduce el efecto en los objetivos de Calidad de Servicio (QoS).

Destacaron las dificultades de investigación relacionadas con las necesidades competitivas de mejorar la calidad de los servicios proporcionados por los servicios en el cloud mientras se reduce el consumo de energía de los recursos de la data center. Propusieron un controlador de recursos y ofrecieron un concepto de controlador central.

Para entender mejor los vectores de ataque para analizar y mejorar la eficiencia del consumo energético en los data centers, hay que partir del modelo de uso que hacen estos de la energía. En la Figura 2-5 se presenta un modelo simplificado del consumo de energía en un Data center donde cada uno de estos niveles es susceptible de ser optimizado con el fin de ayudar en la eficiencia del uso de energía que hace el Data center.

2.4.3.2 Optimización a nivel del Sistema Operativo

A nivel de software, existen numerosas estrategias para intentar reducir y hacer que el consumo de energía en los Data centers sea más eficiente. Por ejemplo, al nivel del sistema operativo que se va a posicionar entre las aplicaciones de los usuarios y el hardware subyacente, se constata que casi el 50% del consumo de energía se realiza entre los “Data-paths” (Intercambios de datos entre procesadores o entre procesadores y memoria/almacenamiento...) y las topologías de “Pipelines” (Orquestación de las secuencias de ejecución de las microinstrucciones), el resto se consume entre el reloj, los diferentes niveles de caché.

En este sentido, encontramos en la literatura varios trabajos que intentan proponer soluciones a este nivel a través de los llamados OSPM (Operating System Power Management) que son un mecanismo utilizado por el Sistema Operativo (SO) para gestionar la energía de la plataforma subyacente y su transición entre diferentes modos de energía. [Herzog et al., 2021] ofrecen “Polar”, un método para determinar automáticamente configuraciones eficientes en energía, así como una implementación en Linux. [Scordino et al., 2018] ilustran cómo el planificador de plazos y el subsistema “cpufreq” pueden ser modificados para crear un enfoque de programación en tiempo real y que tenga en cuenta la energía.

Con el avance de las tecnologías de semiconductores y software, las capacidades de un sistema embebido han crecido incorporando nuevas características y rendimiento.

Los "data centers embebidos", más pequeños, más extendidos y menos conocidos, consumen la mitad de toda la energía del data center, o alrededor del 1% de toda la energía generada en los Estados Unidos. Este tipo de Data center se caracteriza por consumir menos de 50KW de las demandas IT [Embedded Datacenters].

Las tecnologías de recolección de energía basadas en baterías recargables son una opción popular para abordar el problema de suministrar energía continua a dispositivos profundamente implantados como nodos de sensores inalámbricos. [Buschhof et al., 2019] presentaron una idea que permite modelar el uso de energía del hardware y la creación de controladores de dispositivos conscientes de la energía para el SO embebido. De manera similar, [Levy et al., 2017] presentaron "Tock", un SO embebido único para sistemas de baja potencia que utiliza la protección de hardware limitada.

2.4.3.3 Optimización a nivel de Virtualización

La virtualización es una tecnología que ya lleva varias décadas con nosotros y que básicamente permite a un solo servidor físico alojar múltiples sistemas operativos y aplicaciones de manera simultánea. Cada uno de estos sistemas operativos funciona dentro de su propia máquina virtual, que es un espacio aislado que imita un ordenador completo. Existen diferentes categorías de virtualización, cada uno con sus propias técnicas y objetivos [Barret, 2010]. Por ejemplo, la virtualización completa permite que un sistema operativo invitado se ejecute sin modificaciones, mientras que la paravirtualización requiere una adaptación del sistema operativo para optimizar el rendimiento al comunicarse directamente con el hipervisor, que es el software que crea y gestiona las máquinas virtuales. Estos métodos de virtualización son fundamentales para la creación los cloud modernos que conocemos, ya que permiten compartir recursos físicos entre aplicaciones y usuarios de manera eficiente. Además, la virtualización facilita la gestión de recursos como software, hardware, consolidando muchas máquinas "subutilizadas" en un solo destino. La virtualización es una estrategia poderosa en el cloud que permite mejorar la escalabilidad y la disponibilidad de las aplicaciones, y proporcionar una capa de seguridad adicional mediante el aislamiento de las máquinas virtuales.

Justamente el nivel de virtualización es el segundo nivel de optimización a nivel de software que como se ha indicado, utiliza una capa de abstracción sobre el equipo hardware, permitiendo que las características reales de un solo ordenador, almacenamiento, disco, etc., se separen en numerosos ordenadores virtuales. El Monitor de Máquina Virtual / Hipervisor es responsable de la multiplexación de recursos a la máquina virtual y ayuda en la gestión de la energía para realizar operaciones eficientes en dos posibles maneras. El VMM puede actuar como un sistema operativo que tiene en cuenta la energía consumida. Verifica el rendimiento total del sistema y aplica las técnicas de DVFS (Escalado Dinámico de Voltaje y Frecuencia) o cualquier DCD (Desactivación Dinámica de Componentes) a los componentes del sistema.

La otra forma es aprovechar las políticas para la gestión de la energía y el conocimiento de las aplicaciones a nivel del SO. Las llamadas de gestión de energía pueden ser mapeadas desde diferentes máquinas virtuales. Además, se pueden aplicar límites coordinados a nivel de sistema en la energía. Las máquinas virtuales han permitido la creación de nuevas estrategias de desarrollo para añadir nuevas capacidades como, por ejemplo, la consolidación de servidores, la migración transparente de VMs y la computación segura, en los hosts mientras se mantiene la compatibilidad con los sistemas operativos y aplicaciones existentes. Para mitigar los problemas de gestión de energía, [Kang et al. 2021] sugirieron un método de control de energía por VM que permite que el SO invitado de cada VM utilice su estrategia de administración de energía elegida y evita que las VMs similares compitan con la estrategia de control de energía de cada uno. En comparación con el Hypervisor por defecto bajo demanda del “Xen”, el rendimiento virtual (VIP) minimiza el uso de energía y mejora el tiempo de finalización de las aplicaciones intensivas en CPU hasta en un 27% y un 32%, respectivamente, sin violar la SLAs sensibles a la latencia.

Además, [Xiao et al., 2021] examinaron el modelo de programación de VM y el paradigma de virtualización de E/S en términos de optimización de eficiencia energética. Proporcionaron un enfoque de secuenciación de crédito de equidad de energía con un nuevo método de desplazamiento de E/S para lograr un rendimiento rápido de E/S mientras aumentan simultáneamente la conservación de energía. Aparte de esto, [Prabhakaran et al., 2021] introdujeron la calibración de recursos de VM. Crearon un sistema para reducir el uso de energía de los servidores virtuales utilizando una arquitectura de retroalimentación controlada y servicios de monitoreo de energía.

2.4.3.4 Optimización a nivel de Aplicación

Se han propuesto numerosas estrategias para optimizar el consumo de energía a nivel de aplicación. No obstante, la diversidad creciente de cargas de trabajo exige un análisis más exhaustivo a nivel de aplicación, con el objetivo de permitir comportamientos adaptativos y minimizar el consumo energético global. En este contexto, [Ho et al., 2016] llevaron a cabo un estudio centrado en aplicaciones por lotes que operan en Máquinas Virtuales en Data centers. Evaluaron la eficiencia del software, midiendo el rendimiento y el consumo de energía de diferentes tareas y se dieron cuenta de que la configuración de los recursos tenía un impacto significativo en la eficiencia energética, especialmente cuando existían recursos compartidos y entornos heterogéneos. Posteriormente, dividieron las aplicaciones en dos categorías: aplicaciones intensivas en datos [Malik et al., 2019] [Miyazaki, 2018] [Basireddy et al., 2018] [Tiwari et al., 2018] [Ho et al., 2015] y aplicaciones intensivas en comunicación [Jiang et al., 2020], [Vakilinia, 2018]. Las aplicaciones intensivas en datos, que generan, analizan y transmiten enormes volúmenes de datos, consumían grandes cantidades de energía debido a problemas como la gestión, migración y almacenamiento de datos. Por otro lado, las aplicaciones intensivas en comunicación, compuestas por uno o más servicios interdependientes, requerían una gran cantidad de energía debido al tráfico de comunicación entre ellos. Exploraron técnicas de gestión de energía dinámica para ajustar el consumo de energía de un sistema de manera dinámica, en función de sus necesidades en cualquier momento dado. Estas técnicas se aplicaron tanto a aplicaciones intensivas en datos como a aplicaciones intensivas en comunicación para hacerlas más eficientes en términos de energía.

También a nivel de aplicación, si analizamos el (SaaS), vemos que este representa la capa superior de la arquitectura de cloud computing que permite ofrecer software bajo demanda [Kumar & Buyya, 2012]. Los proveedores de SaaS, en general, proporcionan capas adicionales de cloud computing y, por lo tanto, mantienen los datos del cliente y personalizan las aplicaciones para satisfacer las demandas del cliente. Los proveedores de SaaS atienden a una variedad de empresas utilizando la misma infraestructura y software. Este método es claramente más eficiente en términos de energía que instalar varias copias de software en varias infraestructuras, lo que puede reducir la necesidad de nuevo equipo. En este sentido, en el trabajo de [Tchana, 2015] se propuso una solución

para la consolidación dinámica de software con el fin de disminuir el número de VMs utilizadas. La consolidación de software permite asignar y agrupar dinámicamente diferentes aplicaciones de software en la misma VM. Esta estrategia permite colocar varias VMs en menos máquinas reales y por ende reducir el consumo.

Otra estrategia utilizada para optimizar el consumo de energía a nivel de aplicación es la migración de tareas/VMs, aquí encontramos el trabajo de [Samrajesh, 2013] que propone la migración de Elementos de Aplicación de forma consciente del consumo de energía, esto ayuda a gestionar de forma más óptima la carga en los servidores de los Data centers. En su investigación considera factores como el número de componentes conectados a los servidores, el número de personas que intentan acceder a las aplicaciones de software, la tasa de errores de conexión y otros factores importantes. Basándose en estos factores, se deciden qué elementos mover a diferentes servidores. Una vez que todos los componentes de los servidores subutilizados han sido movidos, el servidor se apaga para ahorrar energía.

A las mencionadas migración y consolidación de máquinas virtuales, podemos añadir otras técnicas de resolución de problemas el balanceo de carga y la categorización de esta que se pueden utilizar para la optimización del consumo de energía. Estas técnicas se emplean para la toma de decisión en relación con las máquinas virtuales cuando se alcanzan por ejemplo los umbrales de ciertos criterios como pueden ser el de los recursos de los servidores (CPU, Memoria, las operaciones de entrada-salida...). Estas técnicas son el escenario de un número importante de investigaciones donde se proponen algoritmos de diferentes tipos para gestionar de manera eficiente el consumo de energía en los Data centers.

2.5 Sistemas expertos y Fuzzy logic en el cloud computing

2.5.1 introducción

La historia de la lógica difusa o como su creador la designo inicialmente “La computación y razonamiento con palabras” [Jiang & Tang, 2014] es un ejemplo de resiliencia y revolución en el pensamiento científico y tecnológico. Iniciada por Lotfi Zadeh en 1965 [Zadeh, 1965], [Mittal et al., 2020], esta teoría superó inicialmente el escepticismo y la crítica para convertirse en un pilar fundamental en numerosas áreas de investigación y desarrollo. Zadeh, enfrentando la soledad de sus ideas, perseveró, guiado por su convicción de que los conjuntos borrosos podrían capturar la incertidumbre y la ambigüedad del mundo real de manera más efectiva que los métodos tradicionales. Su trabajo, eventualmente reconocido por la comunidad científica, le valió la medalla de honor del IEEE.

En uno de sus últimos trabajos “Is there a need for Fuzzy logic?”, Lotfi razona como la lógica difusa, llega a ser un sistema que facilita el razonamiento preciso en medio de la imprecisión y la incertidumbre, formalizando las capacidades de toma de decisiones humanas en entornos con información imperfecta. Este enfoque permite un espectro de valores multivaluado de verdad que va más allá del binario, reflejando la naturaleza compleja y matizada de los escenarios del mundo real. Al incorporar principios de graduación, granulación y restricciones generalizadas, la lógica difusa ofrece un marco muy versátil para manejar ambigüedades, haciéndola indispensable en campos que requieren juicios matizados y la gestión de datos inciertos o imprecisos [Zadeh, 2008] [Cordón, 2001].

En las últimas décadas hemos podido presencia como la adopción de la lógica difusa en el cloud computing ha sido un catalizador para la innovación, permitiendo un manejo más sofisticado y eficiente de recursos computacionales, almacenamiento y servicios en el cloud. Su capacidad para manejar la incertidumbre y modelar decisiones basadas en información imprecisa o incompleta ha encontrado aplicaciones prácticas en la optimización de la asignación de recursos, la gestión de la carga de trabajo, la predicción de la demanda de servicios y la mejora de los sistemas de seguridad y privacidad en el cloud entre otros.

2.5.2 Ventajas Distintivas de la Lógica Difusa en el Cloud

La lógica difusa introduce un enfoque muy oportuno para el procesamiento de decisiones y la gestión de sistemas en entornos de cloud computing, caracterizados por su variabilidad y complejidad [Mittal et al., 2020]. Sus principales ventajas incluyen:

- **Manejo de Incertidumbre:** Capacidad para procesar información imprecisa o subjetiva, crucial para la toma de decisiones en el cloud computing.
- **Modelado Intuitivo:** Utiliza variables lingüísticas (como "alto", "medio", "bajo") que reflejan el modo natural en que los humanos describen y razonan sobre el mundo, haciendo los modelos más accesibles y comprensibles.
- **Flexibilidad:** La lógica difusa se adapta a cambios y puede integrarse fácilmente con otras metodologías y tecnologías, incluyendo inteligencia artificial y machine learning, para mejorar la eficiencia y efectividad de los sistemas de cloud computing.

La lógica difusa es valiosa porque emula la lógica humana, permitiendo un enfoque más intuitivo y flexible para resolver problemas. Al reflejar cómo los humanos procesan información ambigua y toman decisiones en situaciones inciertas, facilita soluciones que se acercan más al razonamiento natural. Esto hace de la lógica difusa un marco poderoso para la lógica multivaluada, abordando complejidades del mundo real de manera efectiva y proporcionando un método adaptable para la toma de decisiones en entornos como el cloud computing [Zadeh, 1983] [Cordón, 2001].

2.5.3 Sistemas Expertos Basados en Reglas Difusas, clasificación y aplicación

Los Sistemas Expertos Basados en Reglas Difusas (FRBS) son la manifestación práctica de cómo la lógica difusa se aplica para resolver problemas complejos en el cloud computing. Estos sistemas se componen de tres componentes esenciales [Cordón, 2001]:

- **Fuzificador:** Transforma las entradas cuantitativas en valores difusos mediante funciones de membresía, permitiendo que el sistema procese información cualitativa o imprecisa.

- **Motor de Inferencia:** Utiliza una base de reglas difusas para simular el razonamiento humano, aplicando lógica difusa a las entradas fuzificadas para generar una salida difusa. Este proceso refleja la capacidad humana para tomar decisiones basadas en información imprecisa o incompleta.
- **Defuzificador:** Convierte las salidas difusas del motor de inferencia en valores cuantitativos concretos, facilitando decisiones y acciones específicas.

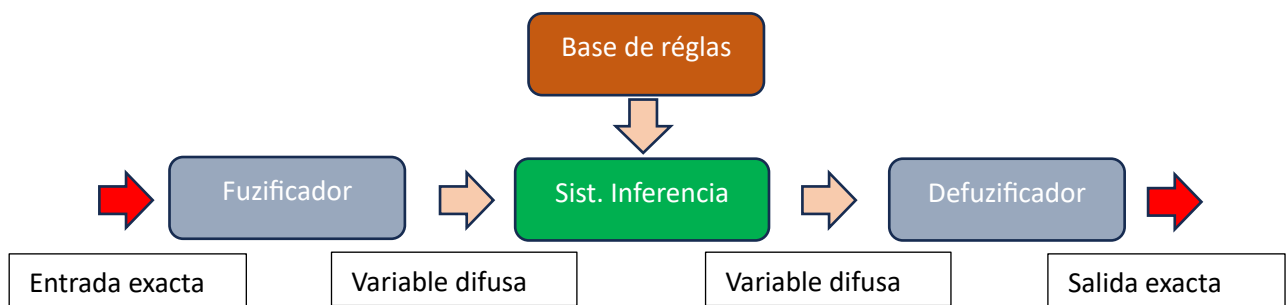


Figura 2-6 Estructura general de un FRBS

2.5.4 Clasificación de los sistemas basados en reglas difusas

Dada la diversidad de problemas que los FRBS pueden abordar y las distintas maneras en que se pueden configurar y optimizar, surge la necesidad de clasificar y categorizar estas implementaciones para entender mejor sus capacidades, limitaciones y áreas de aplicación óptima. Esta clasificación no solo ayuda a los investigadores y desarrolladores a elegir el enfoque adecuado para sus necesidades específicas, sino que también facilita la evolución continua de la tecnología de lógica difusa, permitiendo la comparación y combinación de distintas técnicas para abordar desafíos aún más complejos.

En este contexto, se presenta en la figura a continuación, un enfoque general para clasificar y categorizar las diferentes implementaciones de los FRBS que encontramos en la literatura, considerando criterios como el manejo de la incertidumbre, el método de generación de reglas, el enfoque de toma de decisiones y la estructura del sistema.



Figura 2-7 Categorías del FRBS

Cada uno de estos criterios refleja una dimensión importante de la flexibilidad y la capacidad de los FRBS para adaptarse a distintos entornos y requisitos. A continuación, presentamos una visión de conjunto de estas clasificaciones junto con sus aplicaciones más relevantes:

2.5.4.1 Por el Manejo de la Incertidumbre y la Función de Pertenencia

En el trabajo de [Wu & Mendel, 2023] se tratan dos primeras clasificaciones de los enfoques para FRBS, que son el tipo 1 y el tipo 2.

- **FRBS Tipo-1:** Los FRBS de Tipo 1 utilizan conjuntos difusos de Tipo 1, donde

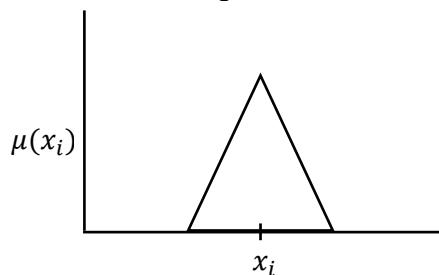


Figura 2-8 Ejemplo de un Fuzzy set de tipo 1

cada elemento del universo de discurso tiene un grado de pertenencia entre $[0,1]$ a un conjunto difuso. Este grado indica cuán verdadero es que el elemento pertenezca al conjunto. Los FRBS de Tipo 1 son efectivos para modelar problemas con cierto grado de incertidumbre, pero asumen que los grados

de pertenencia son conocidos con precisión y no tienen en cuenta la incertidumbre en estos grados.

- **FRBS Tipo-2:** Los FRBS de Tipo 2 extienden los FRBS de Tipo 1 al introducir conjuntos difusos de Tipo 2, en los cuales los grados de pertenencia mismos son difusos, es decir, cada elemento del universo de discurso tiene un grado de pertenencia que es un conjunto difuso en lugar de un valor preciso. Esto permite modelar una capa adicional de incertidumbre que no puede ser capturada por los FRBS de Tipo 1. Los FRBS de Tipo 2 son particularmente útiles en situaciones donde hay una gran incertidumbre o cuando la información sobre el problema es muy imprecisa o difícil de medir. [Jiang & Tang, 2014] utilizaron este tipo, junto con autómatas para desarrollar un modelo de computación con palabras, esto permite aplicaciones como el reconocimiento de lenguajes naturales con y sin contexto además de otros tipos de reconocimiento de patrones.

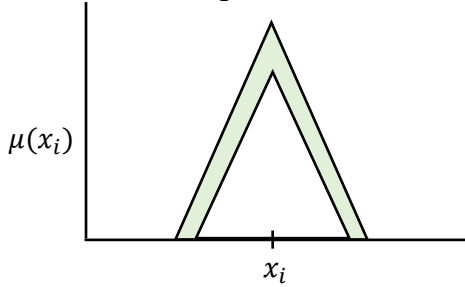


Figura 2-9 Ejemplo de un Fuzzy set de tipo 2

[Kong et al., 2011] propusieron un algoritmo de programación dinámica en línea para data centers virtualizados, buscando equilibrar el balanceo de carga y la predicción de esta, empleando lógica difusa Tipo I y Tipo II.

2.5.4.2 Por el Método de Generación de Reglas

- **Basados en Heurísticas:** Se basan en reglas prácticas obtenidas a través de la experiencia o el conocimiento experto. No garantizan la mejor solución posible, pero ofrecen una forma rápida y efectiva de resolver problemas complejos con una buena aproximación. Son especialmente útiles en dominios donde el conocimiento humano sobre el problema es claro y bien definido, y donde la interpretación de las reglas por humanos es importante. En este sentido, entre otros, podemos encontrar el trabajo de [Fatimeh et al., 2016] que Aplicaron la optimización por PSO difuso para mejorar la asignación de máquinas virtuales y la utilización de recursos en la computación en nube o el de [Christian et al.,

2016], donde Investigaron la selección de servicios de almacenamiento en el cloud usando lógica difusa para reflejar las preferencias imprecisas de los usuarios y también el de [Boominathan y Aramudhan, 2016] que Implementaron el algoritmo “FireFly Colony” en Data centers para consolidación de servidores a través de la asignación inteligente de máquinas virtuales, optimizando así los recursos y el consumo de energía.

- **No Heurísticas:** Utilizan algoritmos y técnicas matemáticas para generar reglas basadas en el análisis de datos. Estos métodos pueden encontrar soluciones óptimas o casi óptimas sin depender directamente del conocimiento experto humano. Son ideales en situaciones donde hay disponibles grandes volúmenes de datos y el patrón específico del problema no es intuitivamente claro o es demasiado complejo para ser definido directamente por expertos humanos. Aquí también podemos encontrar varios trabajos que ilustran este enfoque, como el de [Priya et al. 2017] diseñaron una inferencia difusa Max Min para mejorar la programación de recursos, optimizando sin considerar atributos cualitativos para la calidad del servicio. El de [Zulkar Nine et al., 2013] que Desarrollaron un algoritmo de equilibrio de carga dinámico basado en lógica difusa que elige la máquina virtual adecuada para nuevos trabajos en data centers, considerando memoria, espacio en disco y ancho de banda, o el de [Venkatesa et al., 2012], donde Introdujeron un mecanismo que combina redes neuronales y lógica difusa para asignar recursos en servicios de nube, utilizando un algoritmo genético para ajustar solicitudes de usuarios con recursos disponibles.

2.5.4.3 *Por el Enfoque de Toma de Decisiones*

- **MADM:** Los Métodos de Toma de Decisiones Multi-Atributo (MADM, por sus siglas en inglés) o Multi-Criterio (MCDM) son técnicas utilizadas para evaluar y priorizar alternativas cuando se deben considerar múltiples criterios de decisión. Estos métodos son esenciales en situaciones donde las decisiones no se pueden tomar basándose en un solo criterio debido a la complejidad y la naturaleza multidimensional de los problemas. Por ejemplo, en [Srimannarayana Gran et al., 2015] se Propone un modelo de toma de decisiones multi-atributo difuso para la

evaluación del rendimiento de proveedores de servicios en el cloud, utilizando números difusos con intervalos en criterios como escalabilidad y tiempo de respuesta del servicio para reducir la ambigüedad. Por su lado [Mya y Thein, 2011] Sugieren un esquema basado en neuro-difuso para la predicción de carga de máquinas virtuales y la gestión de migración, integrando la predicción del uso de recursos con AHP para la consistencia de pesos y Fuzzy TOPSIS para la clasificación

- **Jerárquicos:** Se refieren a un enfoque estructurado para descomponer problemas de decisión complejos en partes más manejables, a menudo representados en una estructura de árbol. Estos métodos permiten a los decisores desglosar un problema general en niveles, donde cada nivel representa un conjunto de elementos que son más fáciles de analizar y comparar. El Proceso de Jerarquía Analítica (AHP) es un ejemplo prominente de un método jerárquico. [Xu & Fortes, 2010] Proponen un sistema de gestión de recursos de dos niveles para data centers utilizando lógica difusa para mejorar la eficiencia y rendimiento, minimizar los costes de comunicación entre las VMs y reducir el consumo total de energía. [Baliyan & Kumar, 2016] Presentan un sistema de lógica difusa jerárquico con una función de pertenencia gaussiana, denominado HFS-SWAaaS, diseñado para evaluar servicios en el cloud. Abordan la complejidad de los atributos de calidad, incluyendo atributos únicos específicos de las Aplicaciones Web Semánticas.

2.5.4.4 *Por la estructura del sistema*

- **De una etapa:** En este caso, el proceso de inferencia se realiza en un solo paso. Esto significa que se toman un conjunto de entradas, se aplican las reglas difusas correspondientes, y se genera una salida directa. Son más simples y directos, adecuados para problemas con una estructura menos compleja donde la relación entre entradas y salidas es relativamente directa y menos ambigua. Se suelen utilizar en automatización de procesos industriales donde las entradas se pueden mapear directamente a salidas mediante reglas claras.
- **Multietapa o en cascada:** Implementan un proceso de inferencia más complejo y detallado, dividido en varias etapas o niveles. Cada etapa puede abordar diferentes aspectos del problema o manejar diferentes niveles de abstracción o

incertidumbre, y las salidas de una etapa pueden servir como entradas para la siguiente. Son útiles para problemas más complejos donde una única ronda de inferencia no sería suficiente para capturar todas las dimensiones de la incertidumbre o para cuando se requiere una mayor precisión en el proceso de toma de decisiones. Se suelen utilizar en el procesamiento de lenguaje natural, donde varias capas procesan diferentes aspectos del lenguaje, desde la sintaxis hasta el contexto semántico. [Liang et al., 2018] proponen en su trabajo de 2018 una Red de Fusión Multietapa Recurrente (RMFN, por sus siglas en inglés), que aborda el problema de la fusión de señales multimodales descomponiéndolo en múltiples etapas. Cada etapa se centra en un subconjunto de señales multimodales para una fusión especializada y efectiva. Por su parte [Liu et al, 2020] presentan en su trabajo de 2020 una red de fusión multietapa con un módulo de puerta de olvido para la “sumarización” multimodal de videos en dominios abiertos. Esta “sumarización” multimodal, genera un resumen a partir de información multi-origen (video, audio, transcripción).

2.5.4.5 *Por el Método de Inferencia*

- **Mamdani:** Caracterizados por su simplicidad y su capacidad para modelar el conocimiento experto de manera intuitiva. En un sistema Mamdani, tanto las variables de entrada como de salida se tratan con conjuntos difusos, y el resultado se obtiene mediante un proceso que incluye la fusificación de las entradas, la aplicación de un conjunto de reglas difusas, la agregación de los resultados de las reglas y, finalmente, la defusificación para obtener un valor numérico. Este tipo de sistema es muy utilizado en aplicaciones de control donde la interpretación lingüística de las reglas es importante [Nguyen et al., 2019].
- **Takagi-Sugeno-Kang (TSK):** Utilizan un enfoque diferente para la salida de las reglas. Mientras que en Mamdani la salida de cada regla es un conjunto difuso, en T-S-K la salida es una función matemática, típicamente un polinomio lineal de las variables de entrada. Esto permite que los sistemas T-S-K manejen sistemas dinámicos y complejos de manera más eficiente, siendo especialmente útiles en modelado y optimización. Aunque pueden ser menos interpretables que los sistemas Mamdani, ofrecen una mayor precisión y flexibilidad en muchos casos

[Zhao et al., 2018]. Se suelen utilizar en la modelización financiera para predecir índices de mercado, integrando salidas numéricas precisas dentro de un marco difuso.

2.5.4.6 *Por la Adaptabilidad y Aprendizaje*

- **Estáticos:** Utilizan un conjunto de reglas y funciones de membresía que no cambian con el tiempo. Estos sistemas son diseñados y ajustados manualmente por expertos para una tarea específica y operan bajo la suposición de que la dinámica del sistema o del entorno no varía significativamente. Su principal ventaja es la simplicidad y la estabilidad en entornos controlados o bien entendidos. En el trabajo de [Dey et al. 2013] donde se diseña e implementa un controlador de nivel de agua basado en lógica difusa, se demostró la eficacia de un sistema FRBS estático en aplicaciones de control cuando el entorno operativo es relativamente constante.
- **Adaptativos o Dinámicos:** En contraste, los sistemas FRBS adaptativos son capaces de ajustar sus reglas y funciones de membresía automáticamente en respuesta a cambios en los datos de entrada o en el entorno. Esto se logra a través de mecanismos de aprendizaje y adaptación, tales como algoritmos genéticos, redes neuronales o técnicas de aprendizaje profundo. Estos sistemas son particularmente útiles en entornos dinámicos y complejos donde las condiciones pueden variar significativamente con el tiempo. La detección de fraude en tiempo real [Raj & Portia, 2011] o la predicción de la velocidad del tráfico [Gholami et al., 2021] son aplicaciones típicas de este enfoque, donde los patrones de comportamiento fraudulento pueden evolucionar, requiriendo un ajuste continuo de las reglas.

2.5.4.7 *Por la Fuente de Conocimiento*

- **Basados en Conocimiento Experto:** Los sistemas FRBS basados en conocimiento experto se construyen utilizando el conocimiento y la experiencia humana directamente para definir las reglas y las funciones de membresía. Estos sistemas

dependen de expertos en el dominio de aplicación para formular las reglas que describen cómo se deben interpretar las entradas y producir las salidas. La ventaja de este enfoque es que permite incorporar directamente la intuición y el conocimiento humano detallado en el sistema, lo cual es particularmente útil en dominios donde los datos son escasos o donde la experiencia humana juega un papel crítico en la toma de decisiones. La evaluación de riesgos ambientales es un buen ejemplo de este enfoque [Borri & Concilio, 1998] que ilustra cómo los sistemas FRBS basados en conocimiento experto se aplican para predecir flujos ambientales, utilizando el conocimiento experto o el de [Wang et al., 2022], donde el juicio experto se traduce en reglas para evaluar el impacto de actividades humanas.

- **Basados en Datos:** Por otro lado, los sistemas FRBS basados en datos generan y ajustan sus reglas y funciones de membresía automáticamente a partir de conjuntos de datos. Estos sistemas utilizan técnicas de machine learning y minería de datos para identificar patrones en los datos y construir las reglas del sistema de manera objetiva. La ventaja de este enfoque es que puede adaptarse a complejidades y variaciones en los datos que podrían no ser evidentes para los expertos humanos, permitiendo a los sistemas modelar comportamientos complejos o cambiantes con alta precisión. En [Zonta & Da Costa, 2020] estudio demuestra el uso de sistemas FRBS basados en datos para el mantenimiento predictivo en la Industria 4.0, destacando cómo se pueden extraer reglas difusas directamente de los datos de máquinas y sensores. [Bahani & Moujabbir, 2021] proponen un sistema de clasificación basado en reglas difusas para predecir enfermedades cardíacas. Este sistema ofrece una base de conocimiento interpretable para explicar el proceso de toma de decisiones.

2.5.4.8 *Por la Aplicación*

- **Para Control:** En aplicaciones de control, los FRBS se utilizan para manejar dinámicamente sistemas complejos o procesos donde las relaciones entre variables pueden ser difíciles de modelar con métodos convencionales. Estos sistemas traducen las estrategias de control basadas en la experiencia humana en un conjunto de reglas difusas que pueden manejar la incertidumbre y la

imprecisión inherentes a muchos procesos de control. Casos de uso típicos son la automatización de invernaderos donde por ejemplo [Azaza & Tanougast, 2016] discuten el diseño de un sistema de control inteligente para invernaderos utilizando lógica difusa para optimizar las condiciones ambientales internas, tales como temperatura, humedad, y luz, basándose en las necesidades específicas de las plantas cultivadas, o la conducción autónoma de vehículos [Al-Mahturi & Santoso, 2021].

- **Para Clasificación:** Los FRBS de clasificación se utilizan para asignar entradas a una de varias categorías basadas en un conjunto de reglas difusas. Estos sistemas son particularmente útiles en dominios donde las categorías no están bien definidas o cuando las entradas pueden ser ambiguas o incompletas. La clasificación difusa permite una mayor flexibilidad y tolerancia a la incertidumbre en comparación con los métodos de clasificación tradicionales. Se suele utilizar en sistemas de seguridad, clasificando comportamientos o actividades como normales o sospechosos [Hajek, 2019].
- **Para Predicción:** En el contexto de predicción, los FRBS se emplean para pronosticar futuros estados o valores basándose en datos históricos y actuales. Estos sistemas son capaces de manejar la incertidumbre y la imprecisión en los datos de entrada, lo que los hace adecuados para aplicaciones como la predicción del tiempo, la demanda de energía o las tendencias del mercado [Yang & Jiang, 2022].

2.6 Machine learning en el cloud computing

2.6.1 Introducción

El cloud computing ha transformado el panorama tecnológico al facilitar el acceso a servicios y recursos informáticos virtualizados. En este contexto, el machine learning juega un papel crucial en la optimización y gestión eficiente de los recursos en el cloud. Figura 2-10. Su evolución ha llevado a la industria a explorar nuevos enfoques para abordar desafíos relacionados con la escalabilidad, latencia y adaptabilidad en tiempo real, a la vez que se busca la reducción de costes operativos y la mejora de la calidad del servicio (QoS).

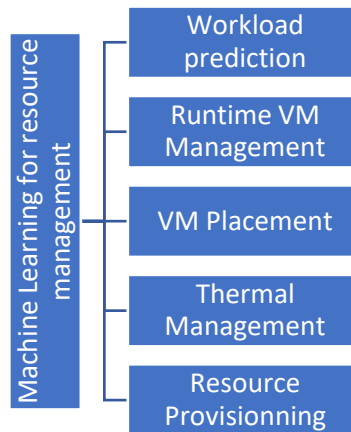


Figura 2-10 Categorización del uso del ML en el cloud computing

Históricamente, la gestión de recursos en entornos de cloud computing se basaba en políticas estáticas, lo cual resultaba insuficiente en escenarios con cargas de trabajo dinámicas. Con el surgimiento del ML, se ha desarrollado una aproximación más flexible y adaptativa para optimizar la gestión de recursos, utilizando modelos guiados por datos y algoritmos de ML. La literatura científica apoya esta tendencia, mostrando el interés en utilizar ML para mejorar el cloud computing. Estudios como [Manvi & Shyam, 2014] han destacado estrategias fundamentales para la administración de recursos, incluyendo la predicción de demandas y la asignación adaptativa de recursos. [Sun et al., 2016] presentan un análisis de los avances recientes en la optimización del uso de recursos, mientras que [Usmani & Singh, 2016; Prado -d et al., 2010; Prado -e et al., 2010] se centran en la consolidación eficiente de máquinas virtuales en data centers ecológicos.

Más recientemente, [Helali & Omri, 2021] y [Khan & Tian, 2021] examinaron la consolidación en cloud data centers distribuidos y las técnicas de ML aplicadas en la gestión de recursos, respectivamente. Estos estudios muestran cómo el ML puede influir positivamente en la eficiencia operativa y la reducción de costes, al tiempo que se alinean con prácticas más sostenibles y respetuosas con el medio ambiente.

A medida que se investiga más profundamente, se observa la convergencia entre ML y otros enfoques, como el Swarm intelligence. Estudios como [Siddiqi et al., 2023], [Murad et al., 2022], y [Kumar & Bhagwan, 2022] abordan la aplicación de Swarm intelligence en el cloud computing, destacando el potencial para resolver problemas complejos relacionados con la asignación de tareas y la toma de decisiones descentralizada. Otros estudios, como [Ghafari et al., 2022] y [Singh et al., 2020], también exploran estas aplicaciones, mientras que [Siddiqi & Uddin, 2023] y [Houssein et al., 2021] destacan los desafíos inherentes a los problemas NP-complete y NP-hard.

El estado del arte del ML en el cloud computing apunta a un futuro donde los enfoques adaptativos y dinámicos son esenciales para una gestión eficiente de recursos [Garcia-Galan -b et al., 2014] [Garcia-Galan -c et al., 2012]. Con la continua investigación y la combinación de diversos métodos, la industria del cloud computing puede anticipar mejoras significativas en la eficiencia, sostenibilidad y calidad de servicio.

2.6.2 Asignación y Ubicación Eficiente de Máquinas Virtuales

La asignación y ubicación estratégica de VM constituyen ejes fundamentales en la administración de los CDCs, encaminados a la distribución efectiva de recursos que manejen las cargas de trabajo fluctuantes y dinámicas de las aplicaciones cliente. Este desafío se centra no solo en asegurar que los "anfitriones" o servidores mantengan la capacidad necesaria para satisfacer demandas computacionales sino también en superar las dificultades impuestas por la variabilidad de dichas demandas. Las aplicaciones, con sus variados requerimientos, necesitan de asignaciones de VM personalizadas capaces de ajustarse a estos patrones cambiantes de necesidad, lo cual introduce una capa adicional de complejidad en la gestión de recursos.

Históricamente, las políticas de ubicación de VM han priorizado métricas de utilización de recursos en tiempo real, como el uso de CPU. Sin embargo, este enfoque, que a menudo descuida las proyecciones de demanda futura, se ha demostrado insuficiente para las necesidades actuales. En este contexto, emerge la importancia de estrategias “*pronosticativas*” que, además de evaluar las necesidades actuales de recursos, puedan anticipar futuras demandas.

Un factor crítico, que a veces se omite en la discusión sobre gestión de recursos, es el ancho de banda de la red. La investigación de [Genez & Bittencourt, 2019] destaca cómo la consideración conjunta del uso de CPU y el ancho de banda puede fundamentar estrategias más efectivas, como aquellas basadas en Optimización por Enjambre de Partículas (PSO) [Garcia-Galan -a et al., 2015], que equilibran restricciones presupuestarias y temporales para la computación distribuida a través de distintos CDCs. Dado el crecimiento esperado del tráfico de Internet impulsado por servicios en el cloud, la presión sobre las redes se intensifica, lo que a su vez afecta los tiempos de migración de VMs y el cumplimiento de los SLA, resaltando la necesidad de enfoques de ubicación de VMs que sean conscientes del ancho de banda.

En este marco, [Shaw et al., 2019] propusieron un modelo predictivo y que tiene en cuenta la red proponiendo un modelo heurístico para planificar la ubicación de VMs y que busca minimizar el consumo de energía y las violaciones de SLA, anticipando necesidades futuras tanto de CPU como de ancho de banda. Aunque este enfoque mejora significativamente la precisión en la programación de recursos y la eficiencia de las asignaciones de VM, no contempla todos los factores críticos, como la memoria RAM, las operaciones de entrada/salida de disco, o las necesidades computacionales específicas de las tareas en ejecución. Esta limitación subraya la intrincada naturaleza de la gestión de VM y la necesidad imperante de estrategias holísticas que integren un espectro más amplio de indicadores de rendimiento, demostrando así la complejidad inherente a la gestión de recursos en entornos de cloud computing y el valor potencial del ML para abordar estos desafíos de manera integral y adaptativa.

2.6.2.1 *Optimización por Asignación de Máquinas Virtuales*

La asignación de Máquinas Virtuales [Prado -c et al., 2011] es otro enfoque en la gestión de recursos de computación en el cloud, centrándose principalmente en la asignación efectiva de máquinas virtuales dentro de máquinas físicas (PM). Este proceso implica no solo la colocación inicial de VMs sino también su reubicación o migración dinámica para optimizar la utilización de recursos y el rendimiento del sistema [Cotes-Ruiz et al., 2017]. La selección e implementación de una estrategia óptima para la asignación de VMs tiene un impacto significativo en la eficiencia operativa, el coste y la calidad del servicio entregado a los usuarios finales.

Las estrategias de asignación de VM varían según sus objetivos. Algunas se centran en la eficiencia energética, buscando reducir el consumo de energía al asignar VM a menos PM y permitir que algunas PM queden inactivas o apagadas. Estas estrategias, conocidas como "Asignación Basada en Energía", son vitales para reducir el impacto ambiental y los costes operativos asociados con el funcionamiento de grandes data centers [Yu & Gao, 2012]. Sin embargo, pueden enfrentarse a desafíos en términos de calidad del servicio (QoS), ya que la reducción de PM puede afectar el rendimiento y la disponibilidad de las VM.

Por otro lado, las estrategias basadas en QoS buscan maximizar el rendimiento y la disponibilidad de los servicios proporcionados por las VM, considerando factores como latencia, ancho de banda y potencia de cómputo [Lin et al., 2018]. Estas estrategias son esenciales para garantizar que los acuerdos de nivel de servicio se cumplan, pero pueden ser menos eficientes energéticamente. Este equilibrio entre eficiencia energética y QoS es un desafío constante en el diseño de estrategias de asignación de VM.

Como estamos viendo, El Cloud Data Center se presenta como un entorno dinámico donde los recursos físicos deben ser gestionados de manera eficiente para satisfacer las crecientes demandas de las VMs. Cada VM tiene requisitos específicos de recursos, como CPU, memoria y almacenamiento, lo que obliga a los administradores a buscar métodos de asignación que maximicen el rendimiento y minimicen el desperdicio de recursos.

De entre las técnicas que encontramos en la literatura que aborda esta problemática encontramos el Constraint Programming que se basa en restricciones

matemáticas para definir las condiciones bajo las cuales se logra una asignación eficaz. En su trabajo [Yu & Gao, 2012] Los autores abordan el problema de la infrautilización en los data centers, lo que conduce a altos costes operativos a pesar de las capacidades de la infraestructura. Al formular el problema de colocación de máquinas virtuales como una variante del problema de empaquetado de contenedores multidimensional, utilizan un solucionador de restricciones para minimizar el número de máquinas físicas que alojan máquinas virtuales. Este método tiene como objetivo reducir los costes operativos de los data centers y mejorar la utilización de recursos.

Sin embargo, hay que señalar ciertas limitaciones de este enfoque como es la complejidad de la programación por restricciones que plantea desafíos en términos de escalabilidad y adaptabilidad a las fluctuaciones de la demanda en el mundo real.

Por otro lado, Stochastic Integer Programming ofrece un enfoque más flexible que el anterior al introducir elementos de probabilidad y estadística para tratar con la incertidumbre inherente en las demandas de las VMs y las capacidades de los PMs. Al utilizar modelos probabilísticos, esta técnica puede adaptarse a condiciones variables y brindar soluciones que optimizan el uso de recursos en un marco estadísticamente óptimo. Como ejemplo de este enfoque, [Lin et al., 2018] en su trabajo profundizan en la Asignación Óptima de Máquinas Virtuales en Cloud computing, abordando la problemática de colocar VMs de manera eficiente en data centers geográficamente dispersos para minimizar los costes de comunicación. Los autores proponen un enfoque de optimización determinista para transformar el problema en un programa lineal entero mixto, que permite calcular una solución óptima global. El enfoque propuesto manifiesta también la limitación en términos de escalabilidad al aumentar la complejidad del problema de optimización con el número de VMs y puede no ser práctico para problemas a gran escala debido al crecimiento exponencial en los requerimientos computacionales.

Aunque ambas estrategias ofrecen soluciones eficaces, comparten una limitación crucial: pueden tener dificultades para adaptarse a condiciones fluctuantes y para predecir demandas futuras basadas en datos históricos. En un mundo donde la tecnología evoluciona rápidamente y las necesidades de los usuarios cambian, la clave para una asignación eficaz de VMs radica en encontrar un equilibrio entre estructura y flexibilidad, aprovechando estrategias que permitan adaptarse y evolucionar junto con el Cloud Data Center.

Los enfoques más modernos para la asignación de máquinas virtuales en entornos de Cloud Data Center están evolucionando gracias a la incorporación de técnicas de machine learning. Estas técnicas permiten a los sistemas adaptarse a entornos cambiantes y hacer predicciones más precisas, superando las limitaciones de métodos convencionales. El uso de machine learning en este contexto tiene como objetivo optimizar la asignación de VMs y mejorar la eficiencia del Cloud Data Center, donde los recursos son cada vez más escasos y las demandas más impredecibles.

Reinforcement Learning (RL), o aprendizaje por refuerzo, es un enfoque que ha ganado popularidad debido a su capacidad para aprender de manera autónoma a través de prueba y error. En este enfoque, el sistema se considera un agente que interactúa con un entorno y recibe recompensas o castigos en función de sus acciones. Con el tiempo, el agente aprende a tomar decisiones que maximizan la recompensa acumulada, ajustándose de forma dinámica a nuevas condiciones. En este caso encontramos el trabajo de [Long et al., 2020] donde en su artículo presentan una estrategia que han llamado “Reinforcement Learning-Based Virtual Machine Placement Strategy o (RLVMP)” con la vocación de reducir el consumo de energía en CDCs. Su estrategia no impone límites de CPU en las máquinas virtuales como medio para ahorrar energía. En cambio, evalúa la degradación en rendimiento “performance loss” de las VMs cuando están densamente asignadas en máquinas físicas (PMs). El RLVMP considera el equilibrio entre la pérdida de rendimiento de las VMs y el ahorro de energía, y emplea una estrategia codiciosa o “greedy strategy” para implementar su resolución. Los autores realizaron sus experimentos mediante simulación con CloudSim y que su algoritmo RLVMP ofrece una mejora en la eficiencia energética en comparación con otros algoritmos clásicos.

Sin embargo, y aunque en su trabajo no se especifican limitaciones de la estrategia propuesta. Típicamente en este caso podemos barajar, limitaciones tales que la escalabilidad para topologías más grandes que las que usaron en las simulaciones de sus Data centers o la aplicabilidad a para diferentes tipos de cargas de trabajo, o incluso la sobrecarga computacional introducida por el propio algoritmo de aprendizaje por refuerzo, sin mencionar las dificultades de implementación de dicha estrategia para un entorno de producción real que una simulación no puede capturar los matices.

Otro enfoque moderno es el uso de Artificial Neural Networks (ANNs). Estas redes imitan el funcionamiento del cerebro humano, con nodos interconectados que procesan información y aprenden a reconocer patrones. En el contexto de la asignación

de VMs, las ANNs pueden analizar grandes cantidades de datos para predecir requerimientos futuros de recursos y estimar el rendimiento de los servidores físicos. Como ejemplo de este enfoque tenemos el trabajo de [Shalu & Singh, 2021] donde en su artículo discuten un enfoque para la asignación de máquinas virtuales en el cloud, utilizando un algoritmo que han llamado “Enhanced-Modified Best Fit Decreasing (E-MBFD)” con el objetivo de optimizar la asignación de VMs a máquinas físicas, y por extensión mejorar la utilización de recursos, la eficiencia energética y el rendimiento de las aplicaciones. El método propuesto incluye la validación cruzada de las asignaciones de VM utilizando una Red Neuronal Artificial (ANN) y aborda el problema de las asignaciones falsas causadas por el uso ineficiente de recursos. Sus resultados sugieren que el algoritmo E-MBFD es superior en la reducción del consumo de energía y la minimización de las violaciones del acuerdo de nivel de servicio en comparación con los métodos tradicionales. En este trabajo tampoco se discuten las limitaciones de la solución propuesta, pero por su naturaleza probablemente tenga limitaciones parecidas al enfoque anterior.

También tenemos el Fuzzy Reinforcement Learning que es un enfoque que combina lógica difusa con aprendizaje por refuerzo para abordar la incertidumbre y la imprecisión en la asignación de VMs. La lógica difusa permite trabajar con grados de verdad en lugar de valores binarios, lo que resulta útil cuando las condiciones del Cloud Data Center no son claramente definidas o son propensas a cambios. Este enfoque permite una toma de decisiones más flexible y puede adaptarse a situaciones en las que las reglas estrictas no son aplicables. Al agregar lógica difusa al aprendizaje por refuerzo, se mejora la capacidad del sistema para lidiar con la ambigüedad y tomar decisiones óptimas en entornos impredecibles. [Jummal & Kumar, 2021] en su trabajo tratan una la asignación óptima de VMs para CDCs y para ello proponen un algoritmo de aprendizaje por refuerzo fuzzy-based SARSA (State-Action-Reward-State-Action) para mapear de manera eficiente las máquinas virtuales a los hosts, con el objetivo de mejorar el uso de recursos y la eficiencia energética. El método propuesto aborda el equilibrio entre minimizar el consumo de energía y mantener la Calidad de Servicio (QoS) deseada para cumplir con los Acuerdos de Nivel de Servicio (SLAs). La integración de un controlador difuso con el aprendizaje por refuerzo se muestra capaz de optimizar la reasignación de VMs, reduciendo el uso de energía y el desperdicio de recursos, especialmente bajo condiciones de carga de trabajo fluctuantes.

Sin embargo, el artículo también reconoce las limitaciones inherentes a este enfoque. Aunque es capaz de manejar cargas de trabajo dinámicas y asegurar la QoS, la complejidad del algoritmo y la necesidad de una afinación precisa del controlador difuso pueden presentar desafíos. Además, la escalabilidad para topologías de más de un Data centers involucrado en las simulaciones no se ha evaluado, lo que reduce su perímetro de acción.

En resumen, los enfoques modernos que utilizan el machine learning están transformando la asignación de VMs en Cloud data centers. Técnicas como Reinforcement Learning, Artificial Neural Networks y Fuzzy Reinforcement Learning brindan soluciones adaptativas y predictivas, permitiendo a los administradores del Cloud Data Center optimizar recursos y mejorar el rendimiento general. La capacidad de aprender y adaptarse es clave para enfrentar los desafíos de entornos dinámicos y necesidades cambiantes en el Cloud computing moderno.

No obstante, existen limitaciones inherentes a estas técnicas. La complejidad algorítmica puede ser un obstáculo para su implementación en entornos de producción reales, además de que su uso suele requerir recursos significativos para el entrenamiento y ejecución. En términos de escalabilidad, tanto RL como ANNs y Fuzzy-RL pueden enfrentar dificultades cuando se aplican a topologías más grandes o a múltiples Data Centers. Por consiguiente, aunque estas técnicas prometen mejorar la eficiencia en la asignación de recursos, es importante tener en cuenta estas limitaciones al considerar su implementación en entornos Cloud.

Continuando con las técnicas utilizadas en asignación de máquinas virtuales llegamos a la siguiente categoría que está basada en el comportamiento de poblaciones. Estas técnicas son métodos avanzados de optimización que comienzan con un grupo inicial de soluciones y evolucionan a través de múltiples generaciones para explorar un amplio espacio de soluciones.

Por una parte, tenemos los Algoritmos Genéticos (GA) que Simulan el proceso de evolución natural, donde las soluciones son tratadas como "individuos" y se someten a operaciones como cruzamiento, mutación y selección natural para buscar soluciones globalmente óptimas. El proceso evolutivo permite explorar una amplia gama de configuraciones, proporcionando una robusta herramienta para resolver problemas complejos. Aquí encontramos el trabajo de [Kaaouache & Bouamama, 2018] donde

presentan un algoritmo genético híbrido para la colocación eficiente de máquinas virtuales en CDCs, con el objetivo de reducir el consumo de energía y las violaciones del acuerdo de nivel de servicio (SLAV). Su estrategia intenta minimizar el número de migraciones de VM, lo que a su vez reduce la carga de la red de comunicaciones. Sin embargo, las limitaciones no se indican explícitamente en el contexto proporcionado. Típicamente, tales algoritmos suelen enfrentar desafíos en escalabilidad, adaptabilidad a diferentes arquitecturas de CDCs y el equilibrio entre el tiempo de optimización y la calidad de la solución.

Por otro lado, la Optimización por Colonias de Hormigas (ACO) imita el comportamiento de las hormigas en busca de alimento. Estas dejan rastros de feromonas, que otros miembros de la colonia siguen para encontrar rutas óptimas. Al simular este comportamiento, el ACO puede ser eficaz en problemas de optimización, especialmente en aquellos que requieren rutas o secuencias óptimas, como el problema del viajante (TSP). Aquí, [Tawfeek et al., 2014] por ejemplo han presentado una propuesta para la Colocación de Máquinas Virtuales (VMP) utilizando Optimización de Colonias de Hormigas (ACO), con el objetivo de minimizar el desperdicio de recursos en los CDCs. La esencia del trabajo radica en su diseño algorítmico que explora el espacio de soluciones para identificar un conjunto de Pareto de soluciones no dominadas, que luego se comparan con algoritmos VMP clásicos. Sin embargo, las limitaciones de su propuesta no se detallan explícitamente en el contexto proporcionado, pero de nuevo, estos algoritmos pueden enfrentar desafíos en escalabilidad, complejidad computacional y adaptabilidad a entornos dinámicos.

Los Algoritmos Miméticos (MA) combinan la evolución genética con técnicas de búsqueda local para refinar soluciones. Están inspirados en la idea de evolución cultural, donde cada individuo de la población puede ser afinado para alcanzar soluciones más precisas. El enfoque mimético es eficaz en problemas que requieren ajustes más precisos tras la exploración global. [Pires & Barán, 2013] presentan su propuesta para la ubicación de Máquinas Virtuales (VMP) en data centers, enfatizando la importancia de considerar los Acuerdos de Nivel de Servicio en el proceso de optimización multiobjetivo. Los autores presentan un algoritmo mimético multiobjetivo que tiene como objetivo optimizar el VMP equilibrando varios factores como el consumo de energía, la utilización de recursos y los requisitos de SLA. El algoritmo propuesto se valida contra una búsqueda exhaustiva de fuerza bruta, demostrando su efectividad y escalabilidad en diferentes

escenarios. Sin embargo, el artículo no trata las posibles limitaciones de la solución propuesta, pero probablemente el algoritmo de búsqueda exhaustiva utilizado para la comparación carece de escalabilidad para instancias más grandes debido a las limitaciones computacionales. El espacio de solución podría estar limitado ya que el número de soluciones factibles disminuye con un aumento en el porcentaje de VMs críticas. La complejidad del algoritmo mimético propuesto puede requerir recursos computacionales significativos para problemas a gran escala. Por último, la formulación actual considera un Acuerdo de Nivel de Servicio simple con solo un nivel de criticidad binario, que puede no capturar la complejidad de los SLAs del mundo real.

El Algoritmo de Luciérnagas (FA) se basa en el comportamiento de las luciérnagas, que emiten luz para atraer a otras. Este algoritmo utiliza este comportamiento para guiar la búsqueda, permitiendo a las soluciones más brillantes o mejores atraer a otras soluciones para concentrarse en las áreas más prometedoras del espacio de soluciones. [Li et al., 2015] en su artículo abordan el problema de colocación de máquinas virtuales en entornos en el cloud, centrándose en un modelo de optimización multiobjetivo que tiene como objetivo minimizar el consumo de energía y el desperdicio de recursos. El núcleo de la solución es el Algoritmo de Luciérnagas Discreto, que simula el comportamiento de las luciérnagas para encontrar la colocación óptima de las máquinas virtuales. La estrategia de movimiento del algoritmo permite que las soluciones menos óptimas (luciérnagas más oscuras) se inclinen hacia las más óptimas (luciérnagas más brillantes), con una estrategia de discretización aplicada para adaptar las posiciones continuas a colocaciones discretas de máquinas virtuales. Además, un mecanismo de búsqueda local acelera el proceso de búsqueda de soluciones. A pesar de sus ventajas, las limitaciones del algoritmo incluyen posibles problemas de escalabilidad en entornos de nube más grandes y la necesidad de más pruebas para validar su efectividad en diversos escenarios.

El Algoritmo de Optimización de Ballenas (WOA) simula el comportamiento de las ballenas al cazar presas. Utiliza este proceso para explorar y explotar el espacio de soluciones de manera eficaz. El enfoque es especialmente útil para problemas que requieren exploración y explotación equilibradas. [Abdel-Basset, 2018] presentan su solución también para optimizar la colocación de máquinas virtuales en entornos cloud, centrándose en la asignación eficiente de ancho de banda. Los autores proponen un Algoritmo de Optimización de Ballenas basado en Lévy mejorado (ILWOA) híbrido

con una nueva política de asignación de ancho de banda (BWAP). Este método tiene como objetivo minimizar el número de máquinas físicas en funcionamiento en los Data centers, abordando el problema de colocación de VM formulado como un problema de asignación de contenedores de tamaño variable (VSBPP). La investigación utiliza la herramienta CloudSim para validar la efectividad del algoritmo en varios conjuntos de datos, demostrando un rendimiento superior en comparación con los algoritmos de optimización clásicos. Sin embargo, el estudio tiene limitaciones. El ILWOA propuesto se prueba principalmente en un entorno simulado, que puede no capturar completamente las complejidades de los data centers en un entorno de producción real. Además, el enfoque en el ancho de banda como único criterio para la colocación de VM podría pasar por alto otros factores críticos como el uso de CPU, memoria y requisitos de almacenamiento.

El Algoritmo Sine-Cosine y el Algoritmo Salp Swarm exploran y explotan el espacio de soluciones utilizando funciones matemáticas y el comportamiento de agrupación de salpas, respectivamente. Estos algoritmos son aplicables en problemas donde el equilibrio entre exploración y explotación es fundamental. [Gharehpasha, 2020] en su artículo Introduce el uso de un algoritmo híbrido que combina el Algoritmo de Seno-Coseno (SCA) y el Algoritmo de Enjambre de Salpas (SSA) con funciones caóticas para la optimización discreta multiobjetivo para abordar el problema de la colocación de máquinas virtuales proponiendo una solución que minimiza las máquinas físicas activas y equilibra la utilización de recursos. Sin embargo, en el mismo artículo se reconoce la complejidad inherente del problema de colocación y los desafíos para lograr un equilibrio perfecto entre la exploración y la explotación en algoritmos metaheurísticos. También sugiere que investigaciones futuras podrían explorar la aplicación del esquema propuesto a otros problemas relacionados con el cloud, como topologías multi-Data center, programación de tareas o seguridad.

El Algoritmo de Manada de Krill (KHA) se inspira en el comportamiento del krill al agruparse para encontrar los mejores recursos. Esta técnica permite que el algoritmo explore y se centre en las regiones más prometedoras del espacio de soluciones, encontrando soluciones óptimas mediante la colaboración y el comportamiento de agrupación. [Baalamurugan & Bhanu, 2020] en su trabajo presentan un algoritmo de manada de krill multiobjetivo para optimizar la colocación de máquinas virtuales en entornos cloud. Enfatizan la importancia de la utilización eficiente de recursos y la reducción del consumo de energía, que son críticos en la gestión de infraestructuras en el

cloud. El algoritmo propuesto tiene como objetivo lograr un equilibrio entre estos objetivos, lo que lleva a un uso más efectivo de los recursos y el ahorro de energía. Sin embargo, las limitaciones de este enfoque también incluyen la complejidad del algoritmo y la necesidad de pruebas extensivas en escenarios del mundo real para validar su efectividad y escalabilidad.

En resumen, las técnicas basadas en población son herramientas poderosas para abordar problemas de optimización complejos. Al simular procesos naturales y aprovechar la colaboración y el comportamiento colectivo, estos algoritmos pueden encontrar soluciones óptimas de manera eficaz y robusta, aunque no por ello no tengan sus limitaciones tanto en implementabilidad o en su aplicabilidad a topologías de CDCs múltiples.

Y llegamos a los algoritmos basados en soluciones únicas y que son un enfoque diferente para la resolución de problemas en comparación con otros métodos que utilizan múltiples soluciones y evolucionan hacia un óptimo global. En estos algoritmos, se parte de una solución inicial y se refina mediante un proceso iterativo para alcanzar el mejor resultado posible.

El EAGLE Algorithm toma como inspiración el método de caza del águila, donde el águila identifica a su presa desde las alturas y luego se lanza en picado para capturarla. De manera similar, el algoritmo EAGLE comienza con una única solución que se identifica como prometedora, y mediante iteraciones sucesivas, refina esa solución en busca del óptimo. Al igual que el águila, el algoritmo tiene un "ojo agudo" para detectar pequeñas mejoras y ajustarlas en cada iteración. A medida que avanza, el algoritmo concentra su atención en un área específica, evitando distracciones y centrándose en alcanzar la solución óptima. [Li & Qian, 2013] en su artículo presentan su solución basada en este algoritmo para el problema de colocación de máquinas virtuales (VMP), con el objetivo de minimizar el consumo total de energía optimizando la implementación de máquinas virtuales en máquinas físicas. Los autores presentan un modelo de partición de espacio multidimensional para caracterizar los estados de uso de recursos de las PMs y proponen el algoritmo EAGLE, que equilibra la utilización de recursos multidimensionales, reduciendo así el número de PMs en funcionamiento y disminuyendo el consumo de energía. Sin embargo, el artículo reconoce la complejidad del problema VMP, que es NP-completo, lo que indica que encontrar una solución óptima es computacionalmente desafiante. Además, las soluciones propuestas se evalúan

mediante simulaciones y experimentos en trazas reales, lo que sugiere que, aunque el algoritmo EAGLE puede ahorrar energía significativa, se basa en suposiciones y modelos específicos que pueden no capturar completamente la dinámica de los data centers del mundo real.

El Imperialist Competitive Algorithm (ICA), se basa en un concepto muy diferente pero igualmente interesante: la historia política y las competencias entre imperios. Este algoritmo simula una competencia entre diferentes potencias imperialistas que buscan expandir su influencia. Comienza con un "imperio" inicial, que se considera la solución principal, y luego genera "colonias" que se expanden en territorios vecinos. A medida que avanza el proceso, las colonias compiten entre sí y con el imperio, refinando continuamente la solución. La competencia imperialista impulsa el proceso de optimización, buscando la solución ideal a través de la expansión, conquista y unificación. [Jamali et al., 2017] utilizan este enfoque en su trabajo, donde el artículo presenta el Algoritmo Competitivo Imperialista (ICA) para la Colocación de Máquinas Virtuales (VMP) en entornos de cloud destinado a optimizar la utilización de recursos y la eficiencia energética. El algoritmo ICA-VMPLC se destaca por su facilidad de movimiento en el vecindario, su rápida tasa de convergencia y la idoneidad de su terminología para el problema en cuestión. Explora el espacio de búsqueda para encontrar una solución de colocación óptima que minimice tanto el consumo de energía como el desperdicio total de recursos. Sin embargo, el artículo no profundiza en los posibles problemas de escalabilidad ni en el rendimiento del algoritmo en entornos de nube a gran escala y reales, que son factores críticos para las aplicaciones prácticas

Estos dos algoritmos demuestran la diversidad de enfoques para resolver problemas complejos. Mientras que el EAGLE Algorithm se inspira en la precisión y el enfoque de la caza del águila, el Imperialist Competitive Algorithm utiliza la dinámica de la expansión y competencia política para refinar soluciones.

La revisión detallada de las diversas estrategias de optimización para la asignación de Máquinas Virtuales en entornos Cloud data centers ha destacado varios enfoques innovadores. Desde los métodos basados en Constraint Programming hasta las técnicas modernas de Machine Learning y algoritmos bioinspirados, cada estrategia presenta sus propios puntos fuertes y debilidades.

Los métodos convencionales, como Constraint Programming y Stochastic Integer Programming, han sido efectivos para abordar problemas específicos de asignación de recursos. Sin embargo, su complejidad y limitada adaptabilidad a entornos en constante cambio plantean desafíos significativos, especialmente en contextos donde la escalabilidad y la capacidad de adaptación son fundamentales. Los enfoques modernos, como Reinforcement Learning, Artificial Neural Networks (ANNs) y Fuzzy Reinforcement Learning, han demostrado ser prometedores gracias a su capacidad para aprender y adaptarse a entornos cambiantes, aunque enfrentan desafíos relacionados con la complejidad algorítmica y la escalabilidad.

Los algoritmos bioinspirados, como los Algoritmos Genéticos, la Optimización por Colonias de Hormigas y la Optimización por Manada de Krill, han mostrado eficacia en la exploración de amplios espacios de soluciones, pero también pueden enfrentar problemas de complejidad y adaptabilidad a entornos de producción reales. Por último, los algoritmos basados en soluciones únicas, como el EAGLE Algorithm y el Imperialist Competitive Algorithm (ICA), aportan una perspectiva diferente, pero también requieren un cuidadoso equilibrio para obtener resultados óptimos.

En este contexto, esta tesis aborda un aspecto crucial que puede superar muchas de estas limitaciones. El uso de sistemas expertos basados en Fuzzy logic y Swarm intelligence para optimizar el consumo de energía renovable en Cloud data centers dispersos geográficamente se presenta como una estrategia prometedora. Al abordar la variabilidad del suministro de energía renovable y la necesidad de soluciones flexibles y adaptativas, esta tesis puede ofrecer un nuevo enfoque que combina la inteligencia colectiva y la lógica difusa para lograr una gestión de recursos más eficiente y sostenible.

El enfoque propuesto se justifica por su capacidad para adaptarse a fluctuaciones en el suministro de energía y por su potencial para optimizar la migración de máquinas virtuales entre data centers dispersos. Al aprovechar el Swarm intelligence para la adquisición de conocimiento y la lógica difusa para abordar la incertidumbre, esta tesis aspira a ofrecer soluciones que no solo reduzcan el consumo de energía, sino que también mejoren la eficiencia general y la sostenibilidad de los Cloud data centers.

2.6.3 Gestión de Máquinas Virtuales en Tiempo de Ejecución para la Optimización Energética

La gestión de máquinas virtuales en tiempo de ejecución aborda uno de los desafíos más críticos en los data centers: la optimización del consumo de energía. Esta optimización se articula principalmente a través de dos estrategias fundamentales: el "Energy Capping" y la "Consolidación de VMs", cada una con sus propios enfoques y desafíos específicos.

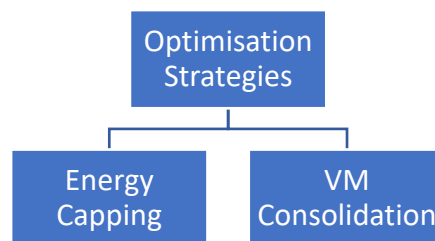


Figure 1. Estrategias de optimización del consumo de energía en los cloud data centers

Energy Capping

La estrategia de "Energy Capping" se centra en limitar los recursos de los hosts cuando se detecta una infrautilización, por ejemplo, mediante la reducción de la frecuencia o ciclos de las CPUs. Este ajuste se realiza a nivel de hardware, lo que permite una disminución directa en el consumo energético. Sin embargo, esta aproximación presenta complicaciones, ya que no considera el entorno multi-VM de los hosts. Si una VM reduce sus necesidades computacionales y se aplica un "Energy Capping", podría inadvertidamente comprometer el rendimiento de otras VMs alojadas en el mismo host, afectando así negativamente los acuerdos de nivel de servicio y los estándares de calidad de servicio (QoS) [Khan & Tian, 2021].

VM Consolidation

Por otro lado, la "Consolidación de VMs" busca una eficiencia operativa agrupando múltiples VMs en el menor número posible de hosts físicos. Este enfoque

reduce la cantidad de hosts activos necesarios, permitiendo el apagado de aquellos que no se necesitan y, por ende, generando un ahorro energético significativo. Este proceso de consolidación es complejo y ha sido objeto de diversas investigaciones. Un ejemplo notable es el trabajo de [Yang et al., 2014], que utiliza el Análisis de Componentes Principales (PCA) para identificar las variables críticas en las decisiones de consolidación y, además, propone un modelo de ML basado en Regresión de Vectores de Soporte (SVR) para optimizar las estrategias de Scheduling de recursos. A pesar de estos avances, la predicción precisa de las necesidades de recursos de una sola VM sigue siendo un desafío, especialmente cuando se consideran factores más allá de la CPU, como la memoria y el almacenamiento, y la naturaleza no lineal de las demandas de los usuarios finales.

En respuesta a estos desafíos, estrategias más sofisticadas que emplean análisis predictivo y múltiples predictores han emergido para facilitar una consolidación de VMs más efectiva, limitando migraciones innecesarias y reduciendo el consumo energético global de los data centers. Un ejemplo innovador de esta aproximación es el estudio de [Nguyen et al., 2017], que propone un enfoque avanzado para la consolidación de VMs. Este método se basa en el análisis predictivo para optimizar tanto la colocación actual como futura de las VMs, buscando reducir el consumo de energía y los costes operativos asociados con las migraciones de VMs. Aunque prometedora, esta línea de investigación sugiere que hay margen para refinar aún más los modelos predictivos, considerando la dinámica fluctuante y compleja tanto de la demanda de recursos como de su utilización efectiva.

En conjunto, estas estrategias subrayan la importancia de una gestión de VMs en tiempo de ejecución que no solo sea eficiente desde el punto de vista energético, sino también adaptativa y sensible a las necesidades cambiantes de los data centers modernos. La incorporación de técnicas avanzadas de machine learning en estos procesos promete mejoras significativas en la eficiencia energética y también en la sostenibilidad general de los entornos de cloud computing.

2.6.3.1 Optimización por migración de máquinas virtuales

La migración de VMs puede definirse como el proceso de transferir una VM de un Host/anfitrión a otro, manteniendo la conexión con la aplicación o cliente y representa

un enfoque que permite mejorar la eficiencia operativa y energética de los data centers, asegurando al mismo tiempo que se mantenga la calidad y la continuidad del servicio ofrecido a los clientes [Wan et al., 2012], [Dong, 2013]. Este proceso se clasifica en tres categorías principales: la migración en vivo, la migración en frío y la migración híbrida.

Migración en Vivo (Live Migration): Se refiere al acto de transferir una VM activa de un servidor físico a otro, causando poca o ninguna interrupción a los servicios en curso. Este mecanismo ayuda en el mantenimiento del hardware, la distribución de carga, la resistencia a fallos y la optimización de energía en data centers [Rosenblum & Garfinkel, 2005] [Minarolli & Freisleben, 2011]. Uno de los principales obstáculos con la migración en vivo es la complejidad de lograr simultáneamente un tiempo de inactividad reducido, una duración total de migración más corta y un uso eficiente de los recursos de CPU y red [Le, 2020]. Si bien minimiza el tiempo de inactividad y garantiza una alta disponibilidad, requiere una infraestructura de red robusta y puede ser más compleja en su implementación [Clark et al., 2005].

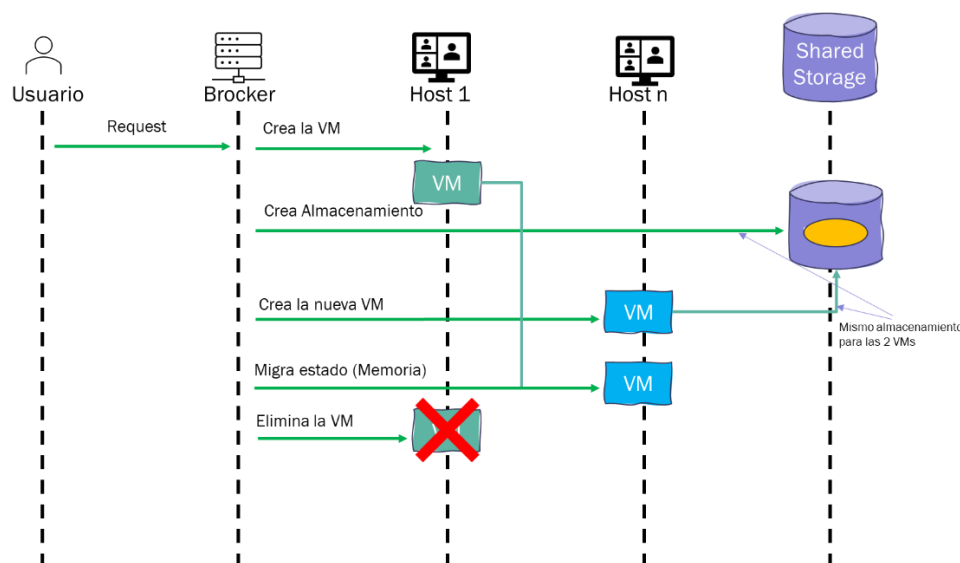


Figura 2-11 Diagrama de secuencia de una migración en vivo

Migración en Frío (Cold Migration): En contraste, la migración en frío es una forma sencilla de trasladar una Máquina Virtual de un host a otro cuando una VM no está realizando ninguna tarea importante en tiempo real, se puede pausar y mover todo lo que está haciendo. Esto significa que la VM no está funcionando durante el traslado, por eso se llama "migración en frío". Aunque este método puede hacer que la VM no esté disponible durante un tiempo más largo [Imran, 2022], también hace que el traslado sea más rápido y se transfiera menos datos. Además de ser un proceso más sencillo. La

migración en frío es especialmente útil en ciertas situaciones como cuando se quiere mover una VM entre máquinas que tienen diferentes tipos de procesadores (Intel/AMD); en este caso no se puede usar la "migración en vivo" (donde la VM sigue funcionando durante el traslado) [Le, 2020].

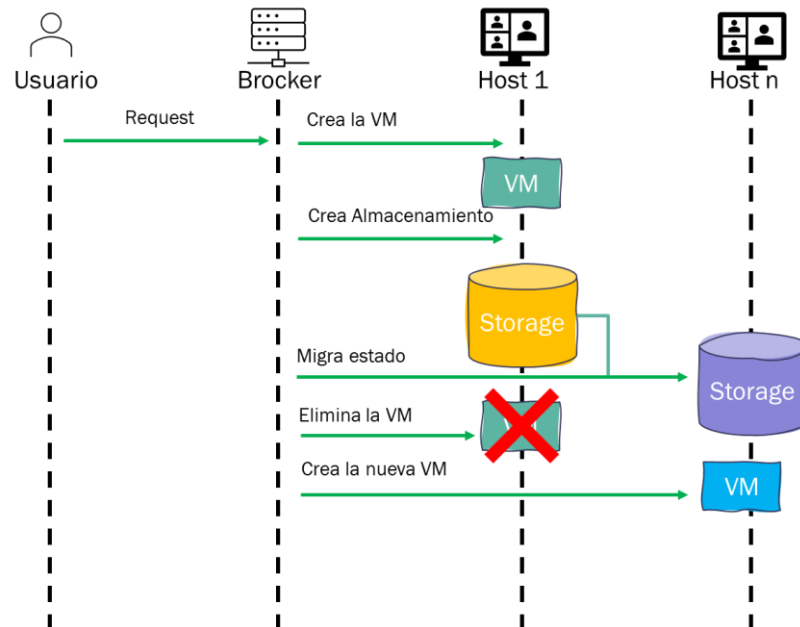


Figura 2-12 Diagrama de secuencia de una migración en frío

Migración Híbrida: Esta modalidad combina elementos de las migraciones en vivo y en frío [Kaur, 2017], realizando parte del proceso con la VM en funcionamiento y otra parte con la VM apagada. Esto ofrece un equilibrio entre minimizar el tiempo de inactividad y la complejidad operativa, aunque requiere una coordinación y planificación más elaboradas para llevar a cabo la migración de manera efectiva [Noshy et al., 2018], [Ahmad et al., 2015].

En la literatura sobre el cloud computing, se han propuesto numerosas técnicas y usos para la migración de máquinas virtuales, desde enfoques tradicionales hasta metodologías influenciadas por el machine learning. Estas técnicas destacan la adaptabilidad necesaria para enfrentar los desafíos de variabilidad de la demanda en los entornos virtuales.

Comenzando por los trabajos más antiguos, [Hines & Gopalan, 2009] discuten en su artículo un enfoque para la migración en vivo de máquinas virtuales basada en post-copy, que ofrece una mejora significativa sobre los métodos tradicionales de Pre-copy. La idea fue la de diferir la transferencia del contenido de la memoria de una máquina virtual

hasta después de haber enviado su estado del procesador al host objetivo. Esta estrategia tiene como objetivo reducir el tiempo total de migración mientras se mantienen los beneficios de vivacidad de la migración Pre-copy. Los autores proponen una pre-paginación adaptativa para eliminar las transmisiones de páginas duplicadas y “auto inflado dinámico” (DSB) para evitar la transferencia de páginas de memoria libres, acelerando así el proceso de migración con un impacto mínimo en el rendimiento. El artículo reconoce ciertas limitaciones. La efectividad de la pre-paginación adaptativa y el DSB puede variar dependiendo de la carga de trabajo específica y el tamaño del conjunto de trabajo de la VM. Además, el enfoque de post-copy podría potencialmente llevar a un aumento de fallos de página vinculados a la red si no se gestiona con cuidado.

Avanzando en el tiempo, [Sagana & Suganthe, 2013] propusieron en su trabajo lo que han llamado “Optimized iterative pre-copy algorithm” que disminuye la tasa de páginas "sucias" durante la migración. El artículo propone esta solución para atacar el problema de la optimización de las migraciones en vivo en entornos cloud, centrándose en reducir el tiempo total de migración y el tiempo de inactividad. Proponen el algoritmo de reemplazo de páginas WSClock para minimizar las páginas de memoria transferidas. Este enfoque tiene como objetivo mejorar la eficiencia de los data centers y clústeres facilitando el movimiento de instancias del sistema operativo a través de diferentes hosts físicos. Este artículo no proporciona detalles sobre las limitaciones de los métodos propuestos, pero podemos prever que las limitaciones podrían incluir la escalabilidad del algoritmo en diversos entornos o el impacto en los recursos del sistema y los compromisos entre la velocidad de migración y el rendimiento del sistema.

Posteriormente, [Nashaat & Ashry, 2019] presentaron dos algoritmos cooperativos; el “Smart Elastic Scheduling Algorithm (SESA)” y el “Adaptive Worst Fit Decreasing Virtual Machine Placement (AWFDVP)”. Estos algoritmos buscan optimizar la distribución de recursos físicos en sistemas cloud para lograr una consolidación y balanceo óptimos para el consumo de energía, memoria y tiempo de procesamiento. SESA agrupa las máquinas virtuales basándose en parámetros de memoria y CPU, manejando VMs localizadas en una misma instancia y que comparten páginas de memoria como un grupo. Luego, AWFDVP toma decisiones de migración para minimizar las migraciones del sistema, conservar energía y prevenir la degradación del rendimiento de las VMs mientras se mantiene el equilibrio de carga del sistema. En su artículo afirman haber obtenido mejoras significativas en el consumo de energía, el número de

migraciones y la degradación del rendimiento. Sin embargo, también reconocen limitaciones como la complejidad del mecanismo de consolidación dinámica de VM y la necesidad de más investigaciones sobre técnicas de Live Gang Migration (LGM) para optimizar la transferencia de datos durante las migraciones de VMs.

El mismo año, [Shukla et al., 2019] propusieron un enfoque de Migración en Vivo Basado en Pre-copy que opera en múltiples fases para transferir páginas de memoria de manera más eficiente y predecir el comportamiento de las páginas para decidir si enviar ciertas páginas o no. Este método utiliza herramientas de simulación CloudSim para evaluar el rendimiento, con el objetivo de minimizar el Tiempo Total de Migración (TMT) y el Tiempo de Inactividad (DT). Sin embargo, las limitaciones de este enfoque no se discuten explícitamente en el artículo, pero en general, tales estrategias también pueden enfrentar desafíos en la implementación para entornos de producción reales debido su complejidad y a la naturaleza muy aleatoria y dinámica de las cargas de trabajo.

Más recientemente, [Jalaei et al., 2020] propusieron una estrategia basada en el framework Virtual CPU Scheduling for Post-Copy (VCSP), con ello intentan optimizar la migración en vivo de máquinas virtuales en entornos cloud. El framework VCSP tiene como objetivo equilibrar la velocidad de procesamiento de las páginas en la máquina objetivo con su velocidad de transmisión desde la máquina fuente mediante la reducción de la velocidad de la CPU virtual. Esta técnica aborda los problemas comunes en la migración en vivo post-copy, como el tiempo de inactividad del sistema, el tiempo total de migración y la eficiencia del rendimiento. El artículo demuestra que VCSP puede mejorar significativamente estos aspectos, con una reducción de hasta el 8.17% en el tiempo de inactividad del sistema, 30.33% en el tiempo total de migración, 54.49% en el total de páginas transferidas y 23.65% en el rendimiento del sistema. El artículo reconoce ciertas limitaciones de la propuesta VCSP. Señala que reducir la frecuencia de la CPU virtual no es beneficioso universalmente para todos los métodos de transferencia de páginas dentro del enfoque post-copy. Específicamente, la técnica de prefetching utilizada en VCSP puede no ser adecuada para entornos con cargas de trabajo altas debido al posible aumento del tráfico de red y la complejidad de predecir las referencias de páginas.

Este recorrido a través del tiempo muestra cómo las técnicas de migración de máquinas virtuales se han adaptado a las necesidades cambiantes del cloud computing, ofreciendo soluciones cada vez más eficientes y específicas para los retos que presentan los entornos virtuales modernos.

En la literatura también podemos encontrar otro tipo de estrategias para abordar esta problemática de migración de las máquinas virtuales mediante el uso de metaheurística inspirada en el Swarm intelligence.

En el trabajo de [Kansal & Chana, 2016] se presenta una propuesta de implementación de una técnica de Migración de Máquinas Virtuales Consciente de la Energía basada en Optimización de Luciérnagas (FFO-EVMM). La técnica aprovecha el algoritmo inspirado en la naturaleza de la luciérnaga para identificar y migrar las VM más cargadas a los nodos activos menos cargados, logrando así un ahorro de energía significativo y reduciendo el número de migraciones y hosts activos requeridos. Sin embargo, el artículo reconoce la complejidad del problema, señalando que lograr la optimización energética a través de la migración de VM es un problema NP-difícil, que a menudo requiere métodos heurísticos para encontrar soluciones casi óptimas. La técnica FFO-EVMM propuesta, aunque efectiva, se basa en simulaciones con datos teóricos utilizando CloudSim y esto da una idea limitada del comportamiento de este algoritmo en condiciones reales de uso. También la dependencia de enfoques metaheurísticos significa que, si bien la técnica puede producir resultados de alta calidad, no siempre garantiza la mejor solución absoluta debido a la naturaleza probabilística del algoritmo.

En el mismo año [Qinghua et al., 2016] presentaron una técnica de optimización basada en biogeografía (BBO) multiobjetivo llamada VMPMBBO. Los autores abordan el desafío de encontrar soluciones óptimas de manera efectiva y eficiente para la migración de VMs (VMcP). El algoritmo VMPMBBO propuesto trata el VMcP como un sistema complejo y tiene como objetivo optimizar la colocación de máquinas virtuales minimizando tanto el desperdicio de recursos como el consumo de energía. Los experimentos se realizaron utilizando datos sintéticos de test y otros con trazas reales para demostrar la efectividad de su aporte. Sin embargo, el estudio reconoce que el rendimiento del algoritmo depende de la precisión del modelo de consumo de energía y del equilibrio de la utilización de recursos, lo que puede no capturar completamente la naturaleza dinámica de los entornos de nube del mundo real.

En el trabajo de [Rodríguez et al., 2017] tratan de reducir el Retraso en los tiempos de respuesta en el Edge Cloud computing (ECC) considerando tanto estos retrasos en la Transmisión como retrasos de Procesamiento en la VMs. Los autores propusieron un modelo de Optimización por Enjambre de Partículas (PSO) que integra la migración de Máquinas Virtuales y el control de potencia de transmisión para mejorar la calidad del

servicio en escenarios de múltiples cloudlets. Este doble enfoque en los retrasos de cálculo y comunicación demuestra según ellos ser superior a los métodos clásicos. El modelo PSO propuesto logra resultados cercanos a la optimalidad mientras mantiene tiempos de ejecución bajos, incluso bajo cargas altas de procesamiento y transmisión. En relación con las limitaciones, el artículo no las aborda, sin embargo, y teniendo en cuenta la naturaleza del algoritmo PSO utilizado, podemos intuir algunas como las inherentes a los algoritmos metaheurísticos, que no garantizan encontrar la solución óptima global o la posibilidad de quedarse atascado en óptimos locales, especialmente en espacios de búsqueda multidimensionales complejos. También podemos hablar del rendimiento dependiente en gran medida de la configuración de sus parámetros, como el número de partículas, el peso de inercia y las constantes de aceleración, donde configuraciones incorrectas de los parámetros pueden llevar a una convergencia prematura o a un tiempo de cálculo excesivo.

En 2020 encontramos el trabajo de [Hossain et al., 2020] donde los autores proponen el algoritmo Active & Idle Virtual Machine Migration (AIVMM), un método computacional evolutivo inspirado en el Sistema de Colonia de Hormigas (ACS). Este algoritmo tiene como objetivo migrar máquinas virtuales inactivas a servidores inactivos, reduciendo así las interrupciones de energía para las máquinas activas y optimizando el uso de energía. Sin embargo, el documento reconoce limitaciones como la necesidad de realizar más pruebas en escenarios reales para validar la efectividad del algoritmo y los posibles problemas de escalabilidad en entornos más grandes.

También en el mismo año, [Sha et al., 2020] en su artículo presentan una propuesta que aborda la utilización de la migración de VMs para la gestión de energía en CDCs, para ello introducen un Algoritmo metaheurístico de Forrajeo Bacteriano Discreto. Los autores utilizaron el simulador CloudSim para validar la eficiencia de su método y demostraron mejoras en el consumo de energía y el número de migraciones. Sin embargo, las limitaciones del estudio incluyen la necesidad de pruebas en entornos reales para confirmar los resultados del simulador y la aplicabilidad en entornos más complejos a mayor escala.

Los enfoques de migración de máquinas virtuales han evolucionado para abordar las crecientes demandas de los CDCs. Se pueden clasificar en tres tipos principales:

migración en vivo, migración en frío y migración híbrida. Cada uno tiene ventajas y limitaciones inherentes.

La migración en vivo permite la transferencia de VMs entre hosts con mínima o ninguna interrupción del servicio. Sin embargo, su complejidad radica en equilibrar el tiempo de inactividad, la duración de migración y el uso de recursos, requiriendo infraestructura robusta y mayor complejidad operativa. La migración en frío, más sencilla, implica interrupciones mayores, pero reduce la cantidad de datos transferidos y puede ser útil para migraciones entre diferentes arquitecturas. Por otro lado, la migración híbrida combina aspectos de ambos enfoques, permitiendo cierto equilibrio entre complejidad y tiempo de inactividad, aunque necesita una coordinación precisa para ser eficaz.

Los métodos basados en algoritmos post-copy y pre-copy, como los propuestos por Hines & Gopalan y Sagana & Suganthe, buscan optimizar el proceso de migración, reduciendo el tiempo total de migración y el tiempo de inactividad. Sin embargo, enfrentan limitaciones de escalabilidad, manejo de páginas de memoria y compromiso entre velocidad y rendimiento del sistema. En un ámbito más reciente, enfoques como VCSP y SESA/AWFDVP buscan mejorar la eficiencia y el equilibrio del sistema, pero también se encuentran con desafíos como la complejidad en la consolidación dinámica de VMs y las dificultades de aplicación en entornos de producción reales.

Las estrategias basadas en metaheurísticas, como FFO-EVMM y VMPMBBO, ofrecen soluciones para la migración de VMs, inspiradas en Swarm intelligence. Aunque estas técnicas prometen ahorros de energía significativos y optimización de recursos, presentan limitaciones como la dependencia de simulaciones teóricas, el comportamiento probabilístico de los algoritmos y la posible falta de aplicabilidad en entornos muy dinámicos.

Finalmente, métodos inspirados en otros sistemas naturales, como el PSO y ACS, proponen soluciones para optimizar la migración de VMs y el control de potencia para mejorar la calidad del servicio y reducir el consumo energético. A pesar de que estos enfoques muestran resultados prometedores, sus limitaciones incluyen la posibilidad de quedarse atascados en óptimos locales, la sensibilidad a la configuración de parámetros y la necesidad de validación en entornos reales.

En conclusión, aunque la migración de VMs ha demostrado ser una herramienta valiosa para la optimización de recursos en cloud computing y Data centers, las limitaciones persisten, reflejadas en la complejidad operativa, la escalabilidad, la eficiencia del sistema y la aplicación en escenarios del mundo real. Estos desafíos subrayan la importancia de un enfoque continuo para la investigación y el desarrollo de técnicas de migración innovadoras y eficientes.

2.6.3.2 Optimización por balanceo de carga

El balanceo de carga constituye una parte fundamental del cloud computing Figura 2-13, esencial para mantener la eficiencia y la robustez del sistema. Se refiere a la técnica utilizada para distribuir las cargas de trabajo de manera uniforme entre servidores y otros recursos computacionales, con el objetivo de optimizar el uso de recursos, minimizar el tiempo de respuesta y evitar la sobrecarga [Muneeswari et al., 2024]. Esta distribución uniforme garantiza que ningún recurso esté sometido a una carga excesiva mientras otros permanecen infrautilizados, un equilibrio crucial en entornos dinámicos como los CDCs.

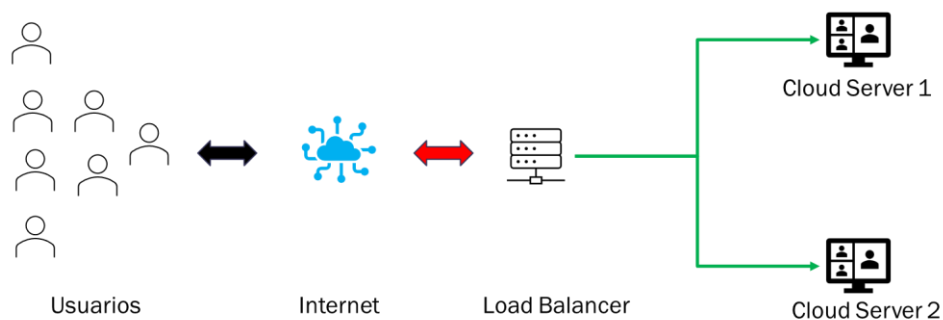


Figura 2-13 Mecanismo de balanceo de carga en el cloud

En el ámbito del cloud computing, la relevancia del balanceo de carga se acentúa debido a la naturaleza dinámica y escalable de los recursos. En estos entornos, los recursos pueden ser agregados o retirados según la demanda, lo que requiere mecanismos de balanceo de carga eficientes y adaptables para evitar cuellos de botella y tiempos de inactividad innecesarios [Shafiq et al., 2022]. La capacidad para equilibrar la carga entre

servidores no solo mejora el rendimiento y la experiencia del usuario final, sino que también contribuye a la eficiencia energética, permitiendo que los servidores que no están en uso entren en un estado de bajo consumo.

El balanceo de carga tiene un impacto directo en la experiencia del usuario y en la calidad del servicio. Una distribución de carga ineficiente puede llevar a tiempos de respuesta prolongados, errores de conexión y una experiencia general insatisfactoria para el usuario. Además, desde una perspectiva de infraestructura, el uso desigual de recursos puede provocar un desgaste acelerado en servidores sobrecargados y un desperdicio de recursos en servidores inactivos, aumentando los costes operativos y disminuyendo la sostenibilidad del sistema [Qazi et al., 2024].

A nivel energético, el balanceo de carga desempeña un papel crítico en la reducción del consumo de energía en los data centers [Mishra et al., 2020; Ali et al., 2020]. La distribución uniforme de las cargas de trabajo evita el uso excesivo de ciertos recursos y permite la desactivación de servidores no utilizados. Esto no solo tiene implicaciones económicas positivas, sino también ambientales, ya que reduce el consumo de energía y las emisiones de carbono asociadas con el funcionamiento de grandes data centers. La eficiencia energética es una prioridad creciente para las empresas tecnológicas, y el balanceo de carga se posiciona como un componente clave para alcanzar estos objetivos.

Sin embargo, la implementación efectiva del balanceo de carga en entornos de cloud computing plantea desafíos significativos. Requiere algoritmos sofisticados que puedan adaptarse a la naturaleza cambiante del entorno en el cloud y responder rápidamente a las variaciones en la carga de trabajo y la disponibilidad de recursos. A medida que el cloud computing evoluciona y crece en escala y complejidad, las técnicas tradicionales de balanceo de carga pueden no ser suficientes para abordar estas necesidades, lo que subraya la importancia de desarrollar enfoques más avanzados y adaptativos para el balanceo de carga.

Aunque el balanceo de carga es crucial para el rendimiento y la eficiencia en el cloud computing, su implementación en un entorno tan dinámico y escalable presenta varios desafíos. Los algoritmos tradicionales de balanceo de carga, que han sido efectivos en sistemas estáticos y menos complejos, a menudo no pueden adaptarse adecuadamente a las condiciones variables de un data center en el cloud [Zhang et al., 2018]. Estos algoritmos suelen basarse en enfoques estáticos que no responden a los cambios en el

entorno, lo que puede llevar a una asignación subóptima de recursos y a una gestión ineficiente del tráfico en la red.

Un problema común con los métodos tradicionales de balanceo de carga es su enfoque estático, que se basa en una instantánea del estado del sistema en un momento dado. Por ejemplo, el método ECMP (Equal-Cost Multi-Path) [Thaler & Hopps, 2000], ampliamente utilizado incluso en la actualidad, emplea un mecanismo de hash estático basado en varios campos de los encabezados de los paquetes, como la dirección IP de origen y destino. Este valor hash se asigna luego a una de las rutas de igual coste hacia el siguiente nodo en la red. La implementación de ECMP resulta particularmente práctica y factible en las infraestructuras de red existentes, ofreciendo una integración sencilla y directa. Sin embargo, pese a sus ventajas en términos de implementación, ECMP presenta limitaciones significativas, especialmente en contextos de redes de data centers, donde es susceptible a provocar congestiones. Esto se debe principalmente a que el mecanismo de hash estático puede llevar a que flujos de datos voluminosos y concurrentes colisionen en un mismo camino, saturando la ruta y deteriorando el rendimiento general de la red. Esta vulnerabilidad subraya la necesidad de considerar alternativas o mejoras que permitan una distribución más dinámica y eficiente del tráfico para evitar estos puntos de congestión críticos.

Esto puede ser problemático en topologías de red donde los recursos y las cargas de trabajo están en constante cambio. Pero Más allá de las consideraciones de topología y tráfico, las técnicas de balanceo de carga actuales tienen varios desafíos a superar; El primero es la necesidad de detectar rápidamente y con precisión la congestión, al tiempo que se mide el ancho de banda del cuello de botella de un camino determinado. Debido a la alta variabilidad y a las ráfagas del tráfico en data centers, junto con los tiempos de ida y vuelta (RTT) de la red a nivel de microsegundos, el balanceo de carga debe ser capaz de responder a la congestión dentro de los tiempos de escala de RTT (rondando los 100 μ s) [Alizadeh et al., 2014].

En segundo lugar, los mecanismos de balanceo de carga deben ser robustos contra el desordenamiento de paquetes. Los paquetes del mismo flujo podrían distribuirse a través de múltiples caminos de red, lo que potencialmente podría llevar a desordenar el Protocolo de Control de Transmisión (TCP). Este desorden no solo reduce el rendimiento del TCP, sino que también añade una carga computacional significativa a los hosts [He et al., 2015].

En tercer lugar, los mecanismos actuales de balanceo de carga a menudo aplican el mismo enfoque algorítmico tanto para flujos cortos como largos. Sin embargo, los flujos cortos tienden a ser más sensibles a la latencia en comparación con los flujos largos, lo que requiere estrategias de manejo diferentes [Wang et al., 2014]. En cuarto lugar, las fallas de enlace son frecuentes y pueden ser disruptivas en los data centers, lo que hace necesario que los algoritmos de balanceo de carga manejen eficazmente la asimetría que estas fallas generan [Alizadeh et al., 2014].

Otros desafíos de “menor orden” de los mecanismos de balanceo de carga en entornos cloud son por ejemplo su implementación dentro de las políticas de switches y routers hardware o como monitorear todos los caminos de enrutamiento hasta los hosts finales o como controlar explícitamente el camino de un flujo a los hosts destino [Zhang et al., 2018].

Para abordar estas limitaciones, se han estado explorando enfoques más dinámicos y adaptativos para el balanceo de carga [Alizadeh et al., 2014; Ghorbani et al., 2015]. Estos nuevos métodos tienen en cuenta el estado actual del sistema y se adaptan a cambios en tiempo real, permitiendo una distribución más equilibrada de las cargas de trabajo. Los algoritmos dinámicos pueden responder a fluctuaciones en la demanda y a cambios en la infraestructura, asegurando que los recursos se utilicen de manera óptima y que se mantenga la calidad del servicio.

Uno de los principales impulsores de esta evolución es la creciente demanda de servicios en el cloud, que exige un rendimiento óptimo y una escalabilidad flexible. A medida que el cloud computing se vuelve más omnipresente y los data centers manejan cantidades cada vez mayores de tráfico, la necesidad de técnicas de balanceo de carga eficientes y adaptables se vuelve aún más crítica. Los enfoques dinámicos y adaptativos permiten a los data centers reaccionar rápidamente a las demandas cambiantes, reduciendo el riesgo de tiempos de inactividad y mejorando la eficiencia energética.

Otra tendencia significativa es la integración de tecnologías, como el Machine learning, en el proceso de balanceo de carga. Las tecnologías de machine learning como las redes neuronales recurrentes (RNN) [Mikolov et al., 2010] y la memoria a largo y corto plazo (LSTM) [Hochreiter & Schmidhuber, 1997] pueden analizar patrones de comunicación a partir de registros históricos para predecir los volúmenes de tráfico

futuros. Al aprovechar estas herramientas, los mecanismos de balanceo de carga pueden anticipar el tamaño de los flujos para mejorar la distribución de la carga.

Además, algunas tecnologías de Machine learning, como el aprendizaje profundo [LeCun, 2015] y el aprendizaje profundo por refuerzo [Mnih et al., 2015], ofrecen la capacidad de gestionar automáticamente las políticas de balanceo de carga. Estas tecnologías permiten la generación automática de políticas de balanceo para cumplir con objetivos de optimización específicos, como minimizar el tiempo promedio de finalización del flujo (FCT), reducir el percentil 99 del FCT y respetar los SLAs. En comparación con las estrategias de balanceo de carga clásicas, los enfoques basados en Machine learning pueden adaptarse mejor a una variedad de arquitecturas de red y patrones de tráfico.

En la literatura también encontramos múltiples trabajos que se basan en Swarm intelligence para abordar el balanceo de carga, como el de [Singhal & Jain, 2015] donde se destaca la importancia de los algoritmos basados en Swarm intelligence, como el Artificial Bee Colony (ABC) y Particle Swarm Optimization (PSO), para mejorar la eficiencia y efectividad de la distribución de carga en los recursos del cloud. En el mismo año [Yadav, 2015] los autores propusieron un enfoque híbrido que combina PSO con el algoritmo de Carga de Ejecución Equitativamente Distribuida (ESCEL) (Híbrido PSO & ESCEL) donde PSO se utiliza para optimizar las tareas en el servidor cloud antes de asignarlas y luego, el servidor asigna las tareas utilizando el enfoque ESCEL. Este enfoque tiene como objetivo optimizar los recursos y producir un tiempo de respuesta más rápido.

[Sharma et al., 2016] en el artículo presentan una propuesta para el balanceo de carga mediante la implementación de un Bat Algorithm. Los autores utilizaron MATLAB para comparar su algoritmo con técnicas existentes como Round Robin y Fuzzy GSO. En su trabajo resaltan el impacto de su algoritmo sobre la migración de tareas y el tiempo de respuesta.

El trabajo de [Gupta & Garg, 2017] donde presentan el algoritmo de “Load Balancing Ant Colony Optimization (LB-ACO)”, que tiene como objetivo minimizar el tiempo de ejecución y mejorar la utilización de recursos. El estudio compara LB-ACO con el algoritmo NSGA-II realizando para ello simulaciones en CloudSim y demostrando que LB-ACO logra un mejor balanceo de carga y una reducción del tiempo de ejecución.

[Parmesivan et al., 2018] en su artículo presentan una evaluación de su algoritmo híbrido de balanceo de carga CDCs basado en el Honey Bee (HB) y el Particle Swarm Optimisation (PSO) y donde demuestran su superioridad sobre los algoritmos tradicionales. Al integrar las fortalezas de ambos HB y PSO, el enfoque híbrido reduce significativamente el tiempo de respuesta de las peticiones al CDC, mejorando así el rendimiento general y la eficiencia global del sistema.

En publicaciones más recientes, encontramos el trabajo de [Ullah et al., 2019] donde presentan una revisión exhaustiva del algoritmo de Colonia de Abejas Artificiales (ABC) y su aplicación en el balanceo de carga dentro de los entornos cloud. Profundizan en la problemática de la distribución de carga entre nodos, enfatizando la importancia de lograr la optimización de recursos y la reducción de energía. El algoritmo ABC, inspirado en el comportamiento de las abejas, se destaca por su robustez, flexibilidad y rápida convergencia, lo que lo convierte en una opción adecuada para abordar problemas complejos que las técnicas clásicas y matemáticas tienen dificultades para resolver.

O el trabajo de [Singh et al., 2019] del mismo año en el que utilizan un algoritmo de balanceo de carga basado en búsqueda de cuervos (CSLBA) para mapear óptimamente las tareas a los recursos, considerando parámetros como el tiempo promedio de finalización, tiempo de espera y tiempo de procesamiento del data center. El CSLBA se compara con el algoritmo de equilibrio de carga basado en optimización de colonia de hormigas (ACOLBA), demostrando un rendimiento superior en varias métricas.

También está el trabajo de [Jena et al., 2022] donde presentan un algoritmo metaheurístico híbrido que combina la Optimización de Enjambre de Partículas Modificada (MPSO) y un algoritmo Q-learning mejorado, denominado QMPSO. La metodología propuesta intenta balancear dinámicamente la carga entre las máquinas virtuales para optimizar la utilización de recursos, maximizar el rendimiento de las VMs y minimizar el tiempo de espera de las tareas.

[Simaiya et al., 2024] proponen un modelo híbrido para el balanceo dinámico de carga en CDCs, integrando aprendizaje profundo con Inteligencia mediante una hibridación de dos técnicas CNN-LSTM (Convolutional Neural Networks (CNN) y Short-Term Memory (LSTM) que a su vez es una Recurrent Neural Network o (RNN)) con otra PSO y un Algoritmo Genético (DPSO-GA). El modelo aborda el balanceo de carga mediante la afinación de hiperparámetros de las redes neuronales y la predicción de

la utilización de recursos. Emplea un enfoque de dos fases: la primera fase utiliza un método híbrido PSO-GA para la optimización de hiperparámetros, y la segunda fase aplica un marco de trabajo CNN-LSTM donde la parte CNN realiza la clasificación de las tareas y el LSTM detecta patrones de comportamiento en el tiempo para predecir el consumo de recursos.

Queda claro que el balanceo de carga es un pilar del cloud computing actual, que distribuye uniformemente las cargas de trabajo entre servidores y otros recursos computacionales, sin ella, el riesgo de sobrecarga, cuellos de botella y tiempos de respuesta prolongados serían mucho mayores. Por eso, la optimización por balanceo de carga es crítica para mantener los data centers en equilibrio y reducir costes operativos.

Una de las grandes ventajas del balanceo de carga es su capacidad para optimizar el uso de recursos. Al distribuir el tráfico entre múltiples servidores, se evita que algunos queden saturados mientras otros permanecen inactivos. Esto no solo mejora el rendimiento general, sino que también reduce el desgaste acelerado en servidores sobrecargados y permite desactivar aquellos que no se están utilizando, contribuyendo así a la eficiencia energética y a la sostenibilidad medioambiental.

Sin embargo, la implementación efectiva del balanceo de carga no está exenta de desafíos. Los algoritmos tradicionales utilizados para implementarla suelen basarse en enfoques estáticos que no responden adecuadamente a las condiciones cambiantes del entorno. En un data center, donde los recursos y las cargas de trabajo están en constante cambio, esta limitación puede provocar puntos de congestión que afectan el rendimiento de la red y, en última instancia, la experiencia del usuario.

Para superar estas limitaciones, se están explorando enfoques más dinámicos y adaptativos. Los algoritmos basados en machine learning, por ejemplo, pueden analizar patrones de comunicación para predecir volúmenes de tráfico futuros. Esto permite anticipar el tamaño de los flujos y ajustar la distribución de la carga para evitar cuellos de botella. Además, el aprendizaje profundo y el aprendizaje por refuerzo están permitiendo la gestión automática de las políticas de balanceo de carga, adaptándose a las necesidades cambiantes de la infraestructura.

Otra tendencia interesante es la adopción de enfoques inspirados en el Swarm intelligence. Algoritmos como el Artificial Bee Colony (ABC) y el Particle Swarm Optimization (PSO) demuestran que el comportamiento de las abejas y otras especies

sociales puede ser una fuente valiosa de inspiración para mejorar la distribución de carga en los data centers. Estos métodos suelen ser más flexibles y robustos que los enfoques tradicionales, ofreciendo soluciones novedosas a problemas complejos.

El balanceo de carga es fundamental para el cloud computing actual donde los enfoques adaptativos y las técnicas basadas en machine learning e Swarm intelligence están allanando el camino hacia sistemas más eficientes y sostenibles. Esto no solo mejora el rendimiento y reduce costes, sino que también contribuye a disminuir el consumo de energía y las emisiones de carbono asociadas con los grandes data centers.

2.6.4 Predicción de Carga de Trabajo en Cloud computing

2.6.4.1 *Optimización por categorización y predicción de carga*

La carga de trabajo o el volumen total de esfuerzo que realiza un servidor en un periodo fijo es un aspecto crítico en la administración de CDCs. Al clasificar y predecir la carga de trabajo, se puede reducir la sobreutilización de los servidores, evitar la reubicación innecesaria de máquinas virtuales y mejorar el consumo energético. Este proceso juega un papel crucial en el funcionamiento eficiente de data centers.

La planificación de recursos, gestión del rendimiento de aplicaciones, dimensionamiento de capacidad y proyección de necesidades futuras de recursos dependen de una adecuada clasificación y caracterización de la carga de trabajo. Esto permite una estimación precisa de la demanda futura de recursos y ayuda a cumplir con los requisitos de calidad del servicio (QoS), asegurando el uso eficaz de los recursos.

En el cloud, las cargas de trabajo se pueden clasificar según paradigmas computacionales, tecnología, recursos y aplicaciones. Dependiendo de la metodología de procesamiento, se dividen en dos categorías: cargas por lotes e interactivas. Según los recursos necesarios, se clasifican como de memoria, CPU, E/S y base de datos. Las clasificaciones por aplicación incluyen web, redes sociales y servicios de video.

El problema del balanceo de carga es como se dijo anteriormente, un NP-complete, lo que lleva al uso métodos heurísticos o estocásticos para abordarlos. Históricamente, la caracterización de la carga de trabajo se realizaba mediante enfoques

estadísticos como la media y la desviación estándar, coeficiente de correlación de Pearson, coeficiente de variación, y ratio pico a media [Derdus et al., 2019]. Estos métodos se utilizaban para una caracterización detallada de recursos solicitados y usados, como CPU, memoria, disco y recursos de red.

Las técnicas de aprendizaje supervisado se han vuelto indispensables en los data centers modernos, proporcionando un enfoque estructurado para categorizar y predecir cargas de trabajo. Este enfoque implica entrenar algoritmos con conjuntos de datos etiquetados, permitiéndoles tomar decisiones informadas basadas en el aprendizaje previo. Las técnicas comunes de aprendizaje supervisado incluyen Máquina de Vectores de Soporte (SVM) [Zhao et al., 2013], Descenso por Gradiente Estocástico (SGD) [Li et al., 2020], Regresión Logística (LR) [Mathematics, 2021], Random Forest (RF) [Cetinski & Juric, 2015], Perceptrón Multicapa (MLP) [Shekhawat et al., 2018] y Red Neuronal de Retropropagación [Sun et al., 2020]. Estos métodos se utilizan ampliamente para clasificar cargas de trabajo, ayudando a optimizar la asignación de recursos en el cloud.

Una vez que las cargas de trabajo se clasifican, se pueden hacer predicciones sobre las demandas futuras, permitiendo a los data centers anticipar las necesidades de recursos de manera más efectiva. Este pronóstico es crítico para mantener el rendimiento del sistema y evitar problemas como violaciones de SLA (Acuerdo de Nivel de Servicio), sobreasignación o sub-asignación. Las predicciones precisas pueden ayudar a los data centers a escalar recursos de manera dinámica y a gestionarlos proactivamente, mejorando así la escalabilidad y el rendimiento. Además, una gestión eficaz de las cargas de trabajo tiene implicaciones significativas para el consumo de energía y los costes operativos, ya que puede llevar a una reducción en la sobreasignación de recursos virtuales. Esto beneficia al medio ambiente al reducir el consumo de energía de los data centers y, al mismo tiempo, reduce los costes operativos, mejora la fiabilidad del sistema y aumenta la satisfacción del cliente.

Encontramos en la literatura diversas técnicas para predecir futuras demandas de carga de trabajo. Los esquemas basados en regresión, como los esquemas basados en ARIMA [Kumar & Mazumdar, 2016], y esquemas basados en Support Vector Regression (SVR) [Barati & Sharifian, 2015], aprovechan enfoques estadísticos para modelar tendencias y hacer predicciones. Estas técnicas son útiles para analizar datos pasados y predecir necesidades futuras de recursos basadas en patrones observados. Los esquemas basados en clasificadores emplean algoritmos de aprendizaje supervisado como esquemas

basados en SVM [Zhong, 2018], esquemas basados en Random Forest (RF) [Cetinski & Juric, 2015], y esquemas basados en Redes Neuronales Artificiales [Yang et al., 2015]. Estos métodos son particularmente efectivos para identificar patrones complejos y hacer predicciones precisas.

Además, los esquemas estocásticos para la predicción de cargas de trabajo, como los esquemas basados en Modelos Ocultos de Markov [Li et al., 2013] y esquemas basados en Modelos de Colas [Jiang et al., 2013], están diseñados para tener en cuenta la variabilidad aleatoria en las operaciones del data center. Estos modelos pueden predecir fluctuaciones en la carga de trabajo y proporcionar información sobre las tendencias de utilización de recursos. Otros enfoques, como esquemas basados en predicción gris [Jheng et al., 2014], esquemas basados en clustering de autocorrelación [Kluge et al., 2010], y esquemas basados en caos [Qazi et al., 2013], ofrecen perspectivas interesantes sobre el pronóstico, aprovechando conceptos matemáticos y estadísticos para predecir demandas de carga de trabajo.

Además, técnicas avanzadas como esquemas basados en aprendizaje profundo [Zhang et al., 2018] y esquemas basados en la teoría Bayesiana [Tian et al., 2016] están ganando popularidad por su capacidad para manejar datos complejos y proporcionar predicciones de alta precisión. Estos métodos, que incluyen Aprendizaje Profundo con Refuerzos y Redes Neuronales, ofrecen un enfoque más matizado para la predicción de cargas de trabajo. Al usar una combinación de técnicas de aprendizaje supervisado y aprendizaje profundo, los data centers pueden pronosticar mejor las demandas futuras y optimizar la asignación de recursos en consecuencia.

La combinación de estas técnicas proporciona otro enfoque para la predicción de cargas de trabajo. Al fusionar varias estrategias, como SVR con filtro de Kalman [Hu et al., 2013], ARIMA con RNN [Janardhanan & Barrett, 2017], o ARIMA con descomposición en wavelet [Fard & Akbari-Zadeh, 2014], los data centers pueden crear modelos de pronóstico robustos. Estas combinaciones pueden ayudar a los data centers a gestionar mejor los recursos, reducir el consumo de energía y mejorar el rendimiento general del sistema, lo que lleva finalmente a una infraestructura en el cloud más eficiente y sostenible.

La naturaleza dinámica y a menudo impredecible de las cargas de trabajo en los entornos de cloud computing plantea un desafío significativo para la eficiencia en el

consumo de energía y la optimización de la utilización de recursos. Las técnicas tradicionales de estimación del uso de energía, que suelen realizarse de manera offline, se quedan cortas frente a la complejidad de las aplicaciones modernas en tiempo real. Este panorama ha impulsado la exploración de nuevos enfoques, especialmente dentro del ámbito del machine learning, para predecir las cargas de trabajo y así mejorar la gestión de la energía en el cloud.

Uno de los retos más acuciantes en el cloud computing es la realización de predicciones precisas y en tiempo real del consumo energético ante cargas de trabajo no lineales. El estudio de [Reiss et al., 2012] revela que los nodos en los clústeres de Google funcionan típicamente al 60% y 50% de su capacidad de CPU y memoria, respectivamente, lo que indica una oportunidad significativa para la optimización. El "Ensemble Learning", como lo proponen [Subirats y Guitart, 2015], surge como una solución prometedora al combinar diferentes modelos predictivos para aumentar la precisión de las predicciones de eficiencia energética. Su enfoque integra técnicas como el Moving Average, suavizado exponencial, regresión lineal y suavizado exponencial doble para prever la eficiencia energética futura de los recursos de VMs. No obstante, la exclusión de métricas importantes como el caché de última línea (LLC) y el rendimiento del disco, que también inciden en el consumo energético, subraya la necesidad de abordajes más holísticos, como se señala en [Sayadnavard et al., 2021].

Otra área de interés es la identificación de perfiles de cargas de trabajo para una gestión de recursos más efectiva. La mayor parte de la investigación se ha centrado en el perfilado de cargas de trabajo de manera offline, un enfoque poco práctico para VMs en entornos de producción. [Bianchini et al., 2020] abordan esta limitación desarrollando un sistema de predicción basado en ML en el cloud de Microsoft Azure, que aprende de datos históricos y facilita predicciones en tiempo real para la gestión de recursos. Sin embargo, este enfoque no contempla la utilización de la memoria, un factor crucial en la planificación de recursos físicos para las VMs.

[Persico et al., 2017] introducen un modelo predictivo para gestionar el autoescalado de aplicaciones web utilizando lógica difusa, destacando la distinción entre enfoques reactivos y proactivos en la gestión de recursos. A diferencia de los métodos reactivos, que asignan recursos basándose en eventos del sistema que superan umbrales

predeterminados, los enfoques proactivos anticipan necesidades de recursos para evitar eventos no deseados. Sin embargo, los métodos estadísticos tradicionales enfrentan limitaciones en la precisión de estas predicciones. [Messias et al., 2016] superan este obstáculo mediante el uso combinado de análisis temporal sobre series temporales y algoritmos genéticos para optimizar las predicciones de los modelos de ML para el autoescalado, mejorando la adaptabilidad frente a variaciones en la carga de trabajo.

Un denominador común en estas investigaciones es el enfoque en la naturaleza distribuida de las cargas de trabajo en data centers, que frecuentemente se modelan como series temporales. Esto abre la puerta al uso de técnicas de análisis de series temporales para predecir el comportamiento futuro de las cargas, facilitando así la toma de decisiones informadas. El análisis de series temporales puede revelar patrones subyacentes, como tendencias, estacionalidad, ciclos y variaciones aleatorias. Investigaciones como [Cao et al., 2014] ejemplifican el uso de conjuntos de predictores que se actualizan dinámicamente para adaptarse a cambios en las tendencias de datos proporcionados por series temporales, demostrando la eficacia de combinar diversas técnicas estadísticas y probabilísticas, como MA, ARIMA, suavizado exponencial, proyección de tendencias y aproximaciones bayesianas, para extraer los patrones inherentes de los datos y mejorar la consolidación de VMs y la gestión general de recursos en el cloud.

2.6.5 Gestión Térmica con ML

La creciente densidad de recursos informáticos en los data centers en el cloud requiere nuevos enfoques para la gestión térmica más allá de los sistemas de enfriamiento tradicionales. Esta subsección profundiza en la extensión de estrategias de gestión térmica que complementan el enfoque de programación predictiva propuesto por [Ilager et al., 2021], centrándose en la reducción de las temperaturas máximas de los hosts y el consumo de energía a través de soluciones avanzadas de enfriamiento, asignación dinámica de recursos, integración con sistemas de edificios inteligentes, utilización de fuentes de energía renovables y técnicas de recuperación de calor.

Soluciones Avanzadas de Enfriamiento

Las tecnologías emergentes de enfriamiento, como el enfriamiento líquido, inmersión en líquido y la aplicación de materiales de cambio de fase, ofrecen métodos más eficientes para la disipación del calor en comparación con el enfriamiento por aire convencional. Estas técnicas permiten un enfriamiento directo de los componentes, reduciendo significativamente el riesgo de sobrecalentamiento y mejorando la eficiencia general del sistema de enfriamiento.

Asignación Dinámica de Recursos

Basándose en la base establecida por la programación predictiva, los algoritmos de asignación dinámica de recursos optimizan aún más la distribución de cargas de trabajo ajustándose en tiempo real tanto a las demandas computacionales como a las condiciones térmicas. Este enfoque ayuda a evitar la formación de puntos calientes dentro del data center y asegura una distribución uniforme de las cargas térmicas, mejorando así la fiabilidad y eficiencia del sistema.

Integración con Sistemas de Edificios Inteligentes

La integración de la gestión térmica de data centers con sistemas de edificios inteligentes presenta una oportunidad para la optimización holística del rendimiento energético y térmico. Al aprovechar las condiciones ambientales externas, como temperaturas ambientales más frescas, los data centers pueden reducir significativamente la dependencia de los sistemas de enfriamiento mecánicos, reduciendo así los costes operativos y el consumo de energía.

Fuentes de Energía Renovable

La adopción de fuentes de energía renovables, como la solar y eólica, para operar sistemas de enfriamiento puede reducir drásticamente el impacto ambiental y los costes operativos asociados con las operaciones de los data centers. Esta estrategia no solo apoya los objetivos de sostenibilidad, sino que también proporciona un medio renovable para alimentar la infraestructura de enfriamiento esencial.

Recuperación y Reutilización del Calor

El concepto de capturar y reutilizar el calor residual de los data centers representa un cambio de paradigma en la gestión térmica. Al utilizar el calor residual para otras

aplicaciones, como la calefacción de distritos, los data centers pueden transformar un subproducto de sus operaciones en un recurso valioso, contribuyendo a la eficiencia energética y la promoción de una economía circular.

Como destacó [Gao, 2014], incluso las mejoras menores en la gestión térmica pueden llevar a ahorros significativos en costes y eficiencias operativas en data centers a gran escala. Las técnicas avanzadas de gestión térmica discutidas aquí construyen sobre el enfoque de programación predictiva de [Ilager et al. 2021], ofreciendo una estrategia integral para abordar el complejo desafío de gestionar la disipación de calor en los data centers en el cloud modernos. Estas innovaciones no solo mejoran la eficiencia y fiabilidad de los data centers, sino que también se alinean con objetivos ambientales y de sostenibilidad más amplios, marcando un paso significativo hacia adelante en la evolución de la gestión de data centers.

2.7 Conclusión

La revisión del estado del arte sobre el cloud computing y la gestión de energía en los Cloud data centers ha revelado avances significativos y desafíos persistentes en el ámbito de la sostenibilidad y eficiencia energética. Las estrategias actuales, como la optimización de la infraestructura, la adopción de fuentes de energía renovable y el uso de inteligencia artificial, contribuyen a mejorar la eficiencia operativa de los CDCs. Sin embargo, estas estrategias presentan limitaciones que requieren enfoques más integrales y adaptativos para abordar la variabilidad y las demandas crecientes.

En particular, se destaca la importancia de la lógica difusa y los sistemas expertos en el cloud computing. La lógica difusa ha demostrado ser eficaz para manejar incertidumbres y mejorar la toma de decisiones en entornos complejos, permitiendo una gestión más flexible y adaptable de los recursos. Este enfoque facilita la migración de máquinas virtuales entre CDCs, optimizando la carga de trabajo y la eficiencia energética. Los Sistemas Expertos Basados en Reglas Difusas (FRBS) han enriquecido la capacidad de decisión en el cloud, contribuyendo al desarrollo de sistemas computacionales más inteligentes y adaptables.

Por otro lado, el machine learning ha desempeñado un papel crucial en la optimización de recursos y la mejora de la flexibilidad en el cloud computing. Las estrategias basadas en machine learning, como el Swarm intelligence y las redes neuronales, han permitido a los CDCs adaptarse dinámicamente a entornos cambiantes, reduciendo el consumo de energía y mejorando la calidad del servicio. No obstante, persisten brechas relacionadas con la escalabilidad y la complejidad algorítmica, lo que subraya la necesidad de enfoques más robustos y eficientes.

Esta tesis doctoral aborda estas brechas al proponer un marco innovador que combina lógica difusa con algoritmos de Swarm intelligence para optimizar la gestión de energía renovable en entornos cloud distribuidos geográficamente. Esta propuesta no solo contribuye a la eficiencia operativa de los CDCs, sino que también promueve la sostenibilidad ambiental al reducir el impacto ecológico mediante la migración de VMS basada en fuentes de energía renovable.

El enfoque de esta tesis promete ser relevante para el campo del cloud computing actual, abordando la necesidad de soluciones adaptativas para enfrentar la variabilidad de

la energía renovable y la gestión eficiente de los recursos en múltiples CDCs dispersos. La combinación de estos elementos crea un marco sólido para mejorar la eficiencia energética y contribuir al desarrollo de modelos de cloud computing más sostenibles.

A pesar de estas contribuciones, algunas limitaciones pueden surgir durante la implementación, como el coste asociado a tecnologías avanzadas y la escalabilidad de los algoritmos propuestos. Validaciones futuras en entornos de producción reales serán esenciales para confirmar la aplicabilidad práctica del enfoque de esta tesis.

En resumen, la revisión del estado del arte muestra que el campo del cloud computing enfrenta desafíos relacionados con la sostenibilidad y la eficiencia energética. Esta tesis doctoral aporta un enfoque innovador al combinar lógica difusa y Swarm intelligence para optimizar el uso de energía renovable en los CDCs. Con esto, se espera contribuir significativamente a la mejora de la eficiencia operativa y a la sostenibilidad en entornos cloud, creando una base para futuras investigaciones y desarrollos en el campo del cloud computing.

3 Contribuciones y aspectos metodológicos

3.1 Introducción

El presente capítulo se enfoca en la descripción de las contribuciones realizadas en el transcurso de esta investigación, así como la metodología utilizada para abordar el problema de la sostenibilidad energética en data centers distribuidos. A lo largo de esta tesis, se exploraron enfoques para optimizar la migración de máquinas virtuales entre data centers geográficamente dispersos, con el objetivo de reducir el consumo de energía y aumentar el uso de fuentes renovables.

Las principales contribuciones se centran en el desarrollo de un sistema experto y un algoritmo de optimización que permiten una migración más eficiente de máquinas virtuales entre CDCs distantes y tomando en cuenta la disponibilidad de energía renovable. Se diseñaron y probaron diferentes enfoques para mejorar la sostenibilidad energética en entornos cloud, con resultados prometedores.

La metodología adoptada en estos estudios incluye el uso simulador Cloudsim y su extensión realizada en el ámbito de una publicación en el “15th International Conference on Signal Processing and Communication Systems (ICSPCS)” en 2021 para modelar escenarios de migración entre múltiples CDCs distantes y evaluar así estrategias de sostenibilidad. Se aplicaron técnicas avanzadas como sistemas de expertos basados en reglas difusas y algoritmos de optimización global para abordar la complejidad inherente a estos entornos.

Esta sección proporciona un panorama general del enfoque adoptado, destacando las contribuciones clave que cubren los objetivos de esta tesis. El análisis detallado y los resultados obtenidos a partir de esta metodología se discutirán en las próximas secciones, donde se explorarán las implicaciones y el impacto de las contribuciones en mayor profundidad.

3.2 Contribución 1 - Sistema experto basado en reglas borrosas para la optimización de energía renovable en CDCs geográficamente distantes

La primera contribución de esta tesis proviene del artículo titulado "Sustainable Expert Virtual Machine Migration in Dynamic Clouds", publicado en la revista "Computers and Electrical Engineering". Esta revista está clasificada en el Journal Citation Reports (JCR) en el primer cuartil (Q1) en Ciencias de la Computación (miscelánea) y en Ingeniería Eléctrica y Electrónica.

El artículo introduce un sistema experto basado en reglas borrosas para optimizar la migración de máquinas virtuales entre data centers en el cloud dispersos geográficamente, teniendo en cuenta la disponibilidad de fuentes de energía renovable. El aporte clave radica en la capacidad de este sistema experto para gestionar la migración dinámica de VMs, mejorando la eficiencia energética y la sostenibilidad de los CDC.

El estudio incluye un análisis comparativo con algoritmos clásicos de migración, con resultados experimentales que evidencian la eficacia del sistema propuesto. Se destaca un uso significativamente mayor de energía renovable sin comprometer la calidad del servicio (QoS) en términos de tiempo de ejecución.

Esta investigación establece las bases para futuras exploraciones. El uso de reglas borrosas permite desarrollar sistemas expertos flexibles, que podrían integrar técnicas de machine learning y Swarm intelligence para mejorar la gestión de recursos en el cloud. El estudio también tiene implicaciones para los proveedores de servicios de TI, al ofrecer estrategias para reducir la huella de carbono y costes operativos mediante una gestión de recursos más consciente de la energía.

Es importante mencionar el trabajo anterior del 2021 y titulado "Sustainability-based Framework for Virtual Machines Migration Among Cloud data centers", publicado por el "Institute of Electrical and Electronics Engineers (IEEE)" en la conferencia "15th International Conference on Signal Processing and Communication Systems (ICSPCS)" en Australia. Este trabajo amplió y extendió el simulador CloudSim para permitir modelar y simular múltiples data centers por una parte y por otra que estos centros tuviesen un carácter renovable y conscientes de la energía, facilitando experimentos con diversas estrategias de migración entre data centers. En, dicho artículo, Se propuso un algoritmo consciente de la energía renovable para la migración de máquinas virtuales entre

diferentes data centers, comparando su desempeño con algoritmos clásicos para validar el modelo de simulación extendido.

En el contexto de esta tesis, esta contribución respalda el primer objetivo, que es diseñar sistemas inteligentes para migración de VM, permitiendo transferir cargas de trabajo entre data centers dispersos geográficamente en función de la disponibilidad de energía renovable. Esto busca no solo mejorar la eficiencia energética, sino también adaptarse a las variaciones en la disponibilidad de fuentes de energía renovable.

3.3 Contribución 2 - Sistema de adquisición del conocimiento para la mejora del comportamiento del sistema borroso

La segunda contribución de esta tesis se encuentra en el artículo titulado "Enhanced Virtual Machine Migration for Energy Sustainability Optimization in Cloud computing through Knowledge Acquisition", mandado a la misma revista "Computers and Electrical Engineering" que el anterior.

Este artículo presenta un enfoque prometedor basado en la adquisición de conocimiento para optimizar la migración de máquinas virtuales en data centers en el cloud, con el fin de maximizar el uso de energía renovable y, en consecuencia, mejorar la sostenibilidad energética de estos entornos. La contribución principal radica en el uso del algoritmo "Knowledge Acquisition version of the Gray Wolf Optimizer (KAGWO)", que utiliza un conjunto de parámetros para generar bases de reglas de conocimiento para un sistema experto con reglas borrosas, permitiendo así un mejor aprovechamiento de la energía renovable en los distintos CDCs, incluso cuando están geográficamente dispersos. Este enfoque ofrece una manera eficiente y flexible de mejorar la gestión energética en arquitecturas complejas, contribuyendo a una operación más sostenible de los data centers en el cloud.

El artículo incluye un análisis comparativo entre el algoritmo KAGWO y otros enfoques, como el Knowledge Acquisition with a Swarm intelligence Approach (KASIA) y el Genetic Algorithm (Pittsburgh). Los resultados experimentales muestran que KAGWO ofrece mejoras sustanciales en sostenibilidad energética, logrando incrementos de rendimiento del 0.53% al 5.23% en comparación con los algoritmos anteriores. Estos resultados son significativos porque demuestran que es posible mejorar la eficiencia energética en CDC sin comprometer la calidad del servicio (QoS).

Esta contribución respalda el segundo objetivo de tu tesis, que es aplicar técnicas de adquisición de conocimiento para mejorar la calidad de las bases de datos de expertos y optimizar el proceso de obtención de estas bases de datos para mejorar la gestión inteligente de la carga en CDC. El trabajo demuestra que es posible optimizar la sostenibilidad energética en el cloud mediante un enfoque basado en conocimiento, contribuyendo así a la creación de modelos de datos en el cloud sostenibles (CDM).

3.4 Aspectos metodológicos

3.4.1 Escenarios de simulación

En ámbito de esta tesis, se han delineado tres escenarios de simulación fundamentales, los cuales se detallan en la Tabla 3-1. La elección de estos escenarios fue realizada teniendo en mente el propósito de abarcar un espectro extenso de casos que reflejen la diversidad en la magnitud de los data centers, así como la variabilidad en las cargas de trabajo que estos gestionan. Es preciso señalar que, en consonancia con las posibilidades que brinda el entorno de simulación CloudSim —que permite una capacidad máxima de hasta 800 hosts con los datos PlanetLab del simulador—, se establecieron tres tipologías de escenarios correspondientes a data centers de pequeña, mediana y gran escala.

	Escenario 1	Escenario 2	Escenario 3
Hosts	265	530	800
VM	350	695	1052
Workload (Cloudlets)	500 - 1,500 - 3,000	1,000 - 2,000 - 5,000	1,500 - 5,000 - 10,000

Tabla 3-1 Escenarios fundamentales de simulación

Cada una de estas configuraciones ha sido ventilada sobre una estructura base de red que se conforma de 4 cloud data centers geográficamente distantes y donde cada una se alimenta de manera parcial o total con energía renovable según la distribución presentada en la figura Tabla 3-1. La arquitectura de estos data centers es la que se muestra a continuación.

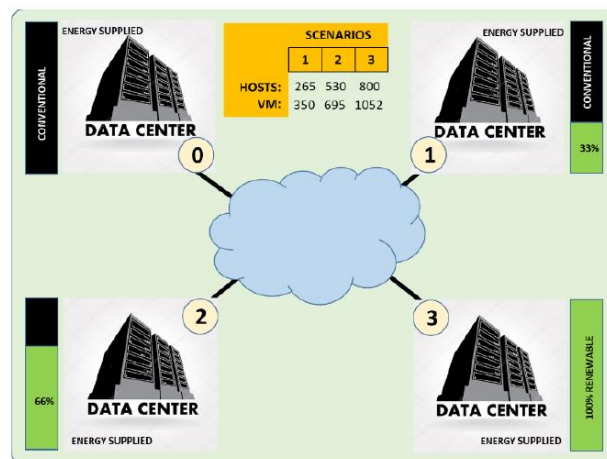


Figura 3-1 configuración de los RCDC

Estos escenarios han sido diseñados para representar una gama variada de tamaños de infraestructura, facilitando así la extracción de resultados concluyentes. De manera específica, en cada uno de los escenarios simulados, se procedió a la asignación aleatoria tanto de hosts como de máquinas virtuales sobre cuatro Cloud data centers. La configuración de consumo de energía de estos centros fue heterogénea; el primero de ellos no disponía de fuentes de energía renovable, mientras que los subsiguientes contaban con un 33%, un 66% y finalmente, un 100% de su energía proveniente de fuentes renovables, respectivamente. Esta distribución fue cambiando entre los diferentes centros sobre 4 horas de simulación. Esto es lo que se refleja en la Figura 3-2.

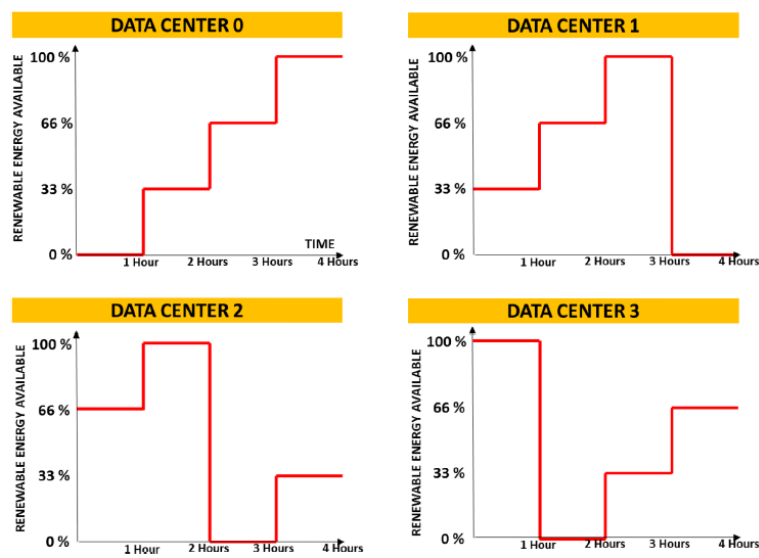


Figura 3-2 Distribución del cambio de disponibilidad de energía renovable en los 4 CDCs utilizados

Rotación Dinámica de Energía Renovable

Un aspecto distintivo y de significativa importancia en el marco de esta tesis es la incorporación de un modelo de cambio dinámico en la disponibilidad de la energía renovable. Dicha dinámica implica una rotación horaria del porcentaje de energía renovable accesible entre los cuatro CDC mencionados. Al concluir el periodo de simulación, que abarca un total de cuatro horas, cada CDC habrá experimentado tres alternancias en cuanto al porcentaje de energía renovable disponible.

Esta rotación dinámica está diseñada para reflejar la realidad operativa de los centros de datos que aprovechan múltiples fuentes de energía renovable, las cuales pueden variar significativamente en disponibilidad a lo largo del tiempo debido a factores como

las condiciones climáticas, la hora del día y la estacionalidad. Al simular estos cambios, se busca evaluar cómo la variabilidad en la disponibilidad de energía renovable afecta la eficiencia operativa y la sostenibilidad de los data centers.

En cada hora de simulación, el porcentaje de energía renovable disponible para cada CDC cambia de acuerdo con un patrón predefinido que asegura que, al final del periodo de simulación, todos los centros hayan experimentado cada nivel de disponibilidad de energía (0%, 33%, 66% y 100%). Este enfoque no solo permite probar la resiliencia y adaptabilidad de los sistemas de planificación inter-cloud, sino que también ayuda a identificar las estrategias más eficientes para maximizar el uso de energías renovables y minimizar el consumo de energía no renovable.

La Figura 3-2 ilustra cómo la disponibilidad de energía renovable se distribuye y rota entre los cuatro CDC a lo largo del tiempo. Esta metodología permite simular escenarios en los cuales los data centers deben adaptarse continuamente a cambios en la disponibilidad de energía, evaluando la efectividad de los algoritmos de planificación para mantener el equilibrio entre eficiencia operativa y sostenibilidad ambiental.

Además, este modelo de rotación dinámica de energía renovable se alinea con la realidad de la generación de energía renovable, que puede ser intermitente y dependiente de diversos factores externos. Al incorporar estas variaciones en la simulación, se proporciona una evaluación más realista y robusta del rendimiento de los CDC bajo condiciones operativas fluctuantes.

La rotación dinámica de energía renovable en los escenarios de simulación es una característica clave de esta tesis, que permite evaluar de manera integral cómo los sistemas de planificación inter-cloud pueden optimizar el uso de recursos renovables, adaptarse a cambios en la disponibilidad de energía y mejorar la sostenibilidad de los centros de datos. Este enfoque asegura que las soluciones propuestas no solo sean teóricamente viables, sino también prácticas y efectivas en entornos operativos reales.

3.4.2 Comparación de Algoritmos

En el contexto de la presente investigación, cuyo objetivo es el de establecer un marco comparativo para evaluar el impacto de políticas de migración de máquinas virtuales en Cloud data centers distribuidos geográficamente y alimentados por distintos porcentajes de energía renovable (independientemente de su fuente), se han seleccionado y analizado una serie de algoritmos [Beloglazov et al., 2012] provistos por el simulador CloudSim en su versión 4.0. Los algoritmos examinados son los siguientes:

Inter Quartile Range (IQR): Utilizado predominantemente en el campo del machine learning para la identificación de outliers, el IQR se aplica en este contexto para la evaluación de la mediana en conjuntos de datos ordenados, enfocándose especialmente en la utilización de recursos en servidores. Este enfoque divide el conjunto de datos en dos medianas, Q1 y Q2, donde Q1 corta el conjunto de datos, y Q2 divide las mitades izquierda y derecha. Este proceso permite una estimación precisa de la dispersión de los datos, facilitando la identificación de servidores sobreutilizados (outliers) mediante el cálculo de un umbral superior. Esto le permite a la política identificar aquellos servidores sobreutilizados para proponer sus máquinas virtuales a migración a otros con menor carga.

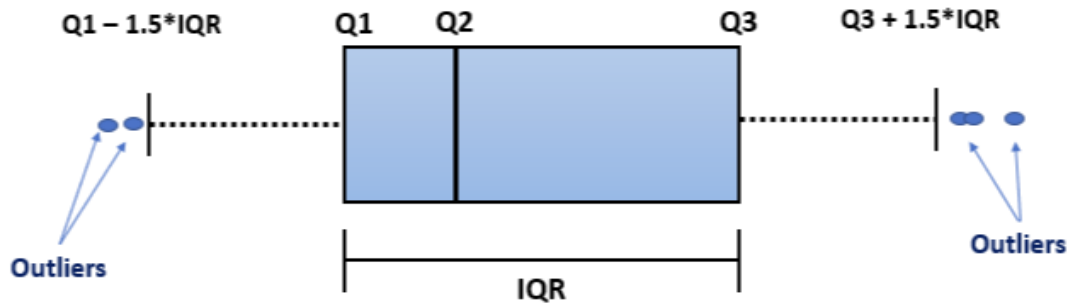


Figura 3-3 Algoritmo del Inter Quartil Range (IQR)

Local Regression (LR): También en el ámbito del Machine Learning, el LR se centra en ajustar la mejor línea de regresión posible a un conjunto de datos, minimizando las distancias cuadráticas entre los puntos y la línea de ajuste. Su aplicación principal en esta investigación es la de predecir patrones de comportamiento en datos no incluidos en el conjunto original. Dentro del marco de políticas de migración de máquinas virtuales, LR es esencial para anticipar situaciones de sobrecarga en servidores, adaptándose a las fluctuaciones en los datos de utilización de recursos.

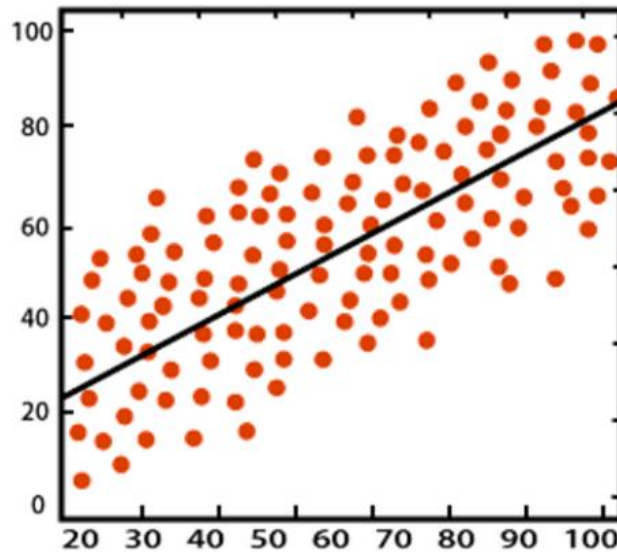


Figura 3-4 Algoritmo del Local Regression (LR)

Local Regression Robust (LRR): Como una extensión del LR, el LRR se muestra más eficaz en situaciones donde los datos presentan una considerable cantidad de “outliers” o que no sigan una distribución lineal. Este método ajusta una curva al conjunto de datos, considerando distintos subconjuntos y empleando un parámetro de tamaño de ventana 'n' para asignar pesos a las muestras. LRR utiliza tanto líneas como parábolas en sus curvas de regresión, ofreciendo una mayor flexibilidad y precisión en la predicción de la sobreutilización de servidores, en comparación con el modelo estándar de LR.

Static Threshold: A diferencia de los métodos anteriores, basados en técnicas estadísticas y de machine learning, esta política de asignación de máquinas virtuales implementa un umbral fijo de utilización de CPU (THR) para detectar la sobrecarga en los servidores. Este umbral, preestablecido y aplicado de manera uniforme, permite identificar servidores que superan un límite predefinido de utilización de recursos.

El algoritmo "**Renewable Optimizer Based on Dynamic Inputs (RODI)**" representa la primera aportación de esta tesis para desarrollar soluciones avanzadas destinadas a la optimización del uso de energía renovable en Cloud data centers. Este se enfoca en la implementación de políticas inteligentes de migración de máquinas virtuales. RODI se concentra en evaluar las mejoras que hemos integrado en CloudSim (como parte de este proyecto de tesis), permitiendo que CloudSim simule migraciones de VMs no solo

dentro de un mismo CDC, sino también extendiéndolas a otros centros geográficamente distantes. La metodología de RODI actúa como un meta-planificador, dirigiendo las migraciones de VMs con base en la disponibilidad de energía renovable en los CDCs simulados. Este enfoque utiliza una estrategia basada en colas y se aplica en contextos donde varios CDCs funcionan con distintos niveles de energía renovable, accesible de manera ininterrumpida.

La operación del algoritmo RODI comienza cuando el meta-planificador, tras recibir una carga de trabajo, indaga sobre la disponibilidad de energía renovable en cada uno de los CDCs. Dada la naturaleza dinámica de esta disponibilidad, es imprescindible realizar actualizaciones periódicas. Con esta información, el meta-planificador organiza los CDCs en función de su nivel de energía renovable, generando una lista priorizada que orienta la asignación de las cargas de trabajo a los CDCs, a sus hosts y finalmente a las VMs.

El meta-planificador procede a evaluar cada CDC de manera secuencial, empezando por aquel con la mayor disponibilidad de energía renovable y continuando con aquellos que tienen menor disponibilidad. Cada CDC examina su capacidad para albergar la carga de trabajo mediante la simulación de la creación de VMs y asegurando que sus hosts mantengan un nivel de uso óptimo, evitando así entrar en un estado de "Sobrecarga".

Si un CDC cuenta con la capacidad necesaria, informa al meta-planificador, procesa la carga de trabajo y luego notifica al meta-planificador al concluir, para que este, a su vez, informe al origen. En cambio, si un CDC no puede albergar la carga de trabajo, informa al meta-planificador sobre el fracaso en la creación de la VM. En tal caso, el meta-planificador continúa el proceso con el siguiente CDC en la lista, priorizado por disponibilidad de energía renovable. Este ciclo se repite hasta encontrar un CDC capaz de gestionar la carga de trabajo. Si no se localiza un CDC adecuado, lo que se manifestaría por fallos reiterados en la creación de VMs, la simulación se detendrá.

La figura siguiente, incluida en la documentación complementaria, presenta de forma detallada este algoritmo a través de un diagrama de secuencia, el cual muestra las interacciones entre los diversos CDCs y sus intermediarios.

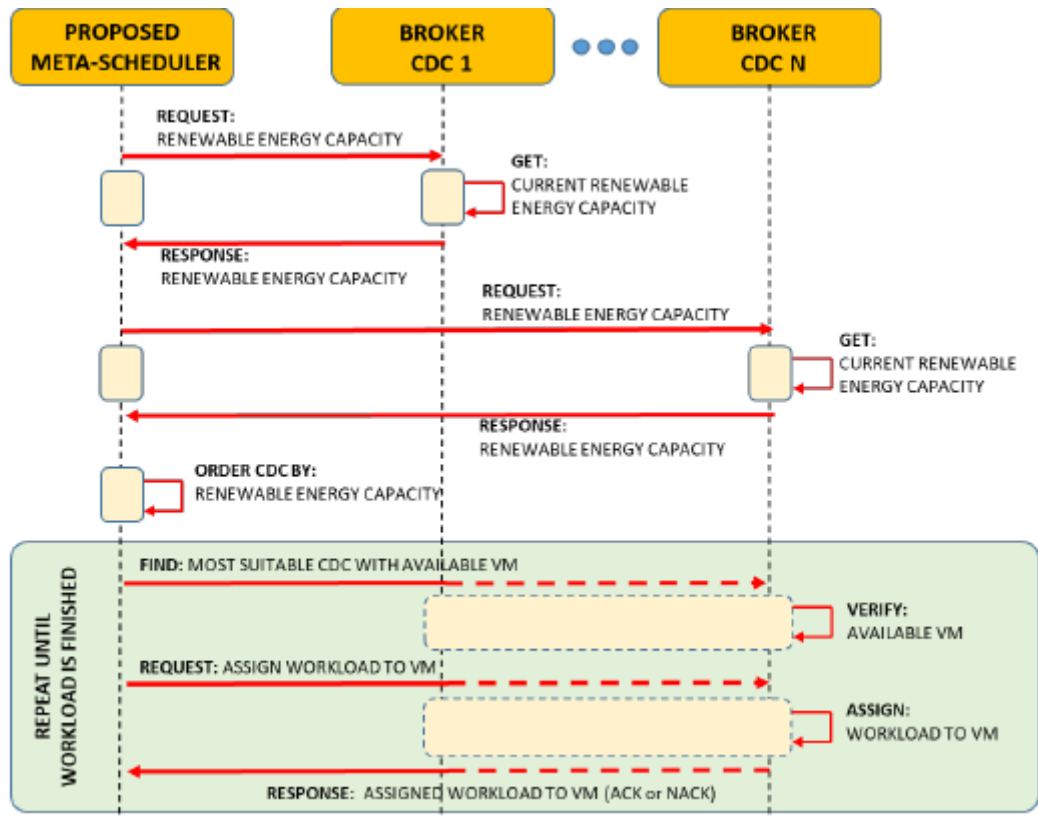


Figura 3-5 Diagrama de secuencia del algoritmo RODI

3.4.3 Evaluación del Rendimiento de Algoritmos mediante Técnicas Paramétricas y No Paramétricas

Con el fin de evaluar el rendimiento de los diferentes algoritmos en la optimización del uso de energía renovable en los Cloud data centers, especialmente para validar el segundo objetivo de esta tesis, se realizaron 30 simulaciones para cada uno de los nueve escenarios propuestos. Esto generó un conjunto de datos con porcentajes de uso de energía renovable que podían ser analizados para determinar si existían diferencias significativas entre los algoritmos.

Para lograr esto, se utiliza el Análisis de Varianza (ANOVA), una técnica estadística diseñada para identificar variaciones significativas entre grupos. ANOVA es ideal para experimentos que involucran múltiples grupos, ya que compara la variación entre grupos con la variación dentro de los mismos. Un resultado significativo de ANOVA sugiere que la variación entre los grupos es mayor que dentro de ellos, indicando que al menos un grupo tiene un resultado diferente. En nuestro contexto, la prueba ANOVA se

usó para evaluar si había diferencias significativas en el uso de energía renovable en los nueve escenarios que involucraban los tres algoritmos.

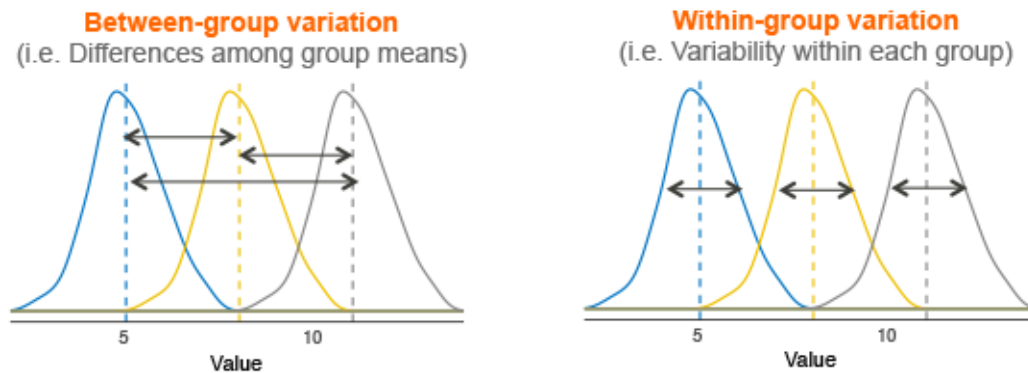


Figura 3-6 análisis de varianza ANOVA

Para que los resultados de ANOVA sean válidos, es esencial que los datos cumplan con ciertos supuestos:

1. Independencia de las Observaciones; Donde las observaciones deben ser independientes entre sí.
2. Normalidad; Donde los datos dentro de cada grupo deben seguir una distribución normal. Esto puede verificarse utilizando la prueba de Shapiro-Wilk.
3. Homogeneidad de Varianzas; Donde las varianzas entre los diferentes grupos deben ser homogéneas. Esto se verifica mediante la prueba de Levene.

Sin embargo, antes de realizar la prueba ANOVA, hay que llevar a cabo varias pruebas preliminares donde verificamos que los datos cumplan con los supuestos anteriores:

- Prueba de Shapiro-Wilk: Utilizada para verificar la normalidad de los datos. Un p-valor mayor que 0.05 indica que los datos siguen una distribución normal.
- Prueba de Levene: Utilizada para comprobar la homogeneidad de las varianzas entre los grupos. Un p-valor mayor que 0.05 indica que las varianzas son homogéneas.

Si los datos no cumplen con estos supuestos, se deben considerar alternativas no paramétricas como:

- Prueba de Wilcoxon: Utilizada para comparar dos grupos cuando los datos no son normales. Se puede aplicar como Wilcoxon rank-sum (independiente) o Wilcoxon signed-rank (pareado).
- Prueba de Kruskal-Wallis: Alternativa no paramétrica a ANOVA para comparar más de dos grupos, utilizada cuando los datos no son normales o las varianzas no son homogéneas.

Para el procedimiento ANOVA, la recopilación y organización de datos fueron cruciales para asegurar su validez y confiabilidad. La estructura de los datos consistía en la variable dependiente, representada por el porcentaje de energía renovable utilizada, y la variable independiente, que representaba las nueve combinaciones de algoritmos y escenarios. Cada combinación tuvo 30 observaciones de las simulaciones, creando un conjunto de datos sólido para el análisis.

Usando Matlab, realizamos la prueba ANOVA para obtener el estadístico F y el valor p, ambos esenciales para interpretar los resultados. El estadístico F mide la proporción de la variación entre grupos con respecto a la variación dentro de los mismos, mientras que el valor p indica la probabilidad de observar los resultados si la hipótesis nula (ninguna diferencia significativa entre grupos) fuera verdadera. Un valor p por debajo de 0.05 sugiere una diferencia significativa entre los grupos, lo que nos lleva a profundizar más en los resultados.

3.4.4 Simulación con CloudSim

3.4.4.1 Extensión del simulador CloudSim

En su versión 4.0 y a día de hoy, CloudSim presenta una limitación significativa en su capacidad para gestionar migraciones de Máquinas Virtuales, estando esta reducida a operaciones dentro de un único Cloud Data Center. Esta restricción es particularmente impactante en entornos donde un CDC utiliza una combinación de fuentes de energía tanto renovables como convencionales. En tales escenarios, los protocolos existentes de optimización de migraciones de VM en CloudSim se aplican uniformemente a todos los tipos de energía. Esta aplicación uniforme no logra diferenciar ni priorizar las fuentes de energía renovables, por lo tanto, no aprovecha completamente los beneficios ambientales

potenciales y las ganancias en eficiencia que podrían lograrse mediante el uso específico de energía renovable en estrategias de migración de VM.

En respuesta a esta limitación, y en el transcurso de la investigación de esta tesis, se implementaron modificaciones sustanciales en CloudSim [Seddiki et al., 2021]. Estas modificaciones fueron fundamentales para extender la funcionalidad de CloudSim y permitirle en definitiva adquirir la capacidad de simular migraciones de máquinas virtuales que esta vez ya son "inter" varios Cloud data centers. Esta capacidad mejorada permite una gestión más dinámica y flexible de las VMs a través de múltiples data centers, fomentando una distribución y utilización más eficientes de los recursos computacionales en las redes de computación en el cloud.

Además, la extensión de la funcionalidad de CloudSim se ejemplifica aún más mediante la introducción del algoritmo RODI presentado en la sección anterior 3.4.2 y que es un algoritmo centrado en la sostenibilidad. Este algoritmo está diseñado de manera que permite gestionar migraciones de VMs entre diferentes CDCs, con un énfasis particular en optimizar el consumo de energía y minimizar el impacto ambiental. El diseño del algoritmo considera métricas de sostenibilidad como la utilización de la energía renovable al mismo tiempo que intenta guardad dentro de niveles adecuados (la potencial degradación) de los parámetros de calidad de servicio QoS, debido a las necesidades de la migración, convirtiéndolo de esta manera en una herramienta robusta para la gestión consciente del medio ambiente en el ecosistema del cloud computing.

Este desarrollo representa un avance significativo hacia la integración de consideraciones de sostenibilidad en el ámbito de del cloud computing. Al permitir que CloudSim admita migraciones de VM entre CDCs e introducir un algoritmo de migración enfocado en la sostenibilidad, esta investigación contribuye al paisaje en evolución de la computación ecológica. Subraya la importancia de desarrollar tecnologías y metodologías de cloud computing que no solo sean tecnológicamente avanzadas, sino también ambientalmente responsables y sostenibles.

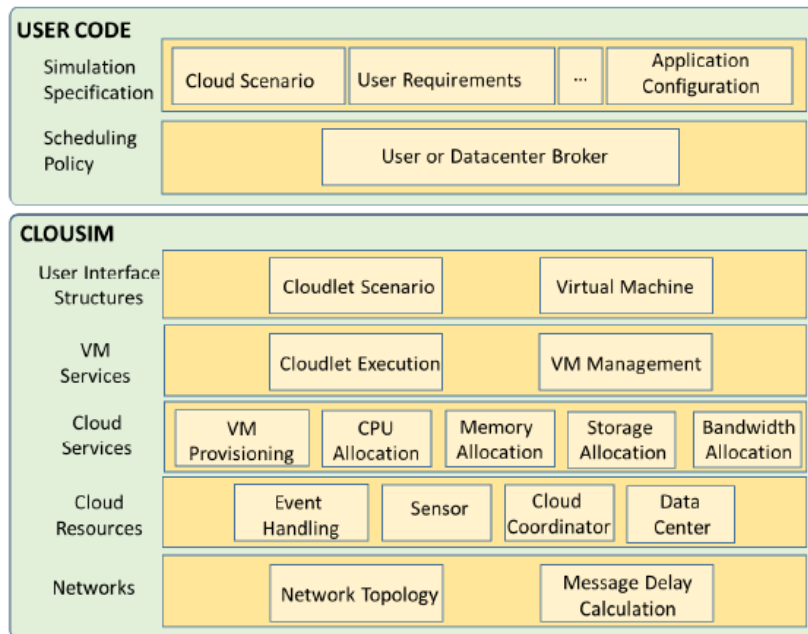


Figura 3-7 Entidades básicas que conforman el "Core" del motor de simulación de CloudSim

Con el fin de dotar al simulador CloudSim con la funcionalidad previamente descrita, se han incorporado nuevas entidades, que se han dividido en dos grandes categorías fundamentales:

- Entidades Conceptuales
- Y Entidades de Simulación

A continuación se presentan las más representativas.

a) Entidades Conceptuales (Figura 3-8):

Estas entidades son vitales para establecer marcos lógicos que reflejen la estructura y la dinámica de los nuevos componentes específicos que permiten la simulación de migraciones de máquinas virtuales "Entre" Cloud data centers:

- **Renewable Cloud Data Center (RCDC):** Este es un nuevo tipo de CDC que utiliza un sistema de suministro de energía híbrido y en tiempo real. Este sistema está compuesto tanto por energías renovables como tradicionales. Este nuevo tipo de CDC Renovable incorpora el uso de una nueva categoría de entidades de "hosts", conocidas como las "Renewable Power Hosts".

- **Renewable Power Host:** Esta entidad, que amplía las capacidades de la entidad existente en el simulador (PowerHostUtilizationHistory) que dota a los hosts de las características de consumir energía, pero también de la capacidad de guardar dicho consumo en el tiempo para poder calcular las diferentes métricas de funcionamiento del hardware. Esta nueva entidad es esencial para abordar la gestión de energía renovable que no era posible anteriormente y que la dota de esta nueva característica para el consumo de diferentes tipos de fuentes de energía.
- **Meta-Scheduler:** Esta entidad es la encargada de implementar las políticas de migración de las máquinas virtuales entre los Cloud data centers. Su principal tarea es la de enrutar el tráfico hacia diferentes destinos dependiendo de la política de migración utilizada. Esta nueva entidad extiende la entidad PowerDataCenter de CloudSim, con lo que nos permite utilizar los algoritmos citados en 3.4.2. Esto es fundamental para el análisis comparativo frente a los algoritmos creados en durante el periodo de investigación de esta tesis, contribuyendo así a la evaluación y validación de las capacidades mejoradas del simulador.

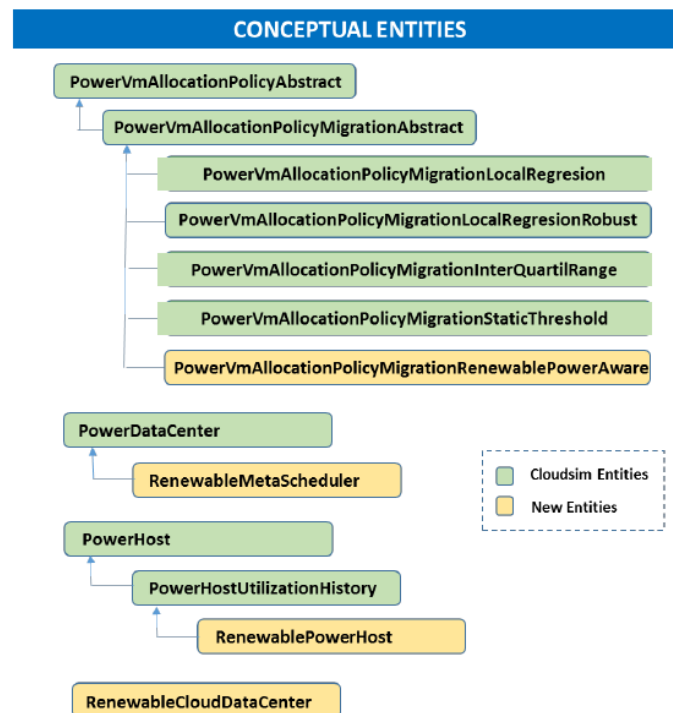


Figura 3-8 Extensión de las entidades conceptuales del CloudSim para integrar la migración "Inter" Cloud data centers

b) Entidades de Simulación:

Estas entidades son cruciales para las operaciones de "RUNTIME" del simulador y Aunque no formen parte del framework central del CloudSim (Core), facilitan la implementación de funcionalidades clave de simulación. Estas incluyen la creación de nuevos tipos de infraestructuras renovables dentro de CloudSim, la generación y gestión de las cargas de trabajo, la aplicación de diferentes estrategias de migración en el meta-Scheduler y el cálculo de los resultados estadísticos. La integración de estas entidades permite una mejora significativa en la eficiencia del consumo energético en los CDCs de energía híbrida. En la Figura 3-9 se resumen las entidades más representativas.

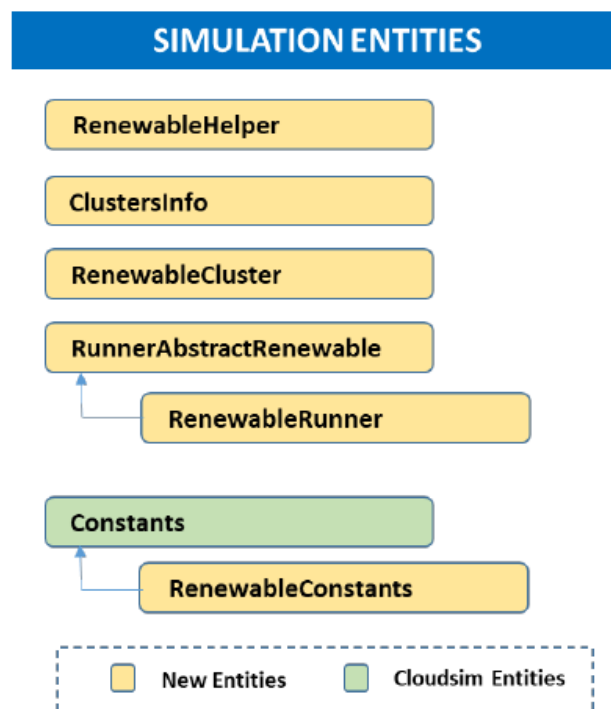


Figura 3-9 Extensión de las entidades de simulación del CloudSim para integrar la migración "Inter" Cloud data centers

3.4.5 Emulación de Hardware real para las simulaciones

CloudSim es un marco de simulación diseñado para infraestructuras y modelado de computación en el cloud, que ofrece capacidades para simulación "Power-Aware" y optimización de recursos. Este marco facilita el uso de diversas configuraciones de servidores en simulaciones, permitiendo así la selección de numerosos modelos de

computadoras del mundo real. En el contexto de esta investigación, se han empleado las siguientes configuraciones de servidores:

- HP ProLiant ML110 G4, equipado con un procesador de doble núcleo Xeon 3040 operando a 1.8 GHz, 4 GB de RAM y ofreciendo 1Gbps de ancho de banda.
- HP ProLiant ML110 G5, con un procesador de doble núcleo Xeon 3075 a 2.6 GHz, 4 GB de RAM y proporcionando 1Gbps de ancho de banda.

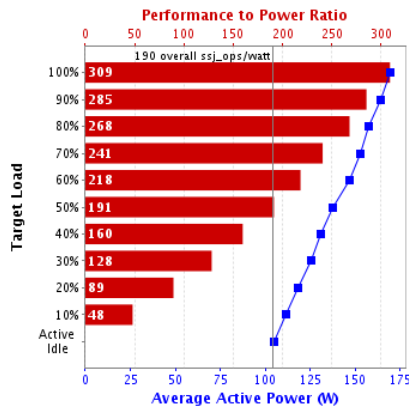


Figura 3-11 Potencia Vs Carga para el HP ProLiant ML110 G4

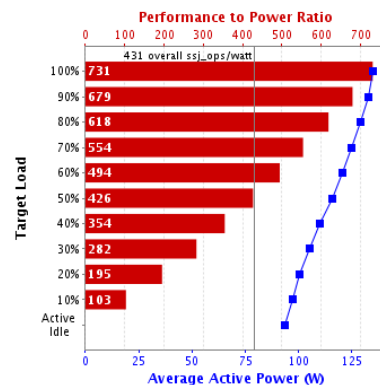


Figura 3-10 Potencia Vs Carga para el HP ProLiant ML110 G5

Las métricas de consumo de energía para estas máquinas son las reales, y fueron obtenidas de la Standard Performance Evaluation Corporation (SPEC) [<https://www.spec.org>], una entidad sin fines de lucro enfocada en crear, mantener y respaldar puntos de referencia estandarizados y herramientas para evaluar el rendimiento y la eficiencia energética de los sistemas informáticos. SPEC proporciona datos de consumo de energía correspondientes a diversas utilidades de la CPU, que varían del 0% al 100% en incrementos del 10%. Para utilidades de la CPU que no se alinean precisamente con estos incrementos, se aplica un enfoque de interpolación lineal para estimar el consumo de energía real.

Además, para mejorar el realismo de las simulaciones, CloudSim ofrece trazas reales de utilización de la CPU a través de 288 entradas, capturando el uso real de la CPU cada cinco minutos, acumulando un conjunto de datos comprensivo de 24 horas para la simulación.

En términos de caracterización de la carga de trabajo, este estudio utiliza cloudlets configurados de la siguiente manera:

- Longitud del Cloudlet: 2500 x tiempo de simulación (MI)
- Tamaño del Archivo del Cloudlet (incluyendo programa y datos): 300 bytes
- Tamaño de Salida del Cloudlet: 300 bytes
- Número de Elementos de Procesamiento: 1

Adicionalmente, y para asegurar la heterogeneidad de las máquinas virtuales empleadas en este estudio, cabe destacar que se han utilizado cuatro tipos diferentes con las siguientes posibles opciones por máquina virtual:

- Número de Elementos de Procesamiento: 1
- MIPS (Millones de Instrucciones Por Segundo): Opciones de 2500 o 2000 o 1000 o 500
- RAM: Configuraciones de 870 MB, 1740 MB y 613 MB
- Ancho de Banda: 100 Mbps
- Capacidad de Almacenamiento: 2.5 GB

Las configuraciones anteriores componen la columna vertebral de las composiciones de arquitecturas de red utilizadas para validar los diferentes algoritmos de migración de máquinas virtuales para demostrar su capacidad en la optimización del uso de energía renovable en los Cloud data centers distantes geográficamente.

3.4.6 Desarrollo de un sistema experto basado en FRBS para implementar la Políticas de Migración

En el ámbito de esta tesis, se introduce un nuevo enfoque centrado en el uso de un Sistema Basado en Reglas Difusas (FRBS) como el algoritmo central para las operaciones de asignación y migración realizadas por un meta-Scheduler para la optimización de energía renovable en los RCDC. El diseño de este componente basado en reglas difusas está conceptualmente influenciado por la configuración modular estándar característica de los Sistemas de Lógica Difusa (FLSs), como se cita en la referencia [Cordón O, 2001]. Esta estructura fundamentalmente abarca cuatro componentes integrales comúnmente asociados con FRBSs: la interfaz de difuminación, base de conocimiento, motor de

inferencia y la interfaz de desfuminación. Estos componentes han sido exhaustivamente explorados y documentados en el contexto de implementaciones de sistemas difusos, como se señala en la referencia [Franke C, 2007].

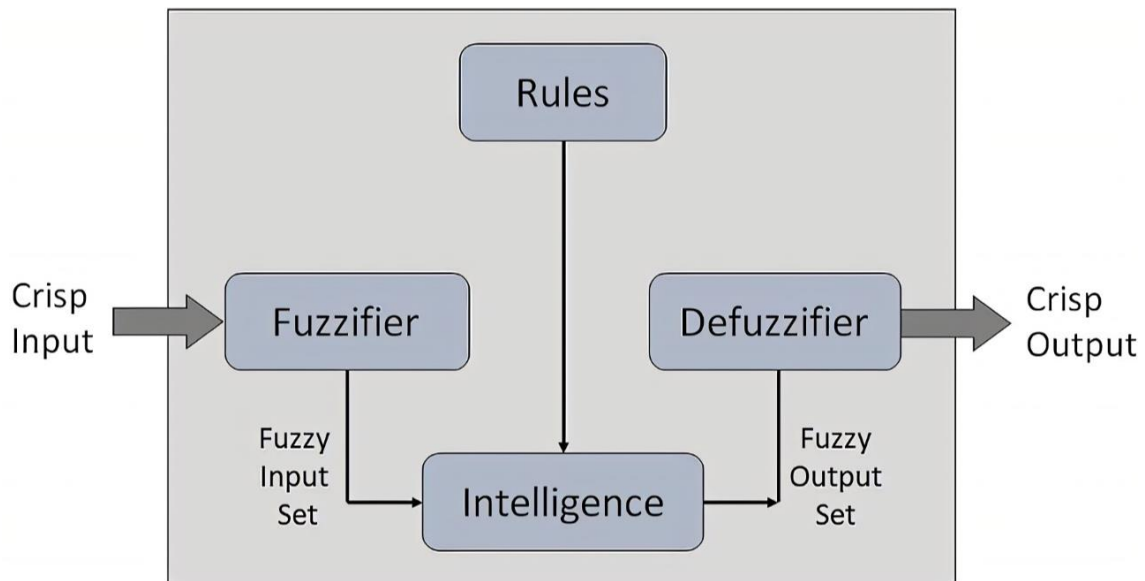


Figura 3-12 Esquema simplificado de un FRBS

Según los principios fundamentales definidos por [Mamdani, 1974], cada regla difusa, denotada como R_i , se compone de dos segmentos distintos: el antecedente y el consecuente. Estos segmentos corresponden a conceptos difusos, con el primero detallando las condiciones para la activación de la regla y el segundo ilustrando la salida o respuesta del planificador. Por lo tanto, una regla dentro de este marco propuesto se representa como:

$$R_i: \text{IF } w_1 \text{ is } A_{1n} \frac{\text{and}}{\text{or}} \dots w_m \text{ is } A_{mn} \text{ THEN } y \text{ is } B_n \quad (1)$$

En esta formulación, w_m simboliza una característica del sistema, mientras que 'y' representa la variable de salida, y A_{mn} y B_n son los conjuntos difusos vinculados con la respectiva característica m y con la salida 'y'. La bondad de una Base de Reglas (RB) para el meta-Scheduler basado en FRBS depende de su capacidad para gestionar óptimamente una amplia variedad de condiciones de incertidumbre al que es aplicado, y que son influenciadas por entradas dinámicas en el modelo de simulación. En esta tesis,

las entradas del sistema difuso experto se han diseñado para lograr tanto un buen rendimiento en el tiempo de ejecución como en el uso de energía renovable en los CDCs. Estas variables de entrada se describen en la tabla siguiente:

Variable de entrada o Feature	Descripción
Cloud Data Center Renewable Availability (CDC-RA)	Porcentaje de energía renovable entregada al CDC
Host Computational Capacity (HCC)	Capacidad computacional máxima expresada en MIPS
Host Computational Availability (HCA)	Capacidad computacional restante después de alojar las máquinas virtuales expresada en MIPS
VM Maximum Computational Needs (VM-MCN)	Necesidades máximas de computación de una máquina virtual expresadas en MIPS
VM Current Computational Needs (VM-CCN)	Capacidades computacionales restantes para las máquinas virtuales expresadas en MIPS

Tabla 3-2 descripción de las variables de entrada del sistema experto basado en FRBS

En este estudio se ha optado por el uso de funciones de membresía gaussianas en lugar de las triangulares, que, aunque son más sencillas y eficientes de utilizar, las gaussianas nos permiten una caracterización más completa del universo de discurso de las variables de entrada (Feature). Cada variable de entrada se modela utilizando una combinación de tres funciones de membresía gaussianas (baja, media y alta), cuya expresión matemática de definición en la siguiente:

$$g_i^w(x) = \frac{1}{\sigma_i(w) \sqrt{2\pi}} \exp \left\{ -\frac{(x - \mu_i^w)^2}{(2\sigma_i(w)^2)} \right\} \{x \in R + |x| \leq 1\} \quad (2)$$

Donde μ_i^w representa la media y $\sigma_i(w)$ la desviación, definen cada conjunto difuso para cada característica w y regla R_i . En consecuencia, una regla R_i se puede representar como una combinación de descripciones de sus características antecedentes y consecuentes de la siguiente manera:

$$R_i = \{g^{(1)}(x), g^{(2)}(x), \dots, g^{(N_F)}(x), \Omega_i\} \quad (3)$$

Donde N_F simboliza el número de características consideradas, y Ω_i representa la parte consecuente de la regla. Para los fines de este estudio, no se consideran diferentes pesos para las reglas, otorgando así la misma influencia a cada regla dentro de la Base de

reglas (RB). Además, la idoneidad de un host para una VM que tiene que migrar se determina como la salida del FRBS, categorizada en cinco conjuntos difusos que van desde "No Muy adecuado" hasta "extremadamente adecuado". Estos se representan con los valores normalizados en la figura siguiente.

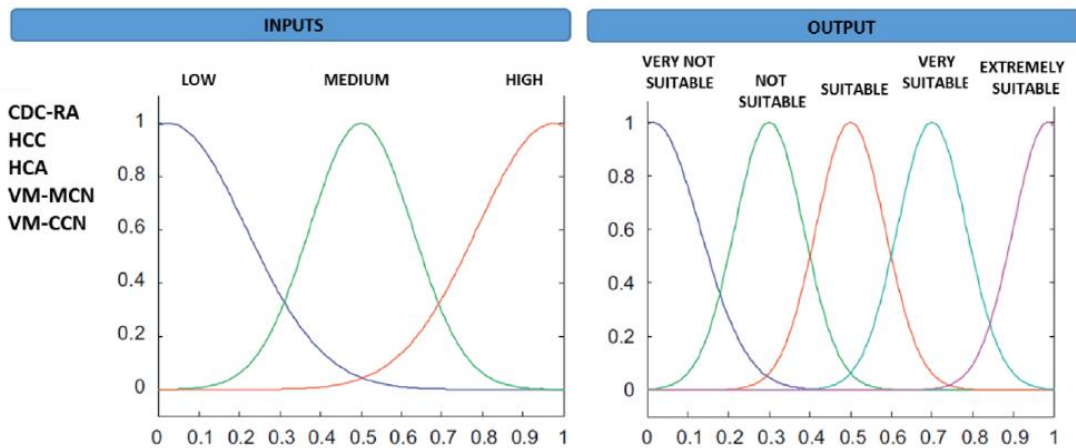


Tabla 3-3 Funciones de membresía para las entradas y las salidas

Es importante destacar que el “conocimiento” utilizado en el ámbito de esta tesis se compone de un conjunto de reglas difusas. Estas reglas establecen una relación entre las variables de entrada y salida del sistema mediante una representación lingüística, lo cual facilita la generación de una respuesta conclusiva para el sistema. Las reglas de un Sistema de Razonamiento Basado en Reglas Difusas (FRBS) representan un elemento central en el conocimiento del sistema experto. La eficacia de este sistema está intrínsecamente vinculada a la calidad de dichas reglas.

RULE	CDC-RA	HCC	HCA	VM-MCN	VM-CCN	AND/OR	OUTPUT
1		High		Low		AND	Extremela Suitable
2		Low		High		AND	Not Very Suitable
3			High		Low	AND	Extremely Suitable
4			Low		High	AND	Not Very Suitable
5	Low						Not Very Suitable
6	Medium						Suitable
7	High						Extremely Suitable

Tabla 3-4 Base de reglas (RB) del sistema experto

La configuración de la base de reglas del sistema FRBS utilizado en esta tesis se basa en la experiencia y conocimiento acumulado por el grupo de investigación TIC-188. La combinación de conocimientos expertos, validación empírica, adaptación continua y evaluación comparativa ha permitido desarrollar un sistema robusto y eficiente para la planificación inter-cloud, estas se detallan en la Tabla 3-4.

Más específicamente, algunas de estas reglas (R1, R2, R3 y R4) se enfocan en aspectos computacionales, mientras que las restantes (R5, R6 y R7) se orientan hacia la gestión de energía renovable. Se optó finalmente por un conjunto limitado de reglas simples, con el objetivo de facilitar su comprensión. Además, se evitó la inclusión de más de dos variables de entrada por regla, para preservar la interpretabilidad y claridad del conjunto de reglas.

También es importante mencionar que se empleó el framework jFuzzyLogic como marco de trabajo principal para el desarrollo de nuestro Sistema Basado en Reglas Difusas (FRBS).

3.4.7 Integración de Swarm intelligence

El uso intensivo de servicios basado en el cloud computing ha supuesto un aumento significativo en el consumo de energía a nivel global, presentando retos notables en términos de eficiencia energética y sostenibilidad. En este contexto, la gestión óptima de recursos (como la tratada en el ámbito de esta tesis) de asignación y migración de Máquinas Virtuales, emerge como una necesidad crítica para mejorar la eficiencia energética de los CDCs, particularmente cuando se trata de optimizar el uso de energías renovables. El Swarm intelligence (Swarm intelligence) representa una aproximación prometedora para abordar dichos desafíos, dada su capacidad para resolver problemas de optimización complejos mediante comportamientos colectivos inspirados en la naturaleza.

Así, los algoritmos de Optimización de Enjambre de Partículas han demostrado ser herramientas eficaces para la optimización en diversos dominios. En este trabajo, nos enfocamos en la utilización de dos variantes avanzadas de PSO: el algoritmo de Optimización por Enjambre de “Knowledge Acquisition with a Swarm intelligence

Approach” (KASIA) [R. Prado, 2010] y el algoritmo de “Knowledge Acquisition with the Gray Wolf Optimizer (KAGWO)” [Seddiki et al., 2024] que deriva del GWO [Mirjalili et al., 2014] que destaca por su equilibrio entre exploración y explotación del espacio de búsqueda, lo cual es crucial para la optimización en entornos complejos como los CDCs geodistribuidos. Su estructura jerárquica, que imita la dinámica social y estrategias de caza de los lobos grises, permite una eficiente adquisición y explotación del conocimiento, lo cual es esencial para el FRBS. Además, la simplicidad en la implementación y la robustez del GWO facilitan su adaptación a diversas aplicaciones de optimización.

Las fortalezas del GWO, combinadas con su capacidad para adaptarse a diferentes escenarios de optimización, lo convierten en una herramienta adecuada para mejorar la programación de tareas en CDCs, aumentando así la eficiencia en el uso de energías renovables. Estos aspectos hacen de GWO una elección sólida para servir de base para el desarrollo de un algoritmo de adquisición del conocimiento para un sistema experto borroso.

Uno de los objetivos de esta tesis es la implementación de bases de conocimiento para un Sistema Basado en Reglas Difusas (Fuzzy Rule-Based System, FRBS) que puedan mejorar significativamente la toma de decisiones en lo que respecta a la asignación y migración de VMs entre CDCs geográficamente dispersos. La implementación de estas bases de conocimiento busca no solo la optimización en la asignación de recursos, sino también una mayor integración de fuentes de energía renovable, contribuyendo así a la sostenibilidad de las infraestructuras de TI.

Posteriormente, la integración de KASIA y del KAGWO con FRBS representa una novedosa metodología que permite a los CDCs, mediante la mejora de las bases de conocimiento anteriores, no solo responder a las cargas de trabajo de manera eficiente, sino también adaptarse dinámicamente a las fluctuaciones en la disponibilidad de energías renovables.

3.4.7.1 *Algoritmos de Swarm intelligence*

3.4.7.1.1 Knowledge Acquisition with a Swarm intelligence Approach (KASIA)

Dentro del contexto de KASIA, las bases de reglas difusas (RB) se conceptualizan como entidades autónomas de conocimiento, las cuales están sujetas a procesos de evaluación y de evolución mediante la aplicación de Swarm intelligence. De manera específica, se ha seleccionado el algoritmo de Optimización por Enjambre de Partículas (PSO) para fundamentar la adquisición de dichas RBs difusas. Este sistema se fundamenta en el comportamiento social de individuos interactuantes dentro de un colectivo o enjambre.

El procedimiento establecido para este enfoque implica la creación de un enjambre de RBs, el cual está constituido por múltiples partículas. Cada partícula se encuentra representada por una matriz denominada P, en la que las filas corresponden a reglas difusas individuales integradas dentro de la partícula RB correspondiente. Se adopta la codificación de Mamdani para cada regla, abarcando tanto los antecedentes y los consecuentes, como los conectivos lógicos implicados.

La fase de inicialización de dichas partículas abarca la asignación de una velocidad inicial, simbolizada como V, a cada una de ellas. Esta velocidad es determinante en la generación de modificaciones en las reglas a medida que se desarrollan las iteraciones del algoritmo. Posterior a esta fase, se procede a identificar las posiciones óptimas para cada partícula de manera individual, así como para el enjambre en su totalidad.

Es importante subrayar que el proceso de evaluación dentro del marco de KASIA se alinea con la evaluación de una RB, teniendo en cuenta las limitaciones y restricciones impuestas a las variables del algoritmo, tal como se detalla en [R. Prado, 2010]. La estructura representativa para la partícula i-esima es la que se expone a continuación.

$$P_i = \begin{bmatrix} a_{1,1}^i & a_{1,2}^i & \cdots & a_{1,n}^i & b_{1,1}^i & c_{1,1}^i \\ \vdots & \vdots & & \vdots & \vdots & \vdots \\ a_{m,1}^i & a_{m,2}^i & \cdots & a_{m,n}^i & b_{m,1}^i & c_{m,1}^i \end{bmatrix} \quad (3)$$

En la estructura de la Base de Conocimiento (KB), el término m hace referencia a la cantidad total de reglas estipuladas, en tanto que el término n se refiere al número de variables de entrada que se incorporan en el sistema. Los caracteres a y b son empleados para simbolizar los antecedentes y los consecuentes respectivamente, siendo este último considerado en singular dentro de dicho contexto. De manera complementaria, el carácter c es utilizado para la representación del conector lógico AND/OR en el marco conceptual establecido.

Es pertinente señalar que las funciones de membresía asignadas a cada variable están ordenadas de manera secuencial, iniciando su numeración en 1. Un valor asignado de 0 indica la no participación de la variable en la regla específica, mientras que la asignación de valores negativos se utiliza para indicar el conjunto complementario de la misma. Es preciso mantener la coherencia de las reglas difusas, lo cual no se sostiene si todos los antecedentes se valoran en 0. En cuanto a los consecuentes, estos pueden recibir igualmente un valor de 0 para señalar la no aplicabilidad de la regla. En lo concerniente a los conectores, un valor asignado de 1 corresponde a la conjunción lógica AND, y un valor de 2 se asocia con la disyunción lógica OR. Durante el proceso de aprendizaje, las funciones de membresía se mantienen invariables, lo cual garantiza la interpretabilidad de las reglas.

El propósito final es la consecución de una matriz representativa de reglas difusas de elevada calidad para un Sistema Basado en Reglas Difusas (FRBS). Con el objetivo de asegurar la calidad y la eficacia de las reglas, se aplican restricciones específicas a cada uno de los elementos constituyentes de la matriz posicional, abarcando tanto antecedentes como consecuentes y conectores. Además, resulta imperativo considerar el procedimiento de actualización posicional del enjambre. Para ello, se introduce la matriz de velocidad, que, dentro de este enfoque metodológico, se define para cada iteración como una matriz que refleja las dimensiones de la partícula para cada una de estas, identificada como V_i .

$$V_i = \begin{bmatrix} v_{1,1}^i & a_{1,2}^i & \cdots & v_{1,n}^i & v_{1,n+1}^i & v_{1,n+2}^i \\ \vdots & \vdots & & \vdots & \vdots & \vdots \\ v_{m,1}^i & v_{m,2}^i & \cdots & v_{m,n}^i & v_{m,n+1}^i & c_{m,n+2}^i \end{bmatrix} \quad (4)$$

Tras la definición y evaluación de las variables pertinentes, se procede a la actualización de la velocidad y la ubicación de cada partícula. Esta actualización se fundamenta en las experiencias acumuladas de índole individual, así como en aquellas derivadas de la dinámica social del enjambre. La actualización de la velocidad toma en consideración factores tales como la inercia propia de la partícula, representada por el símbolo ω , así como la mejor base de reglas (RB) obtenida por la partícula, denotada como P^B , y la mejor posición encontrada por el colectivo del enjambre, señalada como G^B . La actualización de los valores de la velocidad obedece a la siguiente ecuación:

$$V(t + 1) = \omega \otimes V(t) \oplus c_1 r_1 \otimes (P^B(t) - P(t)) \oplus c_2 r_2 \otimes (G^B(t) - P(t)) \quad (5)$$

En el contexto de KASIA, se establece que los coeficientes ponderados c_1 y c_2 , los cuales simbolizan, respectivamente, factores sociales de naturaleza competitiva y cooperativa, se mantienen invariables durante la ejecución del proceso computacional. Paralelamente, r_1 y r_2 corresponden a valores numéricos producidos de manera estocástica, cuyo rango se encuentra comprendido dentro del intervalo cerrado $[0,1]$.

La actualización de la posición de cada partícula se modela por la siguiente ecuación de la manera siguiente:

$$P(t + 1) = P(t) \oplus V(t + 1) \quad (6)$$

El algoritmo KASIA, expuesto en la descripción del Algoritmo 1 abajo, representa una estrategia de optimización metaheurística que emplea una colección de partículas en enjambre para la resolución de problemas. El proceso se inicia con la asignación aleatoria de coordenadas y vectores de velocidad a las partículas constituyentes del enjambre. Mediante una serie de etapas iterativas, se procede a la actualización de las coordenadas

y velocidades de las partículas. Simultáneamente, se lleva a cabo la evaluación de la viabilidad de cada partícula, utilizando para ello una función objetivo que ha sido establecida de antemano, la cual puede ser el tiempo total de procesamiento o el porcentaje de uso de energía renovable, entre otros criterios.

Las posiciones más favorables identificadas por las partículas durante el proceso se registran como el óptimo global y el óptimo individual. Una vez que se alcanza un número determinado de ciclos iterativos, el algoritmo señala la posición óptima global como resultado, la cual se asume como la solución más eficiente hallada por el enjambre.

```
Algorithm 1 KASIA Procedure.
1: Swarm initialization: Nparticles, Nrules, Niter, inertial weight  $\omega$ , c1 and c2 factors.
2: Random setting of RB-Swarm position P.
3: Random setting of velocity V.
4: Apply P and V constraints.
5: Initialize Gbest and Pbest.
6: while Niter do
7:   while Nparticles do
8:     Update P.
9:     Apply constraints to P.
10:    Evaluate particle.
11:    Particles++.
12:   end while
13:   Update Gbest.
14:   while Nparticles do
15:     Update Pbest.
16:     Update V.
17:     Apply constraints to V.
18:     Particles++.
19:   end while
20:   iter++.
21: end while
22: Return: Gbest.
```

Figura 3-13 Pseudo-Código KASIA

De manera general, se puede decir que KASIA armoniza la dinámica de **exploración** del dominio de búsqueda, efectuada a través de las actualizaciones de las posiciones de las partículas, con la dinámica de **explotación** de áreas que se presentan prometedoras. Este balance está dirigido a la consecución de una solución que sea óptima o que se aproxime de manera considerable a la óptima.

3.4.7.2 *Algoritmo Genético Pittsburg*

El enfoque genético de Pittsburgh, tal como lo describe [Smith, 1980], integra principios evolutivos, lógica borrosa y un marco centrado en la población para abordar problemas complejos como el Scheduling de tareas en el cloud computing. En esta estrategia, el sistema genético trata a cada base de reglas (RB) como un cromosoma, donde cada cromosoma representa una RB completa. Se aplican operadores genéticos para evolucionar poblaciones de RBs a lo largo de varias iteraciones. Al final del proceso de optimización, se identifica la RB más óptima para ser incorporada en el sistema borroso. En cada generación, todas las RBs son evaluadas dentro del dominio operativo del sistema borroso. Esta evaluación es facilitada por un sistema de valoración que mide la eficacia de cada RB en función de su adecuación al entorno optimizado. Luego, las RBs se clasifican según su rendimiento, y las de mayor rango se someten a cruces para generar nuevas RBs. Además, las RBs de mejor rendimiento pueden sufrir mutaciones. Mientras tanto, las RBs menos efectivas en la generación son reemplazadas por las recién evolucionadas, con la tasa de reemplazo determinando la magnitud de la sustitución. A través de este proceso iterativo, las RBs que demuestran mejores criterios de adecuación a lo largo de las generaciones se refinan progresivamente, lo que finalmente lleva a la selección de la RB más óptima para servir como base del sistema borroso en la generación final.

Algorithm 2 Pittsburgh Algorithm.

```

1: Inizialization: N Population, N Rules, N Iter, Crossover Rate, Mutation Rate Init,
   Selection Rate, Replacement Rate.
2: Random setting of RB Population P.
3: Initialize  $G^B$ .
4: while N Iter do
5: Update Mutation Prob with Mutation Prob = Mutation Rate Init *  $\exp(\frac{-5t}{N_{iter}})$ 
6: while N Particles do
7: Evaluate fitness (makespan).
8:     Particles++.
9:     end while
10:    Update  $G^B$ .
11: while N Particles * Replacement Rate do
12: Generate Q offspring:
13:     Apply crossover to P.
14:     Apply mutation to Q.
15:     Apply constraints to Q.
16:     end while
17: Update part of P with Q.
18: iter++.
19: end while
20: Return:  $G^B$ 

```

Figura 3-14 Pseudo-Código Pittsburg

La configuración utilizada para este algoritmo ha sido la siguiente:

Simulation s	Particles	Iterations	Crossover Rate	Initial mutation rate	Selection Rate	Replacement Rate
30	64	50	0,8	0,1	0,8	0,8

Tabla 3-5 Configuración Pittsburg

3.4.7.3 Knowledge Acquisition with the Gray Wolf Optimizer (KAGWO)

El algoritmo KAGWO [Seddiki et al., 2024], es una versión avanzada del Optimizador de Lobos Grises (Gray Wolf Optimizer, GWO) que se centra en la adquisición de conocimientos para la optimización del consumo de energía en CDC. Este algoritmo se diseñó específicamente para mejorar la sostenibilidad energética en estos centros al optimizar el uso de energía renovable.

KAGWO funciona recopilando datos sobre la disponibilidad y el uso de energía renovable en los data centers. Utiliza el enfoque de Swarm intelligence, donde las "partículas" o "lobos" en el algoritmo representan bases de reglas difusas que se evalúan y evolucionan para mejorar la base de conocimientos del sistema de reglas difusas (FRBS). Estas mejoras en las bases de conocimientos tienen como objetivo mejorar la toma de decisiones relacionadas con la asignación y migración de máquinas virtuales en los data centers, lo que a su vez conduce a un uso más eficiente de la energía renovable.

3.4.7.3.1 Inspiración del KAGWO

El Algoritmo de Optimización Gray Wolf Optimizer (GWO) se fundamenta en dos pilares esenciales del comportamiento de las manadas de lobos: su jerarquía social y sus estrategias de caza. La jerarquía social de los lobos se distingue por una marcada estructura de dominancia. En esta estructura, los lobos Alfa se sitúan en la cúspide jerárquica, asumiendo el rol de liderazgo en la toma de decisiones que afectan al conjunto del grupo. Los lobos Beta, posicionados en un segundo nivel jerárquico, cumplen funciones de soporte en el proceso decisorio, colaborando estrechamente con los Alfa. En un tercer nivel se encuentran los lobos Delta, y en el escalafón más bajo, los Omega, quienes se someten a las directrices de los rangos superiores. Esta organización jerárquica facilita una coordinación y cooperación eficaces entre los lobos, aspectos cruciales para el éxito de sus actividades de caza y para la supervivencia del grupo como entidad unificada.

Desde el punto de vista de la estrategia de caza, es habitual que la manada de lobos adopte una formación triangular al rodear a su presa, tal como se ilustra en la Figura 3-15. El lobo Alfa, en calidad de miembro dominante, ocupa la posición delantera, mientras que los lobos Beta y Delta se posicionan en los flancos. Por su parte, los lobos Omega asumen roles de soporte. Durante el acercamiento a la presa, los lobos coordinan sus movimientos para conformar un perímetro cerrado, obstruyendo las vías de escape de la presa. Mediante una colaboración y comunicación sincronizadas, dirigen a la presa hacia el centro del triángulo, complicando sus posibilidades de huida. Una vez que la presa está efectivamente cercada, la manada se lanza a la captura, liderada por el lobo Alfa o por otros miembros de rango dominante, mientras que el resto provee apoyo táctico. Esta

metodología demuestra la extraordinaria capacidad de adaptación, inteligencia y cooperación de los lobos, destacando su eficacia como depredadores altamente cualificados.

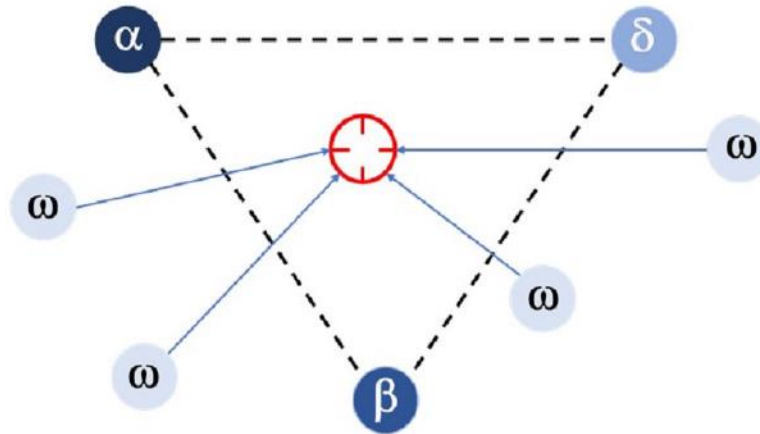


Figura 3-15 Mecanismo de caza de una manada de lobos grises

Este algoritmo incorpora una analogía metafórica en su estructura, conceptualizando el espacio de búsqueda como un extenso paisaje con características topográficas variadas, donde cada pico y valle simboliza posibles soluciones. Dentro de este contexto, el algoritmo efectúa un análisis exhaustivo de múltiples escenarios, inspirándose en la conducta de los lobos en la caza de presas esquivas, con el fin de identificar la solución óptima. La estructura matemática del algoritmo refleja la jerarquía social de una manada de lobos, asignando rangos a las posiciones de los mismos. El lobo Alfa (α) representa la posición de mayor jerarquía, simbolizando la solución más eficaz, seguido por el lobo Beta (β) y el Delta (δ) en el segundo y tercer lugar, respectivamente. Los demás lobos, denominados Omegas (ω), simbolizan el estrato inferior de la jerarquía, correspondiendo a las soluciones menos favorables. Esta representación jerárquica facilita una interpretación estructurada y detallada de los roles asignados a cada miembro de la manada, proporcionando una base para el desarrollo de algoritmos avanzados que replican y se valen de estas dinámicas sociales para optimizar procesos de toma de decisiones.

Se presupone que el algoritmo de optimización se fundamenta principalmente en las tres soluciones preeminentes: alfa, beta y delta, las cuales conforman una configuración triangular en torno a la presa. A lo largo del proceso de búsqueda, en caso

de que surja una nueva solución superior, se procede a una actualización inmediata de las posiciones previamente asignadas a los alfas, betas, deltas y omegas. Este mecanismo de actualización dinámica garantiza la adaptabilidad y capacidad de respuesta del algoritmo frente a variaciones en las condiciones de búsqueda, perfeccionando de manera continua la metodología para alcanzar resultados de optimización altamente eficientes y efectivos.

En el modelado matemático de los mecanismos de caza implementados en el algoritmo, el principal objetivo radica en lograr un cerco estratégico efectivo de la presa. Para ello, se toman en consideración dos ecuaciones fundamentales:

$$\vec{D} = |\vec{C} \cdot \vec{X}_p(t) - \vec{X}(t)| \quad (7)$$

$$\vec{X}(t + 1) = \vec{X}_p(t) - \vec{A} \cdot \vec{D} \quad (8)$$

La primera ecuación (7) calcula el vector distancia \vec{D} , el cual simboliza la separación espacial entre el lobo y su presa. Este vector se deriva de la multiplicación del vector de coeficientes \vec{C} por el vector $\vec{X}_p(t)$, que representa la ubicación de la presa en el momento t, sustraído de la posición simultánea del lobo, indicada por el vector $\vec{X}(t)$. Subsecuentemente, se procede a la actualización de la posición del lobo para la iteración siguiente, expresada como $\vec{X}(t + 1)$. Dicha actualización resulta de la resta del producto del vector de coeficientes \vec{A} y el vector de distancia \vec{D} , obtenido previamente.

Este proceso de actualización posibilita que el lobo modifique su posición de manera dinámica, permitiendo un acercamiento progresivo a la presa, al tiempo que se emplea una estrategia de cerco. Ambas ecuaciones colaboran para formular un enfoque tanto dinámico como estratégico, facilitando al lobo la implementación de tácticas de rodeo durante la cacería. Mediante ajustes continuos en su posición, basados en el vector de distancia, el lobo es capaz de optimizar sus desplazamientos, incrementando así la probabilidad de captura exitosa de la presa. La metodología para obtener los vectores de coeficientes se describe de la siguiente manera:

$$\vec{A} = 2\vec{a} \cdot \vec{r}_1 - \vec{a} \quad (9)$$

$$\vec{C} = 2 \cdot \vec{r}_2 \quad (10)$$

$$a = (2 - t) * \left(\frac{2}{\text{maxIterations}} \right) \quad (11)$$

El vector \vec{a} , experimenta una disminución lineal de 2 a 0 en todas las iteraciones. Tal disminución tiene como propósito moderar la magnitud de los movimientos del algoritmo en proporción al número de iteraciones, facilitando así la **convergencia** del proceso de optimización. Esta reducción progresiva es influenciada significativamente por los vectores aleatorios \vec{r}_1 y \vec{r}_2 , cuya función es asistir al algoritmo en su fase de convergencia, particularmente cuando el valor de \vec{a} se aproxima a cero. La incorporación de este coeficiente decreciente permite al algoritmo adaptar de manera estratégica sus movimientos y moderar los efectos de la aleatoriedad, conduciendo efectivamente a una convergencia hacia soluciones óptimas.

Los vectores \vec{r}_1 y \vec{r}_2 , elementos clave dentro del algoritmo, son vectores aleatorios con distribución uniforme en el intervalo $[0,1]$. Estos parámetros confieren al algoritmo GWO una naturaleza estocástica, introduciendo un elemento de incertidumbre y variabilidad en su comportamiento durante el proceso de búsqueda. Esta característica probabilística y aleatoria del algoritmo facilita la **exploración** de diversas áreas del espacio de búsqueda y **previene** la **limitación** a una **única trayectoria**. La estocasticidad inherente en el algoritmo realza su capacidad adaptativa, permitiéndole identificar soluciones prometedoras en contextos de optimización complejos. Cabe destacar la importancia de la utilización de vectores en las ecuaciones del algoritmo, lo que incrementa la abstracción y generalidad de este, y permite una adaptación efectiva a problemáticas en varias dimensiones. Este enfoque basado en vectores proporciona al algoritmo la flexibilidad y aplicabilidad necesarias para abordar una extensa gama de desafíos en el campo de la optimización.

En cuanto a la metodología operativa del algoritmo GWO, se destaca el uso de una manada de lobos para ejecutar el proceso de caza, en búsqueda de la solución óptima. Dentro de esta manada, las posiciones alfa, beta y delta representan las tres soluciones más eficientes. Estas soluciones son cruciales para la localización de la presa y la actualización de las posiciones de los lobos restantes (omegas), basándose en sus posiciones estimadas. El algoritmo presupone que la solución óptima se encuentra en proximidad a las soluciones alfa, beta y delta, lo que justifica la asignación primordial de recursos computacionales a estas, dada su mayor potencial. Este enfoque refleja el comportamiento estocástico del algoritmo GWO y conlleva la utilización de tres ecuaciones de distancia específicas para las soluciones alfa, beta y delta, adaptando así el proceso de optimización a estas configuraciones más prometedoras.

$$\overrightarrow{D}_\alpha = |\vec{C}_1 \cdot \overrightarrow{X}_\alpha - \vec{X}| \quad (12)$$

$$\overrightarrow{D}_\beta = |\vec{C}_2 \cdot \overrightarrow{X}_\beta - \vec{X}| \quad (13)$$

$$\overrightarrow{D}_\delta = |\vec{C}_3 \cdot \overrightarrow{X}_\delta - \vec{X}| \quad (14)$$

Mediante la utilización de los valores inherentes a estos vectores de distancia, el algoritmo procede a calcular las posiciones que debería adoptar. Esta determinación se realiza bajo la consideración de tres contextos distintos, que incluyen: Seguimiento exclusivo del alfa, Seguimiento exclusivo del beta, y Seguimiento exclusivo del delta. Para cada uno de estos escenarios, se emplean ecuaciones específicas, a través de las cuales el algoritmo determina tres posiciones concretas, cada una correspondiente a las condiciones particulares del escenario en cuestión.

$$\overrightarrow{X}_1 = \overrightarrow{X}_\alpha - \vec{a}_1 \cdot (\overrightarrow{D}_\alpha) \quad (15)$$

$$\overrightarrow{X}_2 = \overrightarrow{X}_\beta - \vec{a}_2 \cdot (\overrightarrow{D}_\beta) \quad (16)$$

$$\overrightarrow{X}_3 = \overrightarrow{X}_\delta - \vec{a}_3 \cdot (\overrightarrow{D}_\delta) \quad (17)$$

Y para considerálas todas ellas al mismo tiempo, el algoritmo propone una media de las tres:

$$\vec{X}(t+1) = \frac{\overrightarrow{X}_1 + \overrightarrow{X}_2 + \overrightarrow{X}_3}{3} \quad (18)$$

El proceso de premediación en el algoritmo garantiza una ponderación adecuada de la influencia de cada posición individual. Este enfoque integrador y equilibrado se beneficia del conocimiento conjunto de las soluciones alfa, beta y delta, proporcionando una perspectiva holística en la resolución de problemas. Para mantener un equilibrio entre las fases de exploración y explotación y eludir las soluciones óptimas locales, se incorporó al algoritmo el vector \vec{A} , cuyo rango fluctúa entre $[-2a, 2a]$. Este rango se determina por un vector aleatorio \vec{r}_1 y muestra una disminución lineal en relación con el vector \vec{a} , que decrece de $[2,0]$ a lo largo del tiempo. Cuando el vector \vec{A} se encuentra en

el intervalo $-1 < \vec{A} < 1$, se interpreta que la posición del lobo está 'dentro' de la distancia \vec{D}_i , indicando un estado de explotación y convergencia en el proceso de búsqueda. En contraste, valores de $\vec{A} > 1$ o $\vec{A} < -1$ sugieren que la posición del lobo está 'fuera' de la distancia \vec{D}_i , lo que señala un estado de exploración y motiva al algoritmo a distanciarse de la presa para prevenir soluciones óptimas locales.

Otro factor de relevancia es el vector de coeficientes \vec{C} , que incluye valores aleatorios en el rango $[0, 2]$. Este parámetro es crucial para intensificar la naturaleza estocástica del algoritmo, especialmente considerando que el vector \vec{A} podría quedar atrapado en soluciones óptimas locales con el tiempo. Mediante la introducción de variaciones aleatorias e independientes a través del vector \vec{C} , el algoritmo logra un balance efectivo en la influencia de la presa (la solución actual) al definir el vector de distancia. Con valores de \vec{C} superiores a 1, los lobos tienden a aproximarse a la presa, mientras que valores inferiores a 1 generan un efecto de alejamiento. Es crucial destacar que los ajustes en \vec{C} son independientes de la iteración actual, aportando así un mecanismo estocástico adicional para enfatizar la fase de exploración. Esta característica no se limita a las iteraciones iniciales, como en el caso del vector \vec{A} , sino que se mantiene a lo largo de todo el proceso, evitando de manera efectiva el estancamiento en óptimos locales.

3.4.7.3.2 KAGWO

En ámbito de esta tesis, se propone el algoritmo KAGWO, concebido como una adaptación prometedora del algoritmo Grey Wolf Optimizer (GWO) para la adquisición de conocimiento. Este algoritmo está diseñado para actuar como un mecanismo de aprendizaje avanzado en sistemas expertos fundamentados en Sistemas de Razonamiento Basado en Reglas Difusas (FRBS). Su aplicación primordial se enfoca en la generación de una Base de Reglas optimizada (RB) para el consumo eficiente de energía renovable en una red de Cloud data centers (RCDCs) modulares distribuidos en diferentes ubicaciones geográficas.

Al igual que en el algoritmo KASIA, la inteligencia empleada para derivar una salida a partir de una entrada específica se basa en la potencialidad y eficacia de la Base de Conocimiento (KB) utilizada. Este capítulo procede a presentar una descripción

matemática formal del algoritmo, integrando el componente de adquisición de conocimiento al framework GWO.

En el marco de esta tesis, se conceptualiza cada RB que constituye la KB como un 'lobo' (solución), representado mediante una matriz de dimensiones $[m \times n + 2]$. En esta matriz, cada fila simboliza la codificación matemática de una regla difusa. Aquí, 'm' representa el número total de reglas, 'n' denota el número de variables de entrada (antecedentes), y los dos parámetros adicionales corresponden al resultado o consecuente de la regla y al operador lógico requerido para completar la estructura de la regla difusa.

$$P_i = \begin{bmatrix} ra_{1,1} & ra_{1,2} & \cdots & ra_{1,n} & rc_1 & ro_1 \\ \vdots & \vdots & & \vdots & \vdots & \vdots \\ ra_{m,1} & ra_{m,1} & \cdots & ra_{m,n} & rc_m & ro_m \end{bmatrix} \quad (19)$$

La partícula P_i , correspondiente a la matriz de Reglas Base (RB), incorpora elementos clave para la representación de las reglas difusas. En esta matriz, el término $ra_{m,n}$ representa el valor difuso codificado del antecedente n de la regla m. Paralelamente, rc_m se refiere al valor difuso codificado correspondiente al consecuente de la misma regla, mientras que ro_m denota el valor difuso codificado atribuido al operador lógico de la regla m. La anatomía de esta estructura de reglas RB es la que se refleja en la siguiente tabla:

Upper				
Elements	Bounds	Lower Bounds	Rules/KB	Rule Anatomy
		[-3 -3 -3 -3 -5		Inputs (5) - Outputs (1) - Logical
7	[3 3 3 3 3 5 2]	2]	7	Operator (1)

Figura 3-16 Anatomía de una regla difusa

Cada fila de la matriz de partículas se compone de siete elementos distintivos, los cuales colectivamente representan una regla difusa completa. Los primeros cinco elementos de esta fila se dedican a los valores codificados de todas las variables de entrada consideradas en el análisis. Estas variables abarcan aspectos críticos como la eficiencia cloud data center, la capacidad de procesamiento de los hosts, la disponibilidad computacional, así como las necesidades máximas y actuales de procesamiento de las Máquinas Virtuales. Es relevante señalar que el valor asignado a estas entradas es

susceptible de variar en función de la granularidad del universo de discurso adoptado y se modela como sigue:

$$ra_{m,n} \in [-MF_a, +MF_a] \quad (20)$$

Donde MF_a es el número de funciones de membresía de los antecedentes.

El sexto elemento representa el valor codificado para el consecuente de la regla difusa y se modela como sigue:

$$rc_{m,n} \in [-MF_c, +MF_c]$$

Donde MF_c representa el número de funciones de membresía para los consecuentes de la regla.

El séptimo y último elemento de la regla representa el operador lógico de esta y puede tomar un valor de 1 para representar la operación lógica AND o un valor 2 para el OR.

$$ro_{m,n} \in [1,2] \quad (21)$$

Pero incluso si las partículas generadas inicialmente pertenecieran a los límites permitidos del espacio de búsqueda, las actualizaciones posteriores de estas matrices a través de la adición de velocidades podrían eventualmente llevar a valores incoherentes de la matriz que alcanzan fuera de los límites iniciales del espacio de búsqueda. Para evitar estas situaciones, los valores anteriores se restringen para asegurar la coherencia de las matrices de la base de reglas durante toda la ejecución del algoritmo. Las restricciones se describen mediante las siguientes ecuaciones:

$$ra_{m,n} = \begin{cases} -fa(MFa), & ra_{m,n} < -fa(MFa) \\ +fa(MFa), & ra_{m,n} \geq +fa(MFa) \end{cases} \quad (22)$$

$$rc_{m,n} = \begin{cases} -fc(MFc), & rc_{m,n} < -fc(MFc) \\ +fc(MFc), & rc_{m,n} \geq +fc(MFc) \end{cases} \quad (23)$$

$$ro_{m,n} = \begin{cases} 1, & ro_{m,n} < 1 \\ 2, & ro_{m,n} \geq 2 \end{cases} \quad (24)$$

Donde $fa(MFa)$ representa el valor de codificación para la función de membresía del antecedente y $fc(MFc)$ representa el valor de codificación para la función de membresía del consecuente. Finalmente, cualquier valor fuera del rango para el valor del operador $ro_{m,n}$ será conformado a los límites establecidos por (24).

Además, el algoritmo considera el espacio de búsqueda como un paisaje con picos y valles, luego considera diferentes soluciones posibles donde las partículas (o lobos) podrían encontrar la presa que están cazando (La mejor solución).

En una modelización matemática de la jerarquía social de la manada, la mejor solución de todas sería el Alfa (α), la segunda mejor sería el Beta (β), la tercera sería Delta (δ) y todas las demás se consideran Omegas (ω).

La suposición es que el Algoritmo de optimización va a estar principalmente influenciado por las tres primeras soluciones que son alfa, beta y delta, creando una especie de triángulo alrededor de la presa o "la mejor solución". Si durante el proceso de caza (búsqueda) surge una nueva mejor solución, el alfa, beta delta y omegas anteriores se actualizarán en consecuencia.

Por otro lado, en la modelización matemática de los mecanismos de caza, la idea es "Rodear" a la presa, se consideran las siguientes dos ecuaciones:

$$D = \left| C \cdot \begin{bmatrix} ra_{1,1}^p ra_{1,2}^p & \cdots & ra_{1,n}^p rc_1^p & ro_1^p \\ \vdots & & & \vdots \\ ra_{m,1}^p ra_{m,2}^p & \cdots & ra_{m,n}^p rc_m^p & ro_m^p \end{bmatrix} - \begin{bmatrix} ra_{1,1}^w ra_{1,2}^w & \cdots & ra_{1,n}^w rc_1^w & ro_1^w \\ \vdots & & & \vdots \\ ra_{m,1}^w ra_{m,2}^w & \cdots & ra_{m,n}^w rc_m^w & ro_m^w \end{bmatrix} \right| \quad (25)$$

$$X(t + 1) = \left| \begin{bmatrix} ra_{1,1}^p ra_{1,2}^p & \cdots & ra_{1,n}^p rc_1^p & ro_1^p \\ \vdots & & & \vdots \\ ra_{m,1}^p ra_{m,2}^p & \cdots & ra_{m,n}^p rc_m^p & ro_m^p \end{bmatrix} - A \cdot \begin{bmatrix} ra_{1,1}^p ra_{1,2}^p & \cdots & ra_{1,n}^p rc_1^p & ro_1^p \\ \vdots & & & \vdots \\ ra_{m,1}^p ra_{m,2}^p & \cdots & ra_{m,n}^p rc_m^p & ro_m^p \end{bmatrix} - \begin{bmatrix} ra_{1,1}^w ra_{1,2}^w & \cdots & ra_{1,n}^w rc_1^w & ro_1^w \\ \vdots & & & \vdots \\ ra_{m,1}^w ra_{m,2}^w & \cdots & ra_{m,n}^w rc_m^w & ro_m^w \end{bmatrix} \right| \quad (26)$$

Donde:

$$X_p(t) = \begin{bmatrix} ra_{1,1}^p ra_{1,2}^p & \cdots & ra_{1,n}^p rc_1^p & ro_1^p \\ \vdots & & & \vdots \\ ra_{m,1}^p ra_{m,2}^p & \cdots & ra_{m,n}^p rc_m^p & ro_m^p \end{bmatrix} \quad (27)$$

Representa la posición de la presa.

$$X_w(t) = \begin{bmatrix} ra_{1,1}^w ra_{1,2}^w & \cdots & ra_{1,n}^w rc_1^w & ro_1^w \\ \vdots & & & \vdots \\ ra_{m,1}^w ra_{m,2}^w & \cdots & ra_{m,n}^w rc_m^w & ro_m^w \end{bmatrix} \quad (28)$$

Representa la posición del "Lobo".

Para realizar la "caza" de la presa (o lo que es lo mismo, encontrar la mejor solución), el algoritmo utiliza una manada de múltiples lobos donde obtendrá el alfa, beta y delta, que son las tres mejores soluciones. El algoritmo las utilizará para encontrar la posición de la presa y también para actualizar la posición de los otros lobos (los omegas) basándose en la posición estimada de estas tres.

Se asume que la mejor solución estará en algún lugar alrededor de las primeras tres soluciones alfa, beta y delta, y los recursos computacionales se asignarán a estas soluciones en lugar de a las otras porque serían las más prometedoras.

Esto significa que habrá tres particularizaciones de la ecuación de distancia para alfa, beta y delta:

$$D_\alpha = \left| C_1 \cdot \begin{bmatrix} ra_{1,1}^\alpha ra_{1,2}^\alpha & \cdots & ra_{1,n}^\alpha rc_1^\alpha & ro_1^\alpha \\ \vdots & & & \vdots \\ ra_{m,1}^\alpha ra_{m,2}^\alpha & \cdots & ra_{m,n}^\alpha rc_m^\alpha & ro_m^\alpha \end{bmatrix} - \begin{bmatrix} ra_{1,1} & ra_{1,2} & \cdots & ra_{1,n} & rc_1 & ro_1 \\ \vdots & & & & & \vdots \\ ra_{m,1} & ra_{m,2} & \cdots & ra_{m,n} & rc_m & ro_m \end{bmatrix} \right| \quad (29)$$

$$D_\beta = \left| C_2 \cdot \begin{bmatrix} ra_{1,1}^\beta ra_{1,2}^\beta & \cdots & ra_{1,n}^\beta rc_1^\beta & ro_1^\beta \\ \vdots & & & \vdots \\ ra_{m,1}^\beta ra_{m,2}^\beta & \cdots & ra_{m,n}^\beta rc_m^\beta & ro_m^\beta \end{bmatrix} - \begin{bmatrix} ra_{1,1} & ra_{1,2} & \cdots & ra_{1,n} & rc_1 & ro_1 \\ \vdots & & & & & \vdots \\ ra_{m,1} & ra_{m,2} & \cdots & ra_{m,n} & rc_m & ro_m \end{bmatrix} \right| \quad (30)$$

$$D_\beta = \left[C_2 \cdot \begin{bmatrix} ra_{1,1}^\delta ra_{1,2}^\delta & \cdots & ra_{1,n}^\delta rc_1^\delta & ro_1^\delta \\ \vdots & & & \vdots \\ ra_{m,1}^\delta ra_{m,2}^\delta & \cdots & ra_{m,n}^\delta rc_m^\delta & ro_m^\delta \end{bmatrix} - \begin{bmatrix} ra_{1,1} & ra_{1,2} & \cdots & ra_{1,n} & rc_1 & ro_1 \\ \vdots & & & & & \vdots \\ ra_{m,1} & ra_{m,1} & \cdots & ra_{m,n} & rc_m & ro_m \end{bmatrix} \right] \quad (31)$$

A partir de los valores obtenidos de las matrices de distancia, el algoritmo establece la ubicación teórica en la que debería encontrarse, asumiendo su seguimiento exclusivo de los líderes alfa, beta o delta, respectivamente. Esta determinación se realiza mediante la aplicación específica de la ecuación de posición, a través de la cual se identifican tres posiciones distintas correspondientes a cada uno de estos casos:

$$X_1 = X_\alpha - a_1 \cdot (D_\alpha) \quad (32)$$

$$X_2 = X_\beta - a_2 \cdot (D_\beta) \quad (33)$$

$$X_3 = X_\delta - a_3 \cdot (D_\delta) \quad (34)$$

Y como se vio anteriormente, se realiza la media de las tres posiciones para determinar la nueva posición:

$$X(t+1) = \frac{X_1 + X_2 + X_3}{3} \quad (35)$$

Para finalizar, hay que recordar que, en el marco de esta tesis, se ha optado por la utilización de funciones gaussianas para la definición de conjuntos difusos, fundamentándose en su eficiencia y simplicidad inherentes. La formulación matemática de dicha función gaussiana se articula de la manera siguiente:

$$\mu_i^{X_m} = \frac{1}{\sqrt{2\pi\sigma_i^{2X_m}}} \exp\left(\frac{-(z - \tau_i^{X_m})}{2\sigma_i^{2X_m}}\right), \{z \in \mathbb{R}, z \leq 1\} \quad (36)$$

Donde $\tau_i^{X_m}$ y $\sigma_i^{X_m}$ denotan la media y la desviación estándar, respectivamente. Aquí, z denota la variable independiente que describe la característica, y m corresponde a la característica actual. La selección de funciones gaussianas está impulsada por su propiedad ventajosa de tener una extensa área de influencia, cubriendo todo el universo de discurso. Esta característica permite que el sistema realice contribuciones valiosas en una amplia gama de condiciones.

En un sistema difuso, cada valor de la variable del sistema se asocia con un grado de pertenencia relacionado con un conjunto difuso, asignado con valores normalizados que van desde 0 (exclusión completa de la variable) hasta 1 (pertenencia total de la variable). Los valores intermedios indican una pertenencia parcial al conjunto, permitiendo un razonamiento preciso y flexible.

3.4.8 Configuración de la Simulación en Matlab

3.4.8.1 Parámetros de Simulación

Ambos algoritmos, KASIA y KAGWO, se sometieron a una serie de simulaciones, consistiendo cada una de ellas en 30 repeticiones. Además, se implementaron 50 iteraciones en cada simulación con el fin de optimizar el rendimiento.

En lo que respecta a KASIA, se utilizó un conjunto de 64 partículas, configuradas de tal manera que se maximizara la eficiencia en el uso de múltiples núcleos de procesamiento. Este enfoque se caracteriza por el empleo de 64 núcleos de la máquina simuladora, operando bajo un esquema de multihilo. Tal disposición permitió la evaluación simultánea de la función de fitness relacionada con el consumo de energía renovable asociado a una base de conocimientos específica. Este método ha demostrado ser eficaz en la optimización del uso de la CPU, distribuyendo la carga de trabajo de manera equitativa entre todos los núcleos disponibles. Las pruebas exhaustivas realizadas han indicado que el número óptimo de partículas, que resulta en el mejor rendimiento, es de 64.

Numero de simulaciones	Numero de partículas	Iteraciones	Paso inicial	Peso final	c1	c2
30	64	50	0,9	0,2	2	2

Figura 3-17 Configuración de la simulación de Matlab para KASIA

Numero de simulaciones	Numero de partículas	Iteraciones
30	64	50

Figura 3-18 Configuración de la simulación Matlab para KAGWO

En KASIA, se empleó un valor máximo inicial de 0.9 para el peso, el cual exhibió una disminución exponencial hacia 0.2 en el transcurso del proceso de optimización. Los coeficientes sociales, identificados como c1 y c2, fueron uniformemente asignados el valor de 2, basándose en los resultados óptimos obtenidos a través de múltiples

experimentaciones. Por otro lado, la configuración adoptada por KAGWO se distingue favorablemente de la utilizada en KASIA debido a su notable simplicidad y eficiencia operativa. Esto se debe a que KAGWO requiere un menor número de parámetros para su proceso de optimización.

A diferencia de KASIA, que exige un ajuste minucioso de varios parámetros, incluyendo el peso, coeficientes sociales y la cantidad de partículas, la metodología de KAGWO se caracteriza por una configuración más sencilla, enfocada únicamente en la determinación del número de simulaciones, lobos e iteraciones necesarias. Esta simplificación en la configuración de KAGWO no solo facilita su implementación, sino que también minimiza la posibilidad de errores y complejidades asociadas a la gestión de parámetros, consolidando a KAGWO como una opción más eficaz y preferente para las actividades relacionadas con la optimización.

3.4.9 Consideraciones Éticas y Limitaciones

Esta sección aborda las consideraciones éticas y las limitaciones inherentes a la investigación presentada en esta tesis, la cual se centra en la optimización del uso de energías renovables en data centers mediante la migración de máquinas virtuales utilizando Sistemas Basados en Reglas Difusas (FRBS) y Algoritmos de Swarm Computing.

3.4.9.1 Aspectos Éticos de la Investigación

En términos éticos, la investigación se adhiere a los principios de integridad científica y responsabilidad. Se ha garantizado que todos los datos y resultados presentados sean precisos y verídicos, y que se han citado adecuadamente todas las fuentes y referencias. Además, la investigación no involucra datos sensibles ni información personal, y se ha llevado a cabo con un enfoque en la sostenibilidad y el impacto ambiental positivo, alineándose con los principios éticos de la investigación responsable y el desarrollo sostenible.

Se ha prestado especial atención a la transparencia y la replicabilidad de los experimentos y simulaciones. Esto incluye una descripción detallada de la metodología

utilizada, las herramientas y tecnologías empleadas, y los procedimientos de análisis de datos, permitiendo que otros investigadores validen o construyan sobre el trabajo realizado.

3.4.9.2 Limitaciones del Estudio y Consideraciones Futuras

A pesar de los esfuerzos para garantizar la rigurosidad y la relevancia del estudio, existen ciertas limitaciones que deben ser reconocidas. En primer lugar, aunque se han utilizado herramientas de simulación avanzadas, los resultados pueden diferir en entornos de data centers reales debido a la variabilidad de factores ambientales y operativos. Por lo tanto, se recomienda la validación de los hallazgos en entornos de producción reales para confirmar su aplicabilidad y eficacia.

Además, aunque los algoritmos de FRBS y Swarm intelligence han demostrado ser efectivos en los escenarios de simulación, su desempeño puede variar en función de la configuración específica del sistema y las condiciones cambiantes del entorno. Es crucial continuar investigando y mejorando estos algoritmos para aumentar su robustez y adaptabilidad.

En cuanto a las consideraciones futuras, se sugiere explorar la integración de otras tecnologías emergentes, como el uso de diferentes tipos de redes neuronales, para investigar si se puede mejorar aún más la eficiencia energética y la gestión de recursos en los data centers. Además, sería beneficioso realizar estudios adicionales que se centren en la escalabilidad e interpretabilidad de estas soluciones en diferentes tamaños y tipos de data centers, incluyendo aquellos con diferentes mezclas de fuentes de energía renovable.

En conclusión, aunque esta investigación presenta avances en la optimización del uso de energías renovables en data centers, es importante reconocer sus limitaciones y las oportunidades para futuras investigaciones. La continuación de este trabajo puede contribuir aún más a la creación de soluciones sostenibles y eficientes para los desafíos energéticos en el ámbito del cloud computing.

3.5 Conclusión

En esta investigación, se ha abordado el problema de la sostenibilidad energética en data centers distribuidos, desarrollando soluciones innovadoras para la migración eficiente de máquinas virtuales (VMs) en entornos cloud. Se ha demostrado que la optimización de estas migraciones no solo puede reducir significativamente el consumo de energía, sino también incrementar el uso de fuentes de energía renovable, mejorando la sostenibilidad de los centros de datos.

Las contribuciones clave de esta tesis se centran en dos desarrollos principales: un sistema experto basado en reglas borrosas (FRBS) y un algoritmo de adquisición de conocimiento (KAGWO) que mejora el comportamiento de estos sistemas expertos. A través de la aplicación de estos enfoques, se ha logrado optimizar la migración de VMs entre data centers geográficamente dispersos, considerando la disponibilidad de energía renovable.

El sistema experto basado en reglas borrosas se presentó como una herramienta eficaz para gestionar la migración dinámica de VMs, mejorando la eficiencia energética y la sostenibilidad de los data centers. Los resultados experimentales han demostrado que el uso de reglas borrosas permite una gestión flexible y eficiente de los recursos energéticos, integrando técnicas de machine learning y Swarm intelligence para una mejor gestión de los recursos en la nube.

Por otro lado, el desarrollo del algoritmo KAGWO representa una evolución significativa del Grey Wolf Optimizer, adaptándolo para la adquisición de conocimiento y la optimización del consumo de energía en data centers. Los resultados comparativos entre KAGWO y otros enfoques han mostrado mejoras sustanciales en la sostenibilidad energética, demostrando que es posible optimizar el uso de energías renovables sin comprometer la calidad del servicio.

La extensión y personalización del simulador CloudSim han permitido modelar y simular múltiples data centers con características de energía renovable, facilitando la evaluación de diversas estrategias de migración y sus impactos en la sostenibilidad. Estos desarrollos han sido fundamentales para validar los algoritmos propuestos y han proporcionado una plataforma robusta para futuros estudios en este campo.

Un aspecto que considerar es la selección de los algoritmos comparados con KAGWO. La elección se basó en su aplicabilidad y relevancia dentro del contexto de la programación de tareas Inter-CDCs. Aunque existen en la literatura algoritmos de optimización más potentes y sofisticados que los utilizados en las comparativas, estos han sido predominantemente aplicados en contextos Intra-CDC, donde las dinámicas y requerimientos son significativamente diferentes.

La adaptación de estos algoritmos más avanzados para un entorno de programación Inter-CDCs habría requerido un tiempo considerable de investigación y desarrollo, un esfuerzo que excede el alcance y los recursos disponibles para esta tesis. Además, la falta de aplicaciones documentadas de estos algoritmos en el contexto específico de la migración de VMs entre data centers geográficamente dispersos añade un nivel adicional de incertidumbre y complejidad.

Incorporar y adaptar dichos algoritmos podría haber introducido riesgos adicionales en términos de viabilidad y resultados, desviando el enfoque principal de esta investigación. La robustez y la eficacia de KAGWO en el manejo de las necesidades específicas de sostenibilidad y eficiencia energética en entornos cloud distribuidos justifican plenamente su selección y priorización en esta tesis.

En resumen, esta tesis ha contribuido al campo de la optimización de energía en data centers distribuidos, proponiendo soluciones avanzadas y eficaces para la migración de VMs. Los desarrollos presentados no solo mejoran la eficiencia energética y la sostenibilidad, sino que también abren nuevas vías para la investigación y aplicación de técnicas avanzadas de inteligencia artificial y optimización en entornos cloud. La continuación de este trabajo puede llevar a avances aún mayores en la gestión sostenible de recursos en la nube, contribuyendo al desarrollo de infraestructuras de TI más eficientes.

4 Resultados y análisis

4.1 Introducción

Como se indica en el capítulo de metodología y con el fin de poder comparar las diferentes estrategias de migración de máquinas virtuales en diferentes data centers distanciados geográficamente, se realizaron varios planes de pruebas correspondientes a los diferentes algoritmos de enrutamiento para los elementos de carga computacionales (Cloudlets) que han de ser servidos en los data centers. Los resultados que se van a presentar en estos capítulos corresponden a los algoritmos:

- Inter Quartile Range (IQR).
- Local Regression (LR).
- Local Regression Robust (LRR).
- Static Threshold (THR).
- Renewable Optimizer Based on Dynamic Inputs (RODI).
- Fuzzy Rules Based System with an expert Knowledge base.
- Fuzzy Rules Based System with KASIA Knowledge acquisition.
- Pittsburgh Algorithm
- Fuzzy Rules Based System with KAGWO Knowledge acquisition.

En los siguientes apartados se presentará la descripción de los escenarios de pruebas, sus resultados experimentales y todas las gráficas que describen el rendimiento de los citados algoritmos comparados con el objetivo de realizar un análisis cualitativo del comportamiento de cada uno, lo que permitirá discutir los resultados y sacar las conclusiones pertinentes en los capítulos siguientes de discusión y conclusiones.

Es importante reiterar que el trabajo publicado en el 2021 y titulado "Sustainability-based Framework for Virtual Machines Migration Among Cloud data centers", publicado por el "Institute of Electrical and Electronics Engineers (IEEE)" en la conferencia "15th International Conference on Signal Processing and Communication Systems (ICSPCS)" en Australia, amplió y extendió el simulador CloudSim para permitir modelar y simular múltiples data centers por una parte y por otra que estos centros tuviesen un carácter renovable y conscientes de la energía, facilitando experimentos con diversas estrategias de migración entre data centers. Sus resultados son los que se presentan en el punto 4.2 Resultados Previos a la contribución 1 - Comparativa de eficiencia renovable para los algoritmos IQR, LR, LRR, THR y RODI.

4.2 Resultados Previos a la contribución 1 - Comparativa de eficiencia renovable para los algoritmos IQR, LR, LRR, THR y RODI

4.2.1 Introducción

Como se explica en la sección de metodología, se utiliza CloudSim para comparar el desempeño de los diferentes algoritmos de repartición de carga sobre los Data Centers ya que CloudSim es un simulador para infraestructura de computación y modelado en el cloud que permite la realización de análisis para Optimización de recursos y consumo de energía. Proporciona multitud de posibles configuraciones reales para los recursos dentro de los Data-Centers como son los hosts con su CPU, RAM, almacenamiento, red... además de simular también toda la capa de virtualización que va sobre los dichos hosts.

Las maquinas (hosts) utilizados para la realización de las diferentes simulaciones han sido las siguientes:

- La HP ProLiant ML110 G4 (Xeon 3040, dual core, 1,8 GHz, 4 GB RAM, y con 1Gbps de ancho de banda).
- Hp ProLiant ML110 G5 (Xeon3075, dual core, 2,6 Ghz, 4Gb RAM, con 1Gbps de ancho de banda)

Los datos de consumo de energía para las maquinas simuladas ha sido obtenido a partir de la “Standard Performance Evaluation Corporation” (SPEC, <https://www.spec.org>), que es una corporación sin ánimo de lucro creada para establecer y mantener tanto datos reales de funcionamiento como herramientas que evalúan el rendimiento de sistemas de computación de nueva generación. El consumo de energía asociado a una utilización específica de la CPU proporcionada por la SPEC oscila desde el 0% al 100% con un paso del 10%. Si la utilización real no es exactamente uno de los valores anteriores, el simulador puede utilizar un método de interpolación lineal que para aproximar el valor real de la energía consumida. Es importante señalar que, para obtener resultados lo más realistas posibles, CloudSim proporciona unas trazas de utilización de CPU de 288 entradas, dichas entradas reflejan una utilización real de la CPU cada 5 minutos, lo que da un total de 24 horas potenciales de simulación real.

En cuanto a la carga que se le inyecta al sistema para las simulaciones, se considera la siguiente configuración de Cloudlets:

- Longitud: $2500 * \text{Tiempo de simulación (MI)}$
- Tamaño del archivo Cloudlets (programa + datos): 300 (bytes)
- Tamaño de salida de Cloudlets: 300 (bytes)
- Número de *Pes*: 1

Respecto a las máquinas virtuales (MV) utilizadas, se generan cuatro tipos de forma aleatoria a partir de las configuraciones siguientes:

- Número de procesadores: 1
- MIPS: 2500, 2000, 1000 o 500
- Memoria RAM: 870, 1740 o 613
- Ancho de banda: 100 Mbps
- Tamaño de las MV: 2,5 Gb

Cabe mencionar que, para las estrategias de migración de máquinas virtuales, CloudSim proporciona un conjunto de implementaciones [39], algunas de las cuales han sido utilizadas para comprarlas con una de las aportaciones de este trabajo y que es el algoritmo “RODI”. Las implementaciones con las que se ha podido comparar son las siguientes:

- **Inter Quartil Range (IQR):** Es una técnica Se utiliza en algoritmos de machine learning para identificar valores atípicos, pero En este caso nos ayuda a calcular la mediana de un conjunto de datos ordenado (dicho conjunto consiste en los datos de utilización de los hosts) donde se intenta averiguar la diferencia entre la primera mitad y la segunda mitad (una medida de propagación) usando otras dos medianas llamadas Q1 y Q2. La primera corta el conjunto de datos por la mitad y los otros dos cortes se realizan en los lados izquierdo y derecho. Esta estrategia calcula un umbral superior que identifica los hosts utilizados en exceso, lo que permite tener un criterio de utilización sobre el que basarse para seleccionar el balanceo de la carga en términos de migración de las máquinas virtuales dentro de los Data-Centers.
- **Local Regression (LR):** Es una técnica utilizada también en algoritmos de machine learning que intenta adaptar la mejor línea de regresión a un conjunto de

muestras utilizando una optimización por mínimos cuadrados de las distancias de forma que pueda predecir un nuevo comportamiento de las muestras fuera del conjunto de datos de entrenamiento. Con esta estrategia de migración de máquinas virtuales se intenta predecir la potencial sobrecarga de los hosts en diferentes situaciones.

- **Local Regression Robust (LRR):** Es un modelo más apropiado que el *LR* en contextos donde hay muchas más muestras atípicas en los datos (El modelo *LR* en casos similares podría derivar en resultados sesgados que no se puedan interpretarse correctamente) y en los que la distribución de estos no sea lineal. Es también una técnica utilizada en Algoritmos de machine learning que intenta ajustar una curva a un conjunto de datos realizando múltiples aproximaciones en diferentes subconjuntos del mismo. Para ello se utiliza un parámetro "n" que controla el tamaño de los subconjuntos de datos a los que se les realiza aproximaciones por regresiones parciales (lineales o parabólicas). El objetivo es el de tener un aproximador de porcentaje de utilización de los hosts que permita decidir sobre el balanceo de las cargas a mandar a los Data-Centers por el bróker.
- **Static Threshold:** Otra estrategia de migración de las máquinas virtuales que utiliza un umbral estático de utilización de CPU (THR) para detectar el porcentaje de utilización de los hosts dentro de los Data-Centers.
- **Renewable Optimizer Based On Dynamic Inputs (RODI):** Representa la primera contribución en el artículo [ref.] y se basa en una familia de técnicas basadas en colas y funciona como se ha mencionado en el capítulo [ref.], en un contexto donde hay varios Data Centers con diferentes niveles de suministro de energía renovable disponible cada Hora.

Cabe destacar que la política de identificación y de selección de máquinas virtuales para su posterior migración se basa en la estrategia del CloudSim del “*tiempo mínimo de migración*” que proviene de la filosofía de “Pre-copy” de las migraciones en tiempo real.

4.2.2 Descripción de los escenarios de simulación

En el contexto de esta comparativa entre estas estrategias de migración de máquinas virtuales, se han considerado los escenarios descritos en la Tabla 4-1.

	Escenario 1	Escenario 2	Escenario 3
Hosts	265	530	800
VMs	350	695	1025
Cloudlets	500 - 1500 - 3000	1000 - 2000 - 5000	1500 - 5000 - 10000

Tabla 4-1: Escenarios de simulación utilizados

Podemos ver en la tabla tres escenarios o categorías para la simulación de experimentos que van desde los pequeños como el escenario 1 donde tenemos 265 hosts con 350 máquinas virtuales repartidas entre cuatro Data Centers geográficamente distantes (ver Figura 4-1). Para esta configuración se establecen a su vez tres situaciones de carga que van desde los 500 Cloudlets, pasando por 1500 y llegando hasta los 3000. Esto se debe a que se pretendía estresar la configuración en tres posibles regímenes de operación y que son bajo (con 500), medio (con 1500) y alto con (3000).

El segundo escenario es un escenario considerado “Medio” en términos de topología de máquinas y de requerimientos de red, donde en este caso tenemos 530 hosts con 695 máquinas virtuales repartidas entre los mismos cuatro Data Centers geográficamente distantes. Igualmente, para este segundo escenario de configuración se establecen tres situaciones de carga a la cual someter a los Data Centers y que en este caso van desde los 1000 Cloudlets, pasando por 2000 y llegando hasta los 5000. Esto se debe igualmente a que se pretende someter al sistema a tres regímenes de operación y que son: bajo (con 1000), medio (con 2000) y alto con (5000).

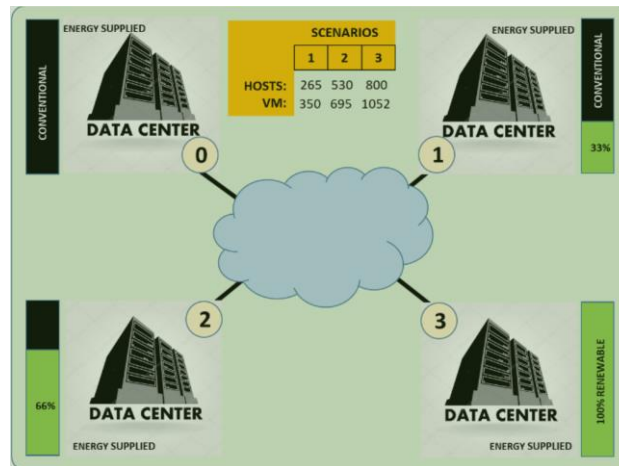


Figura 4-1: Escenarios de simulación

El tercer y último escenario es considerado “Grande” en términos de topología de máquinas y de requerimientos de red y donde en esta vez tenemos 800 hosts con 1025 máquinas virtuales repartidas entre los mismos cuatro Data Centers geográficamente distantes. Esta vez para este último escenario de configuración se establecen también tres situaciones de carga a la cual someter a los Data Centers y que en este caso van desde los 1500 Cloudlets, pasando por 5000 y llegando hasta los 10000. Como es los escenarios precedentes, esto se debe a que se pretende someter al sistema a tres regímenes de operación y que son en este caso: bajo (con 1500), medio (con 5000) y alto con (10000).

4.2.3 Resultados experimentales

4.2.3.1 Resultados globales de uso de energía renovable por algoritmo

En las tablas 2, 3 y 4 se muestran tanto las medias de porcentajes de energía renovable utilizada en los data Centers obtenidas en 30 experimentos consecutivos, como la degradación porcentual de los valores de los niveles de servicio para los algoritmos IQR, LR, LRR, THR y la nueva estrategia RODI en los tres escenarios descritos en el apartado 4.2.2.

En la Tabla 4-2 que se muestra a continuación, se presentan los resultados de las simulaciones del primer escenario (Pequeño) de 265 hosts, 350 máquinas virtuales y cargas de 500, 1000 y 1500 Cloudlets. Se puede observar que para el algoritmo TRH, la media de energía renovable ronda el 50,3% en los tres niveles de carga en los Data

Centers, mientras que para el IQR solo se alcanza el 49,1%. El LRR alcanza una media de 49,7% y el LR un poco menos llegando a un 48,8%. Finalmente, el algoritmo de migración propuesto (RODI) (**Renewable Optimizer Based On Dynamic Inputs**) alcanza un 66,9% de media entre las simulaciones des los tres niveles de carga.

ESCENARIO 1						
HOSTS: 265, VM: 350		ALGORITMO				
Cloudlets	Rendimiento	THR	IQR	LRR	LR	RODI
500	Ren. used (%)	50,75%	49,02%	50,43%	47,72%	67,10%
	SLA Deg. (%)	<0,0001	<0,0001	<0,0001	<0,0001	<0,0001
1500	Ren. used (%)	50,38%	49,54%	49,39%	49,28%	66,28%
	SLA Deg. (%)	<0,0001	<0,0001	<0,0001	<0,0001	<0,0001
3000	Ren. used (%)	49,95%	48,90%	49,21%	49,58%	67,28%
	SLA Deg. (%)	<0,0001	<0,0001	<0,0001	<0,0001	<0,0001

Tabla 4-2: Resultados obtenidos para el escenario 1

En la Tabla 4-3 que se muestran los resultados de las simulaciones del segundo escenario (Mediano) de 530 hosts, 695 máquinas virtuales con cargas de 1000, 2000 y 5000 Cloudlets. Se observa que en este caso para el algoritmo TRH, la media de energía renovable ronda el 49,2% en los tres niveles de carga en los Data Centers, mientras que para el IQR solo se alcanza el 49,1%. El LRR alcanza una media de 48,8% y el LR un poco menos llegando a un 49,6%. Finalmente, el algoritmo de migración propuesto (RODI) (**Renewable Optimizer Based On Dynamic Inputs**) alcanza un 68,1% de media entre las simulaciones des los tres niveles de carga.

ESCENARIO 2						
HOSTS: 530, VM: 695		ALGORITMO				
Cloudlets	Rendimiento	THR	IQR	LRR	LR	RODI
1000	Ren. used (%)	49,22%	48,82%	48,82%	49,11%	68,06%
	SLA Deg. (%)	<0,0001	<0,0001	<0,0001	<0,0001	<0,0001
2000	Ren. used (%)	48,78%	49,10%	49,11%	50,80%	68,01%
	SLA Deg. (%)	<0,0001	<0,0001	<0,0001	<0,0001	<0,0001
5000	Ren. used (%)	49,56%	49,49%	48,63%	48,87%	68,22%
	SLA Deg. (%)	<0,0001	<0,0001	<0,0001	<0,0001	<0,0001

Tabla 4-3: Resultados obtenidos para el escenario 2

Finalmente, en Tabla 4-4 que se muestran los resultados de las simulaciones del tercer escenario (Grande) de 800 hosts, 1052 máquinas virtuales con cargas de 1500, 5000 y 10000 Cloudlets. Se observa que en este caso para el algoritmo TRH, la media de energía renovable ronda el 49,2% en los tres niveles de carga en los Data Centers,

mientras que para el IQR solo se alcanza el 49,1%. El LRR alcanza una media de 48,8% y el LR un poco menos llegando a un 49,6%. Finalmente, el algoritmo de migración propuesto (RODI) (**Renewable Optimizer Based On Dynamic Inputs**) alcanza un 68,1% de media entre las simulaciones des los tres niveles de carga.

ESCENARIO 3						
HOSTS: 800, VM: 1052		ALGORITMO				
Cloudlets	Rendimiento	THR	IQR	LRR	LR	RODI
1500	Ren. used (%)	49,22%	48,82%	48,82%	49,11%	68,06%
	SLA Deg. (%)	<0,0001	<0,0001	<0,0001	<0,0001	<0,0001
5000	Ren. used (%)	48,78%	49,10%	49,11%	50,80%	68,01%
	SLA Deg. (%)	<0,0001	<0,0001	<0,0001	<0,0001	<0,0001
10000	Ren. used (%)	49,56%	49,49%	48,63%	48,87%	68,22%
	SLA Deg. (%)	<0,0001	<0,0001	<0,0001	<0,0001	<0,0001

Tabla 4-4: Resultados obtenidos para el escenario 3

4.2.3.2 Resultados de energía consumida por Data Center

a. Escenario 1 con 500 Cloudlets de carga

Las figuras : Figura 4-2, Figura 4-3, Figura 4-4, Figura 4-5, Figura 4-6, muestran el consumo de energía en los cuatro Data Centers para todos los algoritmos comparados. Se muestra tanto la energía total consumida durante el lapso de simulación (Que en este caso es de 1h de funcionamiento real) como la energía renovable utilizada en cada uno según la configuración del escenario 1 descrita en el apartado 4.2.2 con una carga de 500 Cloudlets.

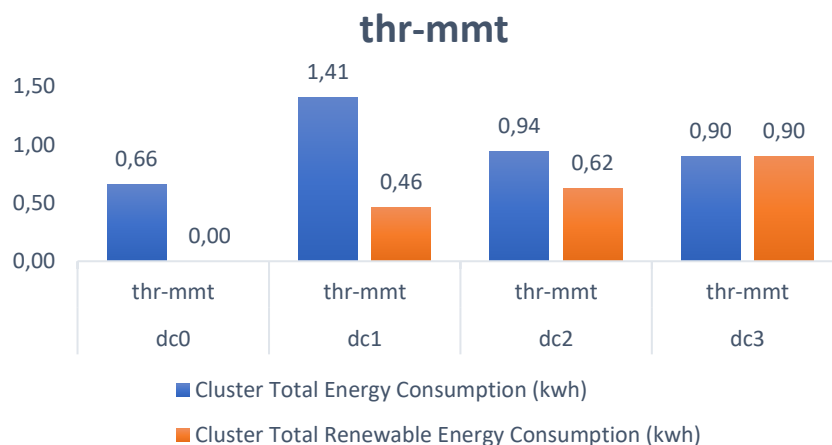


Figura 4-2 Comportamiento del algoritmo THR en el escenario 1 con 500 Cloudlets de carga

Para el algoritmo THR con estrategia MMT, vemos que el consumo de los dc2 y 3 es casi el mismo en términos generales (0,94 kwh y 0,9 kwh) manteniendo una energía renovable proporcional al porcentaje disponible en esos Date Centers y que son el 66% (0,62 kwh) y el 100% (0,9 kwh) respectivamente. Por otro lado, se ve que el mayor consumo de energía total se encuentra en el dc1 que tiene un 33% de renovable (0,46 kwh) y alcanza un consumo total de 1,41 kwh. El menor consumo se registra en el dc0 (0,66 kwh) que no tiene ninguna fuente de energía renovable en su alimentación.

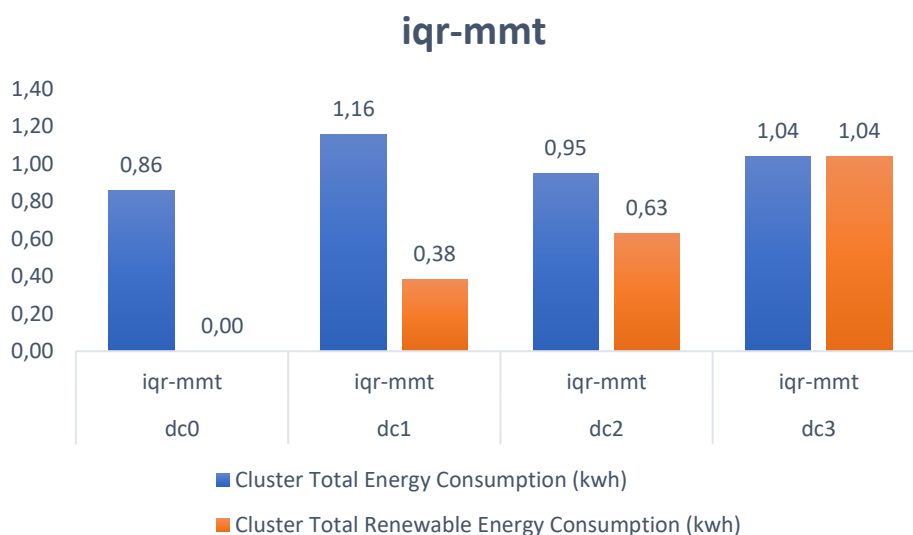


Figura 4-3: Comportamiento del algoritmo IQR en el escenario 1 con 500 Cloudlets de carga

Para el algoritmo IQR con estrategia MMT, se repite un comportamiento parecido al anterior donde vemos que el consumo de los dc2 y 3 es parecido en términos generales (1,04 kwh y 0,95 kwh) manteniendo una energía renovable proporcional al porcentaje disponible en esos Date Centers y que son el 66% (0,63 kwh) y el 100% (1,04 kwh) respectivamente. Por otro lado, se ve que el mayor consumo de energía total se encuentra en el dc1 que tiene un 33% de renovable (0,38 kwh) y alcanza un consumo total de 1,41 kwh. El menor consumo se registra en el dc0 (0,86 kwh) que no tiene ninguna fuente de energía renovable en su alimentación.

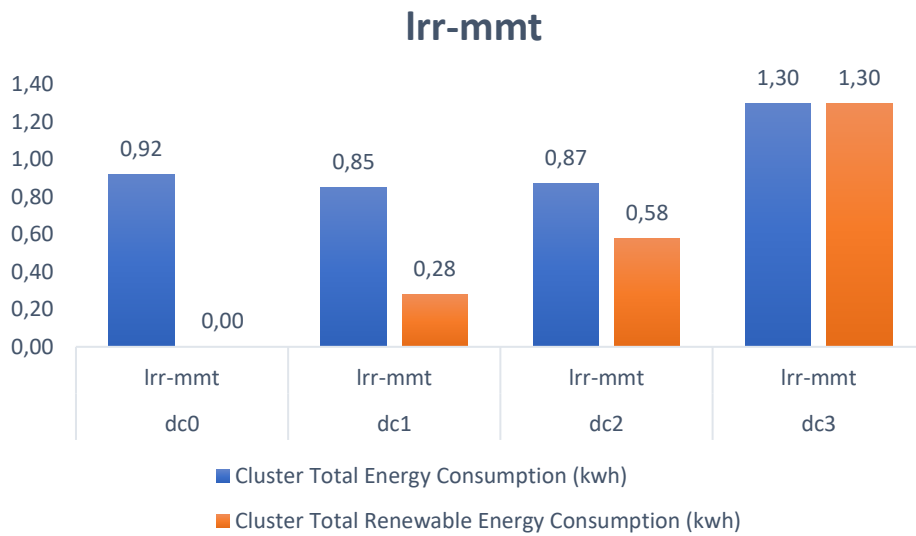


Figura 4-4: Comportamiento del algoritmo LRR en el escenario 1 con 500 Cloudlets de carga

Para el algoritmo LRR con estrategia MMT se observa que tres de los Data Centers muestran un consumo total parecido y estos son el dc 0, 1 y el 2 con registros de 0,92 kwh/0 renovable, 0,85kwh/0,28 kwh renovable y 0,87 kwh/058 kwh renovable. El consumo más importante se registraba en el dc3 con 1,30kwh tanto de consumo total del cual el 100% es en energía renovable.

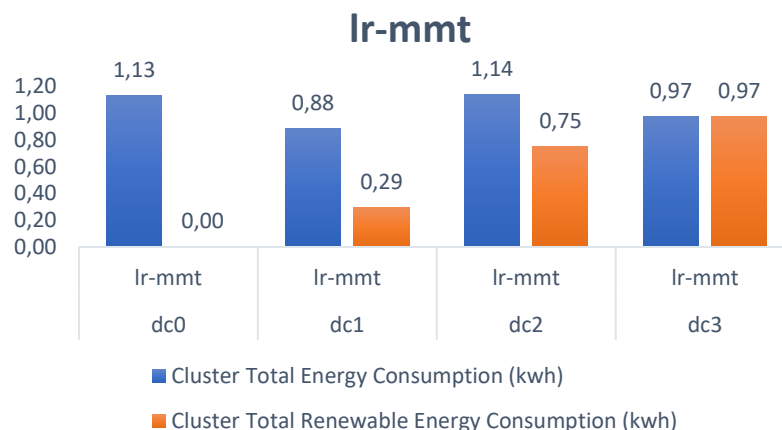


Figura 4-5: Comportamiento del algoritmo LR en el escenario 1 con 500 Cloudlets de carga

El algoritmo LR con MMT muestra un comportamiento diferente al de los anteriores donde el mayor consumo de energía se registra en los dc 0 y 2 con un 1,14 y 1,14 kwh respectivamente y donde en el dc 2 se observa que la proporción de renovables en el consumo alcanza el 0,75 kwh. El dc 1 registra un 0,88 kwh de consumo total con 0,29 kwh de renovable y finalmente el dc 3 registra un 0,97 kwh de consumo total de los cuales el 100% es renovable.

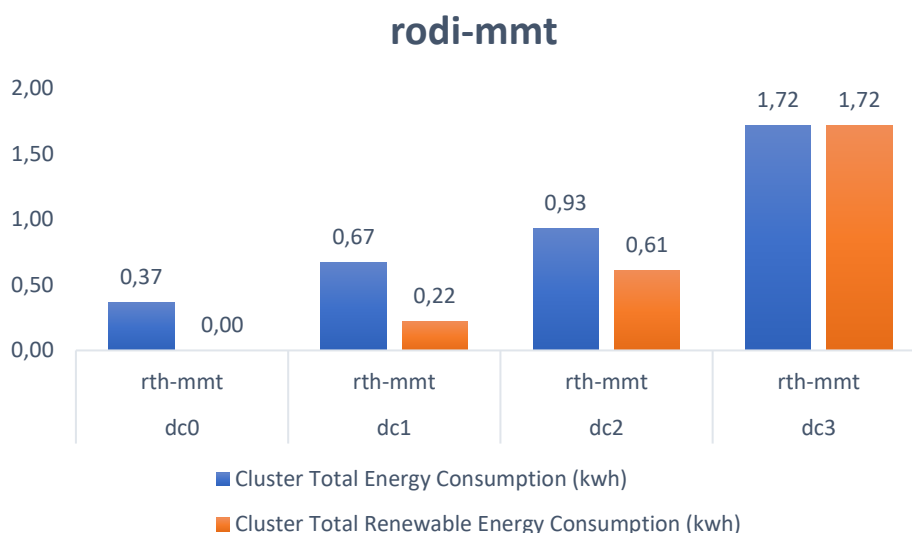


Figura 4-6: Comportamiento del algoritmo RODI en el escenario 1 con 500 Cloudlets de carga

Finalmente llegamos al comportamiento del algoritmo propuesto en el trabajo [ref.] al que se le ha denominado RODI con MMT. En este algoritmo se ve que el consumo

de energía se concentra en los tres Data Centers que disponen de energía renovable en su alimentación, registrando consumos respectivos de energía total y renovable en los Data Centers 1, 2 y 3 de 0,67 kwh/0,22kwh renovable, 0,93 kwh/0,61 kwh renovable y donde el mayor registro de energía de todos se concentraba en el Data Center 3 que contenía el mayor porcentaje de energía renovable disponible de todos los Data Centers. Se puede observar de hecho de que, de todos los algoritmos analizados, RODI-MMT presenta el mayor consumo de energía renovable de todos los escenarios simulados de todos los algoritmos marcando unos notables 1,72 kwh.

b. Escenario 1 con 1500 Cloudlets de carga

Las figuras: Figura 4-7, Figura 4-8, Figura 4-9, Figura 4-10, Figura 4-11 , muestran el consumo de energía en los cuatro Data Centers para todos los algoritmos comparados. Se muestra tanto la energía total consumida durante el lapso de simulación (Que en este caso es de 1h de funcionamiento real) como la energía renovable utilizada en cada uno según la configuración del escenario 1 descrita en el apartado 4.2.2 con una carga de 1500 Cloudlets.

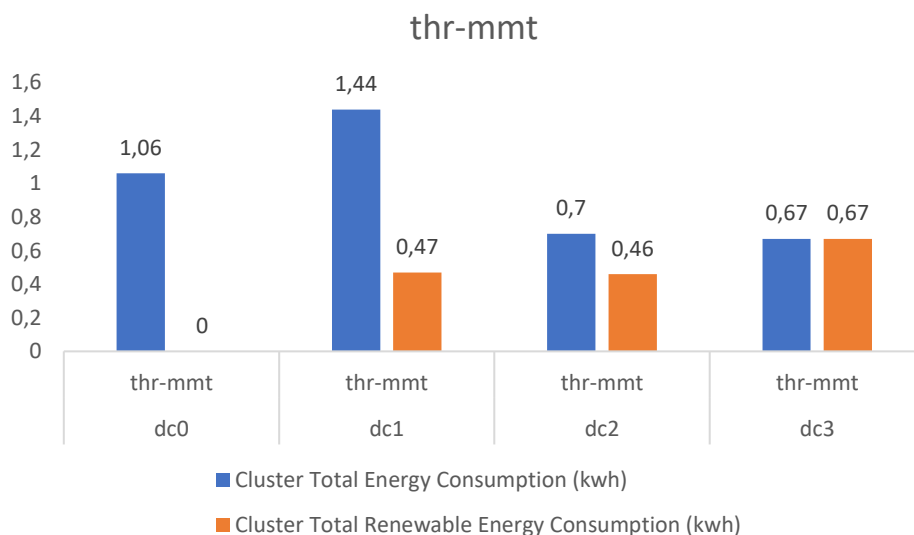


Figura 4-7 Comportamiento del algoritmo THR en el escenario 1 con 1500 Cloudlets de carga

Para el algoritmo THR con estrategia MMT, vemos que el consumo de los dc2 y 3 es casi el mismo en términos generales (0,7 kwh y 0,67 kwh) manteniendo una energía renovable proporcional al porcentaje disponible en esos Date Centers y que son el 66% (0,46 kwh) y el 100% (0,67 kwh) respectivamente. Por otro lado, se ve que el mayor consumo de energía total se encuentra en el dc1 que tiene un 33% de renovable (0,47 kwh) y alcanza un consumo total de 1,44 kwh. El consumo en el dc0 es de 1,06 kwh y este no tiene ninguna fuente de energía renovable en su alimentación. El comportamiento con 1500 Cloudlets de carga se parece en gran medida al observado para 500.

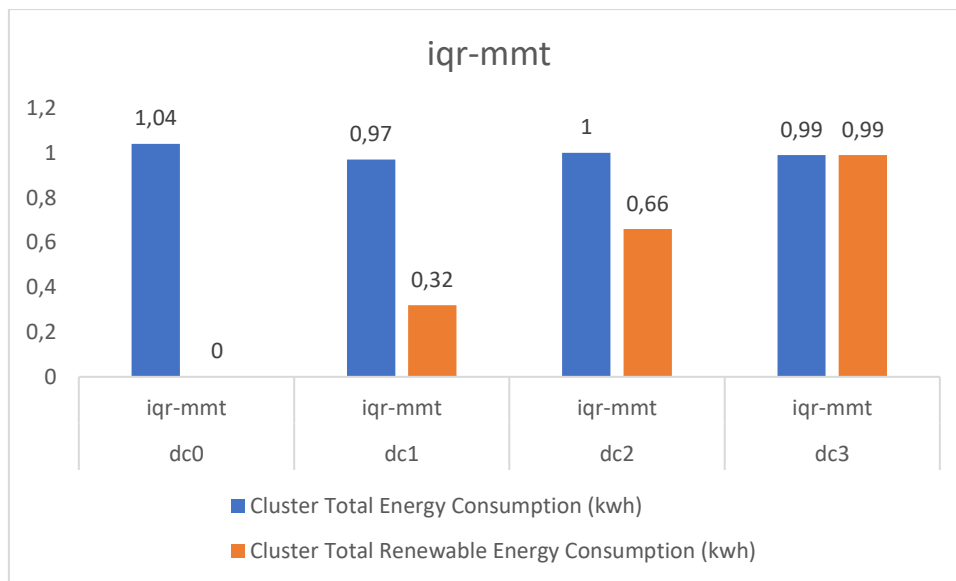


Figura 4-8 Comportamiento del algoritmo IQR en el escenario 1 con 1500 Cloudlets de carga

Para el algoritmo IQR con estrategia MMT, se repite un comportamiento parecido al anterior donde vemos que el consumo de los dc2 y 3 es parecido en términos generales (1 kwh y 0,99 kwh) manteniendo una energía renovable proporcional al porcentaje disponible en esos Date Centers y que son el 66% (0,66 kwh) y el 100% (1 kwh) respectivamente. Por otro lado, se ve que el mayor consumo de energía total se encuentra

en el dc0 que no está alimentado por ninguna fuente renovable. El dc1 registra un consumo de 0,97 kwh en total consumido con 0,32 kwh de consumo renovable.

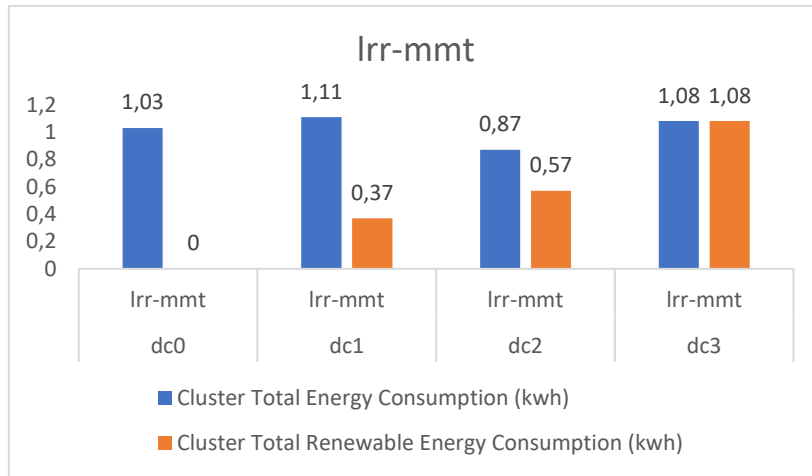


Figura 4-9 Comportamiento del algoritmo LRR en el escenario 1 con 1500 Cloudlets de carga

Para el algoritmo LRR con estrategia MMT se observa un consumo total parecido entre los Data Centers dc 2 y 3 con registros de 0,87 kwh/0,57 kwh renovable y 1,08kwh/1,08 kwh renovable. El consumo más importante se registraba en el dc1 con 1,11 kwh/0,37 kwh renovable. El dc0 registra 1,03 kwh sin renovable al no estar alimentado por esta.

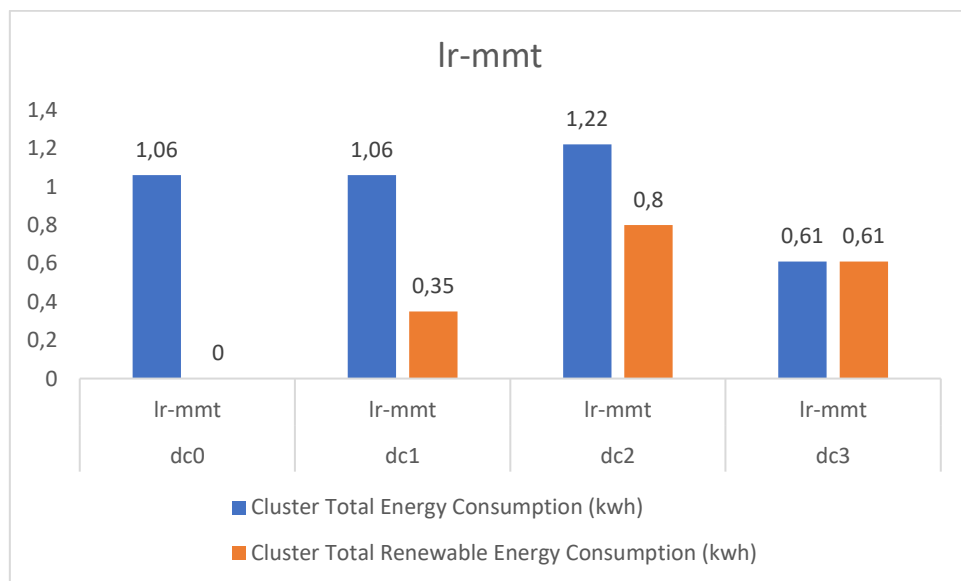


Figura 4-10 Comportamiento del algoritmo LR en el escenario 1 con 1500 Cloudlets de carga

El algoritmo LR con MMT muestra un comportamiento diferente al de los anteriores donde el mayor consumo de energía se registra en los dc 0 y 1 con un 1,06 y 1,06 kwh respectivamente y donde en el dc 1 se observa que la proporción de renovables en el consumo alcanza el 0,35 kwh. El dc 2 registra un 1,22 kwh de consumo total con 0,8 kwh de renovable y finalmente el dc 3 registra un 0,61 kwh de consumo total de los cuales el 100% es renovable. Se ve que, en este escenario, el Data Center con menos consumo de energía es el 3 que se supone que tiene el mayor porcentaje de energía renovable disponible.

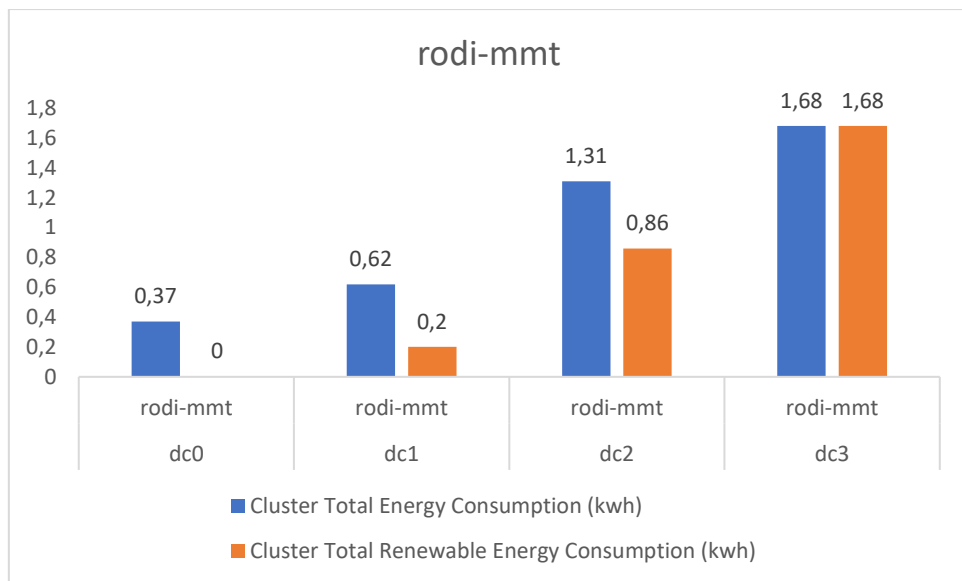


Figura 4-11: Comportamiento del algoritmo RODI en el escenario 1 con 1500 Cloudlets de carga

De nuevo llegamos al comportamiento del algoritmo propuesto en el trabajo [ref.] al que se le ha denominado RODI con MMT. De igual manera, se ve que en este algoritmo el consumo de energía se concentra en los tres Data Centers que disponen de energía renovable en su alimentación, registrando consumos respectivos de energía total y renovable en los Data Centers 1, 2 y 3 de 0,62 kwh/0,2kwh renovable, 1,31 kwh/0,86 kwh renovable y donde el mayor registro de energía de todos se concentraba en el Data Center 3 que contenía el mayor porcentaje de energía renovable disponible de todos los Data Centers. Se puede observar de hecho que, de todos los algoritmos analizados, RODI-MMT presenta el mayor consumo de energía renovable de todos los escenarios simulados de todos los algoritmos marcando unos notables 1,68 kwh.

c. Escenario 1 con 3000 Cloudlets de carga

Las figuras: Figura 4-12, Figura 4-13, Figura 4-14, Figura 4-15, Figura 4-16, muestran el consumo de energía en los cuatro Data Centers para todos los algoritmos comparados. Se muestra tanto la energía total consumida durante el lapso de simulación (Que en este caso es de 1h de funcionamiento real) como la energía renovable utilizada en cada uno según la configuración del escenario 1 descrita en el apartado 4.2.2 con una carga de 3000 Cloudlets.

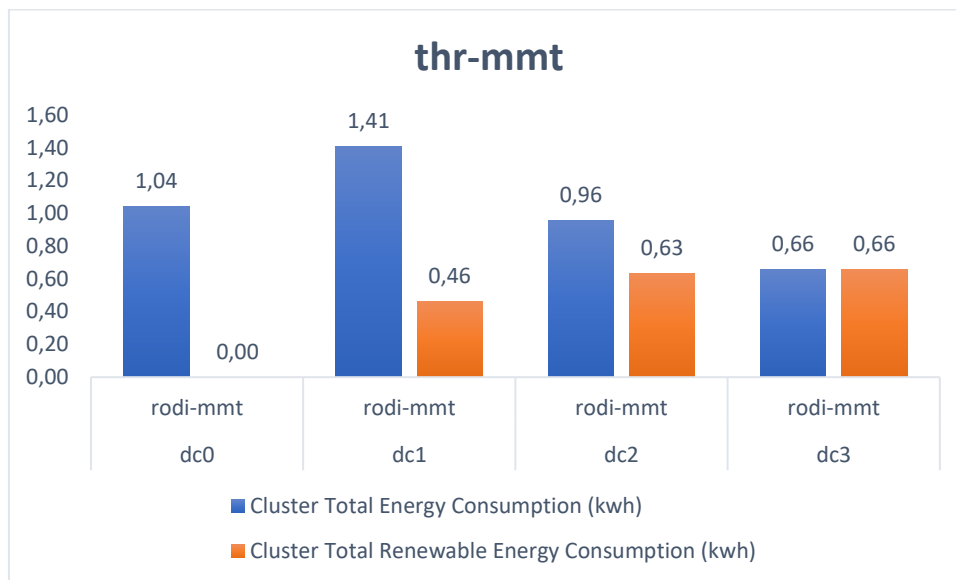


Figura 4-12 Comportamiento del algoritmo THR en el escenario 1 con 3000 Cloudlets de carga

El algoritmo THR con estrategia MMT, se puede ver que en este caso para 3000 Cloudlets de carga, el consumo más importante de energía total se concentra en el dc1 que cuenta con solo el 33% de energía renovable disponible, marcando 1,41 kwh de consumo total y registrando unos 0,46 kwh de proporción renovable. Los dc0 y dc2 registran consumos totales parecidos con la salvedad de que en el dc0 no ha ningún consumo de renovable (al no disponer este de dicho tipo de energía) y en el dc2 se observan unos 0,63 kwh debido a la distribución de renovables disponibles en dicho Data Center y que alcanza el 66%. En cuanto al último Data Center (el dc3), esta marca el menor consumo total de energía tanto en términos totales como en proporción de renovables, registrando 0,66 kwh en términos totales como en renovable.

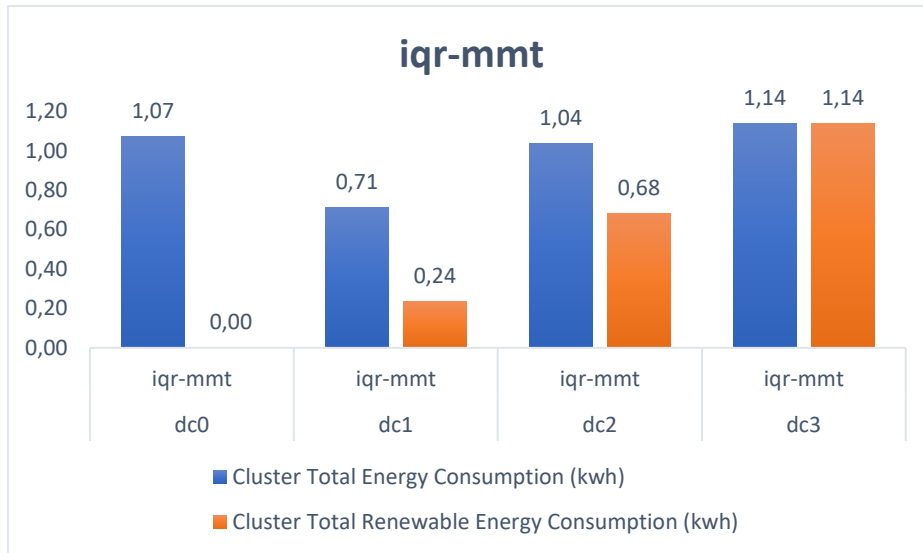


Figura 4-13 Comportamiento del algoritmo IQR en el escenario 1 con 3000 Cloudlets de carga

En este caso para el algoritmo IQR con estrategia MMT se puede observar que el consumo de energía total es muy parecido en todos los Data Centers y que el consumo de la energía renovable es proporcional al porcentaje de esta disponible en cada data center. Se observa entonces un consumo de 1,04 kwh en el dc0, 0,97 kwh totales y 0,32 kwh renovables en el dc1, 1 kwh totales y 0,66 kwh renovables en el dc2 y finalmente 0,99 kwh totales y 0,99 kwh renovables en el dc3.

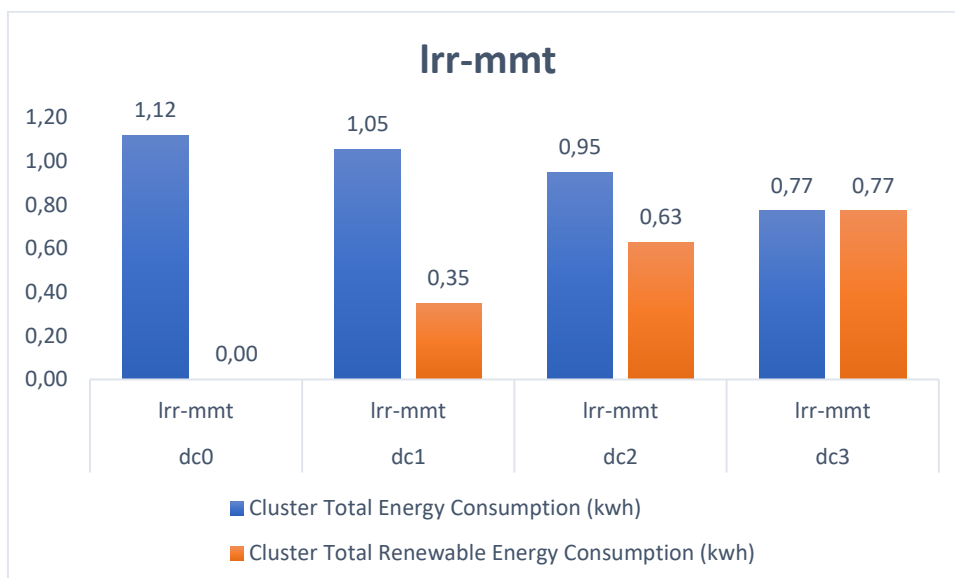


Figura 4-14 Comportamiento del algoritmo LRR en el escenario 1 con 3000 Cloudlets de carga

Para el algoritmo LRR con estrategia MMT y para una carga de 3000 Cloudlets, se puede observar esta vez una distribución de consumo algo diferente de lo que se mostraba en el caso anterior de 1500 Cloudlets. En este caso, se observa un menor consumo en el dc2 que registra 0,87 kwh totales donde 0,57 kwh son renovables, pero también un mayor consumo en el dc3 alcanzando 1,08 kwh totales donde la totalidad es renovable. En cuanto al dc0 y 1, sí que muestran un comportamiento parecido al caso anterior con 1,03 kwh en el primero y 1,11 kwh y 0,37 kwh renovables en el segundo.

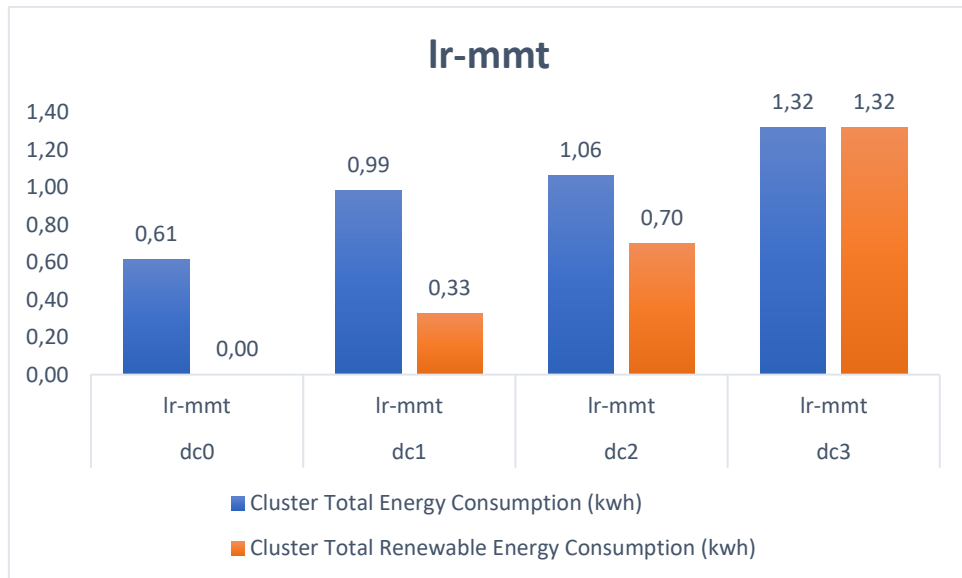


Figura 4-15 Comportamiento del algoritmo LR en el escenario 1 con 3000 Cloudlets de carga

El algoritmo LR con MMT con 3000 Cloudlets muestra un comportamiento bastante diferente que en el escenario anterior. Se observa un aumento lineal del consumo de energía que va de los 0,61 kwh del dc0, pasando por los 0,99 kwh / 0,33 kwh renovables del dc1 y los 1,06 kwh / 0,7 kwh renovables del dc2 y finalmente llegando a los 1,32 kwh / 1,32 kwh renovable del dc3.

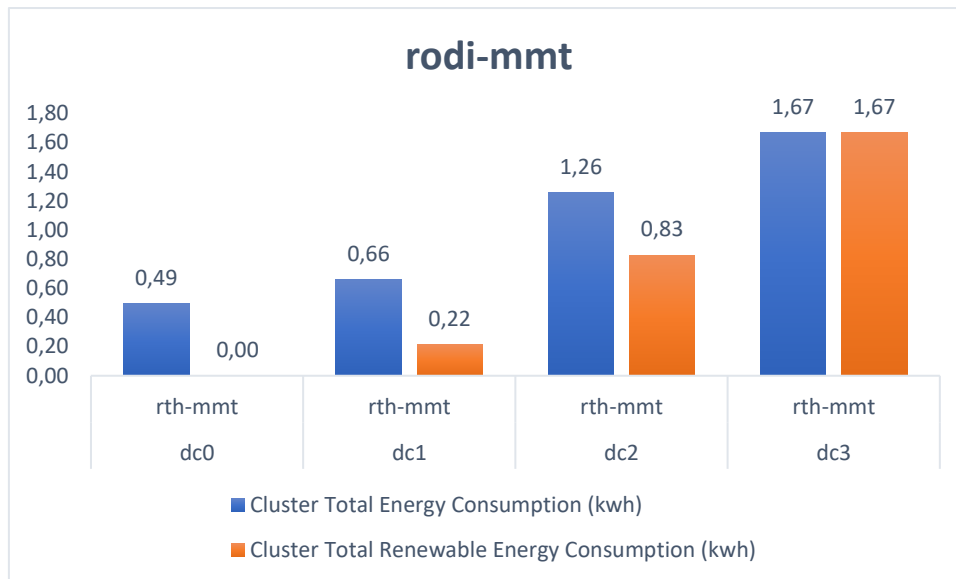


Figura 4-16 Comportamiento del algoritmo RODI en el escenario 1 con 3000 Cloudlets de carga

Finalmente tenemos el algoritmo RODI con estrategia MMT que sigue mostrando un comportamiento muy similar con 3000 Cloudlets a los escenarios de carga precedentes. Se puede ver claramente la tendencia lineal ascendente de consumo en los Data Centers del que tiene menor renovable al que más tiene. Se ve que el dc0 registra 0,37 kwh, el dc1 tiene 0,61 kwh totales y 0,2 kwh renovables, el dc2 marca 1,31 kwh totales de los cuales 0,86 kwh son renovables y finalmente se recoge el mayor consumo en renovables en el dc3 donde se marcan 1,68 kwh.

d. Escenario 2 con 1000 Cloudlets de carga

De nuevo, las figuras : Figura 4-17, Figura 4-18, Figura 4-19, Figura 4-20, Figura 4-21, muestran el consumo de energía en los cuatro Data Centers para todos los algoritmos comparados. Se muestra tanto la energía total consumida durante el lapso de simulación (donde se continúa con 1h de funcionamiento real) como la energía renovable utilizada en cada uno según la configuración del escenario 2 descrita en el apartado 4.2.2 con una carga de 1000 Cloudlets.

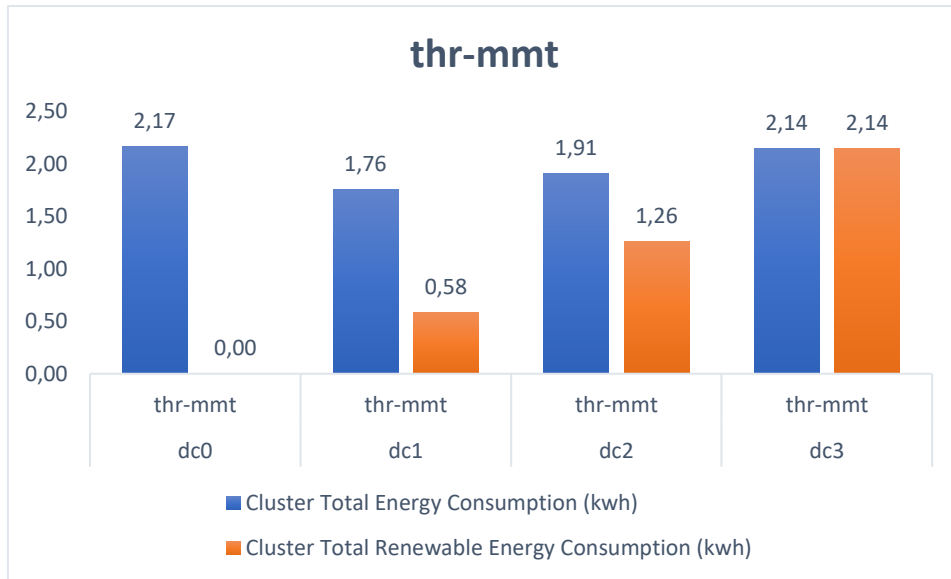


Figura 4-17 Comportamiento del algoritmo THR en el escenario 2 con 1000 Cloudlets de carga

Se empiezan a notar consumos más importantes en este segundo escenario de simulaciones donde tanto el número de hosts como de máquinas virtuales prácticamente se ha duplicado. Con esto vemos que para el algoritmo THR con estrategia MMT se observa un comportamiento diferente al observado en las simulaciones precedentes. Esta vez se observa un comportamiento casi uniforme de consumo de energía sobre todos los Data Centers, donde en los dc0, 2 y 3 se alcanzan 2,17 kwh, 1,91 kwh/1,6 kwh renovables y 2,14 kwh/2,14 kwh renovables respectivamente. El dc1 es que muestra menos consumo de los cuatro, registrando 1,76 kwh/0,58 kwh de renovables.

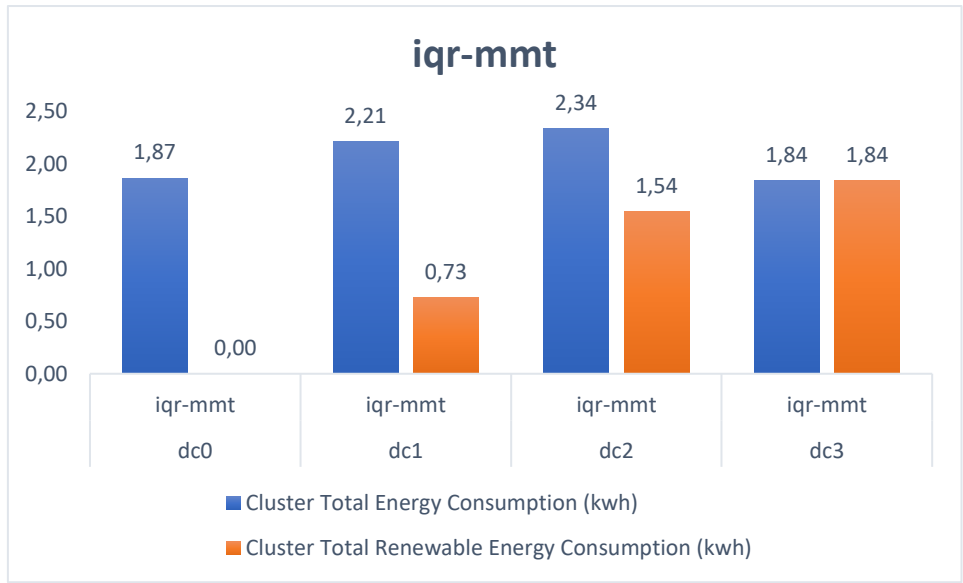


Figura 4-18 Comportamiento del algoritmo IQR en el escenario 2 con 1000 Cloudlets de carga

Se sigue observando el aumento del consumo en los Data Centers debido al aumento de la configuración de simulación. El algoritmo IQR con estrategia MMT presenta en este escenario de simulación un comportamiento diferente al observado en los anteriores escenarios. Los mayores registros de consumo se ven en los dc1 y dc2, marcando 2,21 kwh/0,73 kwh renovables y 2,34 kwh / 1,54 kwh de renovables respectivamente. Los Data Centers dc0 y dc 3 presentan unos consumos menores que los anteriores marcando 1,87 kwh y 1,84 kwh/1,84kwh de renovables respectivamente.

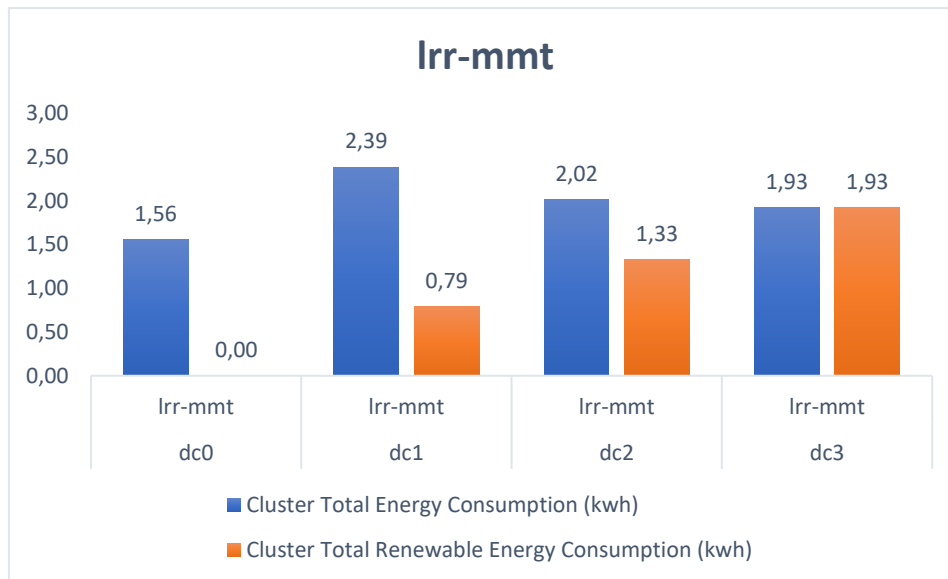


Figura 4-19 Comportamiento del algoritmo LRR en el escenario 2 con 1000 Cloudlets de carga

Continúa la dinámica anterior en términos de aumento de consumo para el algoritmo LRR con estrategia MMT para la carga de 1000 Cloudlets en el escenario 2, pero también se puede observar que el comportamiento del algoritmo sigue mostrando un patrón aleatorio en lo que respecta al consumo en los Data Centers. Se ve que ahora el consumo más importante de energía se registra en el dc1 con 2,39 kwh/0,79 kwh de renovables. Los dc2 y dc3 marcan valores similares de 2,02 kwh/1,33 kwh de renovables y 1,93 kwh/1,93 kwh de renovables respectivamente. El consumo más pequeño se registra en el dc0 con 1,56 kwh.

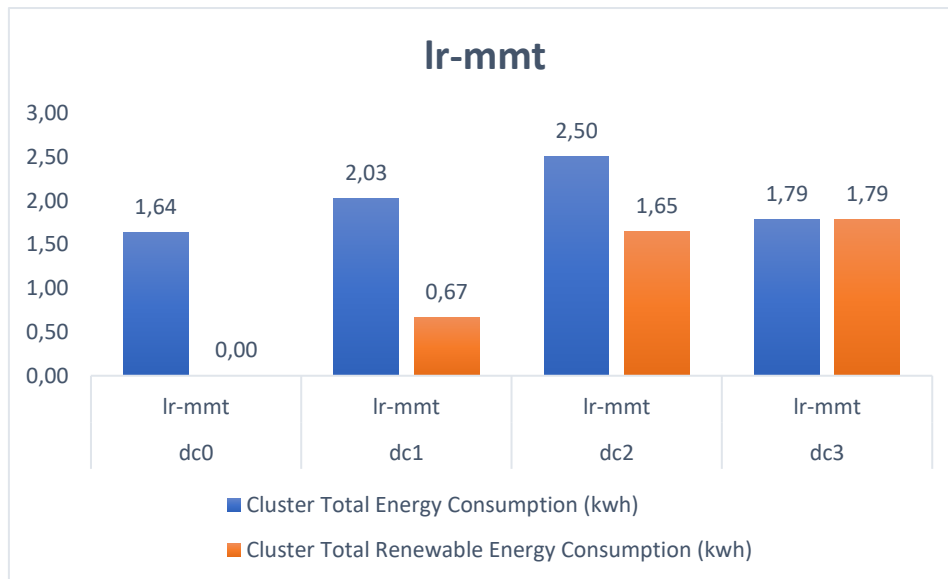


Figura 4-20 Comportamiento del algoritmo LR en el escenario 2 con 1000 Cloudlets de carga

El algoritmo LR con MMT con 1000 Cloudlets en el segundo escenario confirma el aumento generalizado de los valores de consumo en este escenario y de nuevo muestra un comportamiento aleatorio en términos de consumo total y renovable en los diferentes Data Centers. Vemos esta vez que el dc2 muestra el mayor consumo de energía registrando 2,50 kwh/1,65kwh de renovables. Mientras los Data Centers dc1 y dc3 muestran menores consumos donde marcan 2,03 kwh/0,67kwh de renovables y 1,79 kwh/1,79kwh de renovables respectivamente. El menor consumo se registra en el dc0 con 1,64 kwh.

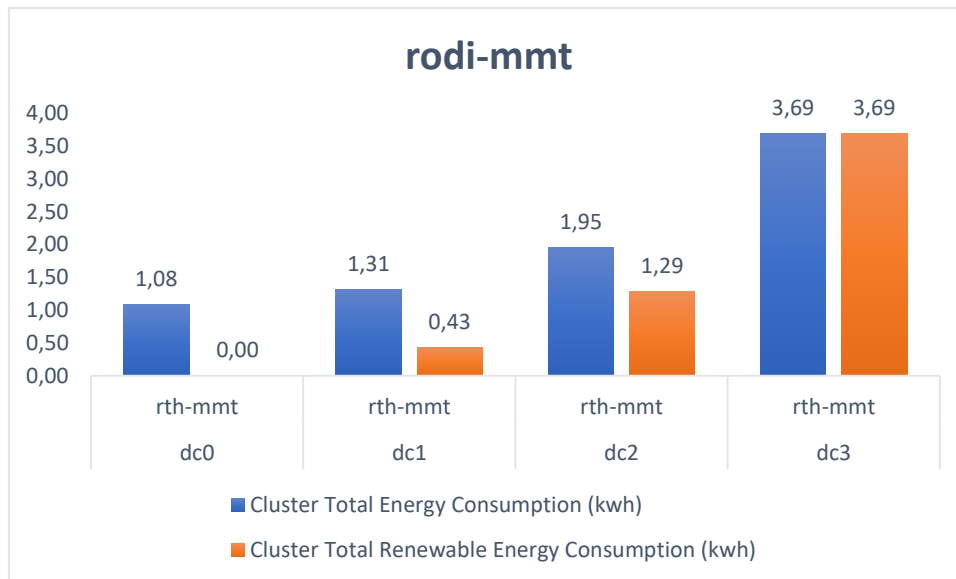


Figura 4-21 Comportamiento del algoritmo RODI en el escenario 2 con 1000 Cloudlets de carga

El algoritmo RODI con estrategia MMT en el escenario 2 con 1000 Cloudlets sigue su desempeño coherente en relación con la distribución del consumo de energía tanto global como renovable; Seguimos viendo un aumento lineal del consumo de energía de forma proporcional al porcentaje de renovable que el Data Center tiene, marcando los resultado más bajos en el dc0 con 1,08 kwh, pasando por el dc1 y dc2 que registran 1,31 kwh/0,43kwh de renovables y 1,95 kwh/1,29kwh de renovables respectivamente y finalizando por le dc3 que marca el mayor consumo de energía donde se registran 3,69 kwh/3,69kwh de renovables.

e. Escenario 2 con 2000 Cloudlets de carga

Las figuras : Figura 4-22, Figura 4-23, Figura 4-24, Figura 4-25, Figura 4-26, muestran el consumo de energía en los cuatro Data Centers para todos los algoritmos comparados. Se muestra tanto la energía total consumida durante el lapso de simulación (donde se continua con 1h de funcionamiento real) como la energía renovable utilizada en cada uno según la configuración del escenario 2 descrita en el apartado 4.2.2 con una carga de 2000 Cloudlets.

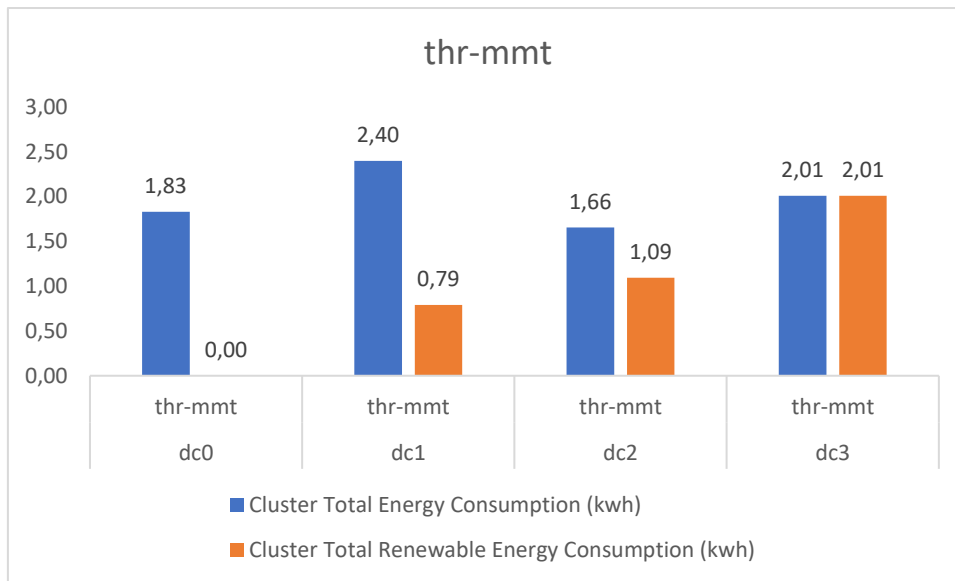


Figura 4-22 Comportamiento del algoritmo THR en el escenario 2 con 2000 Cloudlets de carga

Se sigue mostrando un comportamiento aleatorio en cuanto al consumo de energía en los diferentes Data centers. Esta vez para el algoritmo THR con estrategia MMT se registra el mayor consumo de energía en el dc1 que tiene un 33% de energía renovable disponible, marcando 2,40 kwh/0,79 kwh renovables, seguido por el dc3 que tiene un 100% y que registra a su vez 2,01 kwh/2,01 kwh de renovables. El tercer es el dc0 con 1,83 kwh y finalmente el que registra menos consumo es el dc2 con 1,66 kwh/1,09 kwh de renovables.

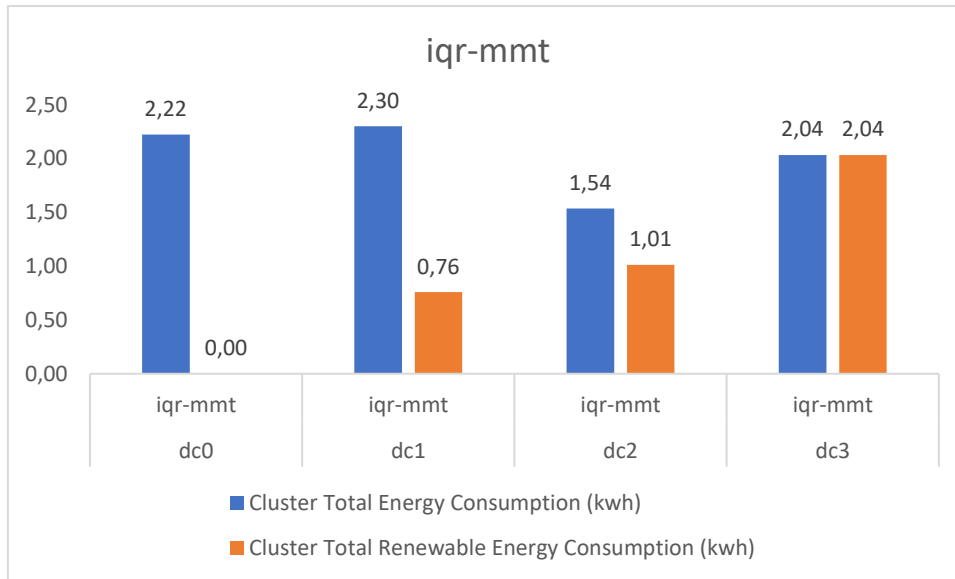


Figura 4-23 Comportamiento del algoritmo IQR en el escenario 2 con 2000 Cloudlets de carga

El algoritmo IQR con estrategia MMT sigue presentando el mismo comportamiento aleatorio que se ha observado hasta ahora. Esta vez es el dc1 el que presenta mayor consumo de energía registrando 2,30 kwh/0,76 kwh de renovables. Luego es el dc0 el que va en segundo lugar con 2,22 kwh y después van el dc3 y dc2 con 2,04 kwh/2,04 kwh de renovables y 1,54 kwh/1,01 kwh de renovables respectivamente.

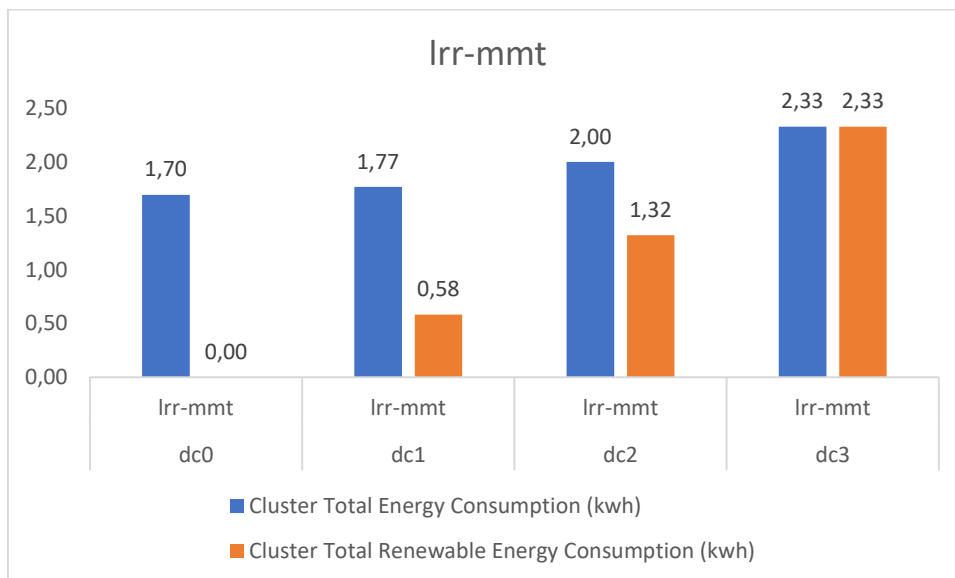


Figura 4-24 Comportamiento del algoritmo LRR en el escenario 2 con 2000 Cloudlets de carga

El algoritmo LRR con estrategia MMT para la carga de 2000 Cloudlets en el escenario 2, Muestra en este caso un comportamiento más lineal donde se registran los menores consumos en el dc0 marcando 1,70 kwh, pasando por el dc1 y dc2 que registran 1,77 kwh/0,58 kwh de renovables y 2,00 kwh/1,32 kwh de renovables respectivamente. Finalmente se observa el mayor consumo en el dc3 donde se marcan 2,33 kwh/2,33 kwh de renovables.

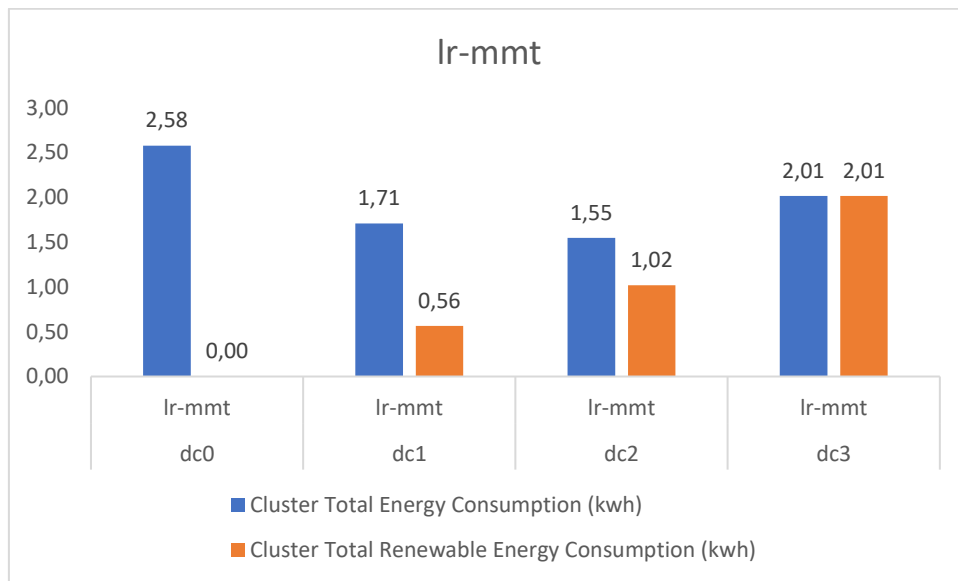


Figura 4-25 Comportamiento del algoritmo LR en el escenario 2 con 2000 Cloudlets de carga

El algoritmo LR con MMT vuelve a mostrar su comportamiento aleatorio en relación con el consumo en los diferentes Data Centers. Esta vez el mayor consumo se registra en el dc0 con 2,58 kwh de consumo total. El segundo mayor consumo se observa en el dc3 con 2,01 kwh/2,01 kwh de renovables. El tercero es el dc1 con 1,71 kwh/0,56 kwh de renovables y en la cola está el dc2 con 1,55 kwh/1,02 kwh de renovables.

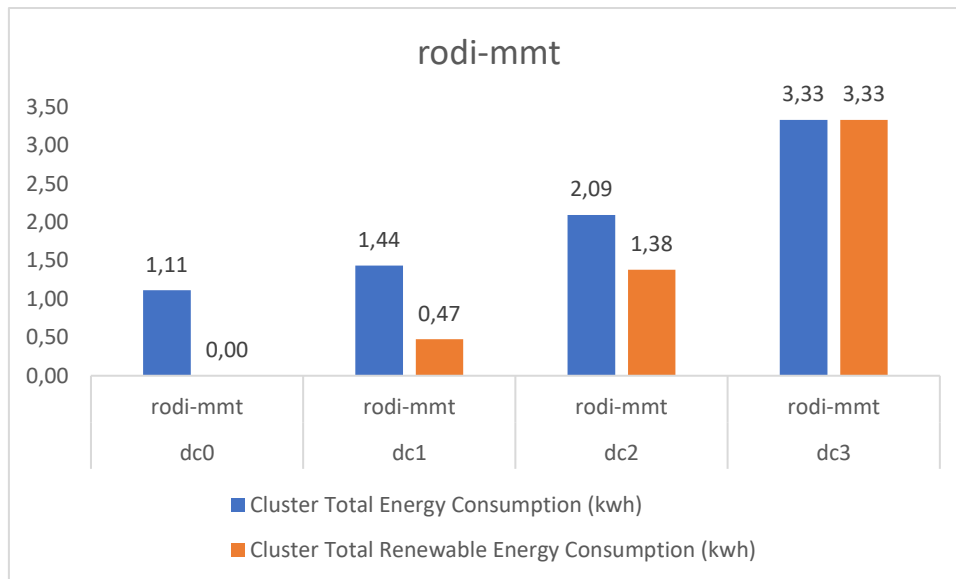


Figura 4-26 Comportamiento del algoritmo RODI en el escenario 2 con 2000 Cloudlets de carga

De nuevo el algoritmo RODI con estrategia MMT en el escenario 2 con 2000 Cloudlets sigue el desempeño coherente en relación con la distribución del consumo de energía tanto global como renovable; Se sigue viendo un aumento lineal del consumo de energía de forma proporcional al porcentaje de renovable que el Data Center tiene, marcando los resultado más bajos en el dc0 con 1,11 kwh, pasando por el dc1 y dc2 que registran 1,44 kwh/0,47kwh de renovables y 2,09 kwh/1,38kwh de renovables respectivamente y finalizando por le dc3 que marca el mayor consumo de energía donde se registran 3,33 kwh/3,33kwh de renovables.

f. Escenario 2 con 5000 Cloudlets de carga

Las figuras : Figura 4-27, Figura 4-28, Figura 4-29, Figura 4-30, Figura 4-31, muestran el consumo de energía en los cuatro Data Centers para todos los algoritmos comparados. Se muestra tanto la energía total consumida durante el lapso de simulación (donde se continua con 1h de funcionamiento real) como la energía renovable utilizada en cada uno según la configuración del escenario 2 descrita en el apartado 4.2.2 con una carga de 5000 Cloudlets.

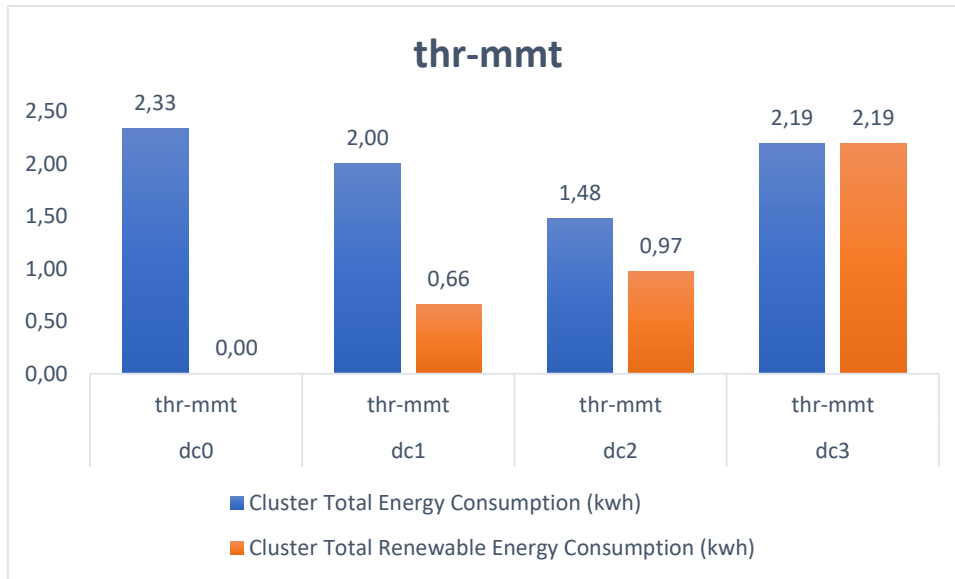


Figura 4-27 Comportamiento del algoritmo THR en el escenario 2 con 5000 Cloudlets de carga

Mismo comportamiento aleatorio en cuanto al consumo de energía en los diferentes Data centers. Esta vez para el algoritmo THR con estrategia MMT se registra el mayor consumo de energía en el dc0 marcando 2,33 kwh, seguido por el dc3 que tiene un 100% y que registra a su vez 2,19 kwh/2,19 kwh de renovables. El tercer es el dc1 con 2,00 kwh/ 0,66 kwh de renovables y finalmente el que registra menor consumo es el dc2 con 1,48 kwh/0,97 kwh de renovables.

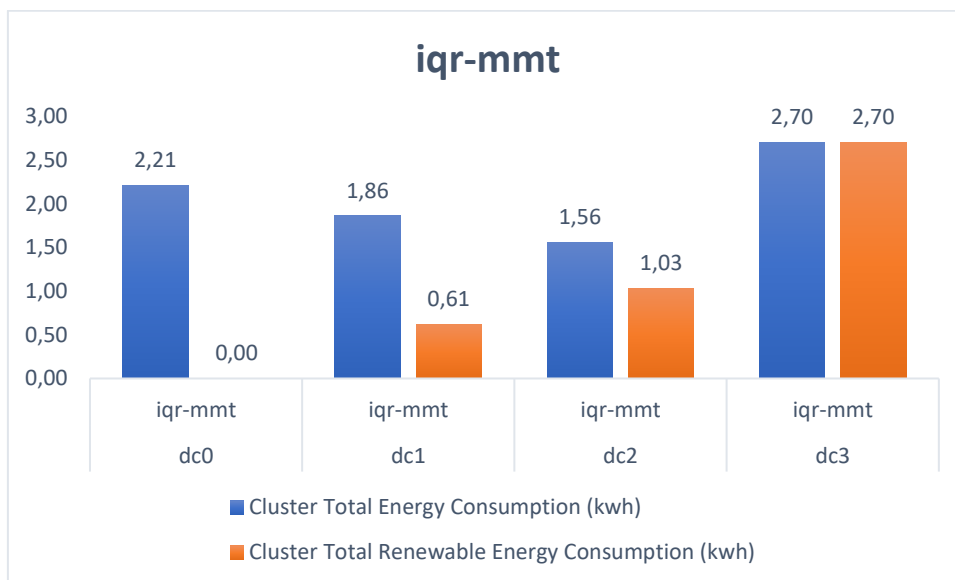


Figura 4-28 Comportamiento del algoritmo IQR en el escenario 2 con 5000 Cloudlets de carga

El algoritmo IQR con estrategia MMT sigue presentando un comportamiento aleatorio parecido al observado hasta ahora. Esta vez es el dc3 el que presenta mayor consumo de energía registrando 2,70 kwh/2,70 kwh de renovables. Luego es el dc0 el que va en segundo lugar con 2,21 kwh y después van el dc1 y dc2 con 1,86 kwh/0,61 kwh de renovables y 1,56 kwh/1,03 kwh de renovables respectivamente.

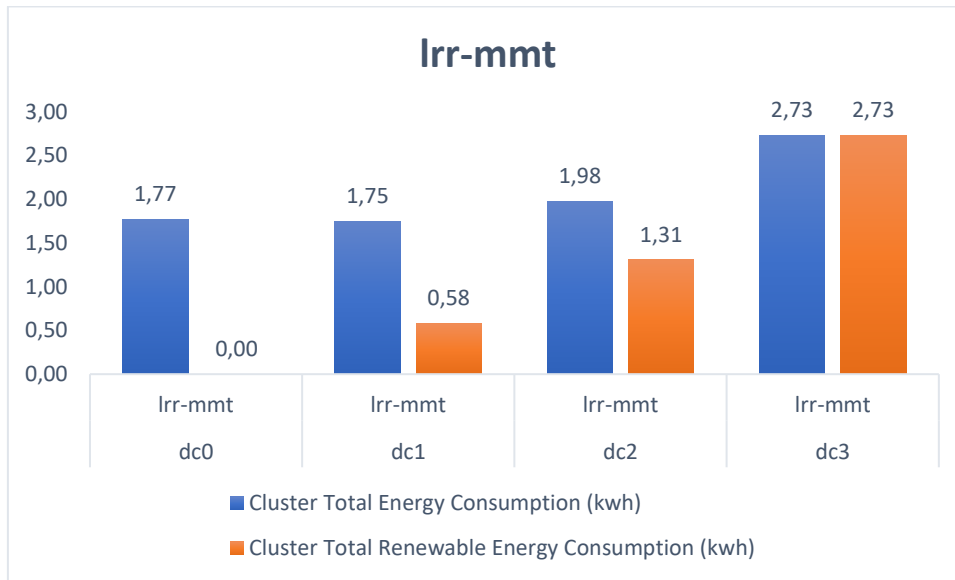


Figura 4-29 Comportamiento del algoritmo LRR en el escenario 2 con 5000 Cloudlets de carga

El algoritmo LRR con estrategia MMT modifica su comportamiento de nuevo en este escenario, donde se registra el mayor consumo de energía en el dc3 donde se marcan 2,73 kwh/2,73 kwh de renovables, seguido por el dc2 con 1,98 kwh/1,31 kwh de renovables y por el dc0 donde se registran unos 1,70 kwh y donde finalmente se registran unos 1,75 kwh/0,58 kwh de renovables en el Data Center de menor consumo.

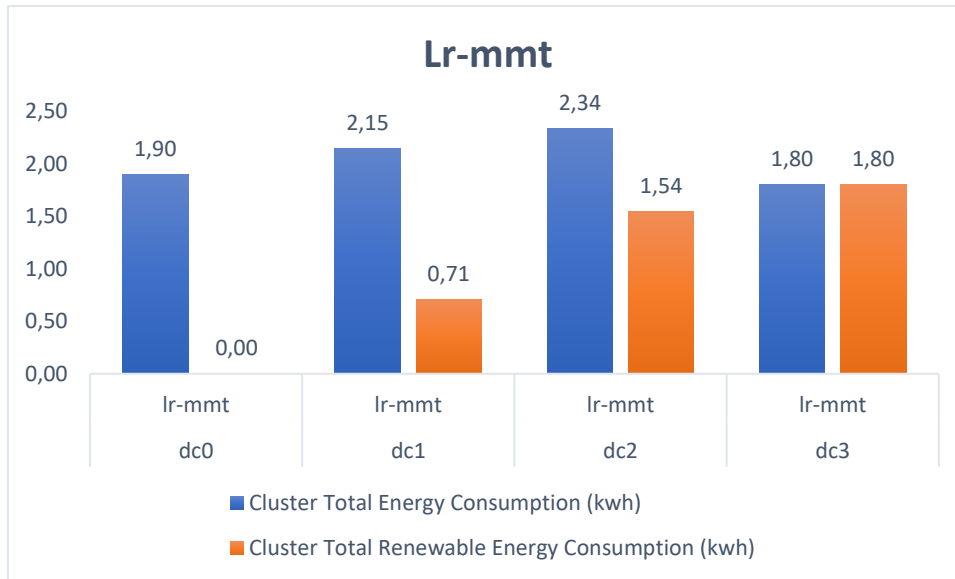


Figura 4-30 Comportamiento del algoritmo LR en el escenario 2 con 5000 Cloudlets de carga

El algoritmo LR con MMT continúa mostrando su comportamiento aleatorio en relación con la energía consumido en los diferentes Data Centers. En este caso se marca un máximo en el dc2 con 2,34 kwh/1,54 kwh de renovables. En segundo lugar, está el dc1 con 2,15 kwh/0,71 kwh de renovables, seguidе del dc0 con 1,90 kwh y finalmente llega el dc3 con 1,80 kwh/1,80 kwh de renovables.

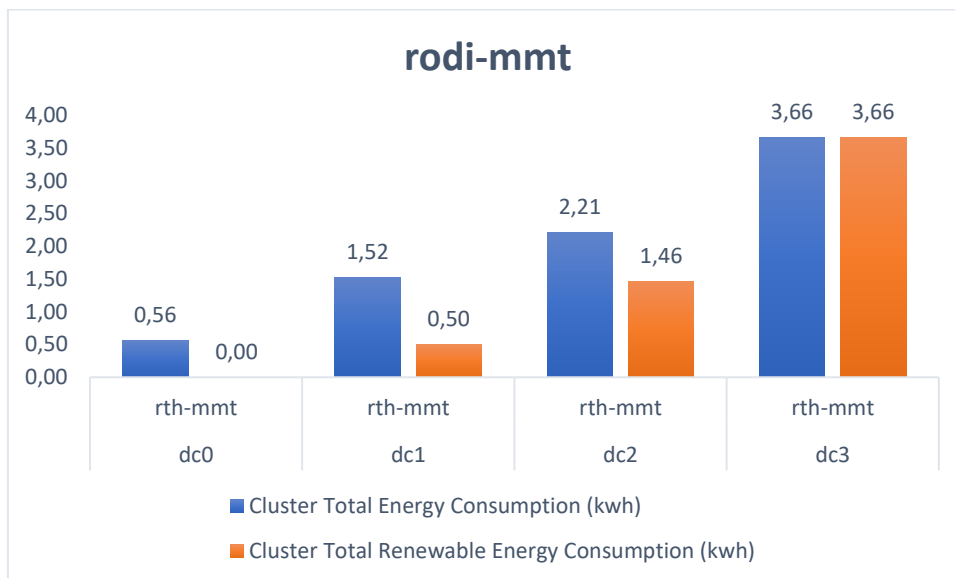


Figura 4-31 Comportamiento del algoritmo RODI en el escenario 2 con 5000 Cloudlets de carga

El algoritmo RODI con estrategia MMT continúa mostrando un comportamiento parecido al observado hasta ahora, marcando un consumo que aumenta proporcionalmente a la disponibilidad de la energía renovable en los Data Centers. Vemos en este escenario que el consumo más importante se registra en el dc3 con 3,66 kwh/3,66 kwh de renovables, después viene el dc2 que marca 2,21 kwh/1,46 kwh de renovables, seguido por el dc1 que registra 1,52 kwh/0,50 kwh de renovables y en último lugar y registrando el menor consumo de energía se encuentra el dc0 con 0,56 kwh.

g. Escenario 3 con 1500 Cloudlets de carga

Las figuras : Figura 4-32, Figura 4-33, Figura 4-34, Figura 4-35, Figura 4-36, muestran el consumo de energía en los cuatro Data Centers para todos los algoritmos comparados. Se muestra tanto la energía total consumida durante el lapso de simulación (donde se continua con 1h de funcionamiento real) como la energía renovable utilizada en cada uno según la configuración del escenario 3 descrita en el apartado 4.2.2 con una carga de 1500 Cloudlets.

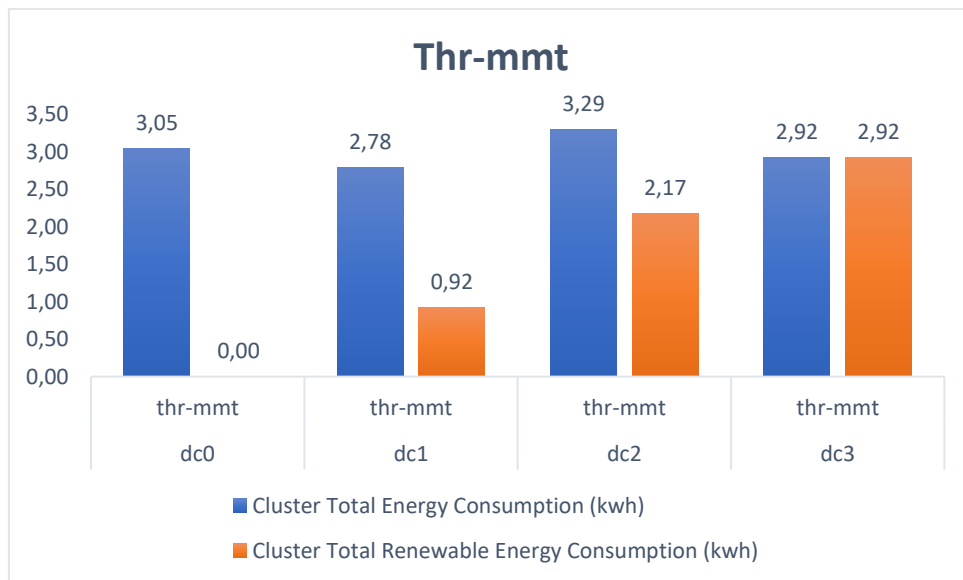


Figura 4-32 Comportamiento del algoritmo THR en el escenario 3 con 1500 Cloudlets de carga

De nuevo cambia el comportamiento del algoritmo THR con estrategia MMT en relación a la energía consumida en los cuatro Data Centers Continuando con la dinámica

observada hasta ahora. En este caso observamos que el mayor consumo de energía se produce en el dc2 con 3,29 kwh/2,17 kwh de renovables, seguido por el dc0 que marca 3,05 kwh, luego va el dc3 con 2,92 kwh/2,92 kwh de renovables y finalmente y en cuarto lugar llega el dc1 que registra 2,78 kwh/0,92 kwh de renovables.

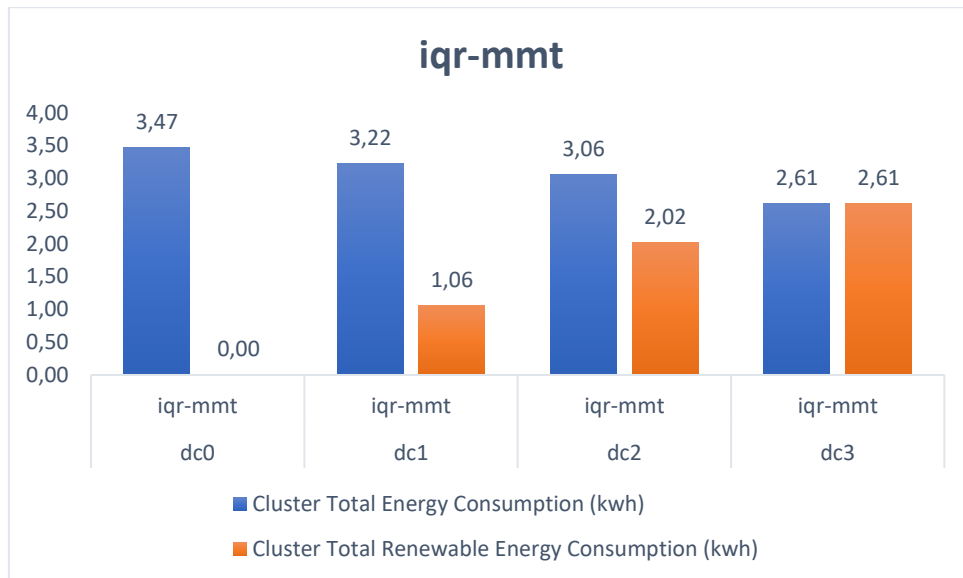


Figura 4-33 Comportamiento del algoritmo IQR en el escenario 3 con 1500 Cloudlets de carga

El algoritmo IQR con estrategia MMT presenta de nuevo un comportamiento diferente con relación al consumo de energía. Vemos que el consumo más importante se registra en el dc0 donde se marcan 3,47 kwh, seguido por el dc1 que marca 3,22 kwh/1,06 kwh de renovables, luego va el dc2 que registra 3,06 kwh/2,02 kwh de renovables y finalmente y con el menor consumo de todos viene el dc3 que registra 2,61 kwh/2,61 kwh de renovables.

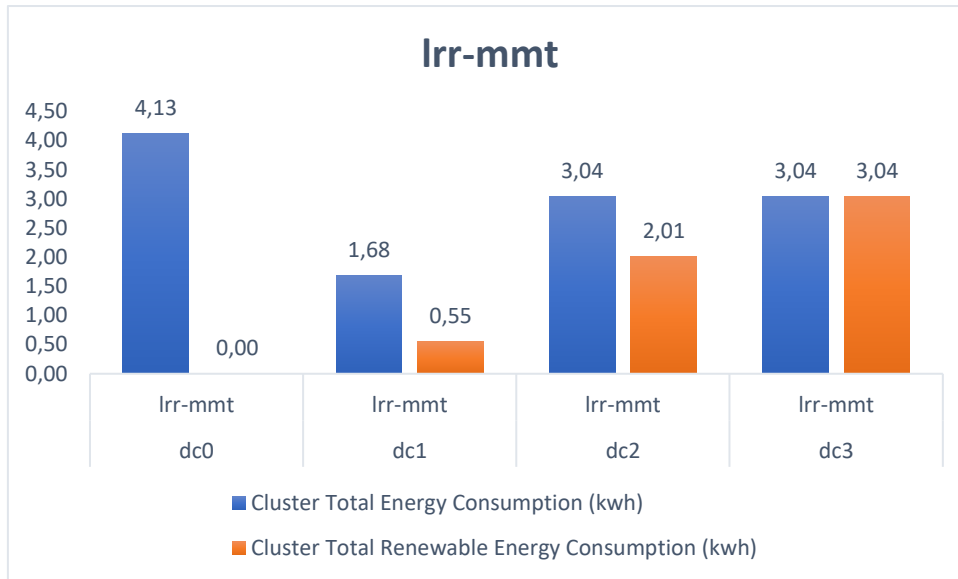


Figura 4-34 Comportamiento del algoritmo LRR en el escenario 3 con 1500 Cloudlets de carga

El algoritmo LRR con estrategia MMT continúa mostrando resultados aleatorios en términos de consumos de energía por los diferentes Data Centers. Esta vez se observa que el consumo más importante se registra en el dc0 donde se marcan 4,13 kwh, seguido por el dc2 que marca 3,04 kwh/2,01 kwh de renovables, luego va el dc3 que registra 3,04 kwh/3,04 kwh de renovables y finalmente y con el menor consumo de todos viene el dc1 que registra 1,75 kwh/0,58 kwh de renovables.

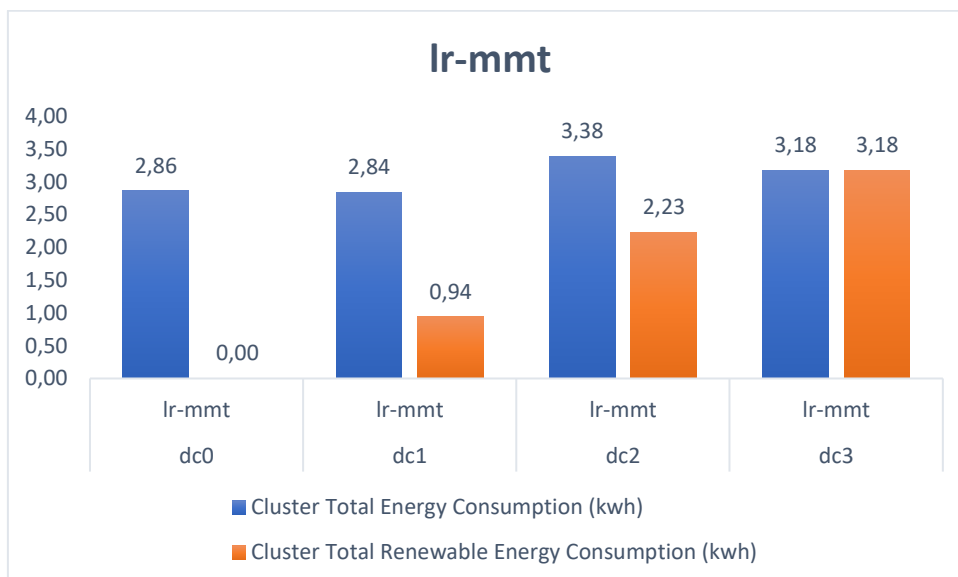


Figura 4-35 Comportamiento del algoritmo LR en el escenario 3 con 1500 Cloudlets de carga

El algoritmo LR con MMT muestra de nuevo unos resultados diferentes a las pasadas simulaciones. Se puede ver que el consumo más importante se registra en el dc2 con 3,38 kwh/2,23 kwh de renovables, seguido por el dc3 que marca 3,18 kwh/3,18 kwh de renovables, luego va el dc0 que registra 2,86 kwh y finalmente y con el menor consumo de todos viene el dc1 que registra 2,84 kwh/0,94 kwh de renovables.

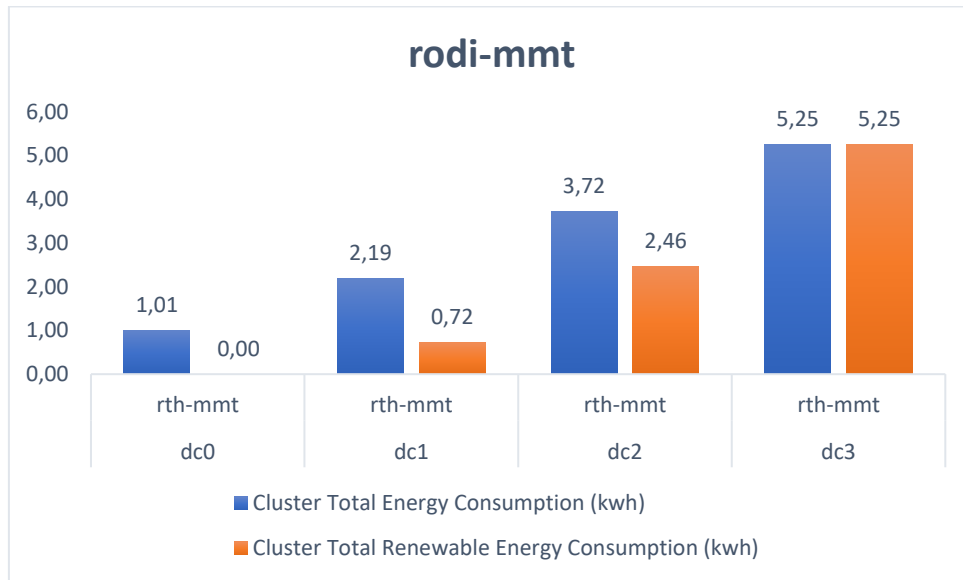


Figura 4-36 Comportamiento del algoritmo RODI en el escenario 3 con 1500 Cloudlets de carga

Se sigue consolidando el comportamiento del RODI con estrategia MMT en este tercer escenario con 1500 Cloudlets. Se puede observar que el consumo más importante se registra en el dc3 con 5,25 kwh/5,25 kwh de renovables, después viene el dc2 que marca 3,72 kwh/2,46 kwh de renovables, seguido por el dc1 que registra 2,19 kwh/0,72 kwh de renovables y en último lugar y registrando el menor consumo de energía se encuentra el dc0 con 1,01 kwh.

h. Escenario 3 con 5000 Cloudlets de carga

Las figuras : Figura 4-37, Figura 4-38, Figura 4-39, Figura 4-40, Figura 4-41, muestran el consumo de energía en los cuatro Data Centers para todos los algoritmos comparados. Se muestra tanto la energía total consumida durante el lapso de simulación (donde se continua con 1h de funcionamiento real) como la energía renovable utilizada

en cada uno según la configuración del escenario 3 descrita en el apartado 4.2.2 con una carga de 5000 Cloudlets.

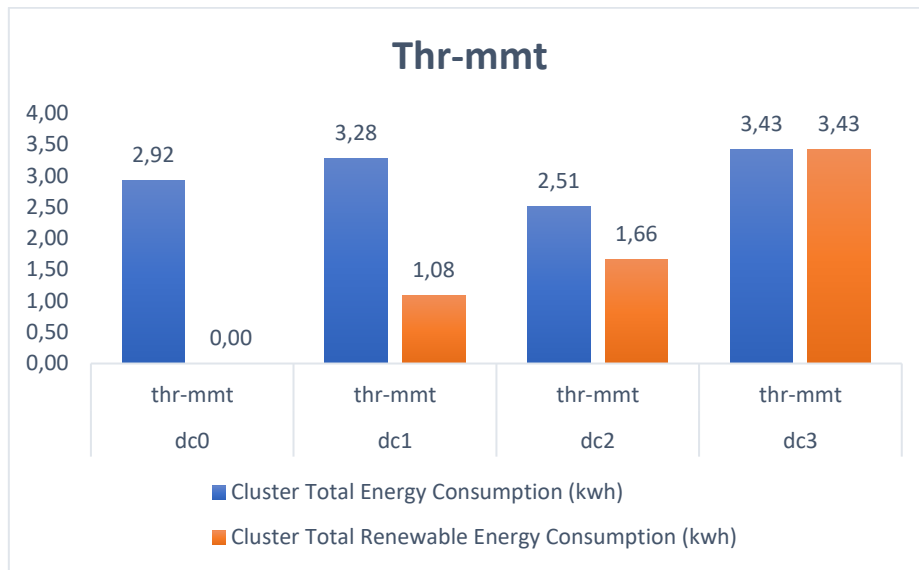


Figura 4-37 Comportamiento del algoritmo THR en el escenario 3 con 5000 Cloudlets de carga

El comportamiento del algoritmo THR con estrategia MMT de la figura anterior muestra que en este caso que el Data Center con mayor consumo de energía es el dc3 que marca 3,43 kwh/3,43 kwh de renovables, seguido por el dc1 con 3,28 kwh/1,08 kwh de renovables, luego va el dc0 con 2,98 kwh y en último lugar viene el dc1 con 3,28 kwh/1,08 kwh de renovables.

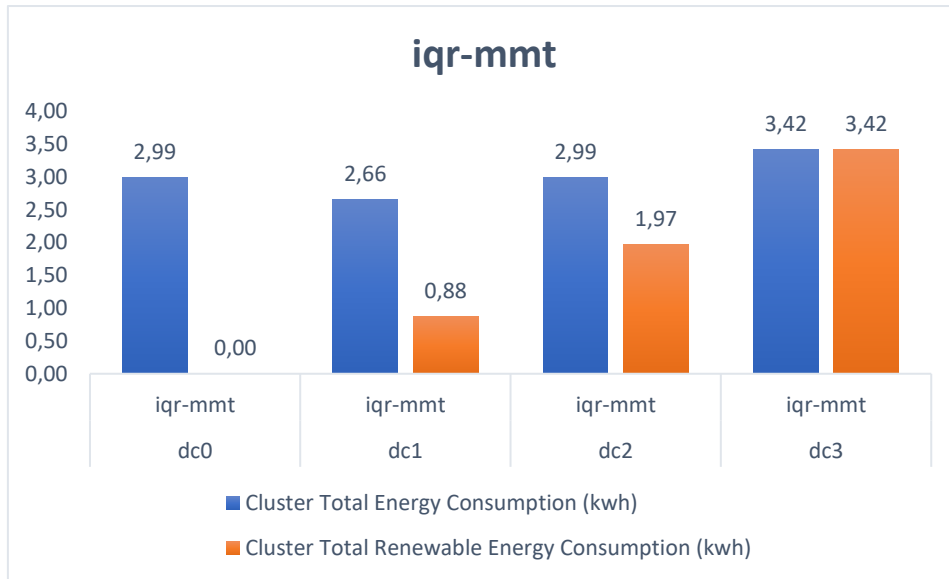


Figura 4-38 Comportamiento del algoritmo IQR en el escenario 3 con 5000 Cloudlets de carga

Para el algoritmo IQR con estrategia MMT que se muestra en la figura anterior podemos ver que esta vez el consumo más importante se registra en el dc3 marcando 3,42 kwh/3,42 kwh de renovables, seguido por los dc0 y dc2 que registran 2,99 kwh y 2,99 kwh/1,97 kwh de renovables respectivamente. Finalmente, y con el menor consumo de todos está el dc1 que marca 2,66 kwh/0,88 kwh de renovables

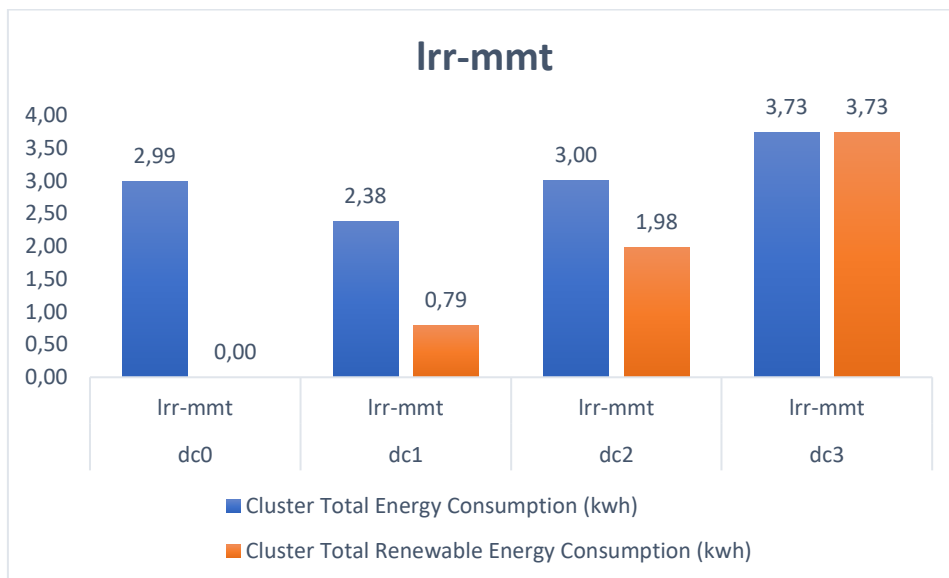


Figura 4-39 Comportamiento del algoritmo LRR en el escenario 3 con 5000 Cloudlets de carga

El algoritmo LRR con estrategia MMT muestra esta vez que el consumo más importante se registra en el dc3 marcando 3,73 kwh/3,73 kwh de renovables, seguido por los dc0 y dc2 que registran 2,99 kwh y 3,00 kwh/1,98 kwh de renovables respectivamente. Finalmente, y con el menor consumo de todos está el dc1 que marca 2,38 kwh/0,79 kwh de renovables

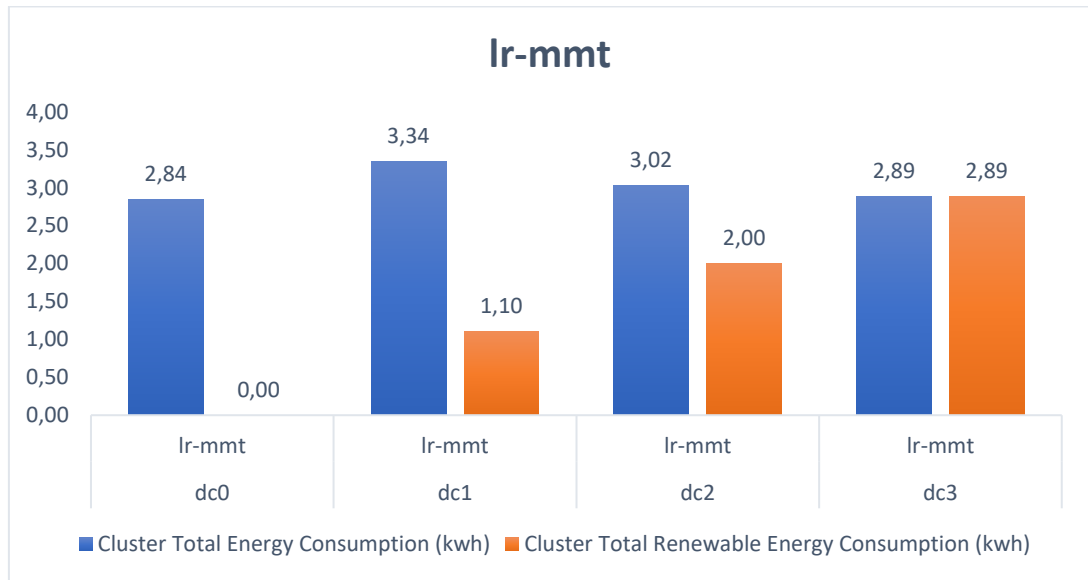


Figura 4-40 Comportamiento del algoritmo LR en el escenario 3 con 5000 Cloudlets de carga

El algoritmo LR con MMT muestra un nuevo orden de consumo de energía en los diferente Data Centers, esta vez es le dc1 que viene en primer lugar marcando 3,34 kwh/1,10 kwh de renovables, seguido por el dc2 3,02 kwh/2,00 kwh de renovables, luego va el dc3 que registra 2,89 kwh/2,89 kwh de renovables y finalmente con el menor consumo de todos está el dc0 que marca 2,84 kwh.

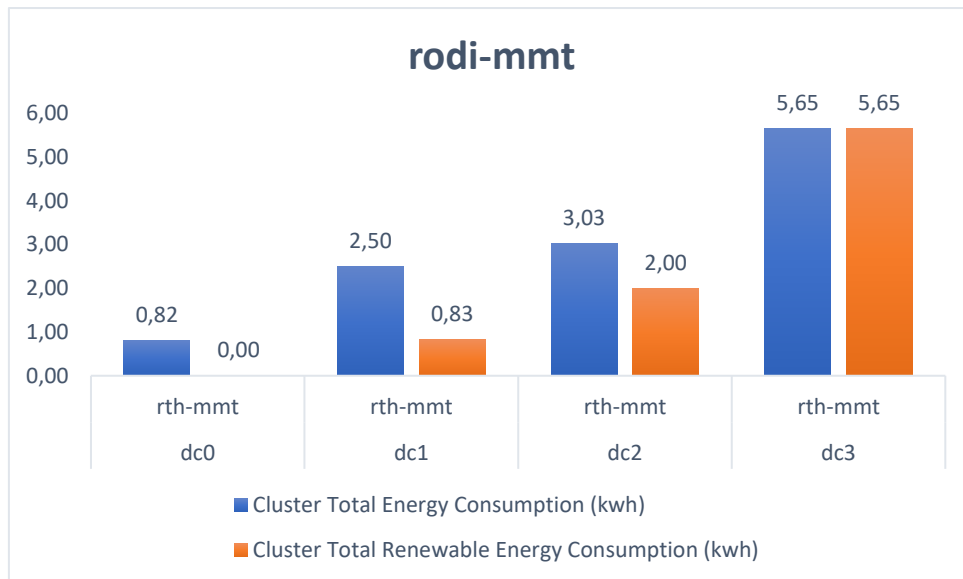


Figura 4-41 Comportamiento del algoritmo RODI en el escenario 3 con 5000 Cloudlets de carga

El comportamiento del algoritmo RODI con estrategia MMT sigue mostrando una coherencia al marcar los valores más importantes de consumo de energía en el dc3 que tiene un 100% de energía renovable asignada al Data Center. Se observa que en el dc3 se registran 5,65 kwh/5,65 kwh de renovables. Después viene el dc2 que marca 3,03 kwh/2,00 kwh de renovables, seguido por el dc1 que registra 2,50 kwh/0,83 kwh de renovables y en último lugar y registrando el menor consumo de energía se encuentra el dc0 con 0,82 kwh.

i. Escenario 3 con 10000 Cloudlets de carga

Las figuras : Figura 4-42, Figura 4-43, Figura 4-44, Figura 4-45, Figura 4-46, muestran el consumo de energía en los cuatro Data Centers para todos los algoritmos comparados. Se muestra tanto la energía total consumida durante el lapso de simulación (donde se continua con 1h de funcionamiento real) como la energía renovable utilizada en cada uno según la configuración del escenario 3 descrita en el apartado 4.2.2 con una carga de 10000 Cloudlets.

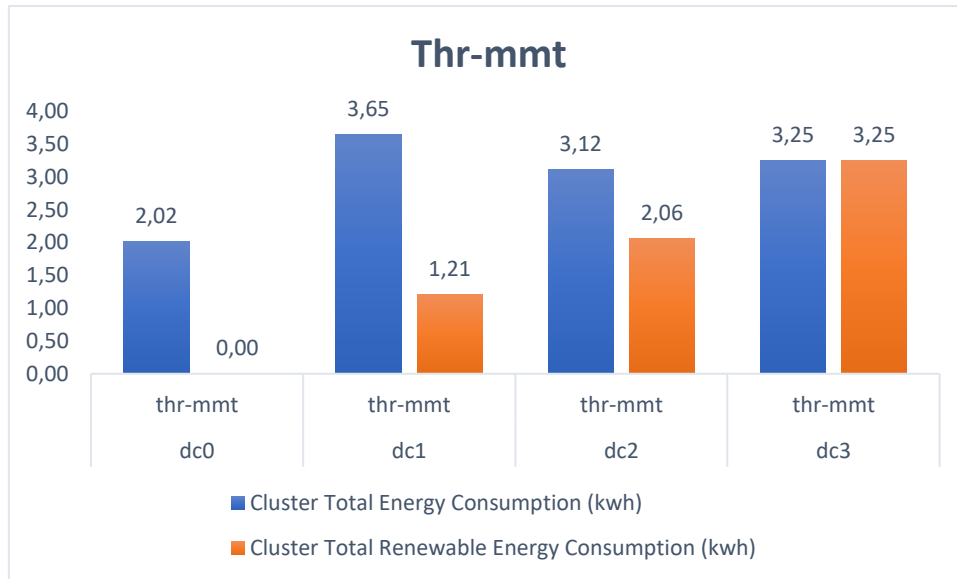


Figura 4-42 Comportamiento del algoritmo THR en el escenario 3 con 10000 Cloudlets de carga

El algoritmo THR con estrategia MMT de la figura muestra en este escenario que el Data Center con mayor consumo de energía es el dc1 que marca 3,65 kwh/1,21 kwh de renovables, seguido por el dc3 con 3,25 kwh/3,25 kwh de renovables, luego va el dc2 con 3,12 kwh/2,06 de renovables y en último lugar está el dc0 con 2,02 kwh.

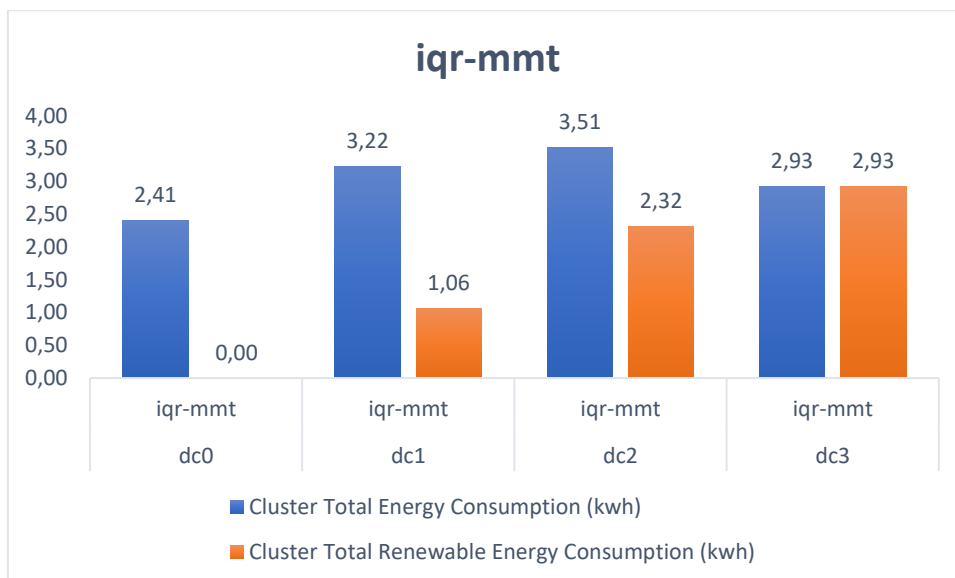


Figura 4-43 Comportamiento del algoritmo IQR en el escenario 3 con 10000 Cloudlets de carga

El algoritmo IQR con estrategia MMT que se muestra en la figura anterior muestra esta vez que el consumo más importante se registra en el dc2 marcando 3,51 kwh/2,32

kwh de renovables, seguido por los dc1 que registra 3,22 kwh/1,06 kwh de renovables, luego va el dc3 que marca 2,93 kwh/2,93 kwh de renovables y finalmente con el menor consumo de todos está el dc0 que registra 2,41 kwh.

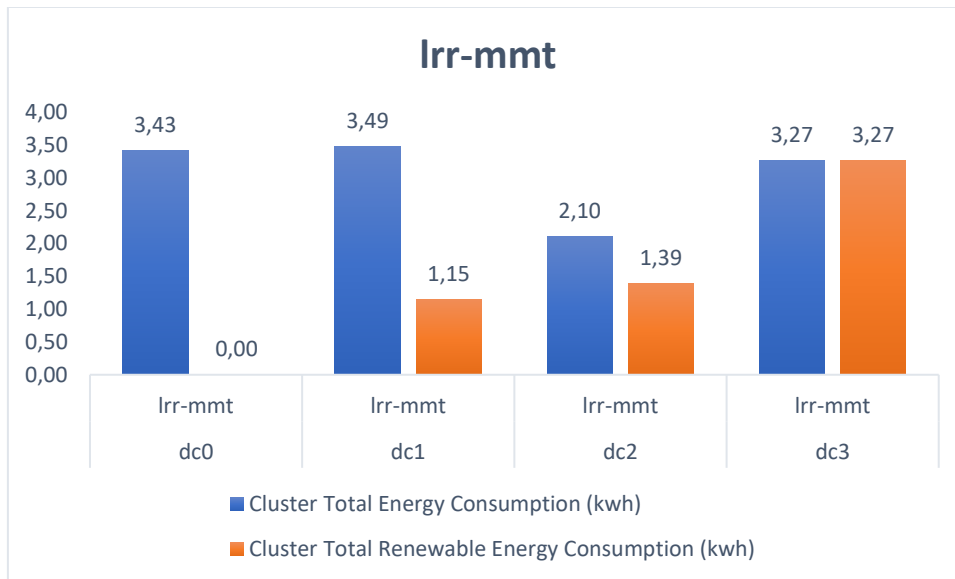


Figura 4-44 Comportamiento del algoritmo LRR en el escenario 3 con 10000 Cloudlets de carga

El algoritmo LRR con estrategia MMT muestra esta vez que el consumo más importante se registra en el dc1 marcando 3,49 kwh/1,15 kwh de renovables, seguido por los dc0 y dc3 que registran 3,43 kwh y 3,27 kwh/3,27 kwh de renovables respectivamente. Finalmente, y con el menor consumo de todos está el dc2 que marca 2,10 kwh/1,39 kwh de renovables.

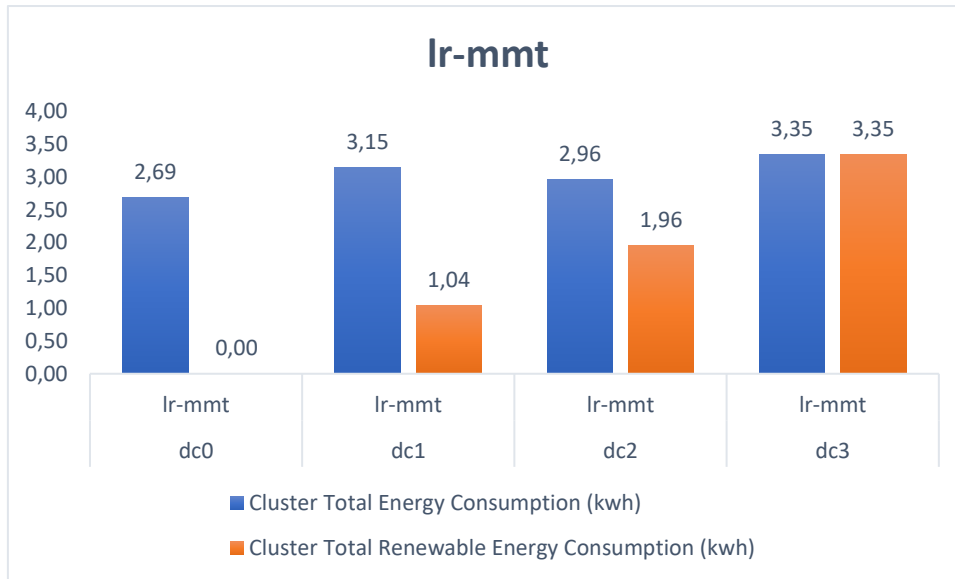


Figura 4-45 Comportamiento del algoritmo LR en el escenario 3 con 10000 Cloudlets de carga

El algoritmo LR con MMT muestra esa vez que el consumo más importante de energía se produce en el dc3 que marca 3,35 kwh/3,35 kwh de renovables, seguido por el dc1 3,15 kwh/1,04 kwh de renovables, luego va el dc2 que registra 2,96 kwh/1,96 kwh de renovables y finalmente con el menor consumo de todos está el dc0 que marca 2,69 kwh.

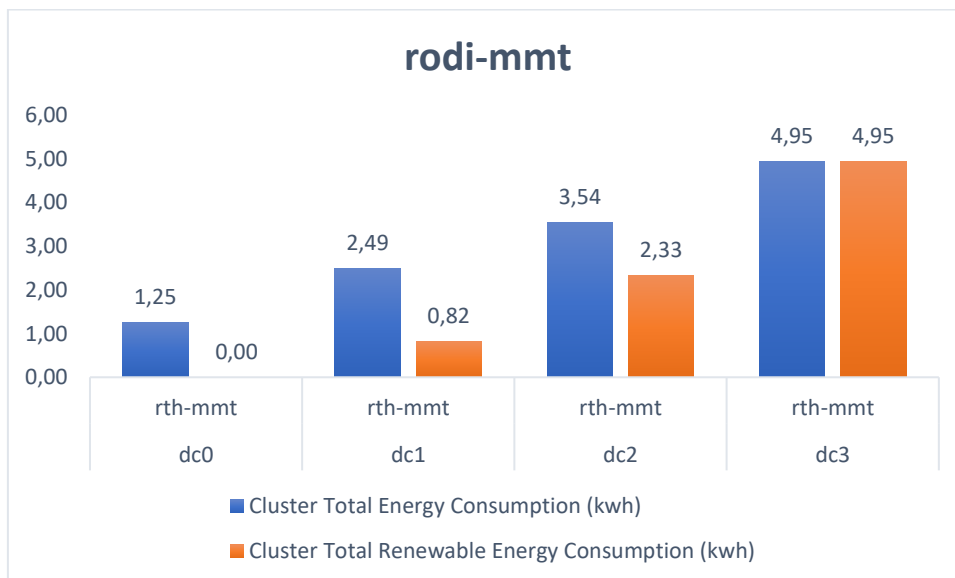


Figura 4-46 Comportamiento del algoritmo RODI en el escenario 3 con 10000 Cloudlets de carga

Se mantiene el comportamiento del algoritmo RODI con estrategia MMT donde se sigue mostrando que los valores más importantes de consumo se registran en los Data Centers con mayor disposición de energía renovable, y con ellos se registran 4,95 kwh/4,95 kwh de renovables en el dc3, seguido por el dc2 que registra 3,54 kwh/2,33 kwh de renovables, luego va el dc1 que marca 2,49 kwh/0,82 kwh de renovables y en último lugar y con el menor consumo de todos está el dc0 que registra 1,25 kwh.

4.3 Resultados contribución 1 - Comparativa de eficiencia renovable para los algoritmos IQR, LR, LRR, THR, RODI y FRBS con conocimiento de experto

4.3.1 Introducción

Al igual que en la sección de resultados “4.2 Resultados Previos a la contribución 1 - Comparativa de eficiencia renovable para los algoritmos IQR, LR, LRR, THR y RODI”, se ha utilizado el simulador CloudSim con los mismos elementos de hardware para la simulación de los hosts, las máquinas virtuales, las cargas de trabajo (Cloudlets) y también las tramas de simulaciones reales (provenientes de SPEC) para la simulación real del trabajo; con la particularidad de que en este caso, el tiempo de simulación es de 4 horas en vez de solo una del experimento anterior. Esto se debe a que se pretenden establecer escenarios de simulación donde la carga de los Cloud data centers vaya variando con el tiempo cada hora de manera que cada uno de ellos pase por porcentajes de energía renovable que van del 0%, pasando por el 33%, luego el 66% y finalmente llegar al 100%. Cada uno de los Data Centers empieza con uno de los valores anteriores y va cambiando con el tiempo al resto de valores.

Respecto a la política de migración de las máquinas virtuales, en este caso se implementa un sistema experto basado en “Fuzzy Rules Based System” y que utiliza como base de conocimientos reglas proporcionadas por un experto. Las variables de entrada que utiliza el sistema experto se describen en la *Tabla 4-5 descripción de las variables de entrada del sistema FRBS*.

Variable de entrada	descripción
Cloud Data Center Renewable Availability (CDC-RA)	Renewable energy percentage supplied to the CDC
Host Computational Capacity (HCC)	Maximum computational capacity in MIPS
Host Computational Availability (HCA)	Remaining computational capacity in the MIPS after holding other VMs
VM Maximum Computational Needs (VM-MCN)	Maximum needs of the Virtual Machine in MIPS
VM Current Computational Needs (VM-CCN)	Remaining computational needs in MIPS for the VM

Tabla 4-5 descripción de las variables de entrada del sistema FRBS

Cabe mencionar que el conocimiento considerado para el sistema experto consiste en un conjunto de reglas difusas que relacionan las variables de entrada y salida del sistema mediante una representación lingüística para obtener una respuesta final. En este trabajo, las reglas utilizadas se obtuvieron a partir de la experiencia de los autores y se

presentan en la Tabla 4-6 Para ser más precisos, se han considerado dos tipos de reglas; las relacionadas con aspectos computacionales y otras relacionadas con aspectos relativos a la energía renovable. Con ello, se ha considerado un conjunto reducido de reglas simples para favorecer una buena comprensión y evitar el uso de más de dos variables de entrada en cada regla individual para así mantener una mejor interpretabilidad de dichas reglas.

Rule	CDC-RA	HCC	HCA	VM-MCN	VM-CCN	AND/OR	OUTPUT
1		High		High		AND	Extremely Suitable
2		Low		Low		AND	Very Not Suitable
3			High		High	AND	Extremely Suitable
4			Low		Low	AND	Very Not Suitable
5	Low						Very Not Suitable
6	Medium						Suitable
7	High						Extremely Suitable

Tabla 4-6 Reglas del Sistema experto

4.3.2 Descripción de los escenarios de simulación

Para este experimento, se han considerado tres escenarios principales de simulación que se describen en la *Tabla 4-7 Escenarios de simulación utilizados*. Estos escenarios se han seleccionado para proporcionar una amplia variedad de situaciones en cuanto al tamaño de los Cloud data centers y también las cargas de trabajo. Para más exactos y teniendo en cuenta la capacidad de CloudSim, que permite hasta 800 hosts (Utilizando las trazas SPEC), se han considerado tres escenarios (pequeño, mediano y grande). Estos escenarios representan diferentes tamaños de infraestructura, ya que permiten obtener resultados concluyentes sin ser exhaustivos y sin dificultar la presentación de los resultados obtenidos.

	Escenario 1	Escenario 2	Escenario 3
Hosts	265	530	800
VM	350	695	1052
Workload (Cloudlets)	500 - 1,500 - 3,000	1,000 - 2,000 - 5,000	1,500 - 5,000 - 10,000

Tabla 4-7 Escenarios de simulación utilizados

Para cada uno de los escenarios, los hosts y las máquinas virtuales se distribuyeron aleatoriamente en cuatro CDCs diferentes. El primero no tiene fuente de energía

renovable, el segundo tiene un 33% de energía renovable, el tercero un 66% y el último un 100%. Sin embargo, es importante mencionar que, a diferencia del anterior experimento, en este caso, se ha considerado un cambio dinámico en la energía renovable disponible. Por lo tanto, el porcentaje de energía renovable rotará en los Data Centers cada hora, lo que significa que al final de las simulaciones (4 horas), cada uno de ellos habrá cambiado tres veces el porcentaje de energía renovable disponible en él.

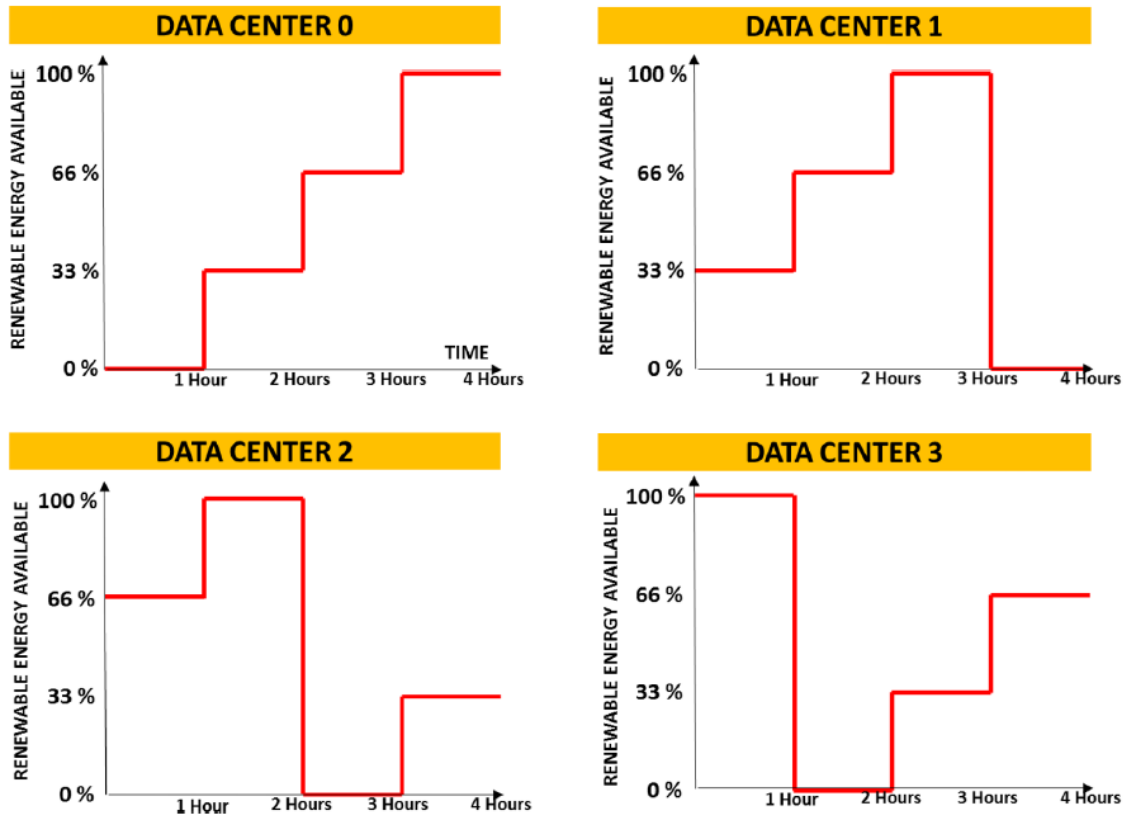


Figura 4-47 Distribución de la disponibilidad en el tiempo de la energía renovable en los Data Centers

Cabe destacar que la disponibilidad de energía renovable considerada es independientemente de su origen y, por lo tanto, no hay dependencia en términos de tipo de energía renovable.

4.3.3 Resultados experimentales

4.3.3.1 Resultados globales de uso de energía renovable por algoritmo

Los escenarios de simulación descritos en la sección anterior se realizaron para obtener los resultados que se muestran en las Tablas 4-6. Estas indican el promedio de 30 experimentos, que corresponden a un umbral de regla general basado en el Teorema del Límite Central, donde la distribución de la muestra se acercará a una distribución normal a medida que aumente el tamaño de la muestra. En las tablas (Tabla 4-8, Tabla 4-9, Tabla 4-10), se puede observar que los algoritmos THR, IQR, IR e IRR conducen consistentemente a un uso del 53-54% de energía renovable independientemente de los escenarios utilizados. Además, el algoritmo anterior presentado en [Seddiki et al., 2022] para verificar las nuevas capacidades de CloudSim ha alcanzado un promedio del 63-64% de uso de energía renovable (promedio de los tres escenarios), lo que representa una mejora clara de aproximadamente el 9-10% con respecto a los algoritmos anteriores. También se puede observar que, en el contexto de las condiciones relacionadas con el entorno dinámico, el meta-planificador basado en FRBS propuesto en este trabajo logra alcanzar el 63-64% del uso de energía renovable, lo que representa un comportamiento similar a los mejores resultados del mejor algoritmo utilizado en el trabajo anterior [Seddiki et al., 2022]. Sin embargo, la mejora alcanzada por el algoritmo basado en FRBS en comparación con el algoritmo anterior se puede observar cuando aumenta el tamaño y la complejidad de la infraestructura en el cloud. Esto significa que el meta-planificador propuesto presenta un buen rendimiento al optimizar la sostenibilidad de los sistemas en el cloud en términos del uso de energía renovable a pesar del conocimiento experto no optimizado utilizado en su base de conocimientos. Finalmente, en relación con el uso de energía renovable expresado en las tablas (Tabla 4-8, Tabla 4-9, Tabla 4-10), también se incluyó un intervalo de confianza del 99%. Por otro lado, es importante señalar que cuantas más migraciones haya entre los data centers en el cloud, más información debe transmitirse a través de las redes de comunicación, lo que puede implicar un deterioro en la experiencia del usuario. Sin embargo, con respecto al SLA expresado en las tablas (Tabla 4-8, Tabla 4-9, Tabla 4-10), el impacto no es significativo.

ESCENARIO 1							
HOSTS: 265, VM: 350		ALGORITHM					
Cloudlet							
s	Performance	THR	IQR	IRR	LR	Previous	FRBS
500	Renewable used	54,09±0,0	53,89±0,0	54,08±0,1	54,40±0,1	63,21±0,1	63,31±0,1
	(%)	9	9	1	3	1	3
1500	SLA Deg. (%)	< 0,0001	< 0,0001	< 0,0001	< 0,0001	< 0,0001	< 0,0001
	Renewable used	54,44±0,0	53,58±0,0	53,80±0,1	53,73±0,1	63,31±0,1	62,75±0,1
3000	(%)	9	8	0	4	2	1
	SLA Deg. (%)	< 0,0001	< 0,0001	< 0,0001	< 0,0001	< 0,0001	< 0,0001
	Renewable used	54,29±0,0	54,08±0,1	54,17±0,1	53,62±0,1	63,91±0,1	62,68±0,1
	(%)	7	1	3	4	0	5
	SLA Deg. (%)	< 0,0001	< 0,0001	< 0,0001	< 0,0001	< 0,0001	< 0,0001

Tabla 4-8 Resultados obtenidos para el escenario 1

ESCENARIO 2							
HOSTS: 530, VM: 695		ALGORITHM					
Cloudlet							
s	Performance	THR	IQR	IRR	LR	Previous	FRBS
1000	Renewable used	53,81±0,0	53,65±0,0	53,59±0,0	53,49±0,1	63,37±0,0	63,71±0,0
	(%)	7	7	7	0	7	7
2000	SLA Deg. (%)	< 0,0001	< 0,0001	< 0,0001	< 0,0001	< 0,0001	< 0,0001
	Renewable used	53,61±0,0	53,66±0,0	53,91±0,0	53,35±0,0	63,19±0,1	63,40±0,0
5000	(%)	7	9	8	7	0	8
	SLA Deg. (%)	< 0,0001	< 0,0001	< 0,0001	< 0,0001	< 0,0001	< 0,0001
	Renewable used	53,83±0,0	53,67±0,0	53,24±0,0	53,56±0,0	63,57±0,0	63,65±0,0
	(%)	9	8	7	9	9	9
	SLA Deg. (%)	< 0,0001	< 0,0001	< 0,0001	< 0,0001	< 0,0001	< 0,0001

Tabla 4-9 Resultados obtenidos para el escenario 2

ESCENARIO 3							
HOSTS: 800VM: 1052		ALGORITHM					
Cloudlet							
s	Performance	THR	IQR	IRR	LR	Previous	FRBS
1500	Renewable used	53,78±0,0	53,67±0,0	53,69±0,0	53,62±0,0	63,54±0,0	64,36±0,0
	(%)	6	5	6	7	7	5
5000	SLA Deg. (%)	< 0,0001	< 0,0001	< 0,0001	< 0,0001	< 0,0001	< 0,0001
	Renewable used	53,86±0,0	53,88±0,0	53,75±0,0	53,58±0,0	63,55±0,0	63,85±0,0
10000	(%)	6	7	7	6	6	7
	SLA Deg. (%)	< 0,0001	< 0,0001	< 0,0001	< 0,0001	< 0,0001	< 0,0001
10000	Renewable used	54,12±0,0	53,90±0,0	53,39±0,0	53,70±0,0	63,56±0,0	64,28±0,0
	(%)	6	7	7	6	9	9
	SLA Deg. (%)	< 0,0001	< 0,0001	< 0,0001	< 0,0001	< 0,0001	< 0,0001

Tabla 4-10 Resultados obtenidos para el escenario 3

A continuación, y en el contexto del segundo escenario con 5000 Cloudlets, se muestra el consumo de energía como para las migraciones de VM. Las figuras (Figura 4-48, Figura 4-50) muestran el comportamiento del sistema basado en reglas difusas y el algoritmo RODI propuesto anteriormente en [Seddiki et al., 2022]. En las figuras (Figura 4-48, Figura 4-50) se puede ver la energía total y renovable consumida en cada data center, mientras que las figuras 8 y 10 muestran cómo ambas estrategias llevan a cabo la migración de VM siguiendo el enfoque renovable.

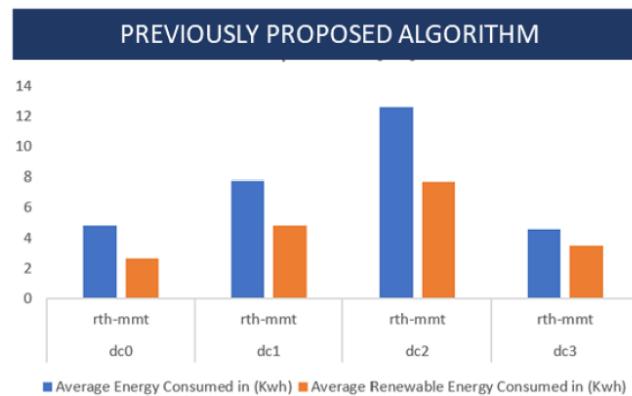


Figura 4-48 Algoritmo RODI en el escenario 2 con 5000 Cloudlets

En relación con la Figura 4-48, se puede observar cómo la mayor parte de la energía consumida (total y renovable) se concentra en los Data Centers DC2 y DC1, seguidos por DC3 y DC0 en este orden específico. En este sentido, la Figura 4-47 presenta el comportamiento de cambio dinámico de la energía renovable en los diferentes Data Centers. Durante las primeras 2 horas de simulación, los data centers DC2 y DC1 tienen los mayores porcentajes promedio de energía renovable disponible de los cuatro Data

Centers. Además, la carga de Cloudlets introducida al sistema es muy importante al principio de la simulación y va decayendo con el paso del tiempo, lo que hace que la mayor parte del procesamiento de esta carga se realice en los data centers DC2 y DC1 y que la carga restante se ventile entre los otros dos Data Centers en la tercera y cuarta hora de simulación.

Este comportamiento se puede explicar claramente mediante la Figura 4-49, donde el primer intervalo (de 0 a 1 hora) muestra que las migraciones más grandes se producen hacia el Data Center DC3, que tiene un 100% de energía renovable disponible, seguido por el DC2 con un 66%, el DC1 con un 33% y finalmente el DC0 con el menor número de migraciones de máquinas virtuales.

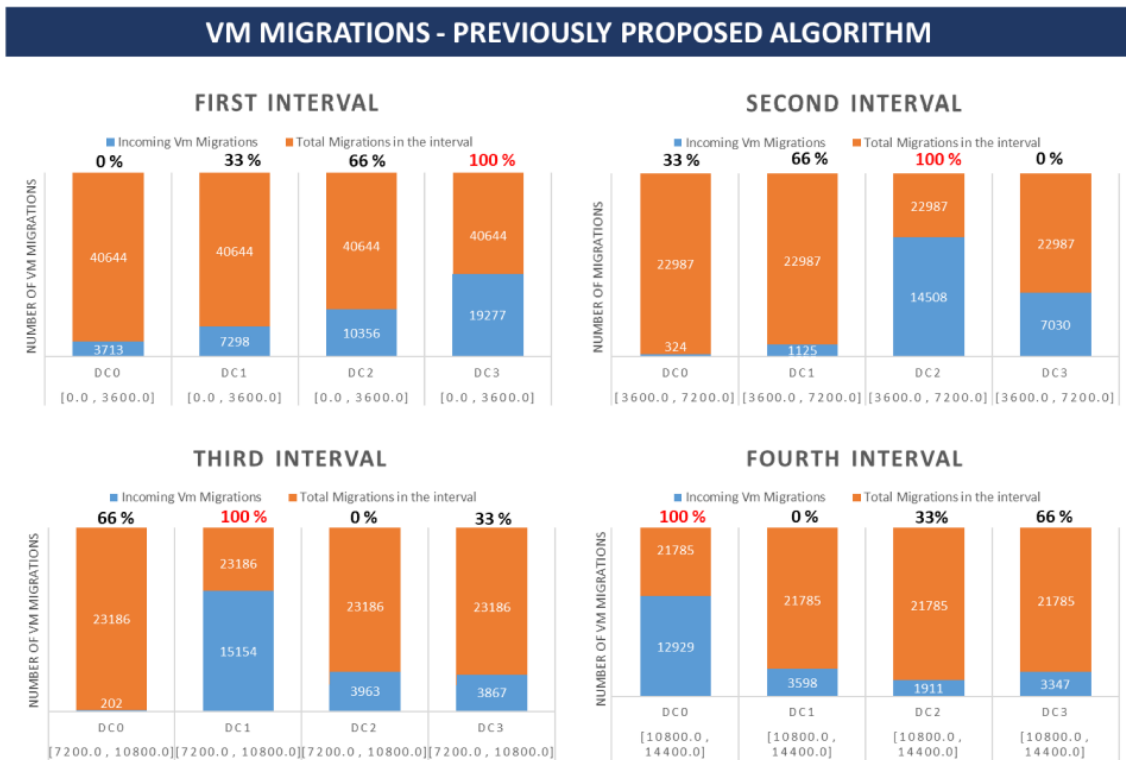


Figura 4-49 Numero de migraciones entrantes Vs Migraciones totales en el escenario 2 para el algoritmo RODI

Durante el segundo intervalo, se puede observar un gran aumento de migraciones, en este caso hacia el data center DC2 (que ahora tiene un 100% de energía renovable) y una gran disminución en las migraciones hacia los otros tres. En el tercer y cuarto intervalo, el algoritmo sigue el mismo patrón que antes, de modo que el Data Center con la mayor cantidad de energía renovable disponible recibe la mayor cantidad de

migraciones. Esto se ve claramente a través del número de estas en el DC1 en el tercer intervalo y el mismo comportamiento para DC0 en el último intervalo.

El planificador basado en FRBS propuesto en este trabajo, que utiliza conocimientos expertos, ha alcanzado el mismo rendimiento que el algoritmo propuesto anteriormente en [Seddiki et al., 2022], con un promedio del 63-64% del uso de energía renovable durante las cuatro horas de simulación y en los cuatro data centers. En este sentido, la Figura 4-50 representa el consumo total y renovable de energía en los diferentes Data Centers.

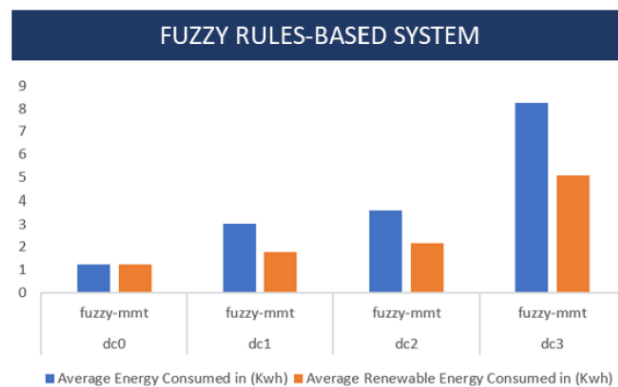


Figura 4-50 Comportamiento del algoritmo basado en FRBS en el escenario 2

En la Figura 4-51 se puede observar que para el *algoritmo basado en FRBS y reglas genéricas*, en el primer intervalo de tiempo (la primera hora), el Data Center con más energía renovable (100%) es el que acapara la mayoría de las migraciones entrantes de las máquinas virtuales, registrando 52.815 migraciones. Se puede observar por otro lado que solo registra 66 migraciones hacia el Data Center DC2 que tiene un 66% de energía renovable disponible, también se ve que solo realiza 30 migraciones hacia el DC1 que tiene un 33% de energía renovable. Finalmente se observa que prácticamente no hay migraciones hacia el DC0 que no tiene energía renovable que lo alimente.

VM MIGRATIONS – FUZZY RULES-BASED SYSTEM

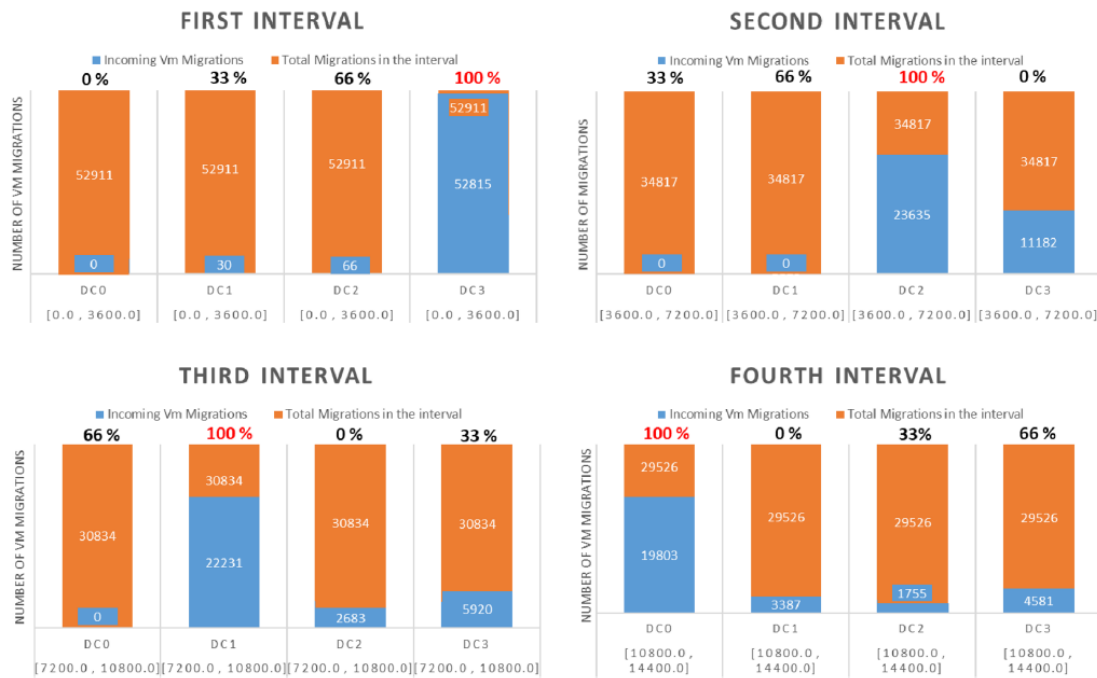


Figura 4-51 Numero de migraciones entrantes Vs Migraciones totales en el escenario 2 para el algoritmo basado en FRBS

En el segundo intervalo de tiempo (la segunda hora de simulaciones) el DC2 pasa de tener un 66% de renovable a un 100% y con ello vemos que el número de migraciones entrantes hacia el pasa de unas pocas 66 migraciones a 23.635, mientras que el DC3 que tenía en la hora previa un 100% de renovable y que ha pasado a tener un 0%, ha visto disminuidas sus migraciones entrantes de 52.815 a 11.182 migraciones.

En los intervalos restantes (el tercero y cuarto), vemos que el comportamiento se mantiene, registrando el mayor número de migraciones entrantes en los Data Centers que tienen mayor energía renovable asignada, pero se debe mencionar que se sigue registrando cierto número de migraciones residuales a los otros Data Centers.

Cabe también mencionar, que el comportamiento descrito en el escenario 2 se repite en todos los otros escenarios presentados en “4.3.2 Descripción de los escenarios de simulación”.

4.4 Resultados contribución 2 - Comparativa de eficiencia renovable FRBS con conocimiento de experto, FRBS con aprendizaje KASIA y FRBS con aprendizaje KAGWO

4.4.1 Introducción

En este trabajo se continua con el uso del CloudSim para realizar las simulaciones, con las mismas configuraciones de Hardware (tipos de máquinas), mismo número de hosts, máquinas virtuales y cargas de trabajo. También se continúan utilizando las trazas reales *SPEC* para las simulaciones.

Todas las simulaciones se realizan para cuatro ciclos de 1h, totalizando 4 horas de funcionamiento donde los cuatro Data Centers utilizados van variando dinámicamente su energía renovable disponibles pasando por cuatro posibles valores que van del 0% al 100% con un paso del 33%.

Se comparan tres técnicas de adquisición del conocimiento, que son KASIA, Pittsburg y la nueva aportación que consiste en el algoritmo KAGWO basado en el algoritmo “Gray Wolf Optimizer”. Todas se utilizan para optimizar las reglas obtenidas en el apartado anterior 4.3.3 y así comparar las diferencias en el desempeño.

Cabe mencionar que tanto las variables de entrada utilizadas para los sistemas FRBS que utilizan estos nuevos algoritmos, como la anatomía y número de las reglas no cambia en este caso, lo único que lo hace son los nuevos valores que estos algoritmos de adquisición del conocimiento proponen para obtener los nuevos sets de reglas que intentan mejorar el funcionamiento anterior de las reglas de propósito general más el sistema FRBS.

Sin embargo, es importante resaltar las configuraciones específicas de los algoritmos de adquisición del conocimiento. Por una parte, tenemos a KASIA que necesita siete parámetros de configuración para la realización de las simulaciones.

Numero de simulaciones	Numero de partículas	Iteraciones máximas	Iteraciones			
			Wmax	Wmin	c1	c2
30	64	50	0,9	0,2	2	2

Tabla 4-11 Configuración para la simulación de KASIA

Configuración de Pittsburg

Simulations	Particles	Iterations	Crossover Rate	Initial mutation Rate	Selection Rate	Replacement Rate
30	64	50	0,8	0,1	0,8	0,8

Tabla 4-12 configuración para la simulación de Pittsburg

Y por otra parte tenemos al KAGWO que necesita menos parámetros de configuración (exactamente 3) y que son:

Numero de simulaciones	Numero de partículas	Iteraciones máximas
30	64	50

Tabla 4-13 Configuración para la simulación de KAGWO

Los escenarios de simulación en este experimento tienen la misma configuración que los utilizados en el apartado 4.3.2.

4.4.2 Descripción de los escenarios de simulación

Los escenarios utilizados tienen la misma configuración que se describe en el apartado 4.3.2.

4.4.3 Resultados experimentales

El objetivo de estos experimentos era el de validar la eficacia del algoritmo KAGWO, que pertenece al campo del Swarm intelligence en el machine learning, para mejorar la solución de referencia establecida por un conjunto de reglas desarrolladas por expertos utilizados por el Sistema Basado en Reglas Difusas (FRBS) del trabajo de [Seddiki et al., 2022]. Para cada uno de los nueve escenarios, se realizaron 30 simulaciones para KASIA, Pittsburgh y KAGWO. La Tabla 10 muestra la media y la desviación estándar de todas las simulaciones para cada escenario, abarcando tanto algoritmos como los resultados del documento anterior.

Hosts	VMs	Cloudlets	Results	KASIA	KASIA Imp.	Pitts.	Pitts. Imp.	KAGWO	KAGWO Imp.
256	350	500	63.31±0.13	65.49±0.02	2.18 (3.45%)	65.47±0.03	2.16 (3.41%)	65.53±0.02	2.22 (3.51%)
256	350	1,500	62.75±0.11	65.51±0.02	2.76 (4.40%)	65.49±0.02	2.74(4.37%)	65.52±0.02	2.77 (4.42%)
256	350	3,000	62.68±0.15	65.50±0.02	2.82 (4.50%)	65.50±0.02	2.82 (4.50%)	65.54±0.02	2.86 (4.57%)
530	695	1,000	63.71±0.07	66.69±0.02	2.98 (4.68%)	66.69±0.02	2.98 (4.68%)	66.73±0.01	3.02 (4.74%)
530	695	2,000	63.40±0.08	66.69±0.01	3.29 (5.19%)	66.70±0.02	3.30 (5.21%)	66.72±0.01	3.32 (5.23%)
530	695	5,000	63.65±0.09	66.69±0.01	3.04 (4.77%)	66.68±0.01	3.03 (4.76%)	66.72±0.01	3.07 (4.83%)
800	1,052	1,500	64.36±0.05	64.67±0.02	0.31 (0.49%)	64.66±0.02	0.30 (0.47%)	64.70±0.01	0.34 (0.53%)
800	1,052	5,000	63.85±0.07	64.68±0.02	0.83 (1.31%)	64.67±0.02	0.82 (1.28%)	64.70±0.02	0.85 (1.34%)
800	1,052	10,000	64.28±0.09	64.69±0.01	0.41 (0.64%)	64.67±0.01	0.39 (0.61%)	64.72±0.02	0.44 (0.68%)

Figura 4-52 Resultados de la comparativa entre los valores de referencias de [D. Seddiki, 2022], KASIA, Pittsburg y KAGWO en términos de porcentaje de energía renovable utilizada

En la evaluación del porcentaje de uso eficiente de energía renovable en diversos escenarios, el algoritmo KAGWO emerge consistentemente como la mejor solución en comparación tanto con los resultados de referencia del documento anterior [Seddiki et al., 2022]. como con el algoritmo KASIA.

En el escenario más pequeño (256 Hosts, 350 máquinas virtuales), KAGWO mostró mejoras notables, logrando un aumento del 3.51% en el uso de energía renovable con 500 cargas de trabajo en el cloud (cloudlets), mientras que Pittsburgh también demostró resultados aceptables con un aumento del 3.41%. Esta tendencia continuó con mayores números de cloudlets, con KAGWO alcanzando una mejora del 4.57%, superando ligeramente el aumento del 4.37% de Pittsburgh. Simultáneamente, KASIA mostró mejoras respetables también; sin embargo, KAGWO superó consistentemente a KASIA en términos de porcentaje de mejora.

Al pasar al escenario medio (530 Hosts, 695 máquinas virtuales), KAGWO continuó mostrando un rendimiento notable, logrando una mejora máxima del 5.23% con 2,000 cloudlets, mientras que Pittsburgh mantuvo su competitividad con una mejora del 5.21%. En comparación, KASIA mostró mejoras competitivas, alcanzando un máximo del 5.19% con 2,000 cloudlets. A pesar de las mejoras notables de KASIA, KAGWO mantuvo una ventaja en términos de aumento porcentual en el uso de energía renovable.

Incluso en el escenario grande (800 Hosts, 1,052 máquinas virtuales), KAGWO superó consistentemente la referencia, con una mejora del 1.34% en el uso de energía renovable con 5,000 cloudlets, mientras que Pittsburgh mostró una mejora del 1.28%. Aunque KASIA alcanzó un aumento del 1.31% con el mismo número de cloudlets,

KAGWO demostró una mejora ligeramente más significativa. Estos resultados resaltan la naturaleza robusta y confiable del algoritmo KAGWO, posicionándolo como una opción destacada para optimizar el uso de energía renovable en un espectro respetable de escenarios.

La superioridad del enfoque KAGWO va más allá de su rendimiento en la utilización de energía renovable; también es evidente en términos de configuración de parámetros, lo que lo convierte en una opción más sencilla y práctica para la implementación. KAGWO logra unos mejores resultados aun con un conjunto de parámetros de configuración más reducido en comparación con las configuraciones más intrincadas de KASIA y Pittsburgh.

La simplicidad en la configuración de los parámetros de KAGWO no solo agiliza el proceso de implementación, sino que también reduce los errores potenciales relacionados con la complejidad. Esta eficiencia en la configuración hace de KAGWO una solución más accesible y fácil de usar, destacando aún más sus ventajas prácticas sobre KASIA y Pittsburgh. En esencia, el enfoque KAGWO no solo sobresale en métricas de rendimiento, sino que también ofrece una experiencia más sencilla y amigable para el usuario, convirtiéndolo en una elección atractiva para aquellos que buscan tanto efectividad como facilidad de implementación.

La comparación gráfica de KASIA, Pittsburgh y KAGWO a lo largo de los nueve sub-escenarios revela patrones de convergencia y dinámicas de rendimiento distintas, como se puede observar en las Figuras Figura 4-53, Figura 4-54, Figura 4-55. En el primer sub-escenario, KASIA y Pittsburgh muestran una convergencia temprana, pero KAGWO presenta una tendencia a la mejora continua a través de iteraciones, superando finalmente a KASIA y Pittsburgh. La superioridad de KAGWO se vuelve estadísticamente evidente cuando las regiones de confianza del 95% no se superponen, destacando su rendimiento consistente. Por el contrario, en el segundo sub-escenario, KASIA, Pittsburgh y KAGWO

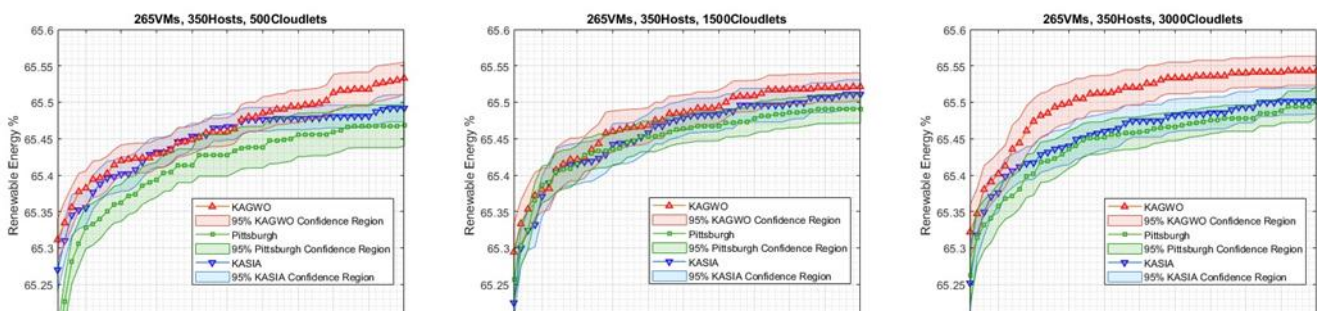


Figura 4-53 Evolución de KASIA, Pittsburgh y KAGWO a través de la adquisición de conocimiento en el escenario pequeño (1.º, 2.º y 3.º sub-escenarios).

muestran una rápida convergencia, con resultados superpuestos. El tercer sub-escenario presencia una convergencia rápida para todos los algoritmos, pero KAGWO destaca con resultados superiores, como se indica por las regiones de confianza no superpuestas.

Se observa una tendencia similar en el cuarto, quinto y sexto sub-escenario, donde KAGWO supera consistentemente a KASIA y Pittsburgh después de su punto de convergencia, excepto en el quinto sub-escenario.

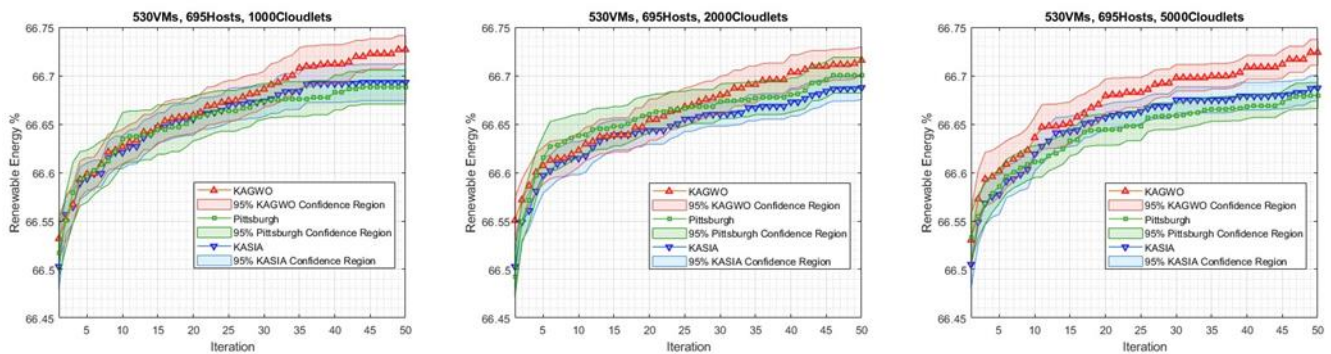
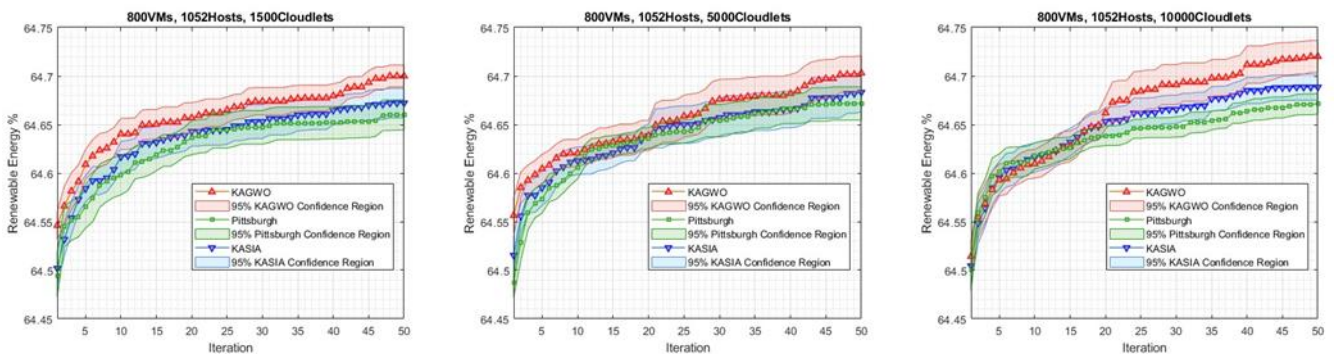


Figura 4-54 Evolución de KASIA, Pittsburgh y KAGWO a través de la adquisición de conocimiento en el escenario medio (4.º, 5.º y 6.º sub-escenarios).

El séptimo sub-escenario refleja este patrón. En el octavo sub-escenario, KAGWO, Pittsburgh y KASIA presentan resultados similares, con regiones de confianza superpuestas. El noveno sub-escenario replica las dinámicas de convergencia del primero, con KAGWO mostrando superioridad sobre KASIA y Pittsburgh. Este análisis gráfico



exhaustivo resalta las variaciones sutiles en el rendimiento a lo largo de los escenarios,

Figura 4-55 Evolución de KASIA, Pittsburgh y KAGWO a través de la adquisición de conocimiento en el escenario grande (7.º, 8.º y 9.º sub-escenarios).

estableciendo a KAGWO como una solución consistentemente efectiva y robusta en comparación con KASIA y Pittsburgh.

KAGWO demuestra consistentemente un rendimiento superior a lo largo de un espectro de escenarios, destacando particularmente en la fase de exploración, donde sus

iteraciones iniciales superan consistentemente a las de KASIA y Pittsburgh. Esta ventaja temprana sienta una base sólida para las iteraciones posteriores, facilitando una fase de explotación más efectiva. A medida que KAGWO progresa, sus últimas iteraciones superan consistentemente a KASIA y Pittsburgh, mostrando su capacidad para la explotación y optimización robustas. En esencia, la fuerza de KAGWO radica en su capacidad no solo para explorar eficazmente el espacio de soluciones desde el principio, sino también para explotar las soluciones descubiertas de manera más eficiente a medida que evoluciona, convirtiéndolo en una opción formidable en diversas tareas de optimización.

Para evaluar el rendimiento de los algoritmos KASIA, Pittsburg y KAGWO en la optimización del uso de energía renovable en los Cloud data centers, se realizaron las siguientes pruebas estadísticas:

Prueba de Normalidad (Shapiro-Wilk): Se realizó la prueba para verificar si los datos seguían una distribución normal. Los resultados de esta prueba para cada conjunto de datos fueron los siguientes:

Escenario	Shapiro-Wilk	Shapiro-Wilk p-value
256 hosts - 350 VMs - 500 Cloudlets	0.994980	0.999926
256 hosts - 350 VMs - 1500 Cloudlets	0.977984	0.769801
256 hosts - 350 VMs - 3000 Cloudlets	0.978899	0.795586
530 hosts - 695 VMs - 500 Cloudlets	0.965746	0.430210
530 hosts - 695 VMs - 2000 Cloudlets	0.967025	0.461304
530 hosts - 695 VMs - 5000 Cloudlets	0.965894	0.433729
800 hosts - 1052 VMs - 1500 Cloudlets	0.974957	0.681451
800 hosts - 1052 VMs - 5000 Cloudlets	0.927000	0.40896
800 hosts - 1052 VMs - 10000 Cloudlets	0.957963	0.274529

Tabla 4-14 Prueba de normalidad de resultados (Shapiro-Wilk) previa al análisis ANOVA

En todos los conjuntos de datos, los p-valores de la prueba de Shapiro-Wilk son mayores que 0.05, lo que sugiere que los datos siguen una distribución normal.

Prueba de Homogeneidad de Varianzas (Levene): Se realizó para comprobar la homogeneidad de las varianzas entre los diferentes grupos:

Estadístico - Levene	Levene p-value
1,65552209	0,109581552

Tabla 4-15 Prueba de homogeneidad de varianza (Levene) Pravia al analisis ANOVA

El p-valor de la prueba de Levene es mayor que 0.05, indicando que las varianzas entre los grupos son homogéneas.

Dado que los datos cumplen con los supuestos de normalidad y homogeneidad de varianzas, procedimos a realizar el ANOVA para comparar los algoritmos. Los resultados del ANOVA mostraron un estadístico significativo y un valor p por debajo de 0.05, sugiriendo diferencias significativas entre los grupos.

Estos hallazgos proporcionaron una visión completa del rendimiento de los tres algoritmos y permitieron identificar las combinaciones de algoritmos y escenarios que produjeron mejoras significativas en la optimización de la energía renovable.

La tabla 11 presenta los resultados de una prueba de ANOVA realizada con datos de varios escenarios, comparando diferentes configuraciones de hosts, máquinas virtuales y cloudlets. El análisis estadístico proporciona información valiosa sobre las variaciones en el rendimiento en diferentes configuraciones, ayudando a comprender de manera integral el sistema estudiado.

Escenarios			KAGWO vs KASIA		KAGWO vs Pittsburgh	
Hosts	VMs	Cloudlets	ANOVA p-value	Significativamente mejor	ANOVA p-value	Significativamente mejor
256	350	500	0.006521	✓	0.000993	✓
256	350	1,500	0.466500	✗	0.036145	✓
256	350	3,000	0.004118	✓	0.005937	✓
530	695	1,000	0.007370	✓	0.001578	✓
530	695	2,000	0.004290	✓	0.199050	✗
530	695	5,000	0.000282	✓	0.000022	✓
800	1,052	1,500	0.004947	✓	0.000142	✓
800	1,052	5,000	0.157650	✗	0.015717	✓
800	1,052	10,000	0.004931	✓	0.000009	✓

Tabla 4-16 análisis de varianza ANOVA realizado con datos de todos los escenarios usando un nivel de significancia $\alpha = 0.05$.

El objetivo principal de estos experimentos fue el de investigar las capacidades de KAGWO, que es un algoritmo basado en Swarm intelligence, para mejorar los resultados de un Sistema Basado en Reglas Difusas (FRBS) que utilizaba una Base de Conocimiento (KB) proporcionada por un experto. La comparación se realizó con otros enfoques,

incluidos KASIA y Pittsburgh, que habían demostrado eficacia en aplicaciones similares. Los resultados de KAGWO obtenidos en la sección anterior mostraron una mejora significativa que oscila entre el 3.51% y el 5.23% para los dos primeros escenarios de simulación. Sin embargo, KAGWO exhibió ciertas limitaciones en el tercer escenario, mostrando solo una modesta mejora entre el 0.53% y el 1.34%. Se realizaron más comparaciones, y se reveló que KASIA y Pittsburgh entregaron resultados similares, pero peores, que KAGWO, con limitaciones análogas en este último escenario.

Estos resultados sugieren aplicaciones interesantes para arquitecturas de pequeña, mediana y gran escala, donde el uso de energía renovable puede optimizarse mediante la implementación de KAGWO para estrategias de intermediación en migraciones de máquinas virtuales entre cloud data centers geográficamente distantes. Al aprovechar las capacidades de KAGWO, se pueden lograr mejoras significativas en los costes de energía, lo que lo convierte en una herramienta valiosa para mejorar la eficiencia energética en entornos cloud.

5 Conclusiones, limitaciones y líneas futuras

5.1 Conclusiones y limitaciones

El desarrollo de esta tesis doctoral culminó en conclusiones que destacan la importancia de nuevas estrategias para la gestión eficiente de carga en entornos de cloud computing basados en energías renovables. Estos logros reflejan el avance en el diseño y empleo de sistemas inteligentes para la migración de máquinas virtuales entre Data Centers geográficamente dispersos, además de la aplicación de técnicas de adquisición de conocimiento para mejorar la calidad de las bases expertas.

La extensión del simulador CloudSim para incorporar la capacidad de simular sistemas de cloud computing con energías renovables fue fundamental para este avance. Gracias a esta base, se desarrollaron y probaron estrategias "Follow the renewables", permitiendo la migración de VMs según la disponibilidad de energía renovable. Este enfoque facilitó la implementación de un primer sistema FRBS con reglas genéricas, mostrando una mejoría en la tasa media de utilización de energía renovable que varía entre el 63% y el 64% en los Cloud data centers simulados.

Además, el desarrollo del algoritmo KAGWO basado en Swarm intelligence para la adquisición del conocimiento y generación de nuevas bases de reglas para el sistema experto anterior con la finalidad de optimizar y mejorar los resultados anteriores basado en conocimientos experto proporcionó resultados significativos para la optimización y sostenibilidad en entornos de cloud computing en nueve escenarios de simulación que cubren una amplio número de casuísticas de escalas de topologías de cloud data centers. KAGWO demostró una mejora de entre el 0.53% y el 5.23% respecto a otros algoritmos como KASIA y Pittsburgh y con menos parámetros de configuración, simplificando así la implementación y reduciendo el riesgo de complejidades.

No obstante, este estudio tiene ciertas limitaciones que deben considerarse para futuras investigaciones. La precisión del modelo energético de los Cloud data centers, basado en software, puede no reflejar con exactitud un entorno real, lo que podría llevar a discrepancias entre los resultados simulados y los obtenidos en la práctica. Además, se desconoce el rendimiento de estos algoritmos en topologías y escalas extremadamente grandes de Data Centers, y la implementación a nivel de aplicación o hardware puede presentar desafíos.

El algoritmo KAGWO, como todo metaheurístico, tiene “los defectos de sus cualidades”: por un lado, proporciona soluciones buenas rápidamente y evita el reto directo de problemas NP-Hard o NP-Completo; pero, por otro lado, no garantiza siempre la solución óptima.

Además, es importante recordar que los Cloud data centers pueden experimentar fluctuaciones de tráfico y diferentes tendencias estacionales, lo cual puede afectar el rendimiento del sistema experto. Si el modelo se entrena con datos de una estación específica, las reglas derivadas pueden no ser adecuadas para el resto de las estaciones.

A pesar de estas limitaciones, esta tesis logra cubrir sus objetivos de manera eficaz. El primer objetivo, centrado en diseñar sistemas inteligentes para la migración de máquinas virtuales con un enfoque en la disponibilidad de energía renovable, se alcanzó mediante el uso de un sistema experto basado en reglas borrosas. Este sistema optimiza el consumo de energía renovable en los Cloud data centers al gestionar las migraciones de VMs entre estos centros según la disponibilidad de energía renovable.

El segundo objetivo, enfocado en aplicar técnicas de adquisición de conocimiento para mejorar la calidad de las bases expertas, se cumplió con el desarrollo y aplicación del algoritmo KAGWO. Este algoritmo metaheurístico produjo nuevo conocimiento para el sistema de reglas borrosas, generando bases de reglas con la capacidad de mejorar aún más el rendimiento inicial del sistema. Esto se tradujo en resultados significativos, reflejando un avance tanto en términos de sostenibilidad como de eficiencia energética.

En conjunto, estos resultados evidencian el logro de los objetivos de la tesis, demostrando cómo las migraciones inteligentes de VMs entre Data Centers pueden ser optimizadas para maximizar el uso de energía renovable, mientras que la adquisición de conocimiento a través de técnicas avanzadas como KAGWO mejora las bases de conocimiento y, por consiguiente, la eficiencia global del sistema. Este éxito respalda la viabilidad de las estrategias propuestas y proporciona un marco sólido para futuros trabajos en este campo.

5.2 Líneas futuras

Para avanzar en la investigación sobre la migración sostenible de máquinas virtuales en CDCs distribuidos, existen varias líneas futuras que merecen ser exploradas. A continuación, se presentan algunas áreas clave, con referencias a trabajos relevantes que pueden ser utilizados como base para investigaciones adicionales.

Mejorar la Interpretabilidad de las Bases de Reglas: Con el aumento en la complejidad y tamaño de los escenarios de data centers, la interpretabilidad de las bases de reglas se vuelve crucial. Cuando el número de reglas y su complejidad crecen, es esencial mejorar la claridad para facilitar la toma de decisiones. Existen varios trabajos que pueden servir de inspiración para abordar este reto, como los de [Alonso et al., 2006; Pancho et al. 2013; Alonso & Magdalena -a 2011; Alonso & Magdalena -b 2011; Alonso et al. 2008; Gacto et al. 2011; García-Galán -d et al. (2020)].

Técnicas de Optimización Multiobjetivo: Otra dirección para futuras investigaciones es el uso de técnicas de optimización multiobjetivo para equilibrar el consumo de energía con otros parámetros, como la interpretabilidad o la reducción del número de migraciones necesarias para optimizar la energía. Trabajos de referencia para esta línea de investigación es el de [Ishibuchi & Nojima, 2007] y [Prado -b et al., 2012].

Migración de Contenedores en lugar de Máquinas Virtuales: Una opción para reducir el tamaño de las migraciones por la red es considerar la migración de contenedores en lugar de máquinas virtuales. Esto podría mejorar la eficiencia y minimizar el tráfico de la red [R.P. Prado -a et al., 2020; Bhardwaj et al., 2023; Smimite & Afdel, 2020].

Estrategias para Adquisición de Conocimiento: Finalmente, una línea de investigación interesante sería explorar nuevas estrategias para la adquisición de conocimiento, especialmente cuando se añade una dimensión temporal al modelo de data centers. En este caso, el problema puede parecerse a una serie temporal, con estacionalidades, tendencias y ruidos. Técnicas como ARIMA, SARIMA, promedios móviles, redes neuronales LSTM (Long Short-Term Memory), o redes neuronales recurrentes (RNN) podrían ayudar a crear conjuntos de reglas adaptables según la temporada o las tendencias observadas. Esta línea futura permitiría que los sistemas expertos basados en reglas pudieran cambiar dinámicamente según las condiciones cambiantes de los data centers [Yao et al., 2023; Banerjee & Mukherjee, 2022; Pandey et al., 2024].

6 Publicaciones

6.1 Derivadas de la tesis

6.1.1 Sustainable expert virtual machine migration in dynamic clouds

[Seddiki et al., 2022] Doraid Seddiki, Sebastián García Galán, J. Enrique Muñoz Expósito, Manuel Valverde Ibañez, Tomasz Marciniak, Rocío J. -Pérez de Prado, “Sustainable expert virtual machine migration in dynamic clouds”, *Computers and Electrical Engineering*, Volume 102, 2022,108257, ISSN 0045-7906, <https://doi.org/10.1016/j.compeleceng.2022.108257>.

Resumen: La flexibilidad de operación bajo demanda en los servicios de cloud computing ha resultado en una gran popularidad y adopción generalizada. Estos servicios integran miles de ordenadores, almacenamiento y redes de comunicación, lo que implica un alto consumo de energía eléctrica. Por lo tanto, los cloud data centers basados en energía renovable están reemplazando las redes eléctricas tradicionales.

En este sentido, la carga de trabajo podría transferirse a diferentes nodos entre varios cloud data centers distribuidos geográficamente en relación con la disponibilidad de energía renovable. Esta publicación un marco de trabajo basado en CloudSim para migraciones de máquinas virtuales entre cloud data centers con respecto a la optimización de la sostenibilidad.

Además, se ha probado un enfoque para migraciones entre cloud data centers basado en un **sistema experto** en varios escenarios con energía renovable dinámicamente disponible. Los resultados experimentales muestran las mejoras del marco propuesto y cómo los sistemas expertos pueden aprovechar la disponibilidad de energía renovable en términos de sostenibilidad mientras preservan la Calidad de Servicio en términos de tiempo de ejecución.

6.1.2 Enhanced Virtual Machine Migration for Energy Sustainability Optimization in Cloud computing through Knowledge Acquisition

Doraid Seddiki, Francisco Javier Maldonado Carrascosaa, Sebastian Garcia Galan, Manuel Valverde Ibñez, Tomasz Marciniak, Nicolas Ruiz Reyes, Enhanced Virtual Machine Migration for Energy Sustainability Optimization in Cloud Computing through Knowledge Acquisition, Computers and Electrical Engineering, 2024 (Under Review)

Resumen: El cloud computing ha revolucionado la forma en que las empresas y organizaciones gestionan sus cargas de trabajo computacionales. Sin embargo, los grandes data centers que soportan los servicios en el cloud consumen mucha energía, lo que convierte la sostenibilidad energética en una preocupación crítica.

Para abordar este desafío, este artículo introduce un enfoque innovador para optimizar el consumo de energía en entornos de cloud computing a través de la **adquisición de conocimiento**. El método propuesto aprovecha una versión de Adquisición de Conocimiento del algoritmo Optimizador de Lobo Gris (KAGWO) para recopilar datos sobre la disponibilidad y el uso de energía renovable dentro de los data centers, contribuyendo a mejorar la sostenibilidad energética en el cloud computing.

Este artículo realiza un análisis comparativo entre el algoritmo KAGWO y el enfoque de Adquisición de Conocimiento con Swarm intelligence (KASIA) para destacar los beneficios y ventajas del primero. Al comparar el rendimiento de KAGWO y KASIA en términos de sostenibilidad energética, el estudio ofrece percepciones valiosas sobre la efectividad de los algoritmos basados en la adquisición de conocimiento para optimizar el uso de energía renovable en entornos de computación en el cloud.

Los resultados demuestran que el algoritmo KAGWO supera a KASIA al ofrecer capacidades más precisas de adquisición de datos, lo que resulta en una mayor sostenibilidad energética. En general, este estudio demuestra mejoras sustanciales de rendimiento que van del 0.49% al 3.30% sobre los puntos de referencia de trabajos anteriores, con una importancia particular encontrada en superar los resultados de KASIA en escenarios pequeños y medianos, mientras que iguala a KASIA en los grandes.

6.1.3 Sustainability-based Framework for Virtual Machines Migration Among Cloud data centers

D. Seddiki, S. G. Galán, E. M. Expósito, M. V. Ibañez, T. Marciniak and R. J. P. De Prado, "Sustainability-based Framework for Virtual Machines Migration Among Cloud data centers," 2021 15th International Conference on Signal Processing and Communication Systems (ICSPCS), Sydney, Australia, 2021, pp. 1-8, doi: 10.1109/ICSPCS53099.2021.9660237

Resumen: Los servicios de cloud computing han ganado gran popularidad y adopción gracias a su funcionamiento flexible bajo demanda. Estos servicios se alojan en cloud data centers que integran miles de ordenadores, redes de almacenamiento y comunicación, cuyo uso conlleva un alto consumo de energía eléctrica. Los cloud data centers basados en energía renovable están reemplazando las redes eléctricas tradicionales con fuentes de energía sostenibles, como la eólica o la fotovoltaica, por ejemplo. En este sentido, en los Data centers Modulares, la carga de trabajo podría transferirse a diferentes nodos entre varios cloud data centers distribuidos geográficamente, basándose en su acceso a la energía renovable para permitir su procesamiento en aquellas ubicaciones donde la energía renovable esté disponible.

En este sentido, para aprovechar este concepto de procesamiento sostenible, además de una gestión eficiente de la carga y los recursos, esta publicación propone un marco de trabajo basado en CloudSim para un sistema basado en la sostenibilidad para la migración de máquinas virtuales entre cloud data centers, que puede usarse para simular diferentes estrategias y hacer conclusiones sobre su comparación.

Además, para probar el comportamiento del marco propuesto, se ha probado un enfoque para las migraciones entre cloud data centers en varios escenarios de migración de máquinas virtuales, de modo que se muestra claramente cómo el marco propuesto mejora las capacidades anteriores de CloudSim, y muestra cómo las políticas de seguimiento de la energía renovable aprovechan la energía renovable en términos de consumo de energía, mientras se preserva la Calidad de Servicio en términos de tiempo de ejecución.

6.2 Relacionadas con la tesis

6.2.1 Multi-Objective Optimization of Virtual Machine Migration among Cloud data centers

F. J. Maldonado Carrascosa, S. García Galán, D. Seddiki, A. Jimenez Sánchez, M. Valverde Ibáñez, A. Marchewka Multi-Objective Optimization of Virtual Machine Migration among Cloud data centers. *Soft computing*. 2024 (Aceptado)

Resumen: La migración de cargas de trabajo entre data centers en el cloud es una tarea en evolución que requiere importantes avances. La incorporación de sistemas difusos tiene el potencial de mejorar el rendimiento y la eficiencia en el cloud computing. Este estudio aborda un problema multiobjetivo, cuyo objetivo es maximizar la interpretabilidad y el porcentaje de energía renovable consumida por un sistema de meta-Scheduling difusa en escenarios en el cloud. Para lograr este objetivo, la investigación propone un enfoque novedoso que utiliza un algoritmo de Adquisición de Conocimiento con Enfoque de Swarm intelligence. Además, se aprovecha un marco construido sobre CloudSim, que incluye capacidades de migración de máquinas virtuales basadas en un sistema experto. Además, se emplea un sistema difuso jerárquico para evaluar la interpretabilidad de la base de reglas, junto con otro algoritmo multiobjetivo, denominado Algoritmo Genético de Clasificación No Dominada II. El marco y el sistema jerárquico se emplean para realizar diversos resultados de simulación relacionados con energía renovable e interpretabilidad, mientras que los algoritmos apuntan a mejorar el rendimiento e interpretabilidad del sistema. Los resultados empíricos demuestran que es posible mejorar el rendimiento de los data centers en el cloud mientras se mejora la interpretabilidad del sistema basado en reglas difusas. El algoritmo multiobjetivo propuesto muestra un rendimiento comparable o superior al del algoritmo genético en diversos escenarios. Los resultados de la simulación indican que se pueden lograr mejoras en el rendimiento de los data centers en el cloud al tiempo que se mejora la interpretabilidad del sistema. El aumento promedio en el índice de interpretabilidad oscila entre 0,6% y 6%, con un aumento correspondiente en el uso de energía renovable que varía entre 5% y 6%.

6.2.2 Knowledge acquisition approaches for virtual machine migration in cloud computing

Francisco Javier Maldonado Carrascosa, Antonio Jimenez Sanchez, Doraid Seddiki, Sebastian Garcia Galan, Manuel Valverde Ibañez, Tomasz Marciniak, Knowledge acquisition approaches for virtual machine migration in cloud computing, Computing and Informatics, 2024. (Aceptado)

Resumen: En los últimos años, el uso de servicios de cloud computing ha aumentado significativamente, colocando una mayor demanda en la carga de trabajo, la infraestructura computacional, el almacenamiento y las redes de comunicación gestionadas por data centers. Este auge ha llevado a los investigadores a mejorar el rendimiento del cloud, centrándose en minimizar el tiempo de ejecución y los costes computacionales. Para abordar este desafío, las estrategias de Scheduling eficientes implican la asignación de la carga de trabajo del cloud entre diferentes data centers a través de la migración de máquinas virtuales. Este artículo examina y compara tres metodologías para abordar este problema: una basada en un Sistema de Inferencia Neuro-Difuso Adaptativo, otra que utiliza un Sistema Difuso de Enjambre, y una tercera que emplea un Algoritmo Genético. El estudio evalúa su efectividad para la programación de flujos de trabajo utilizando un simulador basado en CloudSim en términos de tiempo de ejecución y costes computacionales. Los resultados revelan que el sistema neuro-difuso supera a los sistemas difuso y genético en términos de tiempo de ejecución en los entornos Montage y CyberShake. Demuestra una ventaja en el coste computacional, logrando reducciones del 7.01% y 6.33% para KASIA y del 10.74% y 8.86% para Pittsburgh en Montage y CyberShake, respectivamente. Además, supera al sistema KASIA en un 50% y a Pittsburgh en un 37.5% en cuanto al número de evaluaciones de la base de reglas.

6.2.3 Comparison between Fuzzy and Neuro-Fuzzy Inference Systems in Cloud computing Scheduling

Javier Maldonado Carrascosa, Antonio Jiménez Sánchez, Sebastián García Galán, José Enrique Muñoz Expósito, Doraid Seddiki, and Adam Marchewka, Comparison between Fuzzy and Neuro-Fuzzy Inference Systems in Cloud Computing Scheduling Francisco, Book of Abstracts de EUSFLAT 2023.

Resumen: Hoy en día, los servicios de cloud computing se han vuelto populares debido a su funcionamiento flexible en múltiples campos. Este tipo de servicios opera en data centers que contienen varios equipos computacionales, almacenamiento y redes de comunicación, ofreciendo un rendimiento claramente mejorable en términos de tiempo total de ejecución. Para mitigar este problema, algunos estudios han intentado reducir la cantidad de tiempo de carga de trabajo que un data center tiene que gestionar. En este sentido, la carga de trabajo se puede asignar entre diferentes data centers para minimizar el tiempo de ejecución mediante una programación eficiente.

En este contexto, el uso de Inteligencia Artificial en el cloud computing puede mejorar el rendimiento y la eficiencia del cloud. Así, se han estudiado múltiples técnicas de IA en varios campos para lograr ese objetivo, como sistemas difusos genéticos (GFS) o aquellos basados en optimización por enjambre de partículas (PSO). En este trabajo, se han abordado técnicas de adquisición de conocimiento basadas en sistemas difusos. Además, se ha considerado una comparación entre dos sistemas diferentes. La primera metodología se basa en la Adquisición de Conocimiento con un Enfoque de Swarm intelligence (KASIA), que consiste en la adquisición de conocimiento utilizando optimización de PSO teniendo en cuenta un sistema de inferencia difusa (FIS) que utiliza reglas de Mamdani "si-entonces". Además, este artículo introduce otra metodología basada en un Sistema de Inferencia Neuro-Difusa Adaptativo (ANFIS), que consiste en un FIS llevado a cabo mediante redes adaptativas y que utiliza reglas de Sugeno "si-entonces". Por lo tanto, este artículo estudia la comparación entre KASIA y ANFIS considerando diferentes escenarios de cloud computing con respecto a el Scheduling de tareas para minimizar el tiempo de ejecución de la carga de trabajo. Para ser precisos, el sistema estudiado está compuesto por un meta-Scheduler que distribuye tareas entre data centers de manera eficiente.

Para una gestión adecuada de datos, ambos sistemas son simulados considerando varios escenarios con diferentes niveles de complejidad en un simulador que combina las características de WorkflowSim y CloudSim. Además, este simulador en el cloud utiliza estructuras de flujo de trabajo de Pegasus, como Montage o CyberShake, para recrear un sistema en el cloud con trazas de flujo de trabajo reales, que han sido consideradas en este trabajo. La configuración de los sistemas propuestos es crítica en la etapa de simulación. En este sentido, se han considerado cinco reglas compuestas por cinco antecedentes y un consecuente para la base de reglas (RB) en ambas metodologías. Los antecedentes son los MIPS, la energía consumida en reposo (pow), el tamaño de la tarea (len), la energía máxima consumida (pmax) y la utilización de la CPU (use), mientras que la salida es la selección del data center (sel) para una máquina virtual. Se han considerado tres funciones de membresía para las cinco entradas (bajo, medio y alto), mientras que se han establecido cinco para el consecuente (muy bajo, bajo, medio, alto y muy alto).

Para la técnica KASIA, se ha estudiado una población de 20 partículas y 100 iteraciones para tener un bajo coste computacional. Además, se requieren tres parámetros más para el descubrimiento de reglas de KASIA: la inercia inicial de la partícula, que es 0.9, y los factores social e individual, establecidos en 2. Por otro lado, la técnica ANFIS considera un parámetro adicional además de los de la RB: la variable de épocas, que es el número de veces que se entrena la RB, en este caso, se han contemplado 50 épocas. Cabe destacar que la configuración de ANFIS tiene menos variables que la de KASIA. Además, se han analizado algunos escenarios con una topología de red de 20 hosts virtuales y 20 máquinas virtuales, ya que esta es la configuración estándar del simulador. La salida del simulador es el tiempo transcurrido de la simulación.

Una vez establecidas las configuraciones, se deben definir los flujos de trabajo. En este sentido, se han procesado trazas de trabajo de los proyectos Montage y CyberShake con 100 trabajos para evaluar a los meta-Schedulers. Para cada simulación, se considera la variable makespan como indicador de ajuste para obtener una configuración más precisa y para ver su valor de convergencia final, donde los experimentos corresponden al promedio de la mejor solución de 30 generaciones en cada iteración. Para garantizar el buen funcionamiento del proceso de aprendizaje, ambas metodologías se validan con buenas bases de reglas. Los resultados demuestran que ANFIS supera ligeramente a KASIA (0,3%) en términos de tiempo total de ejecución (makespan) cuando se utilizan configuraciones de KASIA con bajo coste computacional. Además, ANFIS mejora los

resultados de KASIA en un 7.01% y un 6.33% en los escenarios de Montage y CyberShake, respectivamente, en términos de coste computacional en tiempo. Por lo tanto, el meta-Scheduler ANFIS también es mejor en cuanto al número de evaluaciones de la base de reglas, superando la técnica KASIA en un 60% en este sentido.

6.2.4 Optimized interpretability for Expert Virtual Machine Migrations among Data Centers using Fingrams

Doraid Seddiki, Antonio Jiménez Sánchez, Francisco Javier Maldonado Carrascosa, Sebastián García Galán, José Enrique Muñoz Expósito, and Tomasz Marciniak, Optimized interpretability for Expert Virtual Machine Migrations among Data Centers using Fingrams, Book of Abstracts de EUSFLAT 2023

Resumen: Los Cloud data centers integran una gran cantidad de hosts, almacenamiento y redes de comunicación para proporcionar servicios de cloud computing flexibles y robustos a demanda. La operación continua de los CDC implica un alto consumo de energía eléctrica. Por lo tanto, muchos de estos centros utilizan energía renovable para cubrir parte de sus necesidades de suministro eléctrico y, a su vez, este uso de energía renovable permite a los data centers modulares mover la carga de trabajo entre diferentes nodos según la disponibilidad de energía renovable utilizando técnicas de migración de carga.

La inteligencia artificial puede mejorar el rendimiento y la eficiencia en el cloud computing. Este trabajo muestra la optimización de la interpretabilidad de una Base de Reglas (RB) aprovechando un marco basado en CloudSim con capacidades de migración de Máquinas Virtuales basado en un sistema experto y la herramienta de software GUAJE. El marco se utiliza para crear un archivo de datos de prueba de entrada para GUAJE y para obtener los diferentes resultados de simulación. Implementa un sistema basado en reglas difusas (FRBS) del tipo Mamdani como algoritmo para el meta-Scheduler. Las variables de entrada son Disponibilidad de Energía Renovable en Data centers en el cloud (CDC-RA); Capacidad Computacional del Host (HCC); Disponibilidad Computacional del Host (HCA); Necesidades Computacionales Máximas de la VM (VM-MCN); y Necesidades Computacionales Actuales de la VM (VM-CNN). La salida es la idoneidad de un host para recibir una VM que migra. Cada característica se modela con 3 conjuntos de membresía difusa de tipo gaussiano (bajo, medio y alto), excepto la salida, que se modela con 5 conjuntos (muy poco adecuado, no adecuado, adecuado, muy adecuado y extremadamente adecuado).

Las reglas del meta-Scheduler fueron creadas por los autores y se presentan a continuación:

- Si HCC es alto y VM-MCN es baja, entonces la SALIDA es extremadamente adecuada.
- Si HCC es baja y VM-MCN es alta, entonces la SALIDA es muy poco adecuada.
- Si HCA es alta y VM-CNN es baja, entonces la SALIDA es extremadamente adecuada.
- Si HCA es baja y VM-CNN es alta, entonces la SALIDA es muy poco adecuada.
- Si CDC-RA es baja, entonces la SALIDA es muy poco adecuada.
- Si CDC-RA es media, entonces la SALIDA es adecuada.
- Si CDC-RA es alta, entonces la SALIDA es extremadamente adecuada.

Por otro lado, GUAJE se usa para la generación de fingrams, redes que representan gráficamente la interacción entre reglas, y para el cálculo de otros parámetros de calidad sobre la RB, como el Índice de Interpretabilidad (II), cobertura o precisión, para optimizar la interpretabilidad de la RB.

Para probar el comportamiento de las RB, se han llevado a cabo diferentes simulaciones en 3 escenarios. El primer escenario está compuesto por 350 VMs, 265 hosts y 1500 cloudlets; el segundo escenario está compuesto por 695 VMs, 530 hosts y 5000 cloudlets; finalmente, el tercer escenario consiste en 1052 VMs, 800 hosts y 10000 cloudlets.

La RB optimizada obtenida con estos métodos es la siguiente:

- Si HCC es alto y VM-MCN es baja, entonces la SALIDA es extremadamente adecuada.
- Si HCA es alta y VM-CNN es baja, entonces la SALIDA es extremadamente adecuada.
- Si CDC-RA es baja, entonces la SALIDA es muy poco adecuada.
- Si CDC-RA es media, entonces la SALIDA es adecuada.
- Si CDC-RA es alta, entonces la SALIDA es extremadamente adecuada.

Los resultados experimentales muestran que la RB optimizada proporciona resultados de simulación similares o mejorados. Usando el sistema definido, la RB original obtuvo un índice de 0.6 y la optimizada obtuvo un valor de 0.6397, lo que supone un incremento del 6.62%. Utilizando el sistema de cálculo de II de la herramienta GUAJE, la RB original obtuvo un valor de 0.631, mientras que la optimizada obtuvo un II de 0.777, lo que representa un aumento del 23.14%. Para todos los escenarios, el consumo total de energía y de energía renovable permanece similar, pero en el tiempo de ejecución hay cambios significativos. En el primer escenario, el tiempo de ejecución mejora un 1.2%. En el segundo escenario, el tiempo de ejecución se reduce un 1.6%. En el tercer escenario, el tiempo de ejecución mejora un 1.83%. Como conclusión, se pueden obtener mejoras en el rendimiento de los data centers en el cloud mientras se mejora la interpretabilidad del FRBS correspondiente.

7 Bibliografia

1. [Abdel-Basset, 2018] Abdel-Basset, M., Abdle-Fatah, L., Sangaiah, A.K. An improved Lévy based whale optimization algorithm for bandwidth-efficient virtual machine placement in cloud computing environment. *Cluster Computing* 22, 8319–8334 (2018)
2. [Ahamad et al., 2022] S. Ahamad, M. Mohseni, V. Shekher, G. F. Smaisim, A. Tripathi and J. Alanya-Beltran, "A Detailed Analysis of the Critical Role of artificial intelligence in Enabling High-Performance Cloud computing Systems," 2022 2nd International Conference on Advance Computing and Innovative Technologies in Engineering (ICACITE), Greater Noida, India, 2022, pp. 156-159, doi: 10.1109/ICACITE53722.2022.9823679
3. [Ahmad et al., 2015] Ahmad, R. W., Gani, A., Hamid, S. H. A., Shiraz, M., Yousafzai, A., & Xia, F. (2015). A survey on virtual machine migration and server consolidation frameworks for cloud data centers. *Journal of network and computer applications*, 52, 11-25.
4. [Ajana & Singh, 2019] Anjana, & Singh, A. (2019). Security concerns and countermeasures in cloud computing: a qualitative analysis. *International Journal of Information Technology*, 11(4), 683-690.
5. [Ali et al, 2016] A. Ali, Li Lu, Y. Zhu, and J. Yu. An Energy Efficient Algorithm for Virtual. Machine Allocation in Cloud Datacenters. *CCIS 626*, pp. 61–72, 2016. DOI:
6. [Ali et al., 2020] Ali, S., Hafeez, Y., Jhanjhi, N. Z., Humayun, M., Imran, M., Nayyar, A., ... & Ra, I. H. (2020). Towards pattern-based change verification framework for cloud-enabled healthcare component-based. *Ieee Access*, 8, 148007-148020.
7. [Alizadeh et al., 2014] Alizadeh, M., Edsall, T., Dharmapurikar, S., Vaidyanathan, R., Chu, K., Fingerhut, A., ... & Varghese, G. (2014, August). CONGA: Distributed congestion-aware load balancing for datacenters. In *Proceedings of the 2014 ACM conference on SIGCOMM* (pp. 503-514).
8. [Al-Mahturi & Santoso, 2021] Ayad Al-Mahturi, Fendy Santoso, Matthew A. Garratt, Sreenatha G. Anavatti, Chapter 2 - Modeling and control of a quadrotor unmanned aerial vehicle using type-2 fuzzy systems, Editor(s): Anis Koubaa, Ahmad Taher Azar, In *Advances in Nonlinear Dynamics and Chaos (ANDC)*,

- Unmanned Aerial Systems, Academic Press, 2021, Pages 25-46, ISBN 9780128202760, <https://doi.org/10.1016/B978-0-12-820276-0.00009-1>.
9. [Alonso & Magdalena -a, 2011] J. M. Alonso and L. Magdalena, "Generating understandable and accurate fuzzy rule-based systems in a java environment", Lecture Notes in Artificial Intelligence - 9th International Workshop on Fuzzy logic and Applications LNAI6857, pp. 212-219, 2011.
 10. [Alonso & Magdalena -b, 2011] J. M. Alonso and L. Magdalena, "HILK++: an interpretability-guided fuzzy modeling methodology for learning readable and comprehensible fuzzy rule-based classifiers", *Soft Computing*, vol. 15, no. 10, pp. 1959-1980, 2011.
 11. [Alonso et al., 2006] J.M Alonso, L. Magdalena, S. Guillaume, "A hierarchical fuzzy system for assessing interpretability of linguistic knowledge bases in classification problems", Conference Paper, January 2006.
 12. [Alonso et al., 2008] J. M. Alonso, L. Magdalena and S. Guillaume, "HILK: a new methodology for designing highly interpretable linguistic knowledge bases using the fuzzy logic formalism", *International Journal of Intelligent Systems*, vol. 23, no. 7, pp. 761-794, 2008.
 13. [Analysts, 2021] Analysts, G.I.: I. Internet Data Centers - Global Market Trajectory & Analytics. (2021)
 14. [Armbrust et al., 2010] Armbrust, M., et al. (2010). Cloud computing: The Business Perspective. *ACM Transactions on Computer Systems (TOCS)*, 28(4), 1-39.
 15. [Azaza & Tanougast, 2016] M. Azaza, C. Tanougast, E. Fabrizio, A. Mami, Smart greenhouse fuzzy logic based control system enhanced with wireless data monitoring, *ISA Transactions*, Volume 61, 2016, Pages 297-307, ISSN 0019-0578, <https://doi.org/10.1016/j.isatra.2015.12.006>.
 16. [B. V. Vivek and S. M. Kusuma, 2023] B. V. Vivek and S. M. Kusuma, "An Effective Quality Machine Learning to Predict Industry Equipment Failures," 2023 IEEE International Conference on Electronics, Computing and Communication Technologies (CONECCT), Bangalore, India, 2023, pp. 1-6, doi: 10.1109/CONECCT57959.2023.10234823.
 17. [Baalamurugan & Bhanu, 2020] Baalamurugan, K.M., Vijay Bhanu, S.: A multi-objective krill herd algorithm for virtual machine placement in cloud computing. *J. Supercomputing*. 76, 4525–4542 (2020)

18. [Bach-Nutman, 2020] Bach-Nutman, M. (2020). Understanding The Top 10 OWASP Vulnerabilities. ArXiv, abs/2012.09960.
19. [Bahani & Moujabbir, 2021] Khalid Bahani, Mohammed Moujabbir, Mohammed Ramdani, An accurate fuzzy rule-based classification systems for heart disease diagnosis, *Scientific African*, Volume 14, 2021, e01019, ISSN 2468-2276, <https://doi.org/10.1016/j.sciaf.2021.e01019>.
20. [Baliga, 2011] Baliga, J., Ayre, R.W.A., Hinton, K., Tucker, R.S. Green cloud computing: Balancing energy in processing, storage, and transport. *Proceedings of the IEEE* 99, 149–167 (2011)
21. [Baliyan & Kumar, 2016] Baliyan, N., & Kumar, S. (2016). A Hierarchical Fuzzy System for Quality Assessment of Semantic Web Application as a Service. *ACM SIGSOFT Software Engineering Notes*, 41, 1 - 7.
22. [Banerjee & Mukherjee, 2022] Banerjee, S., Mukherjee, S. (2022). A Comparative Study of Seasonal-ARIMA and RNN (LSTM) on Time Series Temperature Data Forecasting. In: Ranganathan, G., Bestak, R., Palanisamy, R., Rocha, Á. (eds) *Pervasive Computing and Social Networking. Lecture Notes in Networks and Systems*, vol 317. Springer, Singapore. https://doi.org/10.1007/978-981-16-5640-8_25
23. [Barati & Sharifian, 2015] Barati, M., Sharifian, S.: A hybrid heuristic-based tuned support vector regression model for cloud load prediction. *J. Supercomputing*. 71, 4235–4259 (2015)
24. [Bard, 1996] J. F. Bard, "Heuristic Scheduling Systems with Applications to Production Systems and Project Management [Book Reviews]," in *Proceedings of the IEEE*, vol. 84, no. 2, pp. 324-, Feb. 1996, doi: 10.1109/JPROC.1996.482234.
25. [Barret, 2010] Diane Barrett, Gregory Kipper, 1 - How Virtualization Happens, Editor(s): Diane Barrett, Gregory Kipper, *Virtualization and Forensics*, Syngress, 2010, Pages 3-24, ISBN 9781597495578, <https://doi.org/10.1016/B978-1-59749-557-8.00001-1>.
26. [Basireddy et al., 2018] Reddy Basireddy, K., Wachter, E.W., Al-Hashimi, B.M., Merrett, G. Workload-Aware runtime energy management for HPC Systems. *Proceedings – 2018 International Conference on High Performance Computing and Simulation, HPCS* 292–299 (2018) <https://doi.org/10.1109/HPCS.2018.00057>

27. [Belady & Malone, 2007] Belady, C., & Malone, C. (2007). Metrics and an Infrastructure Model to Evaluate Data Center Efficiency.
28. [Beloglazov et al., 2011] Beloglazov, A., Buyya, R., Lee, Y.C., Zomaya, A.A.: Taxonomy and survey of energy-efficient data centers and cloud computing systems. *Adv. Computers* 82, 47–111 (2011)
29. [Bharany et al., 2022] Bharany S, Sharma S, Khalaf OI, Abdulsahib GM, Al Humaimeedy AS, Aldhyani THH, Maashi M, Alkahtani H. A Systematic Survey on Energy-Efficient Techniques in Sustainable Cloud Computing. *Sustainability*. 2022; 14(10):6256. <https://doi.org/10.3390/su14106256>
30. [Bhardwaj et al., 2023] Bhardwaj, A., Singh, A.P., Sharma, P., Abid, K., Gupta, U. (2024). Performance Evaluation of Virtual Machine and Container-Based Migration Technique. In: Swaroop, A., Polkowski, Z., Correia, S.D., Virdee, B. (eds) *Proceedings of Data Analytics and Management. ICDAM 2023. Lecture Notes in Networks and Systems*, vol 785. Springer, Singapore. https://doi.org/10.1007/978-981-99-6544-1_41
31. [Bianchini et al., 2020] Ricardo Bianchini, Marcus Fontoura, Eli Cortez, Anand Bonde, Alexandre Muzio, Ana-Maria Constantin, Thomas Moscibroda, Gabriel Magalhaes, Girish Bablani, and Mark Russinovich. 2020. Toward ML-centric cloud platforms. *Commun. ACM* 63, 2 (February 2020), 50–59. <https://doi.org/10.1145/3364684>
32. [Bilal et al., 2014] K. Bilal, S.U.R. Malik, O. Khalid, A. Hameed, E. Alvarez, V. Wijaysekara et al. A taxonomy and survey on green data center networks. *Future Generation Computer Systems*, vol.36, pp. 189–208, 2014.
33. [Borri & Concilio, 1998] Dino Borri, Grazia Concilio, Emilia Conte, A fuzzy approach for modelling knowledge in environmental systems evaluation, *Computers, Environment and Urban Systems*, Volume 22, Issue 3, 1998, Pages 299-313, ISSN 0198-9715, [https://doi.org/10.1016/S0198-9715\(98\)00045-3](https://doi.org/10.1016/S0198-9715(98)00045-3).
34. [Brogi et al., 2019] Brogi A, Forti S, Guerrero C, Lera I (2019) How to place your apps in the fog—state of the art and open challenges
35. [Brown, 2007] Brown R. Report to congress on server and data center energy efficiency public law 109-431. *Environ. Prot.*, pp. 109:431, 2007.
36. [Buschhoff et al., 2019] Buschhoff, M., Falkenberg, R., & Spinczyk, O. (2019). Energy-aware device drivers for embedded operating systems. *ACM SIGBED Review*, 16(3), 8-13.

37. [Cao et al., 2014] Cao, J., Fu, J., Li, M., & Chen, J. (2014). CPU load prediction for cloud environment based on a dynamic ensemble model. *Software: Practice and Experience*, 44.
38. [Cetinski & Juric, 2015] Cetinski, K., Juric, M.B.: AME-WPC: Advanced model for efficient workload prediction in the cloud. *J. Netw. Comput. Appl.* 55, 191–201 (2015)
39. [Chopra & Gupta, 2023] Chopra, P., Gupta, M. (2023). Fuzzy logic and ANN in an Artificial Intelligent Cloud: A Comparative Study. In: Rajakumar, G., Du, KL., Rocha, Á. (eds) *Intelligent Communication Technologies and Virtual Mobile Networks. ICICV 2023. Lecture Notes on Data Engineering and Communications Technologies*, vol 171. Springer, Singapore. https://doi.org/10.1007/978-981-99-1767-9_41
40. [Clark et al., 2005] Clark, Christopher & Fraser, Keir & Hand, Steven & Hansen, Jacob & Jul, Eric & Limpach, Christian & Pratt, Ian & Warfield, Andrew. (2005). *Live Migration of Virtual Machines*.
41. [Compton, 2002] Compton K, Hauck S (2002) Reconfigurable computing: a survey of systems and software. 34(2):171–210
42. [Cook, 2018] Cook, B. (2018). Formal reasoning about the security of amazon web services. In *Computer Aided Verification: 30th International Conference, CAV 2018, Held as Part of the Federated Logic Conference, FloC 2018, Oxford, UK, July 14-17, 2018, Proceedings, Part I 30* (pp. 38-47). Springer International Publishing.
43. [Cordón, 2001] Cordón, O., Herrera, F., Hoffmann, F., & Magdalena, L. (2002). Genetic Fuzzy Systems - Evolutionary Tuning and Learning of Fuzzy Knowledge Bases. *Advances in Fuzzy Systems - Applications and Theory*.
44. [Cotes-Ruiz et al., 2017] I.T. Cotes-Ruiz, R.P. Prado, S. García-Galán, J.E. Muñoz-Expósito, N. Ruiz-Reyes Dynamic Voltage Frequency Scaling Simulator for Real Workflows Energy-Aware Management in Green Cloud computing *Plos One*, vol. 12, no. 1, 2017 DOI: 10.1371/journal.pone.0169803
45. [Ctrls, 2020] Solar Powered Datacenters Drive Sustainable Growth - CtrlS Blog: <https://www.ctrls.in/blog/solar-powered-datacenters-drive-sustainable-growth/>
46. [Derdus et al., 2019] Derdus, K.M., Omwenga, V., Ogao, P. Statistical Techniques for Characterizing Cloud Workloads: A Survey. *International Journal of Computer and Information Technology* 2279–0764(2019)

47. [Dey et al., 2013] Dey, Namrata & Mandal, Ria & Subashini, M. (2013). Design and Implementation of a Water Level Controller using Fuzzy logic. *International Journal of Engineering and Technology*. 5. 2277-2285.
48. [Dong, 2013] Dong, Y. (2013). U.S. Patent No. 8,533,713. Washington, DC: U.S. Patent and Trademark Office.
49. [Dorigo & Gambardella, 1997] Dorigo, M., & Gambardella, L. M. (1997). Ant colony system: A cooperative learning approach to the traveling salesman problem. *IEEE Transactions on Evolutionary Computation*, 1(1), 53-66.
50. [Embedded Data Centers] Embedded Data Centers: | Products | ENERGY STAR. https://www.energystar.gov/products/office_equipment/data_center_storage/data_center_energy_efficiency/embedded_data_centers
51. [Fard & Akbari-Zadeh, 2014] Fard, A.K., Akbari-Zadeh, M.R.: A hybrid method based on wavelet, ANN and ARIMA model for short-term load forecasting. *J. Experimental Theoretical Artif. Intell.* 26, 167–182 (2014)
52. [Fiona, 2021] Fiona, B., Ballarat, C.: *International Review of Energy Efficiency in Data Centres Acknowledgements*. (2021)
53. [Flynn, 1966] Flynn, M. J. (1966). Very high-speed computing systems. *Proceedings of the IEEE*, 54(12), 1901-1909.
54. [Foster et al., 2001] Foster I, Kesselman C, Tuecke S (2001) The anatomy of the grid. *Hand Clin* 17(4):525–532
55. [Gacto et al., 2011] M. J. Gacto, R. Alcalá and F. Herrera, "Interpretability of linguistic fuzzy rule-based systems: An overview of interpretability measures", *Inf. Sci.*, vol. 181, no. 20, pp. 4340-4360, 2011.
56. [Garcia-Galan - e et al., 2009] S. Garcia, A. Fernandez, J. Luengo, and F. Herrera, "A study of statistical techniques and performance measures for genetics-based machine learning: accuracy and interpretability," *Soft Comput.*, 13(10), 959–977, 2009.
57. [Garcia-Galan -a et al., 2015] S. García-Galán, R.P. Prado, J.E. Muñoz Expósito, Rules discovery in fuzzy classifier systems with PSO for scheduling in grid computational infrastructures, *Applied Soft Computing*, Volume 29, 2015, Pages 424-435, ISSN 1568-4946, <https://doi.org/10.1016/j.asoc.2014.11.064>.
58. [Garcia-Galan -b et al., 2014] S. García-Galán, R.P. Prado, J.E. Muñoz-Expósito *Swarm Fuzzy Systems: Knowledge Acquisition in Fuzzy Systems and its*

- Applications in Grid Computing IEEE-Transactions on Knowledge and Data Engineering, vol. 26, no. 7, pp. 1791-1804, 2014 DOI: 10.1109/TKDE.2013.118
59. [Garcia-Galan -c et al., 2012] S. García-Galán, R.P. Prado, J.E. Muñoz-Expósito Fuzzy Scheduling with Swarm intelligence-Based Knowledge Acquisition for Grid Computing Engineering Applications of Artificial Intelligence, vol. 25, no. 2, pp. 359-375, 2012 DOI: 10.1016/j.engappai.2011.11.002
 60. [Garcia-Galan -d et al., 2020] Sebastian Garcia-Galan, Mouad Seddiki, Rocio P. Prado, Jose Enrique Munoz-Exposito, Adam Marchewka, and N. Ruiz-Reyes Relevance of Using Interpretability Indexes for the Design of Schedulers in Cloud computing Systems IEEE World Congress on Computational Intelligence (WCCI) 2020 19-24th July, 2020. Glasgow (UK), (2020)
 61. [Genez & Bittencourt, 2019] T. A. L. Genez, L. F. Bittencourt, N. L. S. d. Fonseca and E. R. M. Madeira, "Estimation of the Available Bandwidth in Inter-Cloud Links for Task Scheduling in Hybrid Clouds," in IEEE Transactions on Cloud computing, vol. 7, no. 1, pp. 62-74, 1 Jan. -March 2019, doi: 10.1109/TCC.2015.2469650.
 62. [Ghafari et al., 2022] Ghafari, R., Kabutarkhani, F.H., Mansouri, N.: Task scheduling algorithms for energy optimization in cloud environment: a comprehensive review. Clust. Comput. 25(2), 1035–1093 (2022)
 63. [Gharehpasha, 2020] Gharehpasha, S., Masdari, M., Jafarian, A. (2020) Power efficient virtual machine placement in cloud data centers with a discrete and chaotic hybrid optimization algorithm. Cluster Computing 24, 1293–1315
 64. [Gholami et al., 2021] Ali Gholami, Daobin Wang, Seyed Rasoul Davoodi, Zong Tian, An Adaptive Neural Fuzzy Inference System model for freeway travel time estimation based on existing detector facilities, Case Studies on Transport Policy, Volume 9, Issue 4, 2021, Pages 1600-1606, ISSN 2213-624X, <https://doi.org/10.1016/j.cstp.2021.08.009>.
 65. [Ghorbani et al., 2015] Ghorbani, S., Godfrey, B., Ganjali, Y., & Firoozshahian, A. (2015, November). Micro load balancing in data centers with DRILL. In Proceedings of the 14th ACM Workshop on Hot Topics in Networks (pp. 1-7).
 66. [Gill & Tuli, 2020] Gill SS, Tuli S, Toosi AN, Cuadrado F, Garraghan P, Bahsoon R, Lutfiyya H et al (2020) ThermoSim: deep learning based framework for modeling and simulation of thermal-aware resource management for cloud computing environments. J Syst Softw 164:110596

67. [Gill et al., 2019] Gill SS, Garraghan P, Buyya R (2019) ROUTER: Fog enabled cloud based intelligent resource management approach for smart home IoT devices. *J Syst Softw* 154:125–138
68. [Gill et al., 2020] Gill SS, Ouyang X, Garraghan P (2020) Tails in the cloud: a survey and taxonomy of straggler management within large-scale cloud data centres. *J Supercomput* 50:10050–10089
69. [Google, 2012] Google, Google's green data centers. Network pop case study, 2012. URL (<https://www.google.com/about/datacenters/efficiency/external>).
70. [Gupta & Garg, 2017] A. Gupta and R. Garg, "Load Balancing Based Task Scheduling with ACO in Cloud computing," 2017 International Conference on Computer and Applications (ICCA), Doha, Qatar, 2017, pp. 174-179, doi: 10.1109/COMAPP.2017.8079781.
71. [H. Ren, 2023] H. Ren, X. Xu and Y. Zeng, "AI-based Green Energy Saving Solutions for IDC," 2023 International Conference on Mobile Internet, Cloud computing and Information Security (MICCIS), Nanjing, China, 2023, pp. 95-101, doi: 10.1109/MICCIS58901.2023.00021.
72. [Hajek, 2019] Hajek, P. (2019). Interpretable Fuzzy Rule-Based Systems for Detecting Financial Statement Fraud. In: MacIntyre, J., Maglogiannis, I., Iliadis, L., Pimenidis, E. (eds) *Artificial Intelligence Applications and Innovations. AIAI 2019. IFIP Advances in Information and Communication Technology*, vol 559. Springer, Cham. https://doi.org/10.1007/978-3-030-19823-7_36
73. [Hayat et al., 2018] Hayat, B., Kim, K.H. & Kim, KI. A study on fuzzy logic-based cloud computing. *Cluster Comput* 21, 589–603 (2018). <https://doi.org/10.1007/s10586-017-0953-x>
74. [He et al., 2015] He, K., Rozner, E., Agarwal, K., Felter, W., Carter, J., & Akella, A. (2015). Presto: Edge-based load balancing for fast datacenter networks. *ACM SIGCOMM Computer Communication Review*, 45(4), 465-478.
75. [Helali & Omri, 2021] Leila Helali, Mohamed Nazih Omri, A survey of data center consolidation in cloud computing systems, *Computer Science Review*, Volume 39, 2021, 100366, ISSN 1574-0137, <https://doi.org/10.1016/j.cosrev.2021.100366>.
76. [Herzog et al., 2021] Herzog, B., Hügél, F., Reif, S., Hönig, T., Schröder-Preikschat, W.: Automated selection of energy-efficient operating system configurations. *Energy* (2021). <https://doi.org/10.1145/3447555.3465327>

77. [Hines & Gopalan, 2009] Michael R. Hines and Kartik Gopalan. 2009. Post-copy based live virtual machine migration using adaptive pre-paging and dynamic self-ballooning. In Proceedings of the 2009 ACM SIGPLAN/SIGOPS international conference on Virtual execution environments (VEE '09). Association for Computing Machinery, New York, NY, USA, 51–60. <https://doi.org/10.1145/1508293.1508301>
78. [Ho et al., 2015] Ho, T.T.N., Pernici, B.: A data-value-driven adaptation framework for energy efficiency for data intensive applications in clouds. IEEE Conf. Technol. Sustainabil. (2015). <https://doi.org/10.1109/SUSTECH.2015.7314320>
79. [Ho et al., 2016] Ho, T.T.N., Gribaudo, M., Pernici, B.: Characterizing Energy per-Job in Cloud Applications. Electron. 2016. 5, 90 (2016)
80. [Hochreiter & Schmidhuber, 1997] Hochreiter, S., & Schmidhuber, J. (1997). Long short-term memory. Neural computation, 9(8), 1735-1780.
81. [Holland, 1992] Holland, J. H. (1992). Adaptation innatural and artificial systems: An introductory analysis with applications to biology, control, and artificial intelligence. MIT press.
82. [Hossain et al., 2020] M. K. Hossain, M. Rahman, A. Hossain, S. Y. Rahman and M. M. Islam, "Active & Idle Virtual Machine Migration Algorithm- a new Ant Colony Optimization approach to consolidate Virtual Machines and ensure Green Cloud computing," 2020 Emerging Technology in Computing, Communication and Electronics (ETCCE), Bangladesh, 2020, pp. 1-6, doi: 10.1109/ETCCE51779.2020.9350915.
83. [Houssein et al., 2021] Houssein, E. H., Gad, A. G., Wazery, Y. M., & Suganthan, P. N. (2021). Task scheduling in cloud computing based on meta-heuristics: review, taxonomy, open challenges, and future trends. Swarm and Evolutionary Computation, 62, 100841.
84. [Hu et al., 2013] Hu, R., Jiang, J., Liu, G., Wang, L., KSwSVR: A new load forecasting method for efficient resources provisioning in cloud. in Proceedings - IEEE 10th International Conference on Services Computing, SCC 2013 120–127 doi: (2013). <https://doi.org/10.1109/SCC.2013.67>
85. [Imran, 2022] Muhammad Imran, Muhammad Ibrahim, Muhammad Salah Ud Din, Muhammad Atif Ur Rehman, Byung Seo Kim, Live virtual machine migration: A survey, research challenges, and future directions, Computers and

- Electrical Engineering, Volume 103, 2022, 108297, ISSN 0045-7906, <https://doi.org/10.1016/j.compeleceng.2022.108297>.
86. [Ishibuchi & Nojima, 2007] H. Ishibuchi and Y. Nojima, "Analysis of interpretability–accuracy tradeoff of fuzzy systems by multiobjective fuzzy genetics-based machine learning", *International Journal of Approximate Reasoning*, vol. 44, pp. 4-31, 2007.
87. [Ismaeel et al, 2019] Ismaeel, S., Al-Khazraji, A., Miri, A. An efficient workload clustering framework for large-scale data centers. 8th International Conference on Modeling Simulation and Applied Optimization, ICMSAO 2019 (2019) doi: (2019). <https://doi.org/10.1109/ICMSAO.2019.8880305>
88. [Jalaei et al., 2020] Jalaei, N., Safi-Esfahani, F. VCSP: virtual CPU scheduling for post-copy live migration of virtual machines. *Int. j. inf. tecnol.* 13, 239–250 (2021). <https://doi.org/10.1007/s41870-020-00483-z>
89. [Jamali et al., 2017] Jamali, S., Malektaji, S., Analoui, M.: An imperialist competitive algorithm for virtual machine placement in cloud computing. *J. Experimental Theoretical Artif. Intell.* 29, 575–596 (2017)
90. [Janardhanan & Barrett, 2017] Janardhanan, D., Barrett, E. CPU workload forecasting of machines in data centers using LSTM recurrent neural networks and ARIMA models. 12th International Conference for Internet Technology and Secured Transactions, ICITST 2017 55–60 (2018) doi: (2017).
91. [Jena et al., 2022] U.K. Jena, P.K. Das, M.R. Kabat, Hybridization of meta-heuristic algorithm for load balancing in cloud computing environment, *Journal of King Saud University - Computer and Information Sciences*, Volume 34, Issue 6, Part A, 2022, Pages 2332-2342, ISSN 1319-1578, <https://doi.org/10.1016/j.jksuci.2020.01.012>.
92. [Jheng et al., 2014] Jheng, J.J., Tseng, F.H., Chao, H.C., Chou, L. der. A novel VM workload prediction using grey forecasting model in cloud data center. *International Conference on Information Networking* 40–45 doi: (2014). <https://doi.org/10.1109/ICOIN.2014.6799662>
93. [Jiang & Tang, 2014] Yuncheng Jiang, Yong Tang, An interval type-2 fuzzy model of computing with words, *Information Sciences*, Volume 281, 2014, Pages 418-442, ISSN 0020-0255, <https://doi.org/10.1016/j.ins.2014.05.055>.
94. [Jiang et al., 2013] Jiang, J., Lu, J., Zhang, G., Long, G. Optimal cloud resource auto-scaling for web applications. *Proceedings – 13th IEEE/ACM International*

- Symposium on Cluster, Cloud, and Grid Computing, CCGrid 2013 58–65 doi: (2013). <https://doi.org/10.1109/CCGRID.2013.73>
95. [Jiang et al., 2020] Jiang, D., Wang, Y., Lv, Z., Wang, W., Wang, H.: An Energy-Efficient Networking Approach in Cloud Services for IIoT Networks. *IEEE J. Sel. Areas Commun.* 38, 928–941 (2020)
 96. [Jumnal & Kumar, 2021] Jumnal, A., Dilip Kumar, S.M. Optimal VM placement approach using fuzzy reinforcement learning for cloud data centers. *Proceedings of the 3rd International Conference on Intelligent Communication Technologies and Virtual Mobile Networks, ICICV 2021* 29–35 doi: (2021).
 97. [Kaaouache & Bouamama, 2018] Kaaouache, M.A., Bouamama, S.: An energy-efficient VM placement method for cloud data centers using a hybrid genetic algorithm. *J. Syst. Inform. Technol.* 20, 430–445 (2018)
 98. [Kang et al., 2021] Kang, D.G.I.S.T., Alian, K.-D., Kim, M., Huh, D.G.I.S.T.D. KAIST, J. & Sung Kim, N. VIP: Virtual Performance-State for Efficient Power Man-agement of Virtual Machines. *Proceedings of the ACM Symposium on Cloud computing '18* (2021)
 99. [Kansal & Chana, 2016] Kansal, N.J., Chana, I. Energy-aware Virtual Machine Migration for Cloud computing - A Firefly Optimization Approach. *J Grid Computing* 14, 327–345 (2016). <https://doi.org/10.1007/s10723-016-9364-0>
 100. [Kant & Agarwal, 2022] Kant, S., Agarwal, D., Shukla, P.K. (2022). A Survey on Fuzzy Systems Optimization Using Evolutionary Algorithms and Swarm intelligence. In: Bansal, J.C., Engelbrecht, A., Shukla, P.K. (eds) *Computer Vision and Robotics. Algorithms for Intelligent Systems*. Springer, Singapore. https://doi.org/10.1007/978-981-16-8225-4_33
 101. [Karaboga & Basturk, 2007] Karaboga, D., & Basturk, B. (2007). A powerful and efficient algorithm for numerical function optimization: artificial bee colony (ABC) algorithm. *Journal of Global Optimization*, 39(3), 459-471.
 102. [Katal, 2023] Katal, A., Dahiya, S. & Choudhury, T. Energy efficiency in cloud computing data centers: a survey on software technologies. *Cluster Comput* 26, 1845–1875 (2023). <https://doi.org/10.1007/s10586-022-03713-0>
 103. [Kaur, 2017] Kaur, R. A. (2017). Hybrid approach for virtual machine migration in cloud computing environment. *Int. J. Adv. Res. Comput. Sci. Softw. Eng.* 7(30), 10-23956.

104. [Kaur, 2017] Kaur, Ramandeep. (2017). A Hybrid Approach for Virtual Machine Migration in Cloud computing Environment. *International Journal of Advanced Research in Computer Science and Software Engineering*. 7. 30. 10.23956/ijarcsse.v7i9.407.
105. [Kennedy & Eberhart, 1995] Kennedy, J., & Eberhart, R. (1995). Particle swarm optimization. *Proceedings of ICNN'95 - International Conference on Neural Networks*, 1942-1948.
106. [Khalil et al., 2019] Khalil, M. I. K., Ahmad, I., & Almazroi, A. A. (2019). Energy efficient indivisible workload distribution in geographically distributed data centers. *IEEE Access*, 7, 82672-82680.
107. [Khan & Tian, 2021] Tahseen Khan, Wenhong Tian, Guangyao Zhou, Shashikant Ilager, Mingming Gong, Rajkumar Buyya, Machine learning (ML)-centric resource management in cloud computing: A review and future directions, *Journal of Network and Computer Applications*, Volume 204, 2022, 103405, ISSN 1084-8045, <https://doi.org/10.1016/j.jnca.2022.103405>.
108. [Kirti, 2023] Kirti, M. Smriti, K. Shaily and D. Sudha, "Minimization of Energy Consumption in Cloud," 2023 Third International Conference on Advances in Electrical, Computing, Communication and Sustainable Technologies (ICAECT), Bhilai, India, 2023, pp. 1-6, doi: 10.1109/ICAECT57570.2023.10117732.
109. [Kluge et al., 2010] Kluge, F., Uhrig, S., Mische, J., Satzger, B., Ungerer, T. Dynamic workload prediction for soft real-time applications. *Proceedings – 10th IEEE International Conference on Computer and Information Technology, CIT- 7th IEEE International Conference on Embedded Software and Systems, ICES-2010, ScalCom-2010 1841–1848 (2010) doi: (2010).* <https://doi.org/10.1109/CIT.2010.317>
110. [Kumar & Bhagwan, 2022] Kumar, R., Bhagwan, J.: A comparative study of meta-heuristic-based task scheduling in cloud computing. In: Mohan, H.D. (ed.) *Artificial intelligence and sustainable computing: Proceedings of ICSISCET 2020*, pp. 129–141. Springer Singapore, Singapore (2022)
111. [Kumar & Buyya, 2012] Kumar, S., Buyya, R.: *Green cloud computing and environmental sustainability harnessing green. Principles Practices* (2012). <https://doi.org/10.1002/9781118305393.CH16>

112. [Kumar & Mazumdar, 2016] Kumar, A.S., Mazumdar, S. Forecasting HPC workload using ARMA models and SSA. Proceedings – 2016 15th International Conference on Information Technology, ICIT 294–297 (2017) doi: (2016). <https://doi.org/10.1109/ICIT.2016.52>
113. [Kumar, 2019] Kumar, D.D.: Review on task scheduling in ubiquitous clouds. *J. IoT Soc. Mobile Anal. Cloud* 1(1), 72–80 (2019)
114. [Laganà et al., 2018] Laganà D, Mastroianni C, Meo M, Renga D. Reducing the Operational Cost of Cloud Data Centers through Renewable Energy. *Algorithms*. 2018; 11(10):145. <https://doi.org/10.3390/a11100145>
115. [Le, 2020] Tuan Le, A survey of live Virtual Machine migration techniques, *Computer Science Review*, Volume 38, 2020, 100304, ISSN 1574-0137, <https://doi.org/10.1016/j.cosrev.2020.100304>.
116. [LeCun, 2015] LeCun, Y., Bengio, Y., & Hinton, G. (2015). Deep learning. *nature*, 521(7553), 436-444.
117. [Levy et al., 2017] Levy, A., et al. Multiprogramming a 64 kB Computer Safely and Efficiently. Proceedings of the 26th Symposium on Operating Systems Principles (2017) <https://doi.org/10.1145/3132747>
118. [Li & Qian, 2013] Li, X., Qian, Z., Lu, S., Wu, J.: Energy efficient virtual machine placement algorithm with balanced and improved resource utilization in a data center. *Math. Comput. Model.* 58, 1222–1235 (2013)
119. [Li et al., 2013] Li, S. A workload prediction-based multi-VM provisioning mechanism in cloud computing. 1–6 (2013)
120. [Li et al., 2015] Li, X.K., Gu, C.H., Yang, Z.P., Chang, Y.H. Virtual machine placement strategy based on discrete firefly algorithm in cloud environments. 12th International Computer Conference on Wavelet Active Media Technology and Information Processing, ICCWAMTIP (2015). <https://doi.org/10.1109/ICCWAMTIP.2015.7493907>
121. [Li et al., 2020] Li, S., Ben-Nun, T., Girolamo, S., di, Alistarh, D., Hoefler, T. Taming Unbalanced Training Workloads in Deep Learning with Partial Collective Operations. (2020)
122. [Liang et al., 2018] Liang, Paul & Liu, Ziyin & Zadeh, AmirAli & Morency, Louis-Philippe. (2018). Multimodal Language Analysis with Recurrent Multistage Fusion. 150-161. 10.18653/v1/D18-1014.

123. [Lin et al., 2018] Lin, M.-H., Tsai, J.-F., Hu, Y.-C., Su, T.-H. Optimal Allocation of Virtual Machines in Cloud computing. *Symmetry* Vol. 10, Page 756 10, 756 (2018). (2018)
124. [Liu et al., 2011] Liu, P., Zhou, Z., Peng, Z., Liu, B., Jiang, S., & Zhou, W. (2011). PaaS: A revolution in cloud computing. *Journal of Software*, 6(8), 1356-1364.
125. [Liu et al., 2020] Nayu Liu, Xian Sun, Hongfeng Yu, Wenkai Zhang, and Guangluan Xu. 2020. Multistage Fusion with Forget Gate for Multimodal Summarization in Open-Domain Videos. In *Proceedings of the 2020 Conference on Empirical Methods in Natural Language Processing (EMNLP)*, pages 1834–1845, Online. Association for Computational Linguistics.
126. [Lix et al., 2018] Li X et al (2018) Holistic virtual machine scheduling in cloud datacenters towards minimizing total energy. *IEEE Trans Parallel Distrib Syst* 29:1317–1331
127. [Long et al., 2020] Long, S., et al. A Reinforcement Learning-Based Virtual Machine Placement Strategy in Cloud Data Centers. *Proceedings – 2020 IEEE 22nd International Conference on High Performance Computing and Communications, IEEE 18th International Conference on Smart City and IEEE 6th International Conference on Data Science and Systems, HPCC-SmartCity-DSS 2020* 223–230 doi: (2020). <https://doi.org/10.1109/HPCC-SMARTCITY-DSS50907.2020.00028>
128. [Loukis et al., 2019] Loukis, E., Janssen, M., & Mintchev, I. (2019). Determinants of software-as-a-service benefits and impact on firm performance. *Decision Support Systems*, 117, 38-47.
129. [M. Romero et al., 2014] M. Romero, H. Hasselqvist, G. Svensson. Supercomputers keeping people warm in the winter. *Proceedings of the 2nd International Conference on ICT for Sustainability (ICTS)*, August 24–27, 2014. Stockholm, Sweden, pp. 324–32, 2014.
130. [M. Yekta and H. S. Shahhoseini, 2023] M. Yekta and H. S. Shahhoseini, "A Review on Machine Learning Methods for Workload Prediction in Cloud computing," 2023 13th International Conference on Computer and Knowledge Engineering (ICCKE), Mashhad, Iran, Islamic Republic of, 2023, pp. 306-311, doi: 10.1109/ICCKE60553.2023.10326297.

131. [Malik et al., 2019] Malik, M., et al. ECoST: Energy-efficient co-locating and self-tuning mapreduce applications. ACM International Conference Proceeding Series (2019). <https://doi.org/10.1145/3337821.3337834>
132. [Manvi & Shyam, 2014] Sunilkumar S. Manvi, Gopal Krishna Shyam, Resource management for Infrastructure as a Service (IaaS) in cloud computing: A survey, Journal of Network and Computer Applications, Volume 41, 2014, Pages 424-440, ISSN 1084-8045, <https://doi.org/10.1016/j.jnca.2013.10.004>.
133. [Masanet et al., 2020] Masanet, E., Shehabi, A., Lei, N., Smith, S., & Koomey, J. (2020). Recalibrating global data center energy-use estimates. Science, 367(6481), 984-986.
134. [Mathematics, 2021] Mathematics, K.K.-T.J. of C. and & undefined. Forecasting of Cloud computing Services Workload using Machine Learning. turcomat.org 12, 4841–4846 (2021). (2021)
135. [Mather, 2009] Mather, T., Kumaraswamy, S., & Latif, S. (2009). Cloud security and privacy: An enterprise perspective on risks and compliance. O'Reilly Media, Inc.
136. [Meijer, 2010] G. Meijer. Cooling energy-hungry data centers. Science; vol. 328(5976), pp. 318–9, 2010.
137. [Mell & Grance, 2011] Mell, P., & Grance, T. (2011). The NIST definition of cloud computing (NIST Special Publication 800-145). National Institute of Standards and Technology.
138. [Messias et al., 2016] Messias, V.R., Estrella, J.C., Ehlers, R. et al. Combining time series prediction models using genetic algorithm to autoscaling Web applications hosted in the cloud infrastructure. Neural Comput & Applic 27, 2383–2406 (2016). <https://doi.org/10.1007/s00521-015-2133-3>
139. [Mikolov et al., 2010] Mikolov, T., Karafiát, M., Burget, L., Cernocký, J., & Khudanpur, S. (2010, September). Recurrent neural network based language model. In Interspeech (Vol. 2, No. 3, pp. 1045-1048).
140. [Minarolli & Freisleben, 2011] Minarolli, D., & Freisleben, B. (2011, September). Utility-driven allocation of multiple types of resources to virtual machines in clouds. In 2011 IEEE 13th Conference on Commerce and Enterprise Computing (pp. 137-144). IEEE.
141. [Mirjalili & Lewis, 2016] Mirjalili, S., & Lewis, A. (2016). The whale optimization algorithm. Advances in Engineering Software, 95, 51-67.

142. [Mirjalili et al., 2014] Mirjalili, S., Mirjalili, S. M., & Lewis, A. (2014). Grey wolf optimizer. *Advances in Engineering Software*, 69, 46-61.
143. [Mishra et al., 2020] Mishra, S. K., Mishra, S., Alsayat, A., Jhanjhi, N. Z., Humayun, M., Sahoo, K. S., & Luhach, A. K. (2020). Energy-aware task allocation for multi-cloud networks. *IEEE Access*, 8, 178825-178834.
144. [Mittal et al., 2020] Kanika Mittal, Amita Jain, Kunwar Singh Vaisla, Oscar Castillo, Janusz Kacprzyk, A comprehensive review on type 2 fuzzy logic applications: Past, present and future, *Engineering Applications of Artificial Intelligence*, Volume 95, 2020,103916, ISSN 0952-1976, <https://doi.org/10.1016/j.engappai.2020.103916>.
145. [Miyazaki, 2018] Miyazaki, T. Bayesian Optimization of HPC Systems for Energy Efficiency. *Lecture Notes Comput. Sci.* 10876: 44–62 (2018)
146. [Mnih et al., 2015] Mnih, V., Kavukcuoglu, K., Silver, D., Rusu, A. A., Veness, J., Bellemare, M. G., ... & Hassabis, D. (2015). Human-level control through deep reinforcement learning. *nature*, 518(7540), 529-533.
147. [Muneeswari et al., 2024] G. Muneeswari, Jhansi Bharathi Madavarapu, R. Ramani, C. Rajeshkumar, C. John Clement Singh, GEP optimization for load balancing of virtual machines (LBVM) in cloud computing, *Measurement: Sensors*, Volume 33, 2024, 101076, ISSN 2665-9174, <https://doi.org/10.1016/j.measen.2024.101076>.
148. [Munoz et al., 2015] J.E. Munoz, R. P. Prado, S. Garcia-Galan, R. Rodriguez-Reche, Adam Marchewka. Analysis and real implantation of cloud infrastructure for computing laboratories virtualization. 7th International Conference on Image Processing & Communications. Bydgoszcz, Poland, 2015.
149. [Murad et al., 2022] Murad, S. A., Muzahid, A. J. M., Azmi, Z. R. M., Hoque, M. I., & Kowsher, M. (2022). A review on job scheduling technique in cloud computing and priority rule based intelligent framework. *Journal of King Saud University-Computer and Information Sciences*, 34(6), 2309-2331.
150. [Murad et al., 2022] Murad, S.A., Muzahid, A.J.M., Azmi, Z.R.M., Hoque, M.I., Kowsher, M.: A review on job scheduling technique in cloud computing and priority rule based intelligent framework. *J. King Saud Univ. Comput. Inform. Sci.* 34(6), 2309–2331 (2022)

151. [Mya y Thein, 2011] Mya, S., & Ni, L.T. (2011). A Resource Pool Management Model using Fuzzy logic Decision Making. *International Journal of Computer Applications*, 29, 24-31.
152. [Nashaat & Ashry, 2019] Nashaat, H., Ashry, N. & Rizk, R. Smart elastic scheduling algorithm for virtual machine migration in cloud computing. *J Supercomput* 75, 3842–3865 (2019). <https://doi.org/10.1007/s11227-019-02748-2>
153. [Nguyen et al., 2017] N. T. Hieu, M. D. Francesco and A. Ylä-Jääski, "Virtual Machine Consolidation with Multiple Usage Prediction for Energy-Efficient Cloud Data Centers," in *IEEE Transactions on Services Computing*, vol. 13, no. 1, pp. 186-199, 1 Jan. -Feb. 2020, doi: 10.1109/TSC.2017.2648791.
154. [Nguyen et al., 2019] A. -T. Nguyen, T. Taniguchi, L. Eciolaza, V. Campos, R. Palhares and M. Sugeno, "Fuzzy Control Systems: Past, Present and Future," in *IEEE Computational Intelligence Magazine*, vol. 14, no. 1, pp. 56-68, Feb. 2019, doi: 10.1109/MCI.2018.2881644.
155. [Noshy et al., 2018] Noshy, M., Ibrahim, A., & Ali, H. A. (2018). Optimization of live virtual machine migration in cloud computing: A survey and future directions. *Journal of Network and Computer Applications*, 110, 1-10.
156. [Nowotny et al., 2017] Nowotny, H., Scott, P., & Gibbons, M. (2003). Introduction:'Mode 2'revisited: The new production of knowledge. *Minerva*, 41(3), 179-194.
157. [Pancho et al., 2013] D. P. Pancho, J. M. Alonso and L. Magdalena, "Quest for interpretability-accuracy trade-off supported by Fingrams into the fuzzy modeling tool GUAJE", *International Journal of Computational Intelligence Systems*, vol. 6, no. 1, pp. 46-60, 2013.
158. [Pandey et al., 2024] Pandey, B., Shukla, A., Khamparia, A. (2024). An Optimized Hybrid ARIMA-LSTM Model for Time Series Forecasting of Agricultural Production in india. In: Khamparia, A., Pandey, B., Pandey, D.K., Gupta, D. (eds) *Microbial Data Intelligence and Computational Techniques for Sustainable Computing. Microorganisms for Sustainability*, vol 47. Springer, Singapore. https://doi.org/10.1007/978-981-99-9621-6_7
159. [PanneerSelvam, 2017] J. Panneerselvam, L. Liu, J. Hardy and N. Antonopoulos, "Analysis, Modelling and Characterisation of Zombie Servers in

- Large-Scale Cloud Datacentres," in *IEEE Access*, vol. 5, pp. 15040-15054, 2017, doi: 10.1109/ACCESS.2017.2725898.
160. [Parast et al., 2022] Fatemeh Khoda Parast, Chandni Sindhav, Seema Nikam, Hadiseh Izadi Yekta, Kenneth B. Kent, Saqib Hakak, Cloud computing security: A survey of service-based models, *Computers & Security*, Volume 114, 2022, 102580, ISSN 0167-4048, <https://doi.org/10.1016/j.cose.2021.102580>.
 161. [Parmesivan et al., 2018] Parmesivan, Y.A., Hasan, S., & Muhammed, A. (2018). Performance evaluation of load balancing algorithm for virtual machine in data centre in cloud computing.
 162. [Peltz, 2003] Peltz C (2003) Web services orchestration and choreography. *IEEE Internet Comput* 36(10):46–52
 163. [Persico et al., 2017] Grimaldi, Domenico & Pescapè, Antonio & Salvi, Alessandro & Santini, Stefania & Persico, Valerio. (2017). A Fuzzy Approach Based on Heterogeneous Metrics for Scaling Out Public Clouds. *IEEE Transactions on Parallel and Distributed Systems*. PP. 1-1. 10.1109/TPDS.2017.2651810.
 164. [Pham et al., 2017] Pham, V. V. H., Liu, X., Zheng, X., Fu, M., Deshpande, S. V., Xia, W., ... & Abdelrazek, M. (2017, May). PaaS-black or white: an investigation into software development model for building retail industry SaaS. In *2017 IEEE/ACM 39th International Conference on Software Engineering Companion (ICSE-C)* (pp. 285-287). IEEE.
 165. [Pires & Barán, 2013] Pires, F.L., Barán, B. Multi-objective virtual machine placement with service level agreement: A memetic algorithm approach. *Proceedings – 2013 IEEE/ACM 6th International Conference on Utility and Cloud computing, UCC 2013* 203–210 doi: (2013). <https://doi.org/10.1109/UCC.2013.44>
 166. [Prabhakaran et al., 2021] Prabhakaran, G., Selvakumar, S.: An diverse approach on virtual machines administration and power control in multi-level implicit servers. *J. Ambient Intell. Humaniz. Comput.* (2021). <https://doi.org/10.1007/S12652-021-03013-2>
 167. [Prado -b et al., 2012] R.P. Prado, F. Hoffmann, S. García-Galán, J.E. Muñoz-Expósito, T. Bertram On Providing Quality of Service in Grid Computing through Multi-Objective Swarm-Based Knowledge Acquisition in Fuzzy

- Schedulers International Journal of Approximate Reasoning, vol. 53, no. 2, pp. 228-247, 2012 DOI: 10.1016/j.ijar.2011.10.005
168. [Prado -c et al., 2011] R.P. Prado, S. García-Galán, J. E. Muñoz-Expósito, A.J. Yuste Delgado Genetic Fuzzy Rule-Based Scheduling System for Grid Computing in Virtual Organizations Soft Computing, vol. 15, no. 7, pp. 1255-1271, 2011 DOI: 10.1007/s00500-010-0660-5
169. [Prado -d et al., 2010] R.P. Prado, S. García-Galán, J.E. Muñoz-Expósito, A.J. Yuste Delgado Knowledge Acquisition in Fuzzy Rule Based Systems with Particle Swarm Optimization IEEE-Transactions on Fuzzy Systems, vol. 18, no. 6, pp. 1083-1097, 2010 DOI: 10.1109/TFUZZ.2010.2062525
170. [Prado -e et al., 2010] R.P. Prado, S. García-Galán, A.J. Yuste Delgado, J. E. Muñoz-Expósito A Fuzzy Rule-Based Meta-Scheduler with Evolutionary Learning for Grid Computing Engineering Applications of Artificial Intelligence, vol. 23, no. 7, pp. 1072-1082, 2010 DOI: 10.1016/j.engappai.2010.07.002
171. [Prado et al., 2014] R.P. Prado, J.E. Muñoz-Expósito, S. García-Galán Flexible Fuzzy Rule Bases Evolution with Swarm intelligence for Meta-Scheduling in Grid Computing. Computing and Informatics, vol. 3, no. 4, pp. 810-830, 2014.
172. [Qazi et al., 2013] Qazi, K., Li, Y., Sohn, A. PoWER - Prediction of workload for energy efficient relocation of virtual machines. Proceedings of the 4th Annual Symposium on Cloud computing, SoCC 2013 doi: (2013). <https://doi.org/10.1145/2523616.2525938>
173. [Qazi et al., 2024] Faiza Qazi, Daehan Kwak, Fiaz Gul Khan, Farman Ali, Sami Ullah Khan, Service Level Agreement in cloud computing: Taxonomy, prospects, and challenges, Internet of Things, Volume 25, 2024, 101126, ISSN 2542-6605, <https://doi.org/10.1016/j.iot.2024.101126>.
174. [Qinghua et al., 2016] Qinghua Zheng, Rui Li, Xiuqi Li, Nazaraf Shah, Jianke Zhang, Feng Tian, Kuo-Ming Chao, Jia Li, Virtual machine consolidated placement based on multi-objective biogeography-based optimization, Future Generation Computer Systems, Volume 54, 2016, Pages 95-122, ISSN 0167-739X, <https://doi.org/10.1016/j.future.2015.02.010>.
175. [R.P. Prado -a et al., 2020] R.P. Prado, S. García-Galán, J.E. Muñoz-Expósito, Adam Marchewka, N. Ruiz-Reyes Smart Containers Schedulers for

- Microservices Provision in Cloud-Fog-IoT Networks. Challenges and Opportunities Sensors, 2020, 20, 1714 DOI: 10.3390/s20061714
176. [Raj & Portia, 2011] S. Benson Edwin Raj and A. Annie Portia, "Analysis on credit card fraud detection methods," 2011 International Conference on Computer, Communication and Electrical Technology (ICCCET), Tirunelveli, India, 2011, pp. 152-156, doi: 10.1109/ICCCET.2011.5762457.
 177. [Reis, 2023] T. N. F. Dos Reis, M. M. Teixeira and C. De Salles Soares Neto, "Relation of Energy Consumption in Green Cloud computing with Big Data," 2023 IEEE Green Technologies Conference (GreenTech), Denver, CO, USA, 2023, pp. 280-284, doi: 10.1109/GreenTech56823.2023.10173828.
 178. [Reiss et al., 2012] Charles Reiss, Alexey Tumanov, Gregory R. Ganger, Randy H. Katz, and Michael A. Kozuch. 2012. Heterogeneity and dynamicity of clouds at scale: Google trace analysis. In Proceedings of the Third ACM Symposium on Cloud computing (SoCC '12). Association for Computing Machinery, New York, NY, USA, Article 7, 1–13. <https://doi.org/10.1145/2391229.2391236>
 179. [Rittinghouse & Ransome, 2016] Rittinghouse, J. W., & Ransome, J. F. (2016). Cloud computing: Implementation, management, and security. CRC Press.
 180. [Rodrigues et al., 2017] T. G. Rodrigues, K. Suto, H. Nishiyama and N. Kato, "A PSO model with VM migration and transmission power control for low Service Delay in the multiple cloudlets ECC scenario," 2017 IEEE International Conference on Communications (ICC), Paris, France, 2017, pp. 1-6, doi: 10.1109/ICC.2017.7996358.
 181. [Rosenblum & Garfinkel, 2005] Rosenblum, M., & Garfinkel, T. (2005). Virtual machine monitors: Current technology and future trends. Computer, 38(5), 39-47.
 182. [S. Ghazanfari-Rad and S. Ebneyousef, 2023] S. Ghazanfari-Rad and S. Ebneyousef, "A Survey of Renewable Energy Approaches in Cloud Data Centers," 2023 8th International Conference on Technology and Energy Management (ICTEM), Mazandaran, Babol, Iran, Islamic Republic of, 2023, pp. 1-7, doi: 10.1109/ICTEM56862.2023.10083820.
 183. [Sagana & Suganthe, 2013] C. Sagana, M. Geetha and R. C. Suganthe, "Performance enhancement in live migration for cloud computing environments,"

- 2013 International Conference on Information Communication and Embedded Systems (ICICES), Chennai, India, 2013, pp. 361-366, doi: 10.1109/ICICES.2013.6508339
184. [Samrajesh, 2013] Samrajesh, M.D., Gopalan, N.P. Component based energy aware multi-tenant application in software as-a service. 15th International Conference on Advanced Computing Technologies, ICACT 2013 (2013). <https://doi.org/10.1109/ICACT.2013.6710502>
185. [Sayadnavard et al., 2021] Monireh H. Sayadnavard, Abolfazl Toroghi Haghghat, Amir Masoud Rahmani, A multi-objective approach for energy-efficient and reliable dynamic VM consolidation in cloud data centers, Engineering Science and Technology, an-International Journal, Volume 26, 2022, 100995, ISSN 2215-0986, <https://doi.org/10.1016/j.jestch.2021.04.014>.
186. [Scordino et al., 2018] Scordino, C., Abeni, L., Lelli, J.: Energy-aware real-time scheduling in the linux kernel. Proc. ACM Sympos. Appl. Comput. (2018). <https://doi.org/10.1145/3167132.3167198>
187. [Seddiki et al., 2021] D. Seddiki, S. G. Galán, E. M. Expósito, M. V. Ibañez, T. Marciniak and R. J. P. De Prado, "Sustainability-based Framework for Virtual Machines Migration Among Cloud Data Centers," 2021 15th International Conference on Signal Processing and Communication Systems (ICSPCS), Sydney, Australia, 2021, pp. 1-8, doi: 10.1109/ICSPCS53099.2021.9660237
188. [Seddiki et al., 2022] Doraid Seddiki, Sebastián García Galán, J. Enrique Muñoz Expósito, Manuel Valverde Ibañez, Tomasz Marciniak, Rocío J. -Pérez de Prado, "Sustainable expert virtual machine migration in dynamic clouds", Computers and Electrical Engineering, Volume 102, 2022,108257, ISSN 0045-7906, <https://doi.org/10.1016/j.compeleceng.2022.108257>.
189. [Seddiki et al., 2024] Doraid Seddiki, Francisco Javier Maldonado Carrascosaa, Sebastian Garcia Galan, Manuel Valverde Ibñez, Tomasz Marciniak, Nicolas Ruiz Reyes, Enhanced Virtual Machine Migration for Energy Sustainability Optimization in Cloud Computing through Knowledge Acquisition, Computers and Electrical Engineering, 2024
190. [Sha et al., 2020] Sha, J., Ebadi, A. G., Mavaluru, D., Alshehri, M., Alfarraj, O., & Rajabion, L. (2020). A method for virtual machine migration in cloud computing using a collective behavior-based metaheuristics algorithm. Concurrency and Computation: Practice and Experience, 32(2), e5441.

191. [Shafiq et al., 2022] Dalia Abdulkareem Shafiq, N.Z. Jhanjhi, Azween Abdullah, Load balancing techniques in cloud computing environment: A review, *Journal of King Saud University - Computer and Information Sciences*, Volume 34, Issue 7, 2022, Pages 3910-3933, ISSN 1319-1578, <https://doi.org/10.1016/j.jksuci.2021.02.007>.
192. [Shaikh & Meshram, 2021]. Shaikh, A. H., & Meshram, B. B. (2021). Security issues in cloud computing. In *Intelligent Computing and Networking: Proceedings of IC-ICN 2020* (pp. 63-77). Springer Singapore.
193. [Shalu & Singh, 2021] Shalu, Singh, D. Artificial neural network-based virtual machine allocation in cloud computing. (2021).
194. [Sharma et al., 2016] Sharma, S., Luhach, A.K., & Abdhullah, S.S. (2016). An Optimal Load Balancing Technique for Cloud computing Environment using Bat Algorithm. *Indian journal of science and technology*, 9.
195. [Shaw et al., 2019] Rachael Shaw, Enda Howley, Enda Barrett, An energy efficient anti-correlated virtual machine placement algorithm using resource usage predictions, *Simulation Modelling Practice and Theory*, Volume 93, 2019, Pages 322-342, ISSN 1569-190X, <https://doi.org/10.1016/j.simpat.2018.09.019>.
196. [Shekhawat et al., 2018] Shekhawat, V.S., Gautam, A., Thakrar, A. Datacenter Workload Classification and Characterization: An Empirical Approach. 13th International Conference on Industrial and Information Systems, ICIIS 2018 - Proceedings 1–7 (2018) doi: (2018). <https://doi.org/10.1109/ICIINFS.2018.8721402>
197. [Shi et al., 2016] Shi W, Cao J, Zhang Q, Li Y, Xu L (2016) Edge Computing: Vision and Challenges. *IEEE Internet Things J.* 3(5):637–646
198. [Shojafar et al., 2016] Shojafar, M., Canali, C., Lancellotti, R., & Abolfazli, S. (2016). An Energy-aware Scheduling Algorithm in DVFS-enabled Networked Data Centers. *International Conference on Cloud Computing and Services Science*.
199. [Shuja et al., 2012] J. Shuja, S. Madani, K. Bilal, K. Hayat, S. Khan, S. Sarwar. Energy-efficient data centers. *Computing*, vol. 94, pp. 973–94, 2012.
200. [Shuja et al., 2016] J. Shuja, A. Gani, S. Shamshirband, R. Wasim Ahmad, K. Bilal. Sustainable Cloud Data Centers: A survey of enabling techniques and technologies. *Renewable and Sustainable Energy Reviews*, vol. 62, pp. 195-214, 2016. DOI: 10.1016/j.rser.2016.04.034.

201. [Shuja et al., 2016] Shuja, J., et al.: Survey of techniques and architectures for designing energy-efficient data centers. *IEEE Syst. J.* 10, 507–519 (2016)
202. [Shukla et al., 2019] Shukla, R., Gupta, R.K., Kashyap, R. (2019). A Multiphase Pre-copy Strategy for the Virtual Machine Migration in Cloud. In: Satapathy, S., Bhateja, V., Das, S. (eds) *Smart Intelligent Computing and Applications. Smart Innovation, Systems and Technologies*, vol 104. Springer, Singapore. https://doi.org/10.1007/978-981-13-1921-1_43
203. [Siddik et al., 2021] Siddik, M. A. B., Shehabi, A., & Marston, L. (2021). The environmental footprint of data centers in the United States. *Environmental Research Letters*, 16(6), 064017.
204. [Siddiqi & Uddin, 2023] Prity, F. Siddiqi, Gazi, M.H. & Uddin, K.M.A. A review of task scheduling in cloud computing based on nature-inspired optimization algorithm. *Cluster Comput* 26, 3037–3067 (2023). <https://doi.org/10.1007/s10586-023-04090-y>
205. [Siddiqi et al., 2023] Prity, F. Siddiqi., Uddin, K.M.A. & Nath, N. Exploring swarm intelligence optimization techniques for task scheduling in cloud computing: algorithms, performance analysis, and future prospects. *Iran J Comput Sci* (2023). <https://doi.org/10.1007/s42044-023-00163-8>
206. [Simaiya et al., 2024] Simaiya, S., Lilhore, U.K., Sharma, Y.K. et al. A hybrid cloud load balancing and host utilization prediction method using deep learning and optimization techniques. *Sci Rep* 14, 1337 (2024). <https://doi.org/10.1038/s41598-024-51466-0>
207. [Singh et al., 2017] Singh S, Chana I, Singh M (2017) The journey of QoS-aware autonomic cloud computing. *IT Prof* 19(2):42–49
208. [Singh et al., 2019] Singh, Harvinder & Tyagi, Sanjay & Kumar, Pardeep. (2019). Crow Search based Scheduling Algorithm for Load Balancing in Cloud Environment. *International Journal of Innovative Technology and Exploring Engineering*. 8. 1058-1064. 10.35940/ijitee. I7787.078919.
209. [Singh et al., 2020] Singh, H., Tyagi, S., Kumar, P.: Scheduling in cloud computing environment using metaheuristic techniques a survey. In: Mandal, J.K. (ed.) *Emerging technology in modelling and graphics: Proceedings of IEM graph 2018*, pp. 753–763. Springer Singapore, Singapore (2020)
210. [Singhal & Jain, 2015] Singhal, U., & Jain, S. (2015). An analysis of swarm intelligence based load balancing algorithms in a cloud computing

- environment. *International Journal of Hybrid Information Technology*, 8(1), 249-256.
211. [Smimite & Afdel, 2020] Smimite, O., & Afdel, K. (2020). Containers placement and migration on cloud system. arXiv preprint arXiv:2007.08695.
 212. [Smith, 1980] Smith, Stephen. (1980). A learning system based on genetic adaptive algorithms.
 213. [Srimannarayana Gran et al., 2015] S. Grandhi and S. Wibowo, "Performance evaluation of cloud computing providers using fuzzy multiattribute group decision making model," 2015 12th International Conference on Fuzzy Systems and Knowledge Discovery (FSKD), Zhangjiajie, China, 2015, pp. 130-135, doi: 10.1109/FSKD.2015.7381928.
 214. [Srinivasan, 2014] S. Srinivasan. *Cloud Computing Evolution. Cloud Computing Basics*, New York, pp. 1–16, 2014. DOI: 10.1007/978-1-4614-7699-3_1, 2014.
 215. [Srivastava et al., 2023] Agam Srivastava, Animesh Ojha, Archana Shaji, Arti Sharma, Rashmi Pandey, *International Journal of Scientific Research in Computer Science, E., & IJSRCSEIT*, I. T. (2023). A Review of Cloud computing Service Models. *International Journal of Scientific Research in Computer Science, Engineering and Information Technology*. <https://doi.org/10.32628/CSEIT2390386>
 216. [Storn & Price, 1997] Storn, R., & Price, K. (1997). Differential evolution—a simple and efficient heuristic for global optimization over continuous spaces. *Journal of Global Optimization*, 11(4), 341-359.
 217. [Subirats y Guitart, 2015] Josep Subirats, Jordi Guitart, Assessing and forecasting energy efficiency on Cloud computing platforms, *Future Generation Computer Systems*, Volume 45, 2015, Pages 70-94, ISSN 0167-739X, <https://doi.org/10.1016/j.future.2014.11.008>.
 218. [Sun et al., 2016] Xiang Sun, Nirwan Ansari, and Ruopeng Wang. 2016. Optimizing Resource Utilization of a Data Center. *Commun. Surveys Tuts.* 18, 4 (October 2016), 2822–2846. <https://doi.org/10.1109/COMST.2016.2558203>
 219. [Sun et al., 2020] Sun, Q., Tan, Z., Zhou, X.: Workload prediction of cloud computing based on SVM and BP neural networks. *J. Intell. Fuzzy Syst.* 39, 2861–2867 (2020)

220. [Tawfeek et al., 2014] Tawfeek, M.A., El-Sisi, A.B., Keshk, A.E., Torkey, F.A.: Virtual Machine Placement Based on Ant Colony Optimization for Minimizing Resource Wastage. *Commun. Comput. Inform. Sci.* 488, 153–164 (2014)
221. [Tchana, 2015] Tchana, A., et al. Software consolidation as an efficient energy and cost saving solution for a SaaS/PaaS cloud model. *Lecture Notes Comput. Sci.* 9233, 305–316 (2015)
222. [Thaler & Hopps, 2000] Thaler, D., & Hopps, C. (2000). Multipath issues in unicast and multicast next-hop selection (No. rfc2991).
223. [Tian et al., 2016] Tian, C., et al.: Minimizing Content Reorganization and Tolerating Imperfect Workload Prediction for Cloud-Based Video-on-Demand Services. *IEEE Trans. Serv. Comput.* 9, 926–939 (2016)
224. [Tiwari et al., 2018] Tiwari, N., Bellur, U., Sarkar, S., Indrawan, M.: Optimizing MapReduce for energy efficiency. *Software: Pract. Experience.* 48, 1660–1687 (2018)
225. [Toraskar & Borse, 2018] Toraskar, T. R., & Borse, Y. (2018). Implementation of cloud computing service delivery models (IaaS, PaaS) by aws and microsoft azure: a survey. *International Journal of Computer Applications*, 975, 8887.
226. [Ullah et al., 2019] Ullah, A., Nawi, N. M., Uddin, J., Baseer, S., & Rashed, A. H. (2019). Artificial bee colony algorithm used for load balancing in cloud computing. *IAES International Journal of Artificial Intelligence*, 8(2), 156.
227. [Usmani & Singh, 2016] Zoha Usmani, Shailendra Singh, A Survey of Virtual Machine Placement Techniques in a Cloud Data Center, *Procedia Computer Science*, Volume 78, 2016, Pages 491-498, ISSN 1877-0509, <https://doi.org/10.1016/j.procs.2016.02.093>.
228. [Vakilinia, 2018] Vakilinia, S.: Energy efficient temporal load aware resource allocation in cloud computing datacenters. *J. Cloud Comput.* 7, 1–24 (2018)
229. [Vaquero et al., 2008] Vaquero, L. M., Rodero-Merino, L., Caceres, J., & Lindner, M. (2008). A break in the clouds: Towards a cloud definition. *ACM SIGCOMM Computer Communication Review*, 39(1), 50-55.
230. [Wan et al., 2012] Wan, X., Zhang, X., Chen, L., & Zhu, J. (2012, May). An improved vTPM migration protocol based trusted channel. In 2012

- International Conference on Systems and Informatics (ICSAI2012) (pp. 870-875). IEEE.
231. [Wang et al., 2014] Wang, W., Sun, Y., Zheng, K., Kaafar, M. A., Li, D., & Li, Z. (2014, October). Freeway: Adaptively isolating the elephant and mice flows on different transmission paths. In 2014 IEEE 22nd international conference on network protocols (pp. 362-367). IEEE.
 232. [Wang et al., 2022] Wang, X., Chen, Y., Jin, J. et al. Fuzzy-clustering and fuzzy network based interpretable fuzzy model for prediction. *Sci Rep* 12, 16279 (2022). <https://doi.org/10.1038/s41598-022-20015-y>
 233. [Wang, 2014] U. Wang. Google, facebook and apple lead on green data centers, 2014. URL <http://www.theguardian.com/sustainable-business/greenpeace-report-google-facebook-apple-green-data-centers>).
 234. [Weinman, 2018] Weinman, J. (2018). The economics of pay-per-use pricing. *IEEE Cloud computing*, 5(5), 101-c3.
 235. [Whitmore et al., 2015] Whitmore A, Agarwal A, Da Xu L (2015) The internet of things—a survey of topics and trends. no. March 2014, pp 261–274
 236. [Wu & Mendel, 2023] D. Wu, R. Peng and J. M. Mendel, "Type-1 and Interval Type-2 Fuzzy Systems [AI- eXplained]," in *IEEE Computational Intelligence Magazine*, vol. 18, no. 1, pp. 81-83, Feb. 2023, doi: 10.1109/MCI.2022.3223496.
 237. [X. Peng and X. Qin, 2020] X. Peng and X. Qin, "Energy Efficient Data Centers Powered by On-site Renewable Energy and UPS Devices," 2020 11th International Green and Sustainable Computing Workshops (IGSC), Pullman, WA, USA, 2020, pp. 1-3, doi: 10.1109/IGSC51522.2020.9291205.
 238. [Xiao et al., 2021] Xiao, P., Ni, Z., Liu, D., Hu, Z.: Improving the energy-efficiency of virtual machines by I/O compensation. *J. Supercomputing*. 77, 11135–11159 (2021)
 239. [Xu & Fortes, 2010] Xu, J., & Fortes, J.A. (2010). Multi-Objective Virtual Machine Placement in Virtualized Data Center Environments. 2010 IEEE/ACM Int'l Conference on Green Computing and Communications & Int'l Conference on Cyber, Physical and Social Computing, 179-188.
 240. [Y. Zhang, 2023] Y. Zhang, F. Hu, Y. Han, W. Meng, Z. Guo and C. Li, "AI-Based Energy-Saving for Fog Computing-Empowered Data Centers," 2023 International Conference on Mobile Internet, Cloud computing and Information

- Security (MICCIS), Nanjing, China, 2023, pp. 16-21, doi: 10.1109/MICCIS58901.2023.00009.
241. [Yadav, 2015] Yadav, A. (2015). Load balancing in cloud computing environment using hybrid approach (ESCEL and PSO) algorithms. *Adv Comput Sci Inf Technol*, 2(8), 10-13.
242. [Yang & Deb, 2009] Yang, X. S., & Deb, S. (2009). Cuckoo search via Lévy flights. *2009 World Congress on Nature & Biologically Inspired Computing (NaBIC)*, 210-214.
243. [Yang & Gandomi, 2012] Yang, X. S., & Gandomi, A. H. (2012). Bat algorithm: a novel approach for global engineering optimization. *Engineering Computations*, 29(5), 464-483.
244. [Yang & Jiang, 2022] Hufang Yang, Ping Jiang, Ying Wang, Hongmin Li, A fuzzy intelligent forecasting system based on combined fuzzification strategy and improved optimization algorithm for renewable energy power generation, *Applied Energy*, Volume 325, 2022, 119849, ISSN 0306-2619, <https://doi.org/10.1016/j.apenergy.2022.119849>.
245. [Yang et al., 2014] Hailong Yang, Qi Zhao, Zhongzhi Luan, Depei Qian, iMeter: An integrated VM power model based on performance profiling, *Future Generation Computer Systems*, Volume 36, 2014, Pages 267-286, ISSN 0167-739X, <https://doi.org/10.1016/j.future.2013.07.008>.
246. [Yang et al., 2015] Yang, Q., et al. Multi-step-ahead host load prediction using autoencoder and echo state networks in cloud computing. *The Journal of Supercomputing* 2015 71:8 71, 3037–3053 (2015)
247. [Yang, 2010] Yang, X. S. (2010). Firefly algorithm, stochastic test functions and design optimization. *International Journal of Bio-Inspired Computation*, 2(2), 78-84.
248. [Yao et al., 2023] Yao, E., Zhang, L., Li, X. et al. Traffic Forecasting of Back Servers Based on ARIMA-LSTM-CF Hybrid Model. *Int J Comput Intell Syst* 16, 65 (2023). <https://doi.org/10.1007/s44196-023-00232-7>
249. [Yousif & Al-Dulaimy, 2017] Yousif, S.A., Al-Dulaimy, A. Clustering Cloud Workload Traces to Improve the Performance of Cloud Data Centers. *Proceedings of the World Congress on Engineering* (2017)

250. [Yu & Gao, 2012] Yu, Y., Gao, Y.: Constraint Programming-Based Virtual Machines Placement Algorithm in Datacenter. *IFIP Adv. Inform. Communication Technol.* 385 AICT, 295–304 (2012)
251. [Yux & Xuey, 2016] Yu X, MI Fellow IEEE, Xue Y (2016) Smart grids: a cyber–physical systems perspective. *Proc IEEE* 104(5):1058–1070
252. [Zadeh, 1965] L.A. Zadeh, Fuzzy sets, *Information and Control*, Volume 8, Issue 3, 1965, Pages 338-353, ISSN 0019-9958, [https://doi.org/10.1016/S0019-9958\(65\)90241-X](https://doi.org/10.1016/S0019-9958(65)90241-X).
253. [Zadeh, 1983] L.A. Zadeh, The role of fuzzy logic in the management of uncertainty in expert systems, *Fuzzy Sets and Systems*, Volume 11, Issues 1–3, 1983, Pages 199-227, ISSN 0165-0114, [https://doi.org/10.1016/S0165-0114\(83\)80081-5](https://doi.org/10.1016/S0165-0114(83)80081-5).
254. [Zadeh, 2008] Lotfi A. Zadeh, Is there a need for fuzzy logic? *Information Sciences*, Volume 178, Issue 13, 2008, Pages 2751-2779, ISSN 0020-0255, <https://doi.org/10.1016/j.ins.2008.02.012>.
255. [Zhang et al., 2018] J. Zhang, F. R. Yu, S. Wang, T. Huang, Z. Liu and Y. Liu, "Load Balancing in Data Center Networks: A Survey," in *IEEE Communications Surveys & Tutorials*, vol. 20, no. 3, pp. 2324-2352, thirdquarter 2018, doi: 10.1109/COMST.2018.2816042.
256. [Zhang et al., 2018] Zhang, Q., Yang, L.T., Yan, Z., Chen, Z., Li, P.: An Efficient Deep Learning Model to Predict Cloud Workload for Industry Informatics. *IEEE Trans. Industr. Inf.* 14, 3170–3178 (2018)
257. [Zhao et al., 2013] Zhao, X., Yin, J., Chen, Z., He, S. Workload classification model for specializing virtual machine operating system. *IEEE International Conference on Cloud computing, CLOUD* 343–350 doi: (2013). <https://doi.org/10.1109/CLOUD.2013.144>
258. [Zhao et al., 2018] K. Zhao, S. Li and Z. Kang, "Takagi-Sugeno Fuzzy Modeling and Control of Nonlinear System with Adaptive Clustering Algorithms," 2018 10th International Conference on Modelling, Identification and Control (ICMIC), Guiyang, China, 2018, pp. 1-6, doi: 10.1109/ICMIC.2018.8530000.
259. [Zheng et al., 2016] Qinghua Zheng, Rui Li, Xiuqi Li, Nazaraf Shah, Jianke Zhang, Feng Tian, Kuo-Ming Chao, Jia Li, Virtual machine consolidated placement based on multi-objective biogeography-based optimization, *Future*

- Generation Computer Systems, Volume 54, 2016, Pages 95-122, ISSN 0167-739X, <https://doi.org/10.1016/j.future.2015.02.010>.
260. [Zhong, 2018] Zhong, W., Zhuang, Y., Sun, J., Gu, J. A load prediction model for cloud computing using PSO-based weighted wavelet support vector machine. *Applied Intelligence* 2018 48:11 48, 4072–4083 (2018)
261. [Zoie, 2017] R. C. Zoie, R. Delia Mihaela and S. Alexandru, "An analysis of the power usage effectiveness metric in data centers," 2017 5th International Symposium on Electrical and Electronics Engineering (ISEEE), Galati, Romania, 2017, pp. 1-6, doi: 10.1109/ISEEE.2017.8170650.
262. [Zonta & Da Costa, 2020] Tiago Zonta, Cristiano André da Costa, Rodrigo da Rosa Righi, Miromar José de Lima, Eduardo Silveira da Trindade, Guann Pyng Li, Predictive maintenance in the Industry 4.0: A systematic literature review, *Computers & Industrial Engineering*, Volume 150, 2020, 106889, ISSN 0360-8352, <https://doi.org/10.1016/j.cie.2020.106889>.
263. [Zulkar Nine et al., 2013] M. S. Q. Zulkar Nine, M. A. K. Azad, S. Abdullah and R. M. Rahman, "Fuzzy logic based dynamic load balancing in virtualized data centers," 2013 IEEE International Conference on Fuzzy Systems (FUZZ-IEEE), Hyderabad, India, 2013, pp. 1-7, doi: 10.1109/FUZZ-IEEE.2013.6622384. 10.1007/978-981-10-2209-8 6

8 Tabla de Tablas

Tabla 3-1 Escenarios fundamentales de simulación.....	104
Tabla 3-2 descripción de las variables de entrada del sistema experto basado en FRBS	120
Tabla 3-3 Funciones de membresía para las entradas y las salidas	121
Tabla 3-4 Base de reglas (RB) del sistema experto.....	121
Tabla 3-5 Configuración Pittsburgh	129
Tabla 4-1: Escenarios de simulación utilizados.....	151
Tabla 4-2: Resultados obtenidos para el escenario 1.....	153
Tabla 4-3: Resultados obtenidos para el escenario 2.....	153
Tabla 4-4: Resultados obtenidos para el escenario 3.....	154
Tabla 4-5 descripción de las variables de entrada del sistema FRBS.....	190
Tabla 4-6 Reglas del Sistema experto	191
Tabla 4-7 Escenarios de simulación utilizados.....	191
Tabla 4-8 Resultados obtenidos para el escenario 1	194
Tabla 4-9 Resultados obtenidos para el escenario 2.....	194
Tabla 4-10 Resultados obtenidos para el escenario 3.....	195
Tabla 4-11 Configuración para la simulación de KASIA.....	199
Tabla 4-12 configuración para la simulación de Pittsburgh	200
Tabla 4-13 Configuración para la simulación de KAGWO	200
Tabla 4-14 Prueba de normalidad de resultados (Shapiro-Wilk) previa al análisis ANOVA	204
Tabla 4-15 Prueba de homogeneidad de varianza (Leneve) Pravia al análisis ANOVA	205
Tabla 4-16 análisis de varianza ANOVA realizado con datos de todos los escenarios usando un nivel de significancia $\alpha = 0.05$	205

9 Tabla de Ilustraciones

Figura 2-1 Evolución del Cloud Computing	30
Figura 2-2 Perímetros de control por usuario final y proveedor de servicios para los tres principales modelos del cloud computing [Parast et al., 2022]	34
Figura 2-3 Anatomía del consumo de energía en los data centers y en los servidores	38
Figura 2-4 Evolución del consumo de energía en diferentes tipologías de data centers	40
Figura 2-5 Modelo simplificado que presenta los niveles donde se utiliza la energía en los data centers	43
Figura 2-6 Estructura general de un FRBS	51
Figura 2-7 Categorías del FRBS	52
Figura 2-8 Ejemplo de un Fuzzy set de tipo 1	52
Figura 2-9 Ejemplo de un Fuzzy set de tipo 2	53
Figura 2-10 Categorización del uso del ML en el cloud computing	60
Figura 2-11 Diagrama de secuencia de una migración en vivo	76
Figura 2-12 Diagrama de secuencia de una migración en frío	77
Figura 2-13 Mecanismo de balanceo de carga en el cloud	83
Figura 3-1 configuración de los RCDC	104
Figura 3-2 Distribución del cambio de disponibilidad de energía renovable en los 4 CDCs utilizados	105
Figura 3-3 Algoritmo del Inter Quartil Range (IQR)	107
Figura 3-4 Algoritmo del Local Regression (LR)	108
Figura 3-5 Diagrama de secuencia del algoritmo RODI	110
Figura 3-6 análisis de varianza ANOVA	111
Figura 3-7 Entidades básicas que conforman el "Core" del motor de simulación de CloudSim	114
Figura 3-8 Extensión de las entidades conceptuales del CloudSim para integrar la migración "Inter" Cloud data centers	115
Figura 3-9 Extensión de las entidades de simulación del CloudSim para integrar la migración "Inter" Cloud data centers	116
Figura 3-11 Potencia Vs Carga para el HP ProLiant ML110 G4	117
Figura 3-10 Potencia Vs Carga para el HP ProLiant ML110 G5	117
Figura 3-12 Esquema simplificado de un FRBS	119
Figura 3-13 Pseudo-Código KASIA	127
Figura 3-14 Pseudo-Código Pittsburg	129
Figura 3-15 Mecanismo de caza de una manada de lobos grises	131
Figura 3-16 Anatomía de una regla difusa	136
Figura 3-17 Configuración de la simulación de Matlab para KASIA	141
Figura 3-18 Configuración de la simulación Matlab para KAGWO	141
Figura 4-1: Escenarios de simulación	152
Figura 4-2 Comportamiento del algoritmo THR en el escenario 1 con 500 Cloudlets de carga	155
Figura 4-3: Comportamiento del algoritmo IQR en el escenario 1 con 500 Cloudlets de carga	155
Figura 4-4: Comportamiento del algoritmo LRR en el escenario 1 con 500 Cloudlets de carga	156
Figura 4-5: Comportamiento del algoritmo LR en el escenario 1 con 500 Cloudlets de carga	157
Figura 4-6: Comportamiento del algoritmo RODI en el escenario 1 con 500 Cloudlets de carga	157

Figura 4-7 Comportamiento del algoritmo THR en el escenario 1 con 1500 Cloudlets de carga	158
Figura 4-8 Comportamiento del algoritmo IQR en el escenario 1 con 1500 Cloudlets de carga	159
Figura 4-9 Comportamiento del algoritmo LRR en el escenario 1 con 1500 Cloudlets de carga	160
Figura 4-10 Comportamiento del algoritmo LR en el escenario 1 con 1500 Cloudlets de carga	160
Figura 4-11: Comportamiento del algoritmo RODI en el escenario 1 con 1500 Cloudlets de carga	161
Figura 4-12 Comportamiento del algoritmo THR en el escenario 1 con 3000 Cloudlets de carga	162
Figura 4-13 Comportamiento del algoritmo IQR en el escenario 1 con 3000 Cloudlets de carga	163
Figura 4-14 Comportamiento del algoritmo LRR en el escenario 1 con 3000 Cloudlets de carga	163
Figura 4-15 Comportamiento del algoritmo LR en el escenario 1 con 3000 Cloudlets de carga	164
Figura 4-16 Comportamiento del algoritmo RODI en el escenario 1 con 3000 Cloudlets de carga	165
Figura 4-17 Comportamiento del algoritmo THR en el escenario 2 con 1000 Cloudlets de carga	166
Figura 4-18 Comportamiento del algoritmo IQR en el escenario 2 con 1000 Cloudlets de carga	167
Figura 4-19 Comportamiento del algoritmo LRR en el escenario 2 con 1000 Cloudlets de carga	168
Figura 4-20 Comportamiento del algoritmo LR en el escenario 2 con 1000 Cloudlets de carga	169
Figura 4-21 Comportamiento del algoritmo RODI en el escenario 2 con 1000 Cloudlets de carga	170
Figura 4-22 Comportamiento del algoritmo THR en el escenario 2 con 2000 Cloudlets de carga	171
Figura 4-23 Comportamiento del algoritmo IQR en el escenario 2 con 2000 Cloudlets de carga	172
Figura 4-24 Comportamiento del algoritmo LRR en el escenario 2 con 2000 Cloudlets de carga	172
Figura 4-25 Comportamiento del algoritmo LR en el escenario 2 con 2000 Cloudlets de carga	173
Figura 4-26 Comportamiento del algoritmo RODI en el escenario 2 con 2000 Cloudlets de carga	174
Figura 4-27 Comportamiento del algoritmo THR en el escenario 2 con 5000 Cloudlets de carga	175
Figura 4-28 Comportamiento del algoritmo IQR en el escenario 2 con 5000 Cloudlets de carga	175
Figura 4-29 Comportamiento del algoritmo LRR en el escenario 2 con 5000 Cloudlets de carga	176
Figura 4-30 Comportamiento del algoritmo LR en el escenario 2 con 5000 Cloudlets de carga	177

Figura 4-31 Comportamiento del algoritmo RODI en el escenario 2 con 5000 Cloudlets de carga	177
Figura 4-32 Comportamiento del algoritmo THR en el escenario 3 con 1500 Cloudlets de carga	178
Figura 4-33 Comportamiento del algoritmo IQR en el escenario 3 con 1500 Cloudlets de carga	179
Figura 4-34 Comportamiento del algoritmo LRR en el escenario 3 con 1500 Cloudlets de carga	180
Figura 4-35 Comportamiento del algoritmo LR en el escenario 3 con 1500 Cloudlets de carga	180
Figura 4-36 Comportamiento del algoritmo RODI en el escenario 3 con 1500 Cloudlets de carga	181
Figura 4-37 Comportamiento del algoritmo THR en el escenario 3 con 5000 Cloudlets de carga	182
Figura 4-38 Comportamiento del algoritmo IQR en el escenario 3 con 5000 Cloudlets de carga	183
Figura 4-39 Comportamiento del algoritmo LRR en el escenario 3 con 5000 Cloudlets de carga	183
Figura 4-40 Comportamiento del algoritmo LR en el escenario 3 con 5000 Cloudlets de carga	184
Figura 4-41 Comportamiento del algoritmo RODI en el escenario 3 con 5000 Cloudlets de carga	185
Figura 4-42 Comportamiento del algoritmo THR en el escenario 3 con 10000 Cloudlets de carga	186
Figura 4-43 Comportamiento del algoritmo IQR en el escenario 3 con 10000 Cloudlets de carga	186
Figura 4-44 Comportamiento del algoritmo LRR en el escenario 3 con 10000 Cloudlets de carga	187
Figura 4-45 Comportamiento del algoritmo LR en el escenario 3 con 10000 Cloudlets de carga	188
Figura 4-46 Comportamiento del algoritmo RODI en el escenario 3 con 10000 Cloudlets de carga	188
Figura 4-47 Distribución de la disponibilidad en el tiempo de la energía renovable en los Data Centers.....	192
Figura 4-48 Algoritmo RODI en el escenario 2 con 5000 Cloudlets	195
Figura 4-49 Numero de migraciones entrantes Vs Migraciones totales en el escenario 2 para el algoritmo RODI	196
Figura 4-50 Comportamiento del algoritmo basado en FRBS en el escenario 2	197
Figura 4-51 Numero de migraciones entrantes Vs Migraciones totales en el escenario 2 para el algoritmo basado en FRBS.....	198
Figura 4-52 Resultados de la comparativa entre los valores de referencias de [D. Seddiki, 2022], KASIA, Pittsburg y KAGWO en términos de porcentaje de energía renovable utilizada	201
Figura 4-53 Evolución de KASIA, Pittsburgh y KAGWO a través de la adquisición de conocimiento en el escenario pequeño	202
Figura 4-54 Evolución de KASIA, Pittsburgh y KAGWO a través de la adquisición de conocimiento en el escenario medio	203
Figura 4-55 Evolución de KASIA, Pittsburgh y KAGWO a través de la adquisición de conocimiento en el escenario grande	203

10 Tabla de Abreviaciones

Abreviatura	Significado
FRBS	Fuzzy Rule-Based System
IQR	Interquartile Range
LR	Linear Regression
LRR	Linear Regression Robust
THR	Threshold Regression
KASIA	KASIA Algorithm
KAGWO	KAGWO Algorithm
CDC	Cloud Data Center
MDC	Modular Data Center
VM	Virtual Machine
IA	Inteligencia Artificial
DVFS	Dynamic Voltage and Frequency Scaling
SLA	Service Level Agreement
TI	Tecnología de la Información
SC	Soft-Computing
FL	Fuzzy logic
EC	Evolutionary Computation
RB	Rule-Based
PAIDI	Plan Andaluz de Investigación, Desarrollo e Innovación
TIC	Tratamiento de Señales en Sistemas de Telecomunicación
ANEP	Agencia Nacional de Evaluación y Prospectiva
JITEL	Jornadas de Ingeniería Telemática
EUSFLAT	European Society for Fuzzy logic and Technology
JCR	Journal Citation Reports
Q1	Quartile 1
NASA	National Aeronautics and Space Administration
CPU	Central Processing Unit
MDCE	Matlab Distributed Computing Environment
IBM	International Business Machines
EPA	Environmental Protection Agency
VMM	Virtual Machine Monitor
IDC	International Data Corporation
PUE	Power Usage Effectiveness
SI	Sistema de Información
IEEE	Institute of Electrical and Electronics Engineers
TCP	Transmission Control Protocol
IP	Internet Protocol
ARPANET	Advanced Research Projects Agency Network
HTTP	Hypertext Transfer Protocol
HTML	Hypertext Markup Language
P2P	Peer-to-Peer
GDPR	General Data Protection Regulation

CSP	Cloud Service Provider
AWS	Amazon Web Services
DC	Data Center
HVAC	Heating, Ventilation, and Air Conditioning
CC	Cloud Computing
SO	Sistema Operativo
OSPM	Open Source Power Management
DCD	Data Center Design
VIP	Very Important Person
PSO	Particle Swarm Optimization
MADM	Multiple Attribute Decision Making
MCDM	Multiple Criteria Decision Making
AHP	Analytic Hierarchy Process
TOPSIS	Technique for Order of Preference by Similarity to Ideal Solution
HFS	Hierarchical Fuzzy System
RMFN	Renewable Modular Fuzzy Network
TSK	Takagi-Sugeno-Kang
NP	Non-Parametric
RAM	Random Access Memory
RL	Reinforcement Learning
RLVMP	Reinforcement Learning Virtual Machine Placement
MBFD	Modified Best Fit Decreasing
ANN	Artificial Neural Network
SARSA	State-Action-Reward-State-Action
GA	Genetic Algorithm
SLAV	Service Level Agreement Violation
ACO	Ant Colony Optimization
TSP	Traveling Salesman Problem
VMP	Virtual Machine Placement
MA	Multi-Agent
FA	Fuzzy Allocation
WOA	Whale Optimization Algorithm
ILWOA	Improved Whale Optimization Algorithm
BWAP	Best Worst Allocation Problem
VSBP	Virtual Machine Selection and Bin Packing Problem
SCA	Sine Cosine Algorithm
SSA	Social Spider Algorithm
KHA	Krill Herd Algorithm
EAGLE	Enhanced Archive-based Genetic Algorithm for Large-Scale Optimization
ICA	Imperialist Competitive Algorithm
VMPLC	Virtual Machine Placement using Learning Classifier
PCA	Principal Component Analysis
SVR	Support Vector Regression
AMD	Advanced Micro Devices

DSB	Dual Stack Buffer
SESA	Simple Economic Schedule Algorithm
AWFDVP	Adaptive Workload Forecasting and Dynamic Voltage Scaling for Energy Efficient Data Centers
LGM	Least Recently Used
TMT	Task Migration and Threshold
DT	Decision Tree
VCSP	Virtual Cloud Service Providers
FFO	Firefly Optimization
EVMM	Enhanced Virtual Machine Migration
BBO	Biogeography-Based Optimization
VMPMBBO	Virtual Machine Placement using Biogeography-Based Optimization
ECC	Elliptic Curve Cryptography
AIVMM	Artificial Immune Virtual Machine Migration
ACS	Artificial Chemical Search
ECMP	Equal-Cost Multi-Path
RTT	Round-Trip Time
RNN	Recurrent Neural Network
LSTM	Long Short-Term Memory
FCT	Flow Completion Time
ABC	Artificial Bee Colony
LB	Load Balancing
NSGA	Non-dominated Sorting Genetic Algorithm
HB	Hybrid Bee
CSLBA	Cloud Service Load Balancing Algorithm
ACOLBA	Ant Colony Load Balancing Algorithm
MPSO	Modified Particle Swarm Optimization
QMPSO	Quantum-behaved Particle Swarm Optimization
CNN	Convolutional Neural Network
DPSO	Distributed Particle Swarm Optimization
SVM	Support Vector Machine
SGD	Stochastic Gradient Descent
RF	Random Forest
MLP	Multilayer Perceptron
ARIMA	AutoRegressive Integrated Moving Average
LLC	Low-Level Cache
CS	Cloud Service
BA	Business Analytics
GWO	Grey Wolf Optimizer
ICSPCS	International Conference on Signal Processing, Communication and Computing Systems
CDM	Cloud Data Management
RCDC	Renewable Cloud Data Center
ANOVA	Analysis of Variance
MI	Multiple Instruction

MIPS	Million Instructions Per Second
RA	Risk Assessment
HCC	Hybrid Cloud Computing
HCA	High-Performance Computing
MCN	Metropolitan Area Network
CCN	Cooperative Cloud Network
KB	Knowledge Base
MMT	Minimum Migration Time
IR	Information Retrieval
IRR	Internal Rate of Return
SARIMA	Seasonal Autoregressive Integrated Moving Average
OWASP	Open Web Application Security Project
HPC	High-Performance Computing
HPCS	High-Performance Cloud Service
OI	Optimization Inference
GM	General Motors
AS	Algorithmic State
THH	Three Hybrid Heuristic
