



Diseño e implementación de un sistema de balanceo de carga basado en inteligencia artificial mediante aprendizaje por refuerzo para redes de un proveedor de servicios de internet (ISP).

Chafla Carguacundo, Karem Estefania y Guamán Pila, Vinicio Paul

Departamento de Ciencias de la Computación

Carrera de Tecnologías de la Información

Trabajo de integración curricular, previo a la obtención del título de Ingeniero/a en Tecnologías de la Información

Ing. Bustos Sánchez, Cristian Santiago, PhD.

31 de julio de 2026



Certificado de análisis
Compilatio Magister+ | ESPE-ECU

TRABAJO INTEGRACION CURRICULAR CHAFLA - GUAMAN

ID : fdbcce116b51847a646bb10435940495238f6b31



8%

Textos sospechosos

Nombre del fichero : TRABAJO INTEGRACION CURRICULAR CHAFLA - GUAMAN.txt
Tamaño del archivo original : 1,67 MB
Número de palabras : 16.904

Depositante : CRISTIAN SANTIAGO BUSTOS SÁNCHEZ
Fecha de depósito : 24 de julio de 2026
Tipo de carga : interface
fecha de fin de análisis : 24 de julio de 2026

Resumen (sección 1/2)

Localización de los textos sospechosos en el documento :



Incluido en el porcentaje de textos sospechosos :



Similitudes

<1%

Pasajes con similitudes a fuentes encontradas en diferentes colecciones.



Detección de IA

7%

Textos estilísticamente próximos a un texto generado por una IA.

Este índice es un indicador y no una prueba. Comprueba con el autor si domina los conocimientos mencionados en el documento.



Idiomas no reconocidos

1%

Pasajes en los que parte del vocabulario utilizado no forma parte del diccionario de la lengua.

Puede tratarse de un intento del autor de modificar el texto para evitar ser detectado.



Validar únicamente en FirmaEC.
Firmado electrónicamente por:
CRISTIAN SANTIAGO
BUSTOS SANCHEZ

Ing. Bustos Sánchez, Cristian Santiago, PhD.

Director de Trabajo de Titulación



Departamento de Ciencias de la Computación

Carrera de Ingeniería en Tecnologías de la Información

Certificación

Certifico que el trabajo de integración curricular: **“Diseño e implementación de un sistema de balanceo de carga basado en inteligencia artificial mediante aprendizaje por refuerzo para redes de un proveedor de servicios de internet (ISP)”** fue realizado por los señores **Chafila Carguacundo, Karem Estefania y Guamán Pila, Vinicio Paul**, el mismo que cumple con los requisitos legales, teóricos, científicos, técnicos y metodológicos establecidos por la Universidad de las Fuerzas Armadas ESPE; habiéndose efectuado una revisión integral de forma y de fondo conforme a los lineamientos y formatos institucionales vigentes. Además, fue revisado y analizada en su totalidad por la herramienta de prevención y/o verificación de similitud de contenidos; razón por la cual me permito acreditar y autorizar para que se lo sustente públicamente.



Validar únicamente en FirmaDC.
Firmado electrónicamente por:
**CRISTIAN SANTIAGO
BUSTOS SANCHEZ**

Sangolquí, 31 de Julio de 2026

Ing. Bustos Sánchez, Cristian Santiago, PhD.

Director de Trabajo de Titulación

C.C.: 1500863111



Departamento de Ciencias de la Computación

Carrera de Ingeniería en Tecnologías de la Información

Responsabilidad de Autoría

Nosotros, **Chafla Carguacundo, Karem Estefania y Guamán Pila, Vinicio Paul**, con cédulas de ciudadanía n° 0604052068 y 1755160981, declaramos que el contenido, ideas y criterios del trabajo de integración curricular: **Diseño e implementación de un sistema de balanceo de carga basado en inteligencia artificial mediante aprendizaje por refuerzo para redes de un proveedor de servicios de internet (ISP)**, es de nuestra autoría y responsabilidad, cumpliendo con los requisitos legales, teóricos, científicos, técnicos, y metodológicos establecidos por la Universidad de las Fuerzas Armadas ESPE, respetando los derechos intelectuales de terceros y referenciando las citas bibliográficas.

Sangolquí, 31 de Julio de 2026



Validar únicamente en FirmaEC.
Firmado electrónicamente por:
**KAREM ESTEFANIA
CHAFLA CARGUACUNDO**

Chafla Carguacundo, Karem Estefania

C.C.: 0604052068



Validar únicamente en FirmaEC.
Firmado electrónicamente por:
**VINICIO PAUL GUAMAN
PILA**

Guamán Pila, Vinicio Paul

C.C.: 1755160981



Departamento de Ciencias de la Computación

Carrera de Ingeniería en Tecnologías de la Información

Autorización de Publicación

Nosotros **Chafla Carguacundo, Karem Estefania y Guamán Pila, Vinicio Paul**, con cédulas de ciudadanía n° 0604052068 y 1755160981 ,autorizamos a la Universidad de las Fuerzas Armadas ESPE publicar el trabajo de integración curricular: **Título:** Diseño e implementación de un sistema de balanceo de carga basado en inteligencia artificial mediante aprendizaje por refuerzo para redes de un proveedor de servicios de internet (ISP), en el Repositorio Institucional, cuyo contenido, ideas y criterios son de nuestra responsabilidad.

Sangolquí, 31 de Julio de 2026



Chafla Carguacundo, Karem Estefania

C.C.: 0604052068



Guamán Pila, Vinicio Paul

C.C.: 1755160981

Dedicatoria

Dedico este trabajo, en primer lugar, a Dios, por darme la vida, la fortaleza y la sabiduría para superar cada desafío y permitirme alcanzar una de las metas más importantes de mi formación profesional. A mis queridos padres, Alfonso y Elina, por su amor, sacrificio y apoyo incondicional durante toda mi vida. Gracias por ser el pilar fundamental de mis sueños y enseñarme que con esfuerzo y perseverancia todo es posible. A mis hermanos, a mi abuelita, a mi sobrino y a mis amigos, por su cariño, confianza y palabras de aliento que me motivaron a seguir adelante en cada etapa de este proceso. A mi novia Alexandra, por su amor, paciencia, comprensión y constante apoyo durante este camino. Gracias a todos por acompañarme en cada reto y celebrar conmigo este importante logro.

Guamán Pila, Vinicio Paul

A mi familia, dedicó este trabajo, ellos son el pilar fundamental de mi vida y la razón por la que nunca dejé de luchar por mis sueños. A mi madre, por enseñarme a no rendirme nunca y a afrontar cada obstáculo con fuerza y esperanza. Por ser mi refugio, por sostener mi mano en los momentos más complicados y por recordarme siempre que todo se puede lograr con esfuerzo y constancia. A mi hermana Erika, por ser mi guía y por ser el ejemplo que siempre admiré. Su pasión y entrega me mostraron que los sueños no tienen fronteras cuando uno trabaja con amor y constancia. A mi hermana Alison, por ser mi motor para seguir adelante cada día, por acompañarme en los momentos más difíciles y recordarme, con su cariño, que nunca estaba sola. A mi papá, Francisco, por su sacrificio y dedicación para sacar adelante a nuestra familia junto a mi mamá. Su ejemplo de responsabilidad, trabajo y constancia me ha servido de inspiración para ser la persona que soy hoy. Por último, dedico este logro a mis amigos, quienes hicieron de esta etapa una experiencia única e inolvidable.

Chafra Carguacundo, Karem Estefania

Agradecimiento

Agradezco, en primer lugar, a Dios, por bendecirme con la oportunidad de culminar esta etapa de mi vida y por acompañarme en cada paso de este camino. Mi más profundo agradecimiento a mis padres, Alfonso y Elina, a mis hermanos, a mi abuelita, a mi sobrino y a mi novia Alexandra, por su apoyo incondicional, comprensión y motivación constante para alcanzar este objetivo. Expreso un especial agradecimiento a mi tutor el Ing. Cristian Santiago Bustos Sánchez, PhD., por su tiempo, dedicación, orientación y valiosos conocimientos compartidos durante el desarrollo de esta investigación. Su guía fue fundamental para la culminación de este trabajo. Finalmente, agradezco a la Universidad de las Fuerzas Armadas ESPE y a todos los docentes que contribuyeron a mi formación académica y profesional, brindándome las herramientas necesarias para enfrentar los retos de mi carrera.

Guamán Pila, Vinicio Paul

Dios, gracias por darme la vida, gracias por darme salud, fuerza y sabiduría para superar cada dificultad y alcanzar uno de los objetivos más importantes de mi vida. Mi familia, por ser el pilar esencial en cada paso de mi camino. Mis padres por ese amor incondicional, por el esfuerzo, el sacrificio diario para darme la oportunidad de crecer tanto personal como profesionalmente.

Mucho de lo que soy y este logro son el resultado de los valores que me inculcaron. A mis hermanas, Erika y Alison, por estar siempre a mi lado, por motivarme a no darme por vencida y por acompañarme en este proceso con su cariño, comprensión y apoyo incondicional.

Manifiesto mi especial agradecimiento al Ing. Cristian Santiago Bustos Sánchez, PhD., director de esta investigación, por su guía, apoyo y profesionalismo a lo largo del desarrollo de este trabajo. Sus valiosas recomendaciones, experiencia y dedicación permitieron enriquecer el presente estudio y alcanzar los objetivos planteados de manera satisfactoria. Por último, quiero dar las gracias a mis amigos y a todas aquellas personas que de una forma u otra estuvieron a mi lado en esta etapa y me animaron, apoyaron y confiaron en mí.

Chafra Carguacundo, Karem Estefania

Índice de contenidos

Certificación	3
Responsabilidad de Autoría	4
Autorización de Publicación	5
Dedicatoria	6
Agradecimiento	7
Resumen	15
Abstract	16
Capítulo I	17
Introducción	17
Antecedentes	18
Planteamiento del problema	18
Formulación del problema o pregunta de investigación	20
Objetivos	21
Objetivo general:	21
Objetivos específicos:	21
Justificación	21
Capítulo II	22
Estado del arte	22
Inteligencia Artificial aplicada a la gestión de redes de telecomunicaciones	22
Automatización inteligente en redes ISP	24
Balanceo de carga en redes de proveedores de servicios de Internet	24
Aprendizaje automático aplicado a redes de telecomunicaciones	25
Aprendizaje por refuerzo aplicado al balanceo de carga	26

Algoritmo Q-Learning para la optimización del balanceo de carga	26
Implementaciones de balanceo de carga sobre MikroTik.....	27
Análisis de investigaciones relacionadas	27
Vacío de investigación	29
Marco teórico.....	32
Redes de Proveedores de Servicios de Internet (ISP)	32
Balanceo de carga	34
Calidad de Servicio (QoS)	36
Inteligencia Artificial	39
Aprendizaje por Refuerzo	41
Componentes del Aprendizaje por Refuerzo	42
Algoritmo Q-Learning	48
Marco conceptual	60
Capítulo III.....	61
Metodología.....	61
Enfoque de investigación.....	61
Variables de investigación	61
Variable independiente	62
Variables dependientes	62
Procedimiento	62
Método de evaluación	64
Relación de la investigación con la metodología CRISP-DM	66
Tipo y diseño de investigación.....	66
Población y muestra.....	67
Técnica e instrumento.....	68
Técnicas de Recolección de Datos	68

Instrumentos.....	68
Estructura del Sistema de Balanceo de Carga – Q-Learning	73
Validación y confiabilidad del instrumento.....	77
Escenarios de Prueba.....	78
Escenario 1: Funcionamiento Normal	78
Escenario 2: Alta Demanda	79
Escenario 3: Congestión de WAN1	80
Escenario 4: Congestión de WAN2	82
Escenario 5: Falla parcial de enlace.....	84
Distribución de Utilización de Enlaces.....	86
Consideraciones éticas	89
Capítulo IV	89
Análisis y discusión de los resultados	89
Método tradicional.....	90
Prueba de latencia y pérdida de paquetes	90
Prueba de análisis de ruta mediante traceroute.....	91
Prueba de variación de retardo (Jitter)	92
Análisis estadístico del throughput de los enlaces WAN	93
Monitoreo de recursos del router.....	94
Utilizando recomendaciones del agente Inteligente(IA).....	95
Prueba de latencia y pérdida de paquetes mediante ping.....	95
Prueba de análisis de ruta mediante traceroute.....	96
Prueba de variación de retardo (Jitter)	97
Análisis estadístico del throughput de los enlaces WAN	98
Monitoreo de recursos del router.....	99
Prueba de tolerancia a fallos (Failover).....	100

Resumen Comparativo	102
Tabla general de resultados del Método tradicional y el sistema basado en Inteligencia Artificial.....	103
Discusión de resultados	105
Comparación del Método tradicional vs Inteligencia artificial	105
Capítulo V	109
Conclusiones	109
Recomendaciones	111
Referencias	111

Índice de tablas

Tabla 1 Comparación de investigaciones relacionadas sobre balanceo de carga basado en inteligencia artificial en redes de telecomunicaciones	30
Tabla 2 Estructura general de la matriz Q del algoritmo Q-Learning	50
Tabla 3 Herramientas utilizadas para la implementación del sistema	68
Tabla 4 Resultados del aprendizaje del agente Q-Learning	88
Tabla 5 Valores Q observados durante el aprendizaje del agente Q-Learning	88
Tabla 6 Resultados de la prueba de latencia y pérdida de paquetes mediante Ping	90
Tabla 7 Resultados de la prueba de análisis de ruta mediante traceroute	91
Tabla 8 Resultados de la prueba de variación de retardo (Jitter)	92
Tabla 9 Mediciones de throughput de los enlaces WAN bajo el método tradicional	93
Tabla 10 Análisis estadístico del throughput de los enlaces WAN bajo el método tradicional	94
Tabla 11 Recursos hardware y estado operativo del router MikroTik	94
Tabla 12 Resultados de la prueba de latencia y pérdida de paquetes mediante ping con inteligencia artificial	95

Tabla 13 Resultados de la prueba de análisis de ruta mediante traceroute con inteligencia artificial	96
Tabla 14 Resultados de la prueba de variación de retardo (Jitter) con Inteligencia Artificial	97
Tabla 15 Resultados de throughput de los enlaces WAN con inteligencia artificial	98
Tabla 16 Análisis estadístico del throughput de los enlaces WAN con inteligencia artificial	99
Tabla 17 Métricas de rendimiento y utilización de recursos del router con inteligencia artificial	99
Tabla 18 Caída de enlaces LUMEN y TELEFÓNICA	100
Tabla 19 Resumen comparativo de la prueba de tolerancia a fallos	101
Tabla 20 Resumen de costos operativos método tradicional vs IA	102
Tabla 21 Tabla general de resultados del método tradicional y el sistema basado en Inteligencia Artificial	103

Índice de figuras

Figura 1 Topología de la infraestructura de red utilizada para la implementación del sistema de balanceo de carga	20
Figura 2 Arquitectura general del ISP	33
Figura 3 Balanceo de carga.....	34
Figura 4 Flujo de adquisición de métricas.....	37
Figura 5 Ciclo del Aprendizaje por Refuerzo.....	42
Figura 6 Arquitectura del sistema inteligente de balanceo de carga basado en IA	45
Figura 7 Funcionamiento de Q-Learning.....	48
Figura 8 Comunicación del sistema	55
Figura 9 Arquitectura del Sistema Inteligente	58
Figura 10 Arquitectura general del sistema de balanceo de carga basado en inteligencia artificial para redes ISP.....	64

Figura 11 Sistema operativo utilizado en el desarrollo del sistema.....	69
Figura 12 Entorno de desarrollo Visual Studio Code utilizado para la implementación del sistema	70
Figura 13 Configuración del servicio API-SSL en MikroTik para la comunicación segura con el sistema	71
Figura 14 Dashboard de monitoreo en tiempo real del sistema de balanceo de carga inteligente.....	71
Figura 15 Panel de control del dashboard de monitoreo en tiempo real del sistema de balanceo de carga inteligente.....	72
Figura 16 Estructura de archivos y módulos del sistema de balanceo de carga basado en inteligencia artificial.....	73
Figura 17 Arquitectura lógica del sistema de balanceo de carga basado en inteligencia artificial	76
Figura 18 Dashboard inteligente de monitoreo y balanceo de carga basado en Q-Learning...	78
Figura 19 Monitoreo de tráfico de las interfaces WAN en el router MikroTik	78
Figura 20 Dashboard de monitoreo y control del agente Q-Learning para el balanceo de carga	79
Figura 21 Comparación entre las decisiones del agente Q-Learning y el algoritmo tradicional de balanceo de carga	79
Figura 22 Aplicación de políticas de balanceo	80
Figura 23 Evolución del desbalance y la recompensa del agente Q-Learning durante el entrenamiento	81
Figura 24 Estado de ejecución y comparación de decisiones del agente Q-Learning	82
Figura 25 Comparación de decisiones entre el agente Q-Learning y la heurística tradicional de balanceo de carga	83
Figura 26 Interfaz de control y ejecución del balanceo de carga basado en Q-Learning	83

Figura 27	Ejecución del agente Q-Learning con selección de política de balanceo ECMP	84
Figura 28	Monitoreo en tiempo real del balance de carga entre los enlaces WAN	85
Figura 29	Monitoreo de tráfico en tiempo real de las interfaces WAN del Router MikroTik	86
Figura 30	Balance de carga equilibrado obtenido mediante el agente Q-Learning	87
Figura 31	Estado de aprendizaje y selección de acciones del agente Q-Learning	87
Figura 32	Comparación de la prueba de latencia y pérdida de paquetes mediante ping(Tradicional vs IA)	105
Figura 33	Comparación de la prueba de análisis de ruta mediante traceroute (Tradicional vs IA)	106
Figura 34	Comparación de la prueba de variación de retardo(Jitter) (Tradicional vs IA)	107
Figura 35	Comparación estadística del throughput de los enlaces WAN (Tradicional vs IA).	108
Figura 36	Comparación de utilización de recursos del router (Tradicional vs IA)	108

Resumen

Esta investigación se centra en el diseño e implementación de un sistema de balanceo de carga basado en inteligencia artificial (IA) que aplica aprendizaje por refuerzo, específicamente el algoritmo Q-learning, para mejorar la distribución del tráfico dentro de una red real de un Proveedor de Servicios de Internet (ISP). Asimismo, el sistema propuesto será implementado en la infraestructura tecnológica de QuitoFibranet Cia. Ltda. utilizando un nodo MikroTik. A medida que el modelo aprenda de las condiciones de la red a lo largo del tiempo, generará recomendaciones inteligentes de enrutamiento para apoyar a los administradores de red en la toma de decisiones fundamentadas.

La metodología propuesta incluye la evaluación de la topología actual de la red y el análisis de indicadores clave de desempeño, como el throughput, la latencia, la utilización de los enlaces y la pérdida de paquetes. Estos datos se utilizarán para entrenar un agente de aprendizaje por refuerzo capaz de adaptarse a las condiciones cambiantes de la red e identificar, en tiempo real, la ruta de enrutamiento más eficiente. Además, un panel de monitoreo web desarrollado con HTML, CSS, JavaScript y Python permitirá a los administradores supervisar el rendimiento de la red y revisar las recomendaciones generadas por el sistema de inteligencia artificial.

El objetivo principal de esta investigación es determinar si el aprendizaje por refuerzo puede mejorar la calidad del servicio, el rendimiento general de la red, la utilización de los recursos y la eficiencia del balanceo de carga en comparación con los enfoques convencionales. Asimismo, se espera que los resultados sirvan de apoyo para futuras investigaciones y fomenten la adopción de soluciones inteligentes para la gestión de redes de proveedores de servicios de Internet (ISP).

Palabras clave: Balanceo de carga, Inteligencia Artificial, Aprendizaje por Refuerzo, Q-Learning, MikroTik.

Abstract

This research focuses on the design and implementation of an artificial intelligence AI-based load balancing system that applies reinforcement learning, specifically the Q-learning algorithm, to improve traffic distribution within a real-world Internet Service Provider (ISP) network. Also, the proposed system will be implemented on the IT infrastructure of QuitoFibranet Cia. Ltda. using a MikroTik node. As the model learns from network conditions over time, it will generate intelligent routing recommendations to support network administrators in making informed decisions.

The proposed methodology includes evaluating the current network topology and analyzing key performance indicators such as throughput, latency, link utilization, and packet loss. These data will be used to train a reinforcement learning agent capable of adapting to changing network conditions and identifying the most efficient routing path in real time. In addition, a web-based dashboard developed with HTML, CSS, JavaScript, and Python will allow administrators to monitor network performance and review the recommendations produced by the AI system.

The main research's objective is to determine whether reinforcement learning can improve quality of service, overall network performance, resource utilization, and load balancing efficiency when compared with conventional approaches. Therefore, the results are also expected to support future research and encourage the adoption of intelligent solutions for ISP network management.

Keywords: Load balancing, Artificial Intelligence, Reinforcement Learning, Q-Learning, MikroTik.

Capítulo I

Introducción

La transformación digital junto con el crecimiento exponencial del uso de servicios en Internet a nivel mundial ha generado en gran medida un incremento del tráfico y procesamiento de datos, videollamadas, plataformas de streaming, cloud computing, juegos online y servicios empresariales demandan infraestructuras de red robustas las cuales deben ser capaces de garantizar altos niveles de calidad de servicio, rendimiento y disponibilidad. Los Proveedores de Servicios de Internet (ISP) son los responsables de administrar de manera eficiente sus recursos de red para satisfacer las necesidades de sus usuarios y sobre todo dar un servicio de calidad mejorando la experiencia del usuario. Ante esto, el patrón no predecible del tráfico de datos junto con la variabilidad en demanda generan desafíos a considerar con la distribución eficiente de la carga entre uno o varios enlaces de comunicación (Muñoz López, 2018).

Los ISP utilizan configuraciones estáticas de balanceo de carga es decir configuraciones manuales o algoritmos predeterminados. En la actualidad muchas empresas optan por estos métodos tradicionales sin embargo presentan limitaciones al momento de responder a cambios repentinos donde existe alta demanda de tráfico de datos o en ocasiones ante fallas en la misma infraestructura TI lo cual ocasiona reducción de la calidad del servicio, incremento de latencia, congestión y pérdida de paquetes. Bajo estas circunstancias, surge la Inteligencia Artificial como una solución alternativa innovadora para optimizar y administrar mejor el tráfico dentro de las redes de telecomunicaciones. En concreto, el aprendizaje por refuerzo hace posible crear agentes inteligentes los cuales pueden ser capaces de aprender y sobre todo tomar las mejores decisiones en tiempo real. En esta investigación se propone el diseño e implementación de un sistema de balanceo de carga basado en aprendizaje por refuerzo mediante el algoritmo Q-Learning, el cual será aplicado a la infraestructura de red de un ISP real, cuyo objetivo es optimizar la distribución del tráfico mejorando así las métricas de desempeño de la red (Sutton & Barto, 2018).

Antecedentes

En los últimos años, varios estudios han demostrado que tan efectivo es adoptar las nuevas técnicas de Inteligencia Artificial en tareas relacionados a optimización de procesos con redes de computadoras y telecomunicaciones. Dichas investigaciones han demostrado que los algoritmos de aprendizaje automático mejoran la asignación de recursos, también al momento de predecir comportamientos de tráfico y en la automatización de procesos de gestión de red.

Dentro del campo de la infraestructura TI, el aprendizaje por refuerzo ha sido utilizado para resolver problemas en la toma de decisiones en entornos dinámicos, los cuales son caracterizados por presentar cambios constantes y múltiples variables de operación. El balanceo de carga se presenta como una solución dentro de estos escenarios debido a la naturaleza volátil del tráfico de red (Quiroz Pérez, 2025).

En el país, la gran mayoría de pequeños y medianos ISP todavía optan por utilizar configuraciones tradicionales de balanceo de carga las cuales son basadas en reglas estáticas o predefinidas y que son implementadas en equipos de red como MikroTik. Aun cuando estas soluciones cumplen su función en condiciones normales de operación, presentan desafíos y limitaciones frente a escenarios de falla de enlaces, crecimiento de usuarios y alta congestión.

QuitoFibranet Cía. Ltda. es una empresa que cuenta con una infraestructura de red basada en múltiples equipos MikroTik y gran cantidad de enlaces de comunicación, lo cual constituye como un entorno apto para evaluar la implementación práctica de técnicas de IA las cuales son orientadas a la optimización de la distribución del tráfico de red.

Planteamiento del problema

Entorno y síntomas operativos

En la actualidad, los proveedores de servicios de Internet o ISP se enfrentan un constante crecimiento del tráfico de datos esto debido a la alta demanda de servicios en internet. Dicho crecimiento hace un uso significativo de enlaces de red lo cual exige mejores mecanismos para distribuir y optimizar de mejor manera la carga de tráfico de red. En la infraestructura analizada

se encontraron varias salidas a Internet que están gestionadas con equipos MikroTik. Hoy en día, la distribución del tráfico se realiza a través de configuraciones manuales o tradicionales que necesitan de la supervisión continua de los administradores de red.

Principales síntomas operativos identificados:

- Congestión de los dos enlaces durante tiempos de alta demanda.
- Desperdicio de enlaces alternativos disponibles.
- Aumento significativo de la latencia en horas pico.
- Pérdida de paquetes producto a la saturación de recursos.
- Dependencia de personal para ajustes manuales de balanceo.
- Incremento de tiempos de respuesta ante incidentes de red.

Restricciones de los esquemas actuales

Existen desventajas y limitaciones técnicas en los métodos tradicionales de balanceo de carga los cuales afectan en gran medida a la eficiencia de la infraestructura. Dichos mecanismos funcionan utilizando reglas predefinidas que no son eficientes frente a cambios dinámicos del entorno tampoco aprenden del patrón del tráfico. Esto lleva a que, las decisiones de distribución de carga sean ineficientes en escenarios de alta demanda de usuarios. Por otra parte, las configuraciones tradicionales requieren de supervisión física constante por parte del personal encargado lo cual aumenta la complejidad a nivel operativo y también en el aumento de costos (Valle Montes, 2023).

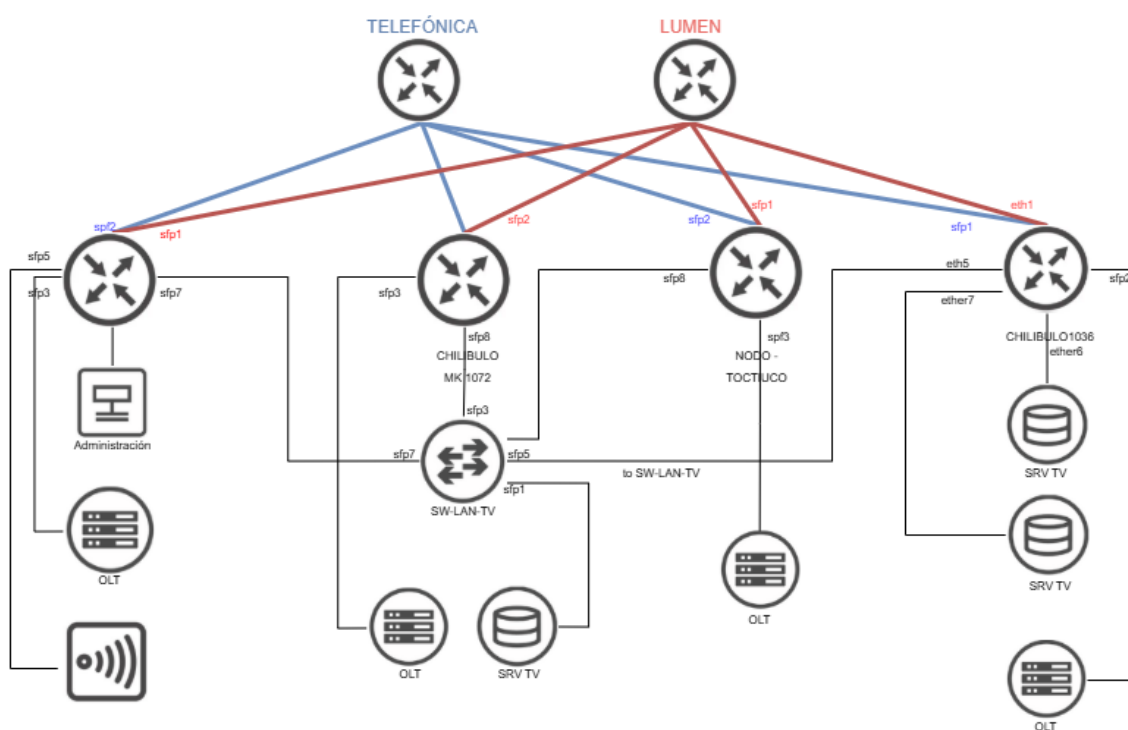
Necesidad de decisiones basadas en métricas

En la actualidad para la administración de redes se requiere de mecanismos que sean capaces de tomar las mejores decisiones basadas en métricas de desempeño fundamentadas. Existen variables que proporcionan información relevante acerca del estado en tiempo real de la infraestructura, por ejemplo, latencia, throughput y pérdida de paquetes. Gracias a estas métricas se logra identificar patrones de comportamiento mediante un análisis continuo y

posterior a ello seleccionar las mejores rutas para la distribución del tráfico. Es por ello que gracias a la implementación de técnicas de IA se logra automatizar estos procesos gracias a agentes que pueden ser capaces de adaptar sus decisiones en base a resultados obtenidos y en el aprendizaje del entorno (Pérez et al., 2025).

Figura 1

Topología de la infraestructura de red utilizada para la implementación del sistema de balanceo de carga



Nota. Elaboración propia

Formulación del problema o pregunta de investigación

¿De qué manera la implementación de un sistema de balanceo de carga basado en Inteligencia Artificial mediante aprendizaje por refuerzo utilizando Q-Learning puede optimizar la distribución del tráfico de red y mejorar la calidad de servicio en la infraestructura de un Proveedor de Servicios de Internet?

Objetivos

Objetivo general:

Diseñar e implementar un sistema de balanceo de carga basado en Inteligencia Artificial mediante aprendizaje por refuerzo, aplicado en la infraestructura de un proveedor de servicios de Internet (ISP) en colaboración con una empresa coparticipante, con el fin de optimizar la distribución del tráfico y mejorar la calidad de servicio, utilizando métricas de rendimiento como throughput, latencia y pérdida de paquetes.

Objetivos específicos:

- Analizar la arquitectura de red del ISP y los puntos críticos de balanceo de carga.
- Diseñar el modelo de aprendizaje por refuerzo para la gestión dinámica del tráfico.
- Implementar el sistema de balanceo de carga en la infraestructura real del ISP.
- Evaluar el desempeño del sistema propuesto frente a métodos tradicionales de balanceo de carga.
- Documentar los resultados y proponer recomendaciones para la integración de IA en redes ISP.

Justificación

Justificación Técnica

En la parte técnica el presente trabajo de investigación busca implementar en infraestructuras de red formas inteligentes ante la toma de decisiones en tiempo real. El agente Q-Learning que se utilizará será capaz de aprender de manera continua mediante el comportamiento de la red, el cual se encargará de seleccionar de manera dinámica las mejores rutas en tiempo real para tener una mejor distribución del tráfico. Esto mejorará indicadores en calidad de servicio y a su vez reducirá problemas de congestión cuando exista alta demanda de usuarios. Por último, permitirá validar la aplicabilidad de estas técnicas de IA en una infraestructura real de producción gracias a la integración con equipos MikroTik.

Justificación Económica y de Viabilidad

En la actualidad el adaptarse a procesos de balanceo de carga automatizados genera reducción en costos operativos significativos puesto que se omite la supervisión y administración manual de los equipos de la infraestructura. En la presente investigación se hará el uso de una infraestructura ya existente lo que incrementa la viabilidad económica del proyecto. Al disminuir el uso de intervención de personal especializado en configuraciones manuales en la infraestructura, el ISP optimiza tanto recursos humanos del personal como también recursos financieros. Por último, el uso de tecnologías de código abierto como Python junto con librerías de IA contribuyen en gran medida a la optimización de costos durante la implementación (Romero & Romero, 2014).

Justificación Organizacional y Académica

En el ámbito organizacional, la presente investigación busca disponer de una herramienta útil y eficaz a la empresa la cual busque optimizar y mejorar procesos como la administración de infraestructura TI y a su vez el fortalecimiento de la calidad de servicio que el ISP ofrece a sus clientes. Dentro del ámbito académico, el presente trabajo de investigación aporta evidencia práctica sobre la aplicación de técnicas de IA en infraestructuras de TI reales y redes de telecomunicaciones, aportando así conocimientos de aprendizaje automático, análisis de datos, desarrollo de software y redes de computadoras. Por último, los resultados obtenidos servirán como referencia para futuras investigaciones relacionadas referentes a automatización de infraestructuras TI, optimización de recursos mediante el uso de IA y redes inteligentes (Fernández Rivera, s. f.).

Capítulo II

Estado del arte

Inteligencia Artificial aplicada a la gestión de redes de telecomunicaciones

El crecimiento exponencial del tráfico de datos, impulsado por la expansión de los servicios en la nube, el Internet de las Cosas (IoT), el video en alta definición y las aplicaciones

en tiempo real, ha incrementado significativamente la complejidad en la administración de las redes de telecomunicaciones. En este contexto, los métodos tradicionales basados en configuraciones estáticas y reglas predefinidas presentan limitaciones para responder de manera eficiente a las variaciones dinámicas del tráfico. Como consecuencia, la Inteligencia Artificial (IA) ha adquirido un papel fundamental en la automatización de procesos de monitoreo, análisis y toma de decisiones dentro de las infraestructuras de red.

La Inteligencia Artificial permite desarrollar sistemas capaces de analizar grandes volúmenes de información provenientes de routers, switches y dispositivos de comunicación, identificando patrones de comportamiento que difícilmente podrían ser detectados mediante técnicas convencionales. Gracias a estas capacidades, es posible optimizar la utilización de recursos, anticipar eventos de congestión y ejecutar acciones correctivas de manera automática, reduciendo los tiempos de respuesta y mejorando la disponibilidad de los servicios de red. Estas características han convertido a la IA en uno de los principales pilares para la evolución hacia redes autónomas.

Dentro de las aplicaciones más relevantes de la Inteligencia Artificial en redes de telecomunicaciones se encuentran la predicción de demanda de tráfico, la detección automática de anomalías, el análisis de fallos, la clasificación de flujos, la asignación dinámica de recursos, la ingeniería de tráfico y el balanceo inteligente de carga. A diferencia de los algoritmos tradicionales, los modelos basados en IA pueden adaptarse continuamente a las condiciones cambiantes de la red, aprendiendo de la experiencia obtenida durante su funcionamiento y mejorando progresivamente la calidad de las decisiones tomadas.

Actualmente, los principales proveedores de servicios de Internet enfrentan escenarios donde la utilización de los enlaces WAN cambia constantemente debido al comportamiento de los usuarios y al crecimiento del consumo de aplicaciones multimedia. Bajo estas condiciones, mantener configuraciones estáticas de balanceo ocasiona que algunos enlaces permanezcan congestionados mientras otros presentan baja utilización, disminuyendo la eficiencia global de

la infraestructura. En respuesta a esta problemática, diversas investigaciones proponen incorporar algoritmos de aprendizaje automático capaces de adaptar automáticamente las políticas de distribución del tráfico según el estado actual de la red.

Automatización inteligente en redes ISP

Los proveedores de servicios de Internet administran infraestructuras caracterizadas por un elevado volumen de tráfico, múltiples enlaces WAN y una gran cantidad de usuarios concurrentes. Tradicionalmente, la operación de estas redes depende de configuraciones manuales realizadas por administradores especializados, quienes modifican parámetros de enrutamiento cuando detectan congestión o degradación del servicio. Las plataformas AIOps integran información proveniente de protocolos como SNMP, NetFlow, Syslog y APIs de administración para construir modelos capaces de identificar patrones de utilización de los enlaces, detectar anomalías y recomendar acciones de optimización antes de que se produzcan interrupciones del servicio. Estas soluciones permiten disminuir la intervención humana y mejorar la utilización de los recursos disponibles mediante procesos de aprendizaje continuo. En el caso específico de los ISP, la automatización inteligente resulta especialmente útil para optimizar la distribución del tráfico entre múltiples proveedores de Internet. Mientras los métodos tradicionales mantienen reglas fijas de encaminamiento, los algoritmos basados en aprendizaje automático pueden modificar dinámicamente las decisiones de balanceo considerando indicadores como el throughput, la latencia, la pérdida de paquetes y la disponibilidad de los enlaces.

Balanceo de carga en redes de proveedores de servicios de Internet

El balanceo de carga constituye una de las estrategias más importantes dentro de la ingeniería de tráfico, ya que permite distribuir las comunicaciones entre varios enlaces de salida con el propósito de incrementar la disponibilidad de la red, reducir la congestión y aprovechar de manera eficiente el ancho de banda contratado. En infraestructuras ISP es habitual disponer de dos o más enlaces WAN proporcionados por diferentes operadores. Esta redundancia

incrementa la disponibilidad del servicio frente a fallos de conectividad; sin embargo, también introduce el desafío de determinar cómo distribuir adecuadamente el tráfico entre todos los enlaces disponibles. Los mecanismos tradicionales de balanceo utilizan algoritmos determinísticos basados en reglas previamente definidas. Entre los más utilizados destacan Equal Cost Multi Path (ECMP), Per Connection Classifier (PCC), NTH, Weighted ECMP y diversas variantes de balanceo por pesos. Estos algoritmos distribuyen las conexiones considerando parámetros previamente configurados por el administrador, sin incorporar mecanismos que permitan aprender del comportamiento histórico del tráfico ni adaptarse automáticamente a condiciones cambiantes. Aunque estas técnicas presentan una implementación sencilla y bajo costo computacional, diversos estudios coinciden en que su desempeño disminuye cuando la demanda de tráfico cambia rápidamente o cuando existen diferencias importantes entre las capacidades de los enlaces disponibles. Bajo estas condiciones pueden producirse situaciones donde un enlace permanezca congestionado mientras otros continúan subutilizados, incrementando la latencia, la pérdida de paquetes y reduciendo la calidad del servicio ofrecido a los usuarios.

Aprendizaje automático aplicado a redes de telecomunicaciones

La creciente complejidad de las redes de telecomunicaciones ha impulsado la incorporación de técnicas de aprendizaje automático para mejorar la administración de los recursos de red. A diferencia de los métodos tradicionales, basados en reglas estáticas definidas por el administrador, el aprendizaje automático permite que los sistemas identifiquen patrones de comportamiento a partir de los datos históricos y operativos, generando modelos capaces de realizar predicciones o tomar decisiones de manera autónoma. En el ámbito de las redes de comunicación, las aplicaciones del aprendizaje automático abarcan la predicción del tráfico, la clasificación de flujos, la detección de anomalías, la identificación de ataques informáticos, la optimización del enrutamiento y la asignación dinámica de recursos. Estas aplicaciones permiten incrementar la eficiencia operativa de las infraestructuras de red y

mejorar indicadores de desempeño como el throughput, la latencia, la utilización de enlaces y la disponibilidad del servicio.

Aprendizaje por refuerzo aplicado al balanceo de carga

El aprendizaje por refuerzo constituye una de las técnicas de Inteligencia Artificial con mayor crecimiento en el área de las telecomunicaciones. Su objetivo consiste en permitir que un agente aprenda una política óptima de decisiones mediante la interacción repetitiva con un entorno dinámico, sin necesidad de disponer previamente de ejemplos etiquetados.

En un problema de balanceo de carga, el entorno corresponde a la infraestructura de red, mientras que el agente representa el algoritmo encargado de decidir cómo distribuir el tráfico entre los enlaces disponibles. Durante cada iteración, el agente observa el estado de la red, ejecuta una acción y posteriormente recibe una recompensa que refleja la calidad de la decisión tomada. A partir de esta retroalimentación, el algoritmo ajusta progresivamente su política de decisión hasta maximizar la recompensa acumulada.

Diversas investigaciones han demostrado que los algoritmos de aprendizaje por refuerzo mejoran indicadores como la utilización de enlaces, la latencia promedio, el tiempo de respuesta y la distribución del tráfico respecto a métodos heurísticos convencionales. Sin embargo, la mayoría de estos estudios se han desarrollado sobre redes móviles, centros de datos o entornos simulados, existiendo todavía un número reducido de implementaciones realizadas sobre redes ISP utilizando equipos comerciales como MikroTik.

Algoritmo Q-Learning para la optimización del balanceo de carga

El funcionamiento de Q-Learning se fundamenta en la construcción iterativa de una tabla Q, donde cada elemento almacena la utilidad esperada de ejecutar una determinada acción bajo un estado específico. Conforme el agente interactúa con el entorno, los valores de la tabla son actualizados utilizando la recompensa obtenida, permitiendo que las decisiones futuras sean progresivamente más eficientes.

En aplicaciones de balanceo de carga, los estados suelen representar condiciones de la red relacionadas con la utilización de los enlaces, la latencia, el throughput, la pérdida de paquetes o la disponibilidad de los proveedores de Internet. Por su parte, las acciones corresponden a modificaciones en las políticas de distribución del tráfico, tales como aumentar la proporción de tráfico enviada hacia un ISP, disminuir la utilización de otro enlace o mantener la configuración actual cuando la red presenta un comportamiento equilibrado.

Implementaciones de balanceo de carga sobre MikroTik

Los equipos MikroTik se han consolidado como una de las plataformas más utilizadas por pequeños y medianos proveedores de servicios de Internet debido a su flexibilidad de configuración, bajo costo y amplio conjunto de funcionalidades de administración de redes. RouterOS incorpora mecanismos de balanceo de carga como PCC, ECMP, NTH, Failover y enrutamiento por políticas, permitiendo distribuir el tráfico entre múltiples enlaces WAN.

La mayoría de las implementaciones reportadas con MikroTik utilizan configuraciones estáticas basadas en PCC o ECMP. Estas técnicas ofrecen resultados satisfactorios cuando el tráfico mantiene un comportamiento relativamente estable; sin embargo, presentan limitaciones frente a escenarios donde la utilización de los enlaces cambia constantemente, ya que la distribución del tráfico depende exclusivamente de reglas configuradas manualmente. Investigaciones recientes comparan diferentes algoritmos implementados sobre MikroTik y concluyen que métodos como PCC, especialmente cuando se combinan con mecanismos de calidad de servicio, logran una distribución más uniforme del tráfico que ECMP tradicional. No obstante, estas propuestas continúan utilizando configuraciones estáticas y no incorporan mecanismos de aprendizaje capaces de adaptar automáticamente las decisiones de balanceo conforme evolucionan las condiciones de la red.

Análisis de investigaciones relacionadas

Durante los últimos años, el empleo de algoritmos de aprendizaje por refuerzo para el balanceo de carga ha experimentado un crecimiento significativo debido al incremento del

tráfico en las redes de comunicación y a la necesidad de implementar mecanismos de toma de decisiones adaptativos. La mayor parte de las investigaciones se ha orientado hacia redes móviles de quinta generación (5G), redes definidas por software (SDN), computación en el borde (Edge Computing) y centros de datos, donde los algoritmos de Inteligencia Artificial buscan optimizar la utilización de los recursos disponibles mediante políticas de control dinámicas.

Wu et al. (2023) realizaron una revisión sistemática sobre la aplicación del aprendizaje por refuerzo en problemas de balanceo de carga dentro de redes de comunicación. Los autores concluyen que los métodos tradicionales basados en reglas presentan dificultades para adaptarse a patrones de tráfico variables, mientras que los algoritmos de aprendizaje por refuerzo logran mejorar la distribución de la carga gracias a su capacidad para aprender continuamente del entorno. Asimismo, identifican que Q-Learning constituye uno de los algoritmos más utilizados debido a su simplicidad de implementación y bajo costo computacional, aunque destacan que el diseño de la función de recompensa continúa siendo uno de los principales desafíos para obtener políticas eficientes.

En el ámbito de la computación en el borde, Esmaeili et al. (2024) propusieron un mecanismo de balanceo dinámico basado en Q-Learning para distribuir solicitudes entre servidores de Edge Computing. Los resultados evidenciaron mejoras en la utilización de recursos, el throughput y el tiempo promedio de respuesta respecto a algoritmos tradicionales de balanceo. Sin embargo, el estudio se desarrolló en un entorno de simulación y no consideró la implementación sobre dispositivos comerciales utilizados por proveedores de servicios de Internet.

Otro aspecto abordado recientemente corresponde al diseño automático de funciones de recompensa. Konar et al. (2023) propusieron un enfoque basado en aprendizaje por refuerzo inverso (Inverse Reinforcement Learning), permitiendo que el sistema aprenda la función de recompensa a partir del comportamiento de expertos en lugar de definirla

manualmente. Los autores demostraron que este enfoque mejora la calidad de las políticas de balanceo en escenarios dinámicos; sin embargo, reconocen que incrementa la complejidad computacional y requiere una cantidad considerable de datos de entrenamiento, lo que limita su aplicación en infraestructuras con recursos restringidos.

Por otra parte, continúan publicándose estudios que comparan algoritmos tradicionales implementados sobre MikroTik, tales como Round Robin, Weighted Round Robin, PCC y ECMP. Estas investigaciones demuestran que la asignación ponderada mejora la utilización de enlaces con diferentes capacidades y reduce parcialmente la congestión. Sin embargo, todas estas técnicas mantienen reglas de distribución previamente configuradas y no incorporan procesos de aprendizaje que permitan modificar automáticamente las decisiones de balanceo conforme evolucionan las condiciones del tráfico.

Vacío de investigación

A partir del análisis realizado se observa que existe una limitada cantidad de investigaciones orientadas a implementar algoritmos de aprendizaje por refuerzo en redes ISP reales utilizando equipos MikroTik. La mayoría de los trabajos se enfoca en simulaciones, redes móviles o arquitecturas SDN, mientras que las soluciones implementadas sobre RouterOS continúan utilizando técnicas tradicionales como PCC, ECMP o Round Robin, las cuales no poseen capacidad de aprendizaje.

En consecuencia, persiste la necesidad de desarrollar sistemas capaces de monitorear continuamente el estado de la red, analizar indicadores de desempeño como el throughput, la latencia, la pérdida de paquetes y la utilización de enlaces, y modificar dinámicamente las políticas de balanceo mediante algoritmos de aprendizaje por refuerzo.

La presente investigación busca contribuir a este vacío mediante el diseño e implementación de un sistema de balanceo de carga basado en Q-Learning para un proveedor de servicios de Internet, utilizando equipos MikroTik como plataforma de enrutamiento. La solución integra el monitoreo de métricas de red mediante SNMP, la comunicación con

RouterOS a través de su API y un agente inteligente desarrollado en Python que determina de manera adaptativa la política de distribución del tráfico entre dos proveedores de Internet. Con ello se pretende mejorar el aprovechamiento de los enlaces disponibles, disminuir la congestión y optimizar el desempeño general de la infraestructura de red.

Tabla 1

Comparación de investigaciones relacionadas sobre balanceo de carga basado en inteligencia artificial en redes de telecomunicaciones

Autor(es)	Año	Tecnología / Escenario	Algoritmo	Principales resultados	Limitaciones
Farag y Stefanovic	2021	Redes IoT	Q-Learning	Mejoró la entrega de paquetes y redujo el retardo mediante selección dinámica de rutas. (arXiv)	Solo evaluado mediante simulación en redes IoT.
Iturria Rivera y Erol- Kantarci	2021	Redes LTE/5G	Clipped Double Q- Learning	Incrementó throughput y redujo latencia, jitter y pérdida de paquetes respecto a algoritmos tradicionales. (arXiv)	Aplicado únicamente en redes celulares.
Wu et al.	2023	Revisión de redes inteligentes	Reinforcement Learning	Concluyeron que RL mejora el balanceo de carga y la adaptación frente a tráfico dinámico. (arXiv)	La mayoría de los trabajos revisados permanecen en simulación.
Lin et al.	2024	Open-RAN / 5G	Federated Deep Q- Learning	Mejoró la calidad del servicio y la distribución de carga entre estaciones base. (arXiv)	Requiere infraestructura Open-RAN y alto poder computacional.

Autor(es)	Año	Tecnología / Escenario	Algoritmo	Principales resultados	Limitaciones
Alizadeh, Lim y Vu	2024	Redes móviles densas	Multi-Agent Q- Learning	Logró balanceo dinámico y rápida convergencia frente a cambios de movilidad. (arXiv)	No implementado en redes ISP tradicionales.
Procedia Computer Science	2023	MikroTik Multi-ISP	PCC + Failover	Mejoró la estabilidad y la disponibilidad utilizando dos ISP en equipos MikroTik. (ScienceDirect)	Utiliza reglas estáticas sin aprendizaje automático.
Kinetik Journal	2025	MikroTik	PCC + CAKE vs ECMP	PCC+CAKE distribuyó el tráfico de manera más uniforme y mejoró throughput, RTT y jitter. (Kinetik)	No incorpora Inteligencia Artificial.
Wadly et al.	2025	Multi ISP	Policy Based Routing	Optimizó el tráfico según el tipo de aplicación y prioridad del servicio. (YSMK)	Las políticas siguen siendo configuradas manualmente.
Karim y Rosadi	2026	MikroTik	Round Robin y Weighted Round Robin	Weighted Round Robin obtuvo menor pérdida de paquetes y mejor utilización del ancho de banda. (Sekawan Journal)	No existe aprendizaje adaptativo.
Mirsal y Agtriadi	2026	MikroTik	PCC vs Queue Tree	PCC presentó mayor throughput para entornos Multi-ISP. (Jurnal Politeknik Kebumen)	Continúa siendo un algoritmo basado en reglas.

Nota. Elaboración propia

A partir de las investigaciones analizadas se observa que el balanceo de carga continúa siendo un tema de gran interés debido al incremento constante del tráfico en las redes de telecomunicaciones. Los trabajos recientes muestran una evolución desde algoritmos tradicionales, como Round Robin, Weighted Round Robin, ECMP y PCC, hacia métodos basados en Inteligencia Artificial, particularmente en aprendizaje por refuerzo.

Asimismo, se evidencia que los algoritmos de aprendizaje por refuerzo, especialmente Q-Learning, ofrecen ventajas importantes frente a los métodos convencionales al adaptarse continuamente a las condiciones de la red sin depender de reglas estáticas. Los estudios reportan mejoras en métricas como el throughput, la latencia, la utilización de enlaces y la pérdida de paquetes. Sin embargo, la mayoría de estas propuestas fueron validadas mediante simulaciones o en escenarios específicos como redes móviles, IoT, 5G o arquitecturas Open-RAN.

En este contexto, la presente investigación propone el diseño e implementación de un sistema de balanceo de carga basado en Q-Learning, integrado con RouterOS mediante la API de MikroTik y el protocolo SNMP para la obtención de métricas en tiempo real. A diferencia de los enfoques tradicionales, el agente inteligente toma decisiones de forma dinámica considerando variables como el throughput, la latencia, la pérdida de paquetes y la utilización de los enlaces, con el objetivo de mejorar la distribución del tráfico entre dos proveedores de Internet y optimizar el desempeño general de la red ISP.

Marco teórico

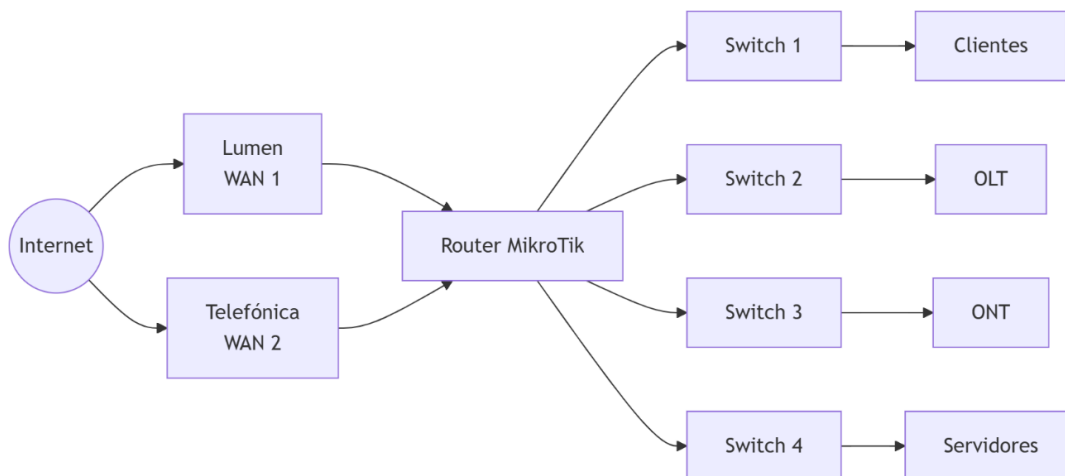
Redes de Proveedores de Servicios de Internet (ISP)

Los Proveedores de Servicios de Internet (ISP, por sus siglas en inglés) son empresas que ofrecen acceso a Internet y otros servicios de telecomunicaciones a usuarios domésticos, empresariales e institucionales. Una infraestructura ISP está conformada por varios equipos de red como pueden ser los routers, switches, servidores, OLT, ONT, sistemas de monitoreo y enlaces de comunicación WAN y LAN que se comunican entre sí, estos a su vez dan servicio a

los clientes que hacen uso de los servicios de los ISP, la disponibilidad del servicio es esencial para dar un funcionamiento eficiente por parte del ISP para la estabilidad de las comunicaciones y el cumplimiento de los niveles de calidad acordados con los clientes (Kurose & Ross, 2021). Una infraestructura ISP se distribuye en varios niveles, cada uno cumple un rol esencial y específico cuya finalidad es dar servicio a usuarios finales, adicional el balanceo de carga se vuelve importante para mejorar la distribución del tráfico y a su vez mejorar la calidad de servicio al usuario final. Estos algoritmos estáticos se basan por lo general en reglas predefinidas, pero tienen limitaciones para adaptarse de manera automática a las condiciones cambiantes del entorno de red. De esta forma, en los últimos años se ha generado un gran interés por implementar técnicas de inteligencia artificial y aprendizaje automático que sean capaces de optimizar de manera dinámica la toma de decisiones en función del comportamiento observado de la infraestructura (Sutton & Barto, 2018).

Figura 2

Arquitectura general del ISP



Nota. Elaboración propia

El objeto del presente trabajo es la infraestructura de QuitoFibranet Cía. Ltda. y en especial en el nodo Toctiuco, donde el acceso a Internet depende de dos enlaces WAN que

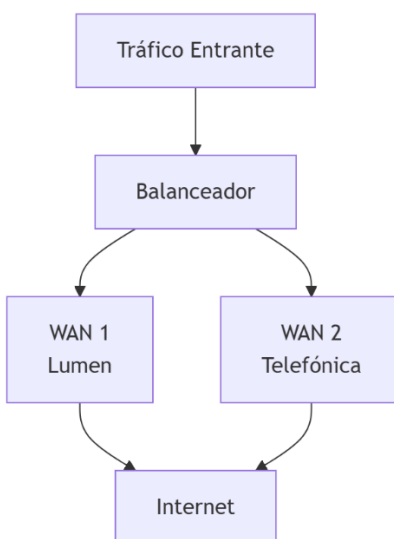
proveen los operadores Lumen y Telefónica. Sobre esta infraestructura se pondrá en marcha un sistema inteligente de balanceo de carga basado en aprendizaje por refuerzo, que tendrá como fin analizar de manera constante el estado de la red y recomendar la distribución más adecuada del tráfico teniendo en cuenta métricas tales como throughput, latencia, pérdida de paquetes y estabilidad operativa.

Balanceo de carga

Una de las estrategias más importantes para administrar de manera eficiente el tráfico en redes de datos es el balanceo de carga (Load Balancing). Su función principal es distribuir de forma equitativa las cargas de trabajo o el tráfico de red entre varios enlaces, servidores o recursos disponibles, evitando que todos los requerimientos se concentren en uno solo y otros queden infrautilizados. Como resultado, se logra una mayor disponibilidad del servicio, un mejor desempeño de la infraestructura y una optimización en el uso del ancho de banda disponible (Kurose & Ross, 2021).

Figura 3

Balanceo de carga



Nota. Elaboración propia

El objetivo principal del balanceo de carga es optimizar la utilización de recursos de red mediante una distribución equitativa del tráfico. Los principales beneficios que ofrece son:

- Evitar saturar los enlaces de comunicación.
- Mejorar la disponibilidad del servicio.
- Acortar los tiempos de respuesta de las aplicaciones.
- Optimizar la utilización del ancho de banda.
- Aumentar la tolerancia a los fallos.
- Asegurar la continuidad operativa de la infraestructura.
- Disminuir la probabilidad de que se degrade el servicio en momentos de alta demanda.

Existen varios mecanismos para hacer balanceo de carga en dispositivos MikroTik. Uno de los más utilizados corresponde al algoritmo Per Connection Classifier (PCC), que realiza la distribución de las conexiones de acuerdo a funciones hash generadas a partir de parámetros como direcciones IP de origen y destino, puertos de comunicación o ambos a la vez. Esta técnica permite mantener una distribución equitativa del tráfico, conservando la persistencia de cada conexión durante toda su vida (Huane Solorzano, 2020).

RouterOS incluye el algoritmo Nth, que está destinado a la distribución de conexiones siguiendo un patrón secuencial predeterminado. Tiene una ventaja que es la facilidad de implementación, sin embargo, este mecanismo no tiene en cuenta ni el comportamiento dinámico de la red ni las condiciones reales de congestión, esto genera que su rendimiento empeore cuando el tráfico es muy variable (Chonillo Bermúdez, 2022).

Muchos ISP también implementan mecanismos de Failover para que, en caso de una falla de uno de los enlaces principales, se garantice la continuidad del servicio. En este caso todo el tráfico se redirige de forma automática al enlace secundario hasta que el enlace principal vuelve a estar disponible. Aunque el Failover aumenta en gran medida la disponibilidad de la red, no se

trata de un mecanismo de optimización del tráfico, este actúa cuando se produce una interrupción del servicio.

Vílchez Espinoza et al. (2020) indican que, a pesar de su amplia utilización, los algoritmos tradicionales presentan una característica común: la mayoría opera mediante reglas predefinidas por el administrador de la red. Esto implica que las decisiones se mantienen constantes y no tienen en cuenta de forma inteligente las continuas variaciones del tráfico, la congestión temporal de los enlaces o el comportamiento histórico de la infraestructura. Estos mecanismos pueden, pues, dar lugar a distribuciones poco eficientes cuando cambian con rapidez las condiciones operativas.

Los últimos avances en inteligencia artificial han hecho posible diseñar nuevas técnicas de balanceo de carga que utilizan aprendizaje automático y que son capaces de ajustar de forma continua las decisiones de distribución del tráfico a medida que cambia el estado de la red. Esta forma de proceder evita tener que definir todas las reglas de decisión de forma manual, dejando que el sistema optimice su comportamiento de forma automática a partir de la experiencia que va adquiriendo durante la operación (Chonillo Bermúdez, 2022).

En el presente trabajo, el balanceo de carga se realizará sobre el nodo Toctiuco de QuitoFibranet Cía. Ltda., con dos enlaces WAN para los proveedores Lumen y Telefónica. A diferencia de los mecanismos tradicionales implementados en RouterOS, la propuesta introduce un agente inteligente basado en Q-Learning que tiene por objetivo analizar de manera paulatina indicadores tales como throughput, latencia, pérdida de paquetes y estabilidad operativa para recomendar la distribución más conveniente del tráfico entre ambos enlaces.

Calidad de Servicio (QoS)

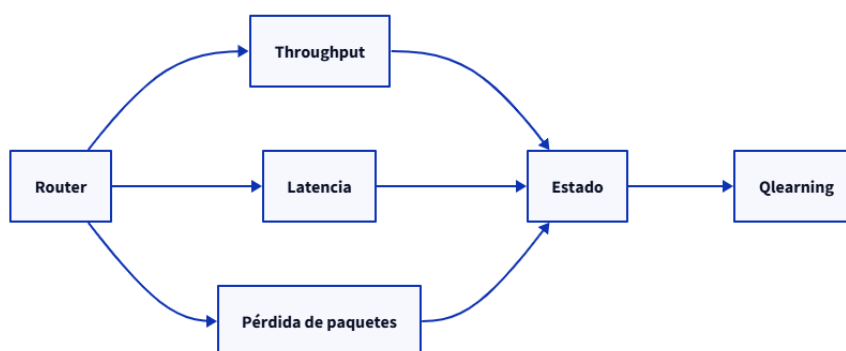
La Calidad de Servicio o también conocida como QoS son los mecanismos, políticas y métricas utilizados para garantizar un determinado nivel de rendimiento dentro de una red de comunicaciones. Su objetivo primordial es gestionar los recursos existentes, garantizando que aplicaciones y servicios sean tratados conforme a sus necesidades de transmisión, y reduciendo

al mínimo problemas como la congestión, los retrasos excesivos y la pérdida de información. En las redes actuales, en particular en los proveedores de servicios de Internet (ISP), la implementación de políticas de QoS es indispensable para tener un servicio estable y cumplir con los niveles de calidad pactados en los SLA (Cisco Systems, 2024).

El tráfico creciente generado por aplicaciones multimedia, servicios en la nube, videoconferencias y plataformas de transmisión en tiempo real ha hecho que sea cada vez más necesario administrar de forma adecuada el ancho de banda disponible. Si una red tiene altos niveles de utilización y no cuenta con mecanismos adecuados de control, pueden ocurrir fenómenos de congestión que afectan a la experiencia del usuario, por ejemplo, mediante el incremento de la latencia, la reducción del throughput y la pérdida de paquetes. Por esta razón, la Calidad del Servicio constituye uno de los elementos fundamentales para el diseño y la operación de infraestructuras de telecomunicaciones (Suqui Carchipulla, 2010).

Figura 4

Flujo de adquisición de métricas



Nota. Elaboración propia

Throughput: El throughput es la cantidad real de información que se transmite de manera satisfactoria entre dos puntos de una red en un tiempo determinado, los bits son representados

por bps, los kilobits por kbps los megabits por Mbps y los gigabits por segundo como Gbps, según la capacidad del enlace evaluado (Kurose & Ross, 2021).

Latencia: En este trabajo, la latencia permitirá evaluar cómo se comportan los dos enlaces WAN durante las pruebas experimentales, y será una de las variables que considere el algoritmo Q-Learning para determinar si procede redistribuir el tráfico al enlace que tenga mejor comportamiento.

Jitter: El jitter representa la variación que existe en la hora de llegada de los paquetes en el transcurso de una comunicación. Aunque se envíen dos paquetes con el mismo intervalo de tiempo, varios factores pueden hacer que lleguen al destino con diferencias temporales, debido a la congestión, cambios de ruta o variaciones en el procesamiento de los dispositivos de red. Aunque el jitter es una métrica muy utilizada para evaluar la Calidad del Servicio, no será considerado como variable experimental en esta investigación, puesto que el objetivo principal es analizar el comportamiento del sistema de balanceo de carga mediante indicadores relacionados con la utilización de enlaces y el desempeño general de la infraestructura (Ortega Ureta, 2019).

Pérdida de paquetes: Es el porcentaje de paquetes en un proceso de transmisión que no llegan de manera correcta a su destino. Esto puede deberse a que haya congestiones, interferencias, errores físicos en las conexiones, averías de hardware, configuración inadecuada o porque se saturó la cola de transmisión. Con gran pérdida de paquetes, las aplicaciones empiezan a sufrir mayor degradación del servicio. En el transcurso de la investigación, la pérdida de paquetes permitirá determinar el nivel de confiabilidad de cada enlace WAN y será una de las variables utilizadas por el agente inteligente para evaluar la calidad de las decisiones de balanceo de carga (León Bolaño & Daza Páez, 2008).

Estabilidad operativa: Esta investigación, además de las métricas tradicionales de QoS, incluye el concepto de estabilidad operativa, entendida como la habilidad del sistema para mantener decisiones consistentes durante la distribución del tráfico, evitando cambios demasiado

frecuentes entre enlaces, esto puede causar oscilaciones innecesarias dentro de la infraestructura. En los sistemas inteligentes basados en aprendizaje por refuerzo, el agente podría ir cambiando de forma continua sus decisiones ante las pequeñas variaciones que se detecten en el estado de la red. Por lo general este comportamiento puede resultar adecuado desde el punto de vista del aprendizaje, en entornos reales puede causar inestabilidad operativa, sobrecargando el procesamiento de los dispositivos y generando fluctuaciones continuas en las rutas de comunicación (León Bolaño & Daza Páez, 2008).

Calidad de Servicio en la presente investigación: La evaluación de la Calidad del Servicio constituye uno de los elementos centrales de la investigación, esto permitirá comparar el desempeño del sistema tradicional de balanceo implementado en RouterOS con la propuesta basada en inteligencia artificial mediante aprendizaje por refuerzo. Los resultados obtenidos permitirán determinar si el algoritmo Q-Learning ayuda a mejorar la utilización de los enlaces WAN del nodo Toctiuco, a reducir la congestión y a mantener una distribución más eficiente del tráfico entre los proveedores Lumen y Telefónica, lo que favorece una mejor calidad del servicio ofrecido por QuitoFibranet Cía. Ltda (Ortega Ureta, 2019).

Inteligencia Artificial

La Inteligencia Artificial se ha ido adoptando en los más diversos sectores, entre los cuales se encuentran la medicina, sector industrial, sector financiero, sector del transporte, la ciberseguridad y las telecomunicaciones. De acuerdo al informe The State of AI 2024 de McKinsey, el 78% de las organizaciones a nivel mundial están empleando IA en al menos una función de negocio, por encima del 55% que había registrado el año pasado. Este aumento muestra que la IA tiene un aporte significativo en la automatización de procesos, en la optimización del uso de los recursos tecnológicos y en la mejora de la toma de decisiones (McKinsey & Company, 2024).

En términos generales, se puede definir a la Inteligencia Artificial como el conjunto de métodos, técnicas y algoritmos computacionales orientados al diseño de sistemas capaces de

percibir información del entorno, analizarla, aprender de la experiencia y realizar acciones que les permitan alcanzar objetivos específicos. A diferencia de los sistemas tradicionales que funcionan sólo con las reglas que ha programado el programador, los sistemas basados en Inteligencia Artificial son capaces de cambiar su comportamiento a medida que obtienen información nueva durante su funcionamiento (Uve et al., 2025).

Las principales ventajas de la Inteligencia Artificial en el contexto de redes de telecomunicaciones son:

- Automatización de procesos de gestión de red.
- Optimización dinámica del ancho de banda.
- Detección precoz de anomalías y fallos.
- Previsión de la conducta del tráfico.
- Reducción de los tiempos de respuesta frente a eventos críticos.
- Mayor disponibilidad de los servicios.
- Optimización de los recursos de cómputo y comunicación.

Estas capacidades permiten que las redes avancen hacia modelos de administración más inteligentes, donde las decisiones operativas ya no dependan sólo de configuraciones manuales, sino que se basen en el análisis continuo de la información del entorno.

Machine Learning: Se caracteriza por ser una de las principales áreas de la IA la cual se encarga de desarrollar algoritmos que son capaces de aprender de manera automática a partir de datos históricos o de la interacción con un entorno, sin que se tenga que programar de manera directa todas las reglas, también se ha utilizado en el ámbito de las telecomunicaciones para optimizar el enrutamiento de paquetes, detectar intrusiones, gestionar recursos de red, predecir la demanda de ancho de banda, clasificar el tráfico y mejorar la calidad del servicio con soluciones inteligentes. Según la forma en que el algoritmo aprende, el Machine Learning se suele dividir en tres categorías principales:

Aprendizaje Supervisado: En el cual el algoritmo aprende a partir de ejemplos que ya han sido etiquetados.

Aprendizaje No Supervisado: Es una técnica cuyo objetivo principal es descubrir patrones ocultos dentro de grandes cantidades de datos.

Aprendizaje por Refuerzo: en el cual un agente aprende a través de la interacción directa con el entorno mediante el uso de recompensas y penalizaciones.

Aprendizaje por Refuerzo

El aprendizaje por refuerzo (Reinforcement Learning, RL) constituye una de las principales ramas del Machine Learning, en la que un agente inteligente aprende a tomar decisiones mediante la interacción continua con un entorno. A diferencia del aprendizaje supervisado, donde el algoritmo se entrena con ejemplos previamente etiquetados, y del aprendizaje no supervisado, cuyo objetivo es identificar patrones o estructuras en los datos sin etiquetas, el aprendizaje por refuerzo se fundamenta en un proceso de prueba y error. En este enfoque, el agente ejecuta acciones, observa las consecuencias de sus decisiones y recibe recompensas o penalizaciones en función de los resultados obtenidos. Con esta retroalimentación, mejora progresivamente su comportamiento hasta encontrar una política de decisión óptima (Sutton y Barto, 2018).

Este modelo toma como referencia la manera en que los seres humanos y otros organismos aprenden de la experiencia. Si una acción deriva en un resultado favorable, hay más probabilidad de que se repita ese comportamiento en circunstancias similares; si una acción tiene consecuencias negativas, el sistema aprende a evitar esa acción en decisiones futuras. Este aprendizaje continuo permite generar modelos que se adapten a los cambios del entorno, sin que el programador tenga que especificar todas las reglas de decisión. En los últimos años, el aprendizaje por refuerzo ha despertado un gran interés por su aplicación en problemas en los que las condiciones varían de manera continua y se necesitan tomar decisiones de forma autónoma. Las áreas más importantes de aplicación son la robótica, la conducción autónoma, la

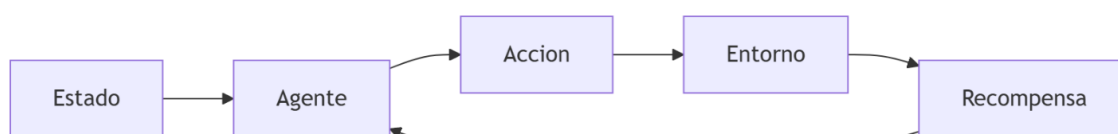
gestión de recursos computacionales, los sistemas financieros, los videojuegos y la gestión inteligente de redes de telecomunicaciones (Russell & Norvig, 2021).

Componentes del Aprendizaje por Refuerzo

El aprendizaje por refuerzo funciona mediante la interacción de varios componentes esenciales que permiten representar el proceso de toma de decisiones.

Figura 5

Ciclo del Aprendizaje por Refuerzo



Nota. Elaboración propia

Agente: El agente es la parte inteligente del sistema y es quien toma las decisiones. Su misión es observar siempre el estado del entorno, elegir una acción y, a continuación, valorar el resultado obtenido mediante una recompensa. A medida que crece el número de interacciones, el agente adquiere más experiencia y, en consecuencia, la calidad de sus decisiones va mejorando de forma gradual (Sutton & Barto, 2018).

En la investigación propuesta, el agente es el algoritmo desarrollado en lenguaje de programación Python que tiene como función analizar el patrón de la infraestructura de red y proponer la distribución más adecuada del tráfico entre los enlaces WAN disponibles.

Entorno: El entorno es el sistema en el que actúa el agente inteligente. En éste se producen todos los cambios que se originan a partir de las acciones ejecutadas por el algoritmo y representa la fuente de información empleada para el proceso de aprendizaje. El entorno de esta investigación está conformado por la infraestructura del nodo Toctiuco de QuitoFibranet Cía. Ltda., entre ellos los routers MikroTik, los enlaces WAN de Lumen y Telefónica y las métricas obtenidas a través de los mecanismos de monitoreo implementados.

Estado: El estado es la representación de la situación del entorno en un momento dado. Explica las condiciones en que el agente ha de decidir. En esta investigación, el estado será formado por variables relacionadas con el desempeño de la red, entre ellas:

- Throughput de la WAN 1.
- Velocidad efectiva del enlace WAN 2.
- Promedio de latencia.
- Pérdida de paquetes
- Grado de utilización de los enlaces.
- Estabilidad en el funcionamiento.

Acción: La acción es la elección que el agente toma sobre el entorno. En aplicaciones de telecomunicaciones, las acciones pueden incluir cambios en políticas de enrutamiento, asignación de recursos o tráfico. En este trabajo de investigación, las acciones a tomar consisten en recomendar el porcentaje de tráfico que debe utilizar cada uno de los enlaces WAN disponibles, con el fin de optimizar el desempeño general de la infraestructura.

Recompensa: La recompensa se calcula como una combinación lineal de métricas relacionadas con el balance de carga, la latencia, la pérdida de paquetes y la disponibilidad de los enlaces. El término principal corresponde a 1–imbalance, donde el desequilibrio de carga se define como:

$$imbalance = \frac{load_1 - load_2}{load_1 + load_2} \quad (1)$$

Siendo $load_1 = RX1 + TX1$ y $load_2 = RX2 + TX2$, que representan la carga total de tráfico (recepción y transmisión) de los enlaces ISP 1 e ISP 2, respectivamente. Esta métrica cuantifica la diferencia relativa de tráfico entre ambos enlaces, tomando valores en el intervalo $[0,1]$, donde un valor cercano a 0 indica una distribución equilibrada de la carga. A este valor se le restan penalizaciones normalizadas asociadas a la latencia y la pérdida de paquetes, y se añade una bonificación cuando el desequilibrio es reducido. Finalmente, la recompensa se limita al rango

$[-1,1]$. Este diseño prioriza el equilibrio de carga como objetivo principal, mientras que la latencia y la pérdida de paquetes actúan como criterios complementarios para la toma de decisiones.

Política de decisión: La política es la estrategia que emplea el agente para elegir la acción que va a realizar en cada estado. Al principio, por falta de experiencia, las decisiones pueden parecer aleatorias, pero conforme se aumenta la interacción con el entorno la política se va haciendo más eficiente. El propósito del aprendizaje por refuerzo es hallar la política óptima, o sea, la que permita maximizar la recompensa acumulada durante todo el proceso de operación del sistema.

Exploración y explotación: Uno de los mayores retos del aprendizaje por refuerzo es encontrar un buen equilibrio entre los procesos de exploración y explotación. La exploración permite que el agente pueda probar nuevas acciones con el fin de encontrar alternativas mejores a las conocidas hasta ahora. Algunas de estas decisiones pueden producir resultados no muy favorables, pero constituyen una fuente importante de aprendizaje.

Ventajas del Aprendizaje por Refuerzo: Principales ventajas aplicadas a redes de telecomunicaciones:

- Capacidad de adaptación a cambios dinámicos en el tráfico.
- Aprendizaje gracias medio de la experiencia.
- Reducir la intervención manual de usuarios físicos.
- Optimización automática de recursos.
- Mayor efectividad en la toma de decisiones.
- Capacidad de operar en entornos con alta variabilidad.

Estas propiedades hacen que el aprendizaje por refuerzo sea una alternativa muy prometedora respecto a los algoritmos tradicionales de balanceo de carga, que por lo general dependen de reglas estáticas establecidas con anterioridad.

Aplicación del Aprendizaje por Refuerzo en la presente investigación

En el presente trabajo el sistema inteligente de balanceo de carga implementado está basado en el aprendizaje por refuerzo. El agente desarrollado monitorea el estado de la infraestructura del nodo Toctiuco, a través de las métricas de throughput, latencia, pérdida de paquetes y estabilidad operativa, las cuales representan el estado del entorno, el algoritmo aprende decisiones basadas en políticas y que son orientadas a maximizar el throughput, disminución de la latencia y pérdida de paquetes, evitando cambios innecesarios en la distribución del tráfico, las decisiones de balanceo se adaptan de manera dinámica a las variaciones de las condiciones de la red, en lugar de depender de reglas estáticas predefinidas, el sistema utiliza el algoritmo Q-Learning, el cual constituye el núcleo del proceso de aprendizaje y de la toma de decisiones en esta investigación (Watkins & Dayan, 1992; Sutton & Barto, 2018).

Arquitectura del sistema

El sistema inteligente de balanceo de carga está compuesto por tres capas principales:

Monitoreo: obtiene métricas de throughput, latencia, pérdida de paquetes y estado de los enlaces desde el nodo Toctiuco.

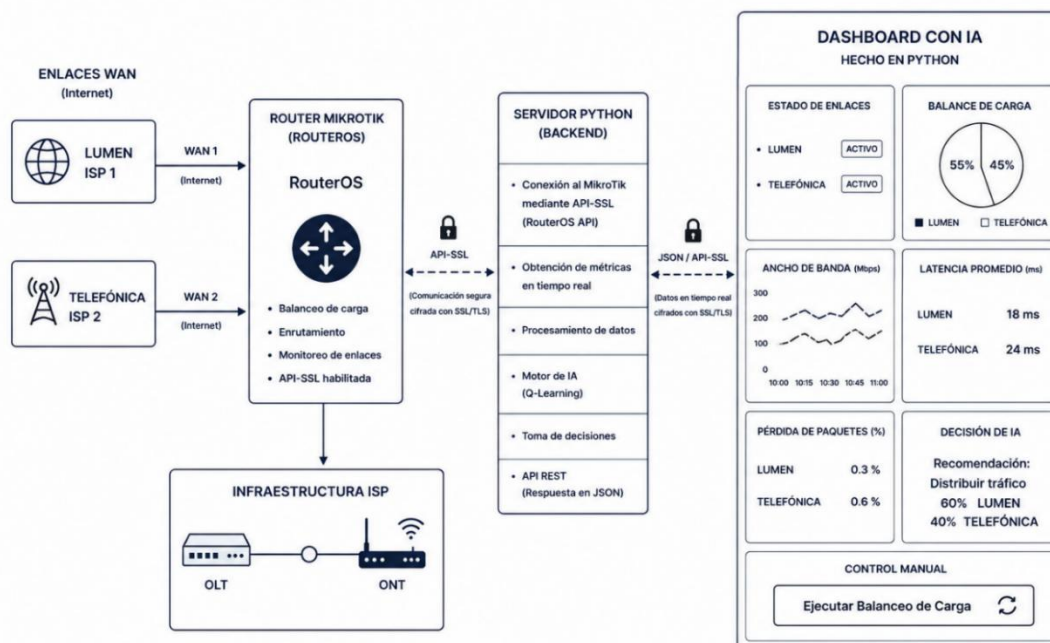
Agente de decisión: un agente Q-Learning que procesa esas métricas como estado del entorno.

Control: aplica las decisiones al balanceo de tráfico del router MikroTik mediante reglas dinámicas, en lugar de usar reglas estáticas predefinidas.

Esta arquitectura permite que el sistema observe continuamente el entorno, tome decisiones en función de su experiencia y aplique cambios en tiempo real.

Figura 6

Arquitectura del sistema inteligente de balanceo de carga basado en IA



Nota. Elaboración propia

En la Figura 6, se muestra la arquitectura del sistema inteligente de balanceo de carga desarrollado para un ISP. El sistema recibe el tráfico de dos enlaces WAN correspondientes a los proveedores Lumen y Telefónica, donde se conectan a un router MikroTik con RouterOS que es el encargado del balanceo de carga y monitoreo de los enlaces. El router cuenta con API-SSL habilitada para la integración con la aplicación desarrollada, estableciendo una comunicación segura con un servidor implementado en python junto con el framework FastAPI, haciendo uso de la librería RouterOS API, mediante la API de RouterOS, el servidor consulta en tiempo real las métricas de la red. Entre las métricas recopiladas se incluyen el estado de los enlaces WAN (activo/inactivo), el throughput de transmisión y recepción (Mbps), la latencia (ms), la pérdida de paquetes (%), estos indicadores que permiten caracterizar el estado actual de la infraestructura y que constituyen la información de entrada que utiliza el agente de aprendizaje por refuerzo para la toma de decisiones. Estas métricas son analizadas por un algoritmo de inteligencia artificial basado en Q-Learning, el cual evalúa las condiciones de la red y genera recomendaciones para

optimizar la distribución del tráfico entre ambos proveedores. Por último, la información se envía al dashboard desarrollado en Python, donde el administrador puede ver en tiempo real los indicadores de desempeño de la red, el estado de los enlaces, las métricas de rendimiento y las recomendaciones generadas por el algoritmo, y también puede ejecutar de manera manual el balanceo de carga cuando sea necesario.

Variables del modelo

En el agente Q-Learning, el estado del modelo se representa con variables derivadas de las métricas de red:

- **Throughput:** carga total en cada ISP medida en bits por segundo.
- **Latencia:** tiempo promedio de respuesta hacia el gateway.
- **Pérdida de paquetes:** porcentaje de paquetes perdidos.
- **Estabilidad operativa:** reflejada en si cada enlace está activo o caído.

Además, el modelo discretiza el grado de desequilibrio de carga, la latencia y la pérdida para construir el espacio de estados del agente.

Objetivo del agente

El objetivo del agente es maximizar la calidad del balanceo de carga. En la presente investigación, el agente busca:

- Equilibrar el tráfico entre ISP1 e ISP2
- Minimizar la latencia
- Reducir la pérdida de paquetes
- Mantener la estabilidad operativa

Para ello utiliza una función de recompensa que prioriza el equilibrio de carga, penaliza degradaciones en latencia/pérdida y castiga enlaces caídos. De esta forma, las decisiones de balanceo se adaptan dinámicamente a las condiciones de la red.

Algoritmo Q-Learning

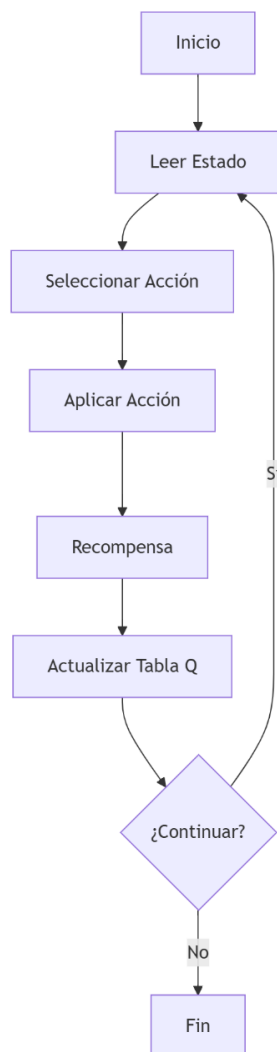
Uno de los métodos de aprendizaje por refuerzo más utilizados dentro del área de la Inteligencia Artificial es el algoritmo Q-Learning, debido a su simplicidad, eficiencia y capacidad para aprender políticas óptimas sin necesidad de disponer de un modelo matemático del entorno. Fue propuesto por Christopher Watkins en 1989 y posterior formalizado junto a Peter Dayan en 1992, y se ha convertido en uno de los algoritmos más usados dentro del aprendizaje por refuerzo, aquí el algoritmo aprende a través de un proceso iterativo en el cual está evaluando los resultados que consigue después de ejecutar una acción determinada. Esta característica es útil en redes de telecomunicaciones, donde el tráfico tiene un comportamiento cambiante y es difícil modelar todas las condiciones posibles (Watkins & Dayan, 1992).

Principio de funcionamiento

El algoritmo Q-Learning funciona mediante la creación de una Tabla Q (Q-Table), el cual se encarga de guardar el valor esperado de realizar una acción determinada en un estado determinado, en este caso en cada posición de la tabla representa el conocimiento adquirido por el agente en el proceso de aprendizaje.

Figura 7

Funcionamiento de Q-Learning



Nota. Elaboración propia

Conforme el agente interactúa con el entorno, la Tabla Q se actualizará de manera paulatina con la información obtenida, esto permite que las decisiones a tomar sean eficientes. Al principio, todos los valores de la tabla son cero o valores aleatorios, pero a medida que se incrementa el número de iteraciones el algoritmo aprende qué acciones generan mayor recompensa y refuerza esas decisiones.

En los problemas de carga balanceada, los estados pueden representar diversas condiciones de la red, y las acciones corresponden a las decisiones posibles para distribuir el

tráfico entre los enlaces disponibles. Así, el agente va aprendiendo cual es la estrategia que permite obtener un mejor rendimiento de la infraestructura.

Tabla Q: La Tabla Q es el dato más importante en la estructura del algoritmo. Su objetivo es guardar el valor esperado de cada acción posible para todos los estados conocidos del entorno. Por lo general una tabla Q puede verse así:

Tabla 2

Estructura general de la matriz Q del algoritmo Q-Learning

Estado	Acción 1	Acción 2	Acción 3
Estado 1	Q_{11}	Q_{12}	Q_{13}
Estado 2	Q_{21}	Q_{22}	Q_{23}
Estado 3	Q_{31}	Q_{32}	Q_{33}

Nota. Elaboración propia

Cada valor guardado representa la utilidad que se espera obtener al ejecutar una acción determinada cuando el sistema está en un estado determinado. En el transcurso del entrenamiento, se van cambiando estos valores para que se acerquen cada vez más a la política óptima.

Ecuación de actualización: El aprendizaje del algoritmo se realiza mediante la siguiente ecuación propuesta por Watkins y Dayan (1992):

$$Q(s, a) = Q(s, a) + \alpha[r + \gamma \max_{a'} Q(s', a') - Q(s, a)] \quad (2)$$

Donde:

Q(s,a): Valor actual asociado al estado s y la acción a.

α (Alpha): Representa a la tasa de aprendizaje.

γ (Gamma): Hace referencia al factor de descuento.

r: Es la recompensa obtenida después de ejecutar la acción.

s': Es el nuevo estado alcanzado.

$\max Q(s',a')$: Es la mayor recompensa estimada para el siguiente estado.

La ecuación permite al agente actualizar su aprendizaje, teniendo en cuenta la recompensa junto con el beneficio esperado de las decisiones que serán futuras. Como consecuencia, el algoritmo no sólo optimiza el resultado actual, sino que aprende estrategias que produzcan mejores beneficios a largo plazo (Sutton & Barto, 2018).

Ecuación para la normalización de la latencia

$$L_n = \min\left(\frac{\frac{L_1 + L_2}{2}}{L_{max}}, 1\right) \quad (3)$$

L_n: Latencia normalizada

L₁: Latencia del enlace ISP 1

L₂: Latencia del enlace ISP 2

L_{max}: Latencia máxima esperada como referencia

min(...,1): Limita el valor máximo a 1

Ecuación para la normalización de la pérdida de paquetes

$$P_n = \min\left(\frac{\frac{P_1 + P_2}{2}}{P_{max}}, 1\right) \quad (4)$$

P_n: Pérdida de paquetes normalizada

P₁: Porcentaje de pérdida del ISP 1

P₂: Porcentaje de pérdida del ISP 2

P_{max}: Pérdida máxima aceptable como referencia

min(...,1): Limita el valor máximo a 1

Ecuación del índice de desequilibrio de carga

$$I = \frac{|L_1 - L_2|}{L_1 + L_2} \quad (5)$$

I: Índice de desequilibrio de carga

L₁: Carga del enlace ISP 1 (Lumen)

L₂: Carga del enlace ISP 2 (Telefónica)

L1+L2/L1-L2: Tráfico total de ambos enlaces

Ecuación de la recompensa total (Reward Function)

$$R = W_{BALANCE} \cdot R_{balance} - W_{LATENCY} \cdot L_n - W_{LOSS} \cdot P_n - Penalty \quad (6)$$

R: Recompensa final entregada al agente

W_BALANCE: Peso del equilibrio de carga

I: Índice de desequilibrio entre enlaces

W_LATENCY: Peso de la latencia

L_n: Latencia normalizada

W_LOSS: Peso de pérdida de paquetes

P_n: Pérdida de paquetes normalizada

Penalty: Penalización por inestabilidad

Ecuación para el cálculo de throughput o ancho de banda utilizado

$$TX_{bps} = \frac{\Delta TX \times 8}{\Delta t} \quad (7)$$

TX_bps: Velocidad de transmisión en bits por segundo

ΔTX: Diferencia de bytes transmitidos entre dos mediciones

8: Conversión de bytes a bits (1 byte = 8 bits)

Δt: Intervalo de tiempo entre mediciones en segundos

Ecuación del decaimiento de exploración

$$\epsilon_t = \max(\epsilon_{min}, \epsilon_0 \times d^t) \quad (8)$$

ε_t: Probabilidad actual de exploración

ε_0: Valor inicial de exploración

d: Factor de reducción

t: Número de episodios realizados

ε_min: Valor mínimo permitido

Parámetros principales

El comportamiento del algoritmo depende en gran medida de la configuración de sus parámetros de aprendizaje.

Tasa de aprendizaje (α): El grado de aprendizaje es el que indica la influencia que la información nueva tiene sobre los conocimientos ya almacenados. Se utiliza $\alpha = 0.35$, este valor se elige como un compromiso entre permitir que el agente incorpore nueva información con rapidez, y evitar cambios bruscos en la tabla Q que puedan generar inestabilidad en el balanceo. Una tasa de aprendizaje moderada permite adaptarse a variaciones de tráfico sin que el agente rehaga su política de forma errática. Valores más altos harían que una única observación altere demasiado la política; valores más bajos la harían demasiado lenta para reaccionar.

Factor de descuento (γ): El factor de descuento determina la importancia que se asigna a las recompensas futuras. Se utiliza $\gamma = 0.95$, este valor es alto porque en redes la calidad de una decisión no se limita al instante actual. El factor de descuento cercano a 1 otorga importancia a las recompensas futuras, lo cual es deseable en telecomunicaciones, ya que una buena configuración de balanceo puede mantener el rendimiento estable en los intervalos siguientes. Así se busca un comportamiento que optimice no solo el presente, sino también el efecto acumulado de las acciones.

Exploración (ϵ): Otro parámetro relevante es ϵ (épsilon), utilizado en la estrategia ϵ -greedy. Se utiliza $\epsilon_{\text{inicial}} = 0.20$, con $\epsilon_{\text{decay}} = 0.98$ y $\epsilon_{\text{min}} = 0.02$, esto significa que al inicio el agente explora un 20 % de las veces, la exploración disminuye gradualmente, siempre conserva al menos 2 % de probabilidad de probar nuevas acciones. Esta configuración permite que el agente pruebe alternativas al comienzo sin comportarse de forma demasiado aleatoria. El decaimiento exponencial hace que el sistema transite de exploración a explotación de forma suave, manteniendo un pequeño margen de adaptación frente a cambios en el tráfico.

Función de recompensa: Una de las partes más importantes de cualquier algoritmo Q-Learning es la función de recompensa. La función de recompensa está definida con pesos y componentes explícitos: $W_{\text{BALANCE}} = 0.80$, $W_{\text{LATENCY}} = 0.15$ y $W_{\text{LOSS}} = 0.05$, los cuales

tienen como objetivo principal el equilibrio de carga entre ISP1 e ISP2, por lo que ese término recibe el peso más alto. La latencia y la pérdida son factores secundarios importantes, pero menos prioritarios, se añade una penalización extra si algún enlace está caído, además, se refuerzan estados muy balanceados con un bono adicional: +0.20 si el desequilibrio es menor al 10 %, +0.10 si el desequilibrio está entre 10 % y 20 %. Finalmente, el valor se limita al rango [-1, 1]. Esta función refleja la prioridad del problema: mantener ambos ISP balanceados, sin dejar de penalizar degradaciones de calidad (latencia/pérdida) y sin olvidar la disponibilidad de los enlaces.

Ventajas del algoritmo Q-Learning

Diversos trabajos han mostrado que Q-Learning ofrece varias ventajas para resolver problemas de optimización en entornos dinámicos (Sutton & Barto, 2018; Watkins & Dayan, 1992).

Se pueden mencionar entre sus principales beneficios:

- No necesita un modelo matemático del entorno.
- Aprende de la experiencia.
- Se adapta a las condiciones cambiantes.
- Optimiza las decisiones.
- Disminuye la necesidad de intervención humana.
- Es una implementación sencilla.
- Se integra con facilidad a los sistemas de monitoreo en tiempo real.

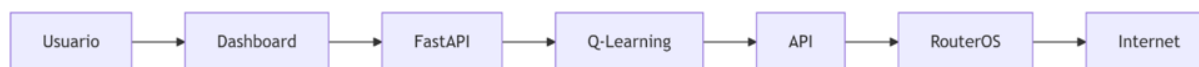
Estas cualidades han posibilitado su uso en múltiples aplicaciones, desde la robótica y los sistemas autónomos hasta los videojuegos, la optimización industrial y la gestión inteligente de redes.

Aplicación de Q-Learning en la presente investigación

El objetivo del presente trabajo ha sido el desarrollo de un sistema inteligente de balanceo de carga para la infraestructura de QuitoFibranet Cía., donde el algoritmo Q-Learning tiene un papel fundamental. Ltda. El agente desarrollado en lenguaje de programación Python, recibe información del nodo Toctiuco, entre otros parámetros, las métricas de throughput, latencia, pérdida de paquetes, utilización de los enlaces WAN ofrecidos por Lumen y Telefónica. Con esta información, el algoritmo identifica el estado presente de la red y revisa la Tabla Q para escoger la acción que brinde el mayor valor anticipado. También se recomienda la distribución del tráfico entre ambos enlaces, con el fin de maximizar el rendimiento global de la infraestructura y mantener una operación estable, cuando se obtiene una nueva decisión también se obtiene una recompensa que la cual depende del comportamiento observado de la red (Teixeira & Trainer, 2019).

Figura 8

Comunicación del sistema



Nota. Elaboración propia

La API de RouterOS en los equipos MikroTik ofrece una interfaz de comunicación que permite el acceso remoto desde aplicaciones desarrolladas en diversos lenguajes de programación como Python, Java, PHP, C# y Go, gracias a esta capacidad, la API de RouterOS es empleada por numerosos sistemas de monitoreo, plataformas de automatización y soluciones que utilizan la IA como alternativa para interactuar con los equipos MikroTik (MikroTik, 2024).

RouterOS: RouterOS es el sistema operativo creado por MikroTik para administrar dispositivos de red. Está basado en el núcleo de Linux e incorpora una amplia variedad de funciones orientadas al enrutamiento, conmutación, seguridad informática, gestión del ancho de banda y monitorización de redes.

Administración mediante API: Estas funcionalidades han hecho que RouterOS sea una de las plataformas más utilizadas por los pequeños y medianos proveedores de servicios de Internet, gracias a su flexibilidad, estabilidad y bajo costo de implementación (MikroTik, 2024).

Funcionamiento de la API: En la API de RouterOS la comunicación funciona por el modelo cliente-servidor. En este caso el router hace de servidor y es una aplicación externa la que establece la conexión mediante las credenciales que ha configurado el administrador. Una vez que el usuario ha sido autenticado, el sistema puede enviar comandos para consultar información o para modificar algunos parámetros del dispositivo.

Con la API se pueden realizar las siguientes operaciones:

- Obtener el estado de las interfaces.
- Comprobar el tráfico de entrada y salida.
- Gestión de rutas.
- Crear o suprimir configuraciones.
- Ejecutar los scripts.
- Obtener información del sistema.

Esto hace posible crear aplicaciones inteligentes que estén conectadas de manera permanente con las instalaciones y no necesiten la intervención de un ser humano.

Integración con Python: Python es un lenguaje de programación muy utilizado para consumir la API de RouterOS gracias a su simplicidad, a la amplia disponibilidad de bibliotecas y a la facilidad que ofrece para integrar procesos de automatización con algoritmos de Inteligencia Artificial. Python puede conectar de forma segura a dispositivos MikroTik y ejecutar operaciones de lectura y escritura sobre diferentes módulos del sistema operativo gracias a bibliotecas diseñadas como puede ser routers (MikroTik, 2024).

Automatización de procesos de red: Uno de los mayores beneficios de emplear la API de RouterOS es que permite automatizar procesos administrativos que, de otra manera, requerirían intervención manual.

Destacan entre estos procesos:

- Vigilancia permanente del estado de la red.
- Métricas de desempeño obtenidas de forma automática.
- Aplicación de configuraciones de balance.
- Cambio de rutas al momento.
- Detección de averías.
- Avisos de alarmas.
- Recopilación de datos para análisis estadístico.

API de MikroTik en la presente investigación

En la presente investigación se empleó la API de RouterOS como medio de comunicación entre el agente inteligente desarrollado en lenguaje de programación Python y los equipos MikroTik instalados en el nodo Toctiuco de QuitoFibranet Cía. Ltda En esta interfaz el sistema obtiene información relacionada con el comportamiento de los enlaces WAN, como métricas de utilización, throughput y estado de las interfaces. Más tarde, el agente basado en Q-Learning examina esos indicadores para decidir cuál es la mejor estrategia de distribución de tráfico. La propuesta incluye un sistema inteligente de apoyo para la toma de decisiones, pero la API deja las puertas abiertas para versiones futuras donde las recomendaciones se puedan ejecutar de forma automática sobre RouterOS, permitiendo una administración dinámica del balanceo de carga (MikroTik, 2024).

Dashboard de Monitoreo

El avance de las infraestructuras tecnológicas ha hecho que cada vez sea más necesario contar con herramientas que permitan una visualización clara y ordenada del comportamiento de

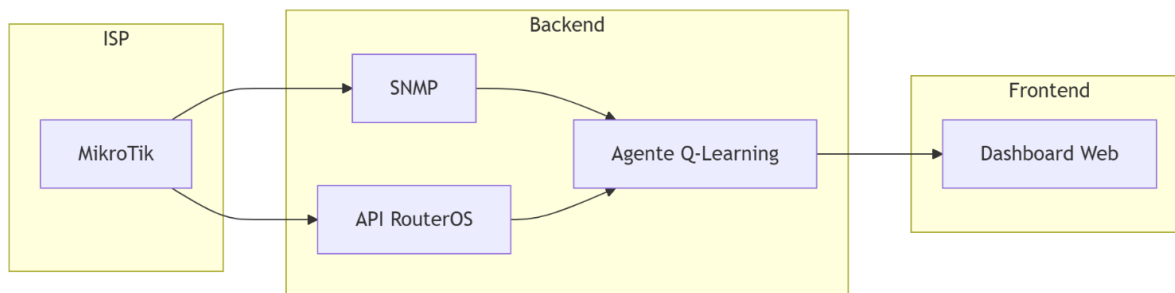
los sistemas de información. En este sentido, los dashboards se presentan como una de las principales soluciones de monitoreo en tiempo real, puesto que sus indicadores relevantes en están en una única interfaz gráfica, facilitando el seguimiento continuo de la infraestructura y apoyando la toma de decisiones por parte de los administradores (Few, 2013).

Para un proveedor de servicios de Internet los dashboards son herramientas fundamentales para supervisar el funcionamiento de los routers, switches, enlaces WAN y sistemas de monitoreo. Estas plataformas permiten que los administradores conozcan el estado de la red en tiempo real, que puedan identificar posibles fallas y tomar medidas correctivas antes de que los usuarios sufran una degradación de calidad del servicio.

Arquitectura del dashboard: Por lo general, un dashboard de monitoreo está formado por tres elementos principales:

Figura 9

Arquitectura del Sistema Inteligente



Nota. Elaboración propia

- Fuente de datos, cuya función es obtener información de los dispositivos de red utilizando protocolos de comunicación o interfaces de programación.
- Servidor de procesamiento, que se encarga de analizar la información recibida, ejecutar algoritmos de procesamiento y preparar los datos para su visualización.

- Interfaz gráfica en donde se presentan los indicadores de manera organizada a través de gráficos, tablas e indicadores visuales.

Tecnologías utilizadas

Se emplearán diferentes tecnologías muy utilizadas en aplicaciones web para el desarrollo del dashboard propuesto en esta investigación. Se utilizará Python como lenguaje de programación para el backend por su facilidad de integración en algoritmos de Inteligencia Artificial, bibliotecas de análisis de datos y comunicación con dispositivos MikroTik mediante la API de RouterOS. HTML será el que aporte la estructura de la interfaz en la web. Con CSS se podrá dar el diseño visual y la presentación de los distintos componentes gráficos. JavaScript añadirá funciones dinámicas que permitan actualizar de forma automática la información mostrada en pantalla, sin que sea necesario recargar la página por completo. Estas tecnologías, una vez incorporadas, harán posible la construcción de una plataforma de monitoreo a la que se podrá acceder desde cualquier navegador web conectado a la red del ISP (Pibaque Villacreses, 2019).

Indicadores visualizados: El dashboard diseñado para el presente trabajo permitirá monitorear de forma continua el comportamiento del sistema inteligente a través de la presentación de distintos indicadores de desempeño.

Se incluirán entre la información mostrada:

- Estado de funcionamiento de los enlaces WAN.
- Rendimiento de cada proveedor.
- Promedio de latencia.
- Pérdida de paquetes.
- Uso de los enlaces.
- Decisión recomendada por el agente inteligente.
- Almacenamiento de recomendaciones usadas.

- Panel del sistema en general.

Importancia para la toma de decisiones: Lo más interesante de un dashboard es que te permite analizar el comportamiento de la red a través de una representación visual de la información. En lugar de revisar distintos registros o hacer consultas manuales en los equipos, el administrador dispone de una vista consolidada que resume el estado operativo de la infraestructura.

Marco conceptual

Agente Inteligente: Es el sistema informático capaz de percibir el entorno y tomar decisiones de acorde a las situaciones presentadas.

Aprendizaje por Refuerzo: Hace referencia al refuerzo y también al castigo dentro del aprendizaje automático.

Balanceo de Carga: Hace referencia al reparto del tráfico de manera equitativa por varios enlaces.

Dashboard: Será la interfaz gráfica donde se podrá visualizar y monitorizar datos.

ISP: También llamado Proveedor de Servicios de Internet.

Latencia: Hace referencia al tiempo que se necesita para que los datos viajen entre dos puntos, es decir inicio a fin.

MikroTik: Hardware y software plataforma para la administración de redes.

Pérdida de paquetes: Los paquetes se pierden cuando se están transmitiendo los datos.

Q-Learning: Algoritmo de refuerzo para optimizar la toma de decisiones.

Throughput: Cantidad efectiva de datos que se envían en un período determinado.

WAN: Red de larga distancia que se emplea para unir distintos puntos geográficos.

Capítulo III

Metodología

Enfoque de investigación

El presente trabajo de investigación toma una postura cuantitativa puesto que, se centra en recopilar, medir y analizar datos numéricos relacionados con el desempeño de una infraestructura de red ISP. Se estudiaron métricas como el throughput, la latencia, la pérdida de paquetes y la utilización de los enlaces, que hacen posible evaluar de forma objetiva el comportamiento del sistema propuesto. El enfoque cuantitativo es adecuado, porque permite comparar los resultados obtenidos mediante los mecanismos tradicionales de balanceo de carga, con los resultados obtenidos mediante la implementación del agente inteligente basado en aprendizaje por refuerzo. Del mismo modo, la investigación posee una vertiente experimental, dado que se implementó un sistema de balanceo de carga basado en aprendizaje por refuerzo en un entorno de operación real, realizándose pruebas controladas para evaluar el impacto sobre las métricas de desempeño de la red. La experimentación abarcó situaciones de operación normal, condiciones de alta demanda y pruebas de tolerancia a fallos, lo cual facilitó la comparación de los resultados obtenidos con el método tradicional de balanceo de carga y la verificación de la efectividad de la solución propuesta. Por último, el trabajo de investigación también será de tipo aplicado puesto que busca resolver una problemática real en una infraestructura tecnológica de un proveedor de servicios de Internet mediante la implementación de una solución basada en Inteligencia Artificial.

Variables de investigación

Con el propósito de evaluar el desempeño del sistema de balanceo de carga basado en inteligencia artificial mediante aprendizaje por refuerzo, se definieron las siguientes variables de investigación.

Variable independiente

La variable independiente corresponde al algoritmo de balanceo de carga, el cual constituye el elemento objeto de estudio en la presente investigación. Se comparó el desempeño del algoritmo basado en aprendizaje por refuerzo mediante Q-Learning con el método tradicional de balanceo de carga implementado en la infraestructura del proveedor de servicios de Internet (ISP). El agente inteligente empleó métricas de red obtenidas en tiempo real para recomendar la política de distribución del tráfico entre los enlaces WAN, adaptando sus decisiones de acuerdo con las condiciones de operación de la red.

Variables dependientes

Las variables dependientes corresponden a las métricas utilizadas para evaluar el desempeño del sistema propuesto y determinar la efectividad del algoritmo implementado.

Throughput: cantidad de datos transmitidos por los enlaces WAN durante un intervalo de tiempo, expresada en Mbps. Esta métrica permitió evaluar la distribución del tráfico entre los proveedores de Internet.

Latencia: tiempo de respuesta de la red obtenido mediante pruebas de conectividad, utilizado para determinar el retardo experimentado durante la transmisión de los paquetes.

Pérdida de paquetes: porcentaje de paquetes que no alcanzaron su destino durante las pruebas realizadas, indicador empleado para evaluar la confiabilidad de la comunicación.

Utilización de enlaces: porcentaje de uso de cada enlace WAN durante la operación del sistema, utilizado para determinar el nivel de equilibrio alcanzado en la distribución del tráfico.

Procedimiento

El procedimiento metodológico desarrollado para la implementación y evaluación del sistema de balanceo de carga basado en aprendizaje por refuerzo se ejecutó mediante las siguientes etapas:

Recolección de métricas de red: Se obtuvieron las métricas de throughput, latencia, pérdida de paquetes y utilización de los enlaces WAN mediante el protocolo SNMP y la API-SSL

de MikroTik. Estas métricas fueron utilizadas para representar el estado de la red y servir como entrada del agente inteligente.

Implementación del agente Q-Learning: Se desarrolló un agente de aprendizaje por refuerzo utilizando Python, el cual se integró con la infraestructura del ISP para procesar las métricas obtenidas y generar recomendaciones de balanceo de carga.

Definición de estados, acciones y función de recompensa: Los estados fueron definidos considerando las condiciones de utilización de los enlaces WAN y las métricas de desempeño de la red. Las acciones correspondieron a las diferentes políticas de balanceo disponibles, mientras que la función de recompensa fue diseñada para favorecer una distribución equilibrada del tráfico, reducir la latencia, minimizar la pérdida de paquetes y mantener la estabilidad del sistema.

Entrenamiento del agente: El agente Q-Learning fue entrenado mediante la interacción continua con el entorno de red. En cada iteración se recopilaron las métricas del estado actual de la infraestructura, el agente seleccionó una acción de balanceo utilizando una política de exploración y explotación, ejecutó la acción correspondiente y calculó una recompensa con base en el desempeño obtenido. Posteriormente, la tabla Q fue actualizada de acuerdo con la ecuación de aprendizaje del algoritmo, permitiendo mejorar progresivamente la selección de acciones durante el proceso de entrenamiento.

Evaluación del sistema: El desempeño del sistema fue evaluado mediante cinco escenarios de prueba: funcionamiento normal, alta demanda de tráfico, congestión del enlace WAN1, congestión del enlace WAN2 y falla parcial de un enlace. En cada escenario se registraron las métricas de throughput, latencia, pérdida de paquetes y utilización de enlaces.

Comparación de resultados: Finalmente, los resultados obtenidos mediante el algoritmo Q-Learning fueron comparados con los obtenidos utilizando el método tradicional de balanceo de carga, con el propósito de determinar las mejoras alcanzadas en los indicadores de desempeño de la red.

Método de evaluación

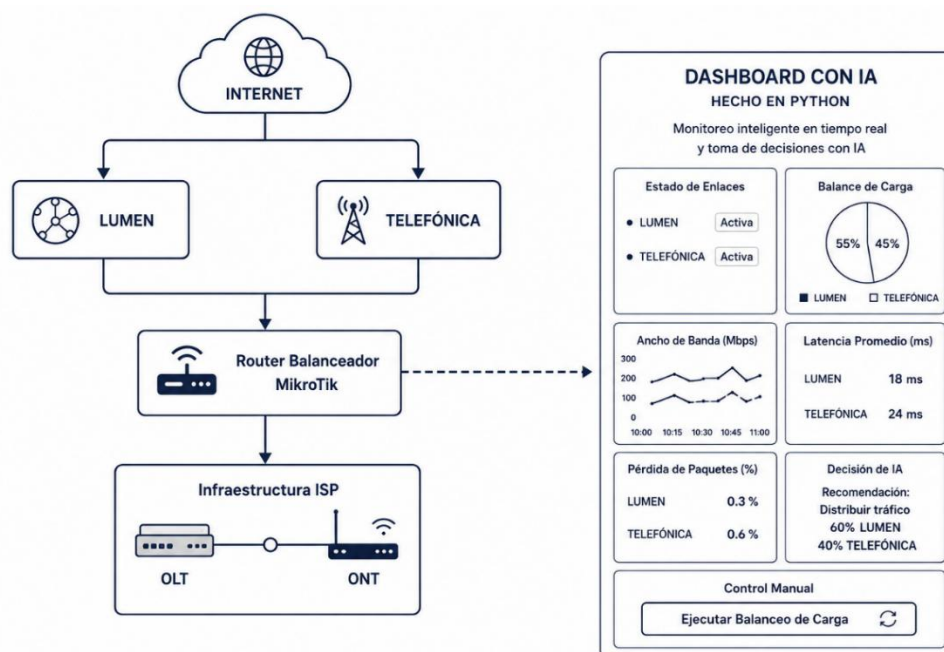
La validación del sistema propuesto se realizó mediante un análisis comparativo entre el método tradicional de balanceo de carga y el sistema basado en aprendizaje por refuerzo implementado en la infraestructura del ISP. Para ello se ejecutaron cinco escenarios de prueba: funcionamiento normal, alta demanda de tráfico, congestión del enlace WAN1, congestión del enlace WAN2 y falla parcial de un enlace.

En cada escenario se realizaron diez mediciones utilizando ambos métodos de balanceo. Durante cada prueba se registraron las métricas de throughput, latencia, pérdida de paquetes y utilización de enlaces mediante herramientas de monitoreo y diagnóstico de red. Posteriormente, los datos obtenidos fueron analizados mediante estadística descriptiva, calculando el promedio de cada indicador para comparar el desempeño del método tradicional con el sistema basado en Q-Learning.

Los resultados permitieron determinar el impacto del algoritmo de aprendizaje por refuerzo sobre la distribución del tráfico y el rendimiento de la infraestructura de red, estableciendo si el sistema propuesto mejoró los indicadores de desempeño respecto al método tradicional.

Figura 10

Arquitectura general del sistema de balanceo de carga basado en inteligencia artificial para redes ISP



Nota. Elaboración propia

En la Figura 10, se muestra la arquitectura general del sistema propuesto de balanceo de carga para la infraestructura del ISP. En este caso los enlaces de Internet suministrados por los proveedores Lumen y Telefónica cuentan con un router MikroTik que actúa como dispositivo central para gestionar y repartir el tráfico de red, en dicho dispositivo se realizará la implementación del algoritmo de aprendizaje por refuerzo que tiene como objetivo analizar métricas de desempeño tales como throughput, latencia y pérdida de paquetes, para seleccionar el enlace con las mejores condiciones de operación. Las decisiones del agente inteligente irán al router de la infraestructura de red que a su vez permitirá optimizar la comunicación entre la red core, los clientes y los diferentes servicios que ofrece. Al mismo tiempo, un dashboard inteligente recoge y muestra las métricas de la red y las decisiones tomadas por el sistema en tiempo real, brindando a los administradores una herramienta para monitorear el comportamiento del balanceo de carga y evaluar el rendimiento de la solución implementada.

Relación de la investigación con la metodología CRISP-DM

El desarrollo del sistema inteligente de balanceo de carga puede enmarcarse dentro de la metodología CRISP-DM (Cross Industry Standard Process for Data Mining), ampliamente utilizada en proyectos de inteligencia artificial y minería de datos. Aunque el presente trabajo se centra en el aprendizaje por refuerzo mediante Q-Learning, las actividades desarrolladas guardan correspondencia con las fases propuestas por dicha metodología.

Comprensión del negocio: Se identificó la problemática asociada al balanceo de carga tradicional en la infraestructura del ISP QuitoFibranet Cía. Ltda., estableciendo como objetivo mejorar la distribución del tráfico y la calidad de servicio mediante técnicas de inteligencia artificial.

Comprensión de los datos: Se analizaron métricas de red como throughput, latencia, pérdida de paquetes y utilización de enlaces obtenidas desde equipos MikroTik mediante SNMP y API-SSL.

Preparación de los datos: Las métricas recopiladas fueron organizadas y utilizadas para representar los estados del entorno requeridos por el agente Q-Learning, permitiendo el proceso de aprendizaje y toma de decisiones.

Modelado: Se desarrolló un agente basado en Q-Learning, definiendo estados, acciones, función de recompensa y parámetros de aprendizaje para optimizar el balanceo de carga entre los enlaces WAN.

Evaluación: El sistema fue evaluado mediante escenarios de funcionamiento normal, alta demanda, congestión y fallos de enlace, comparando los resultados con el método tradicional de balanceo de carga.

Despliegue: La solución fue integrada dentro de la infraestructura real del ISP y complementada con un dashboard web para monitoreo y apoyo a la toma de decisiones.

Tipo y diseño de investigación

Tipo de Investigación

La presente investigación es de tipo aplicada, ya que emplea conocimientos científicos y tecnológicos para desarrollar una solución orientada a optimizar el balanceo de carga en la infraestructura de red de un proveedor de servicios de Internet (ISP). Asimismo, posee un enfoque comparativo, debido a que permite evaluar el desempeño del sistema propuesto basado en aprendizaje por refuerzo frente al método tradicional de balanceo de carga utilizado por el ISP.

Diseño de Investigación

La presente investigación emplea un diseño experimental aplicado basado en escenarios de prueba, con un enfoque de campo, debido a que se implementa un agente basado en aprendizaje por refuerzo (Q-Learning) como mecanismo de recomendación de políticas de balanceo de carga y se evalúa su impacto sobre el desempeño de la red. La investigación es de campo debido a que la recopilación de métricas se realizó sobre la infraestructura tecnológica real del proveedor de servicios de Internet QuitoFibranet Cía. Ltda., considerando parámetros como throughput, latencia, pérdida de paquetes y utilización de enlaces. El componente experimental se evidencia mediante la aplicación controlada de las políticas recomendadas por el agente Q-Learning sobre el equipo MikroTik, permitiendo comparar el comportamiento de la red antes y después del balanceo de carga bajo diferentes escenarios de prueba.

Población y muestra

Población: La población está dada por todo el tráfico de datos que circula por los enlaces principales de acceso a Internet de la infraestructura de QuitoFibranet Cía. Ltda. en el nodo Toctiuco. De la misma manera, a la población se le incorporan los registros de monitoreo producidos por los equipos MikroTik empleados para la administración de la red.

Muestra: La muestra estará constituida por los registros de tráfico obtenidos en los períodos de observación definidos para la investigación.

Los datos serán:

- Throughput enlace WAN1 (Lumen).

- Throughput enlace WAN2 (Telefónica).
- Promedio de la latencia.
- Pérdida de paquetes.
- Enlaces usados.
- Decisiones del agente Q-Learning.

La selección de la muestra se hará mediante un muestreo no probabilístico por conveniencia, ya que los datos se obtendrán de la infraestructura disponible para el proyecto.

Técnica e instrumento

Técnicas de Recolección de Datos

En el presente trabajo de investigación se utilizarán las siguientes técnicas.

Observación Directa: Permite la posibilidad de analizar el comportamiento operativo de la infraestructura de red, así como verificar el funcionamiento de los enlaces WAN.

Monitoreo de Red: Le permite recopilar métricas técnicas en tiempo real mediante herramientas de supervisión y gestión de red.

Análisis Comparativo: Facilita la comparación entre las diferencias que existen entre el balanceo de carga tradicional y el basado en Inteligencia Artificial.

Instrumentos

Se usarán los siguientes instrumentos:

- Router MikroTik del nodo Toctiuco.
- Logs de eventos de red.
- Python scripts.
- FrontEnd será la visualización del dashboard de monitoreo.
- API de MikroTik en este caso RouterOS.

Tabla 3

Herramientas utilizadas para la implementación del sistema

Herramienta	Descripción
Sistema Operativo	Windows Server 2019
Lenguaje	Python 3
Framework Backend	FastAPI
Router	MikroTik CCR1072-1G-8S+
Sistema Router	RouterOS
Comunicación	API-SSL
Librería	RouterOS API
Dashboard	Aplicación Web
Editor	Visual Studio Code
IA	Q-Learning
Administración	WinBox

Nota. Elaboración propia

Las Herramientas y tecnologías utilizadas en el desarrollo e implementación del sistema de balanceo de carga basado en inteligencia artificial se presentan en la Tabla 2. Se utilizó Python 3 como lenguaje de programación principal, FastAPI para el desarrollo de la API REST y RouterOS API mediante API-SSL para la comunicación segura con el router MikroTik CCR1072-1G-8S+ para la construcción de la solución. También se utilizó Visual Studio Code como entorno de desarrollo, WinBox para la administración del router y el algoritmo Q-Learning como parte de la técnica de aprendizaje por refuerzo el cual será el encargado de tomar decisiones y a su vez optimizar el balanceo de carga.

Figura 11

Sistema operativo utilizado en el desarrollo del sistema

Windows specifications

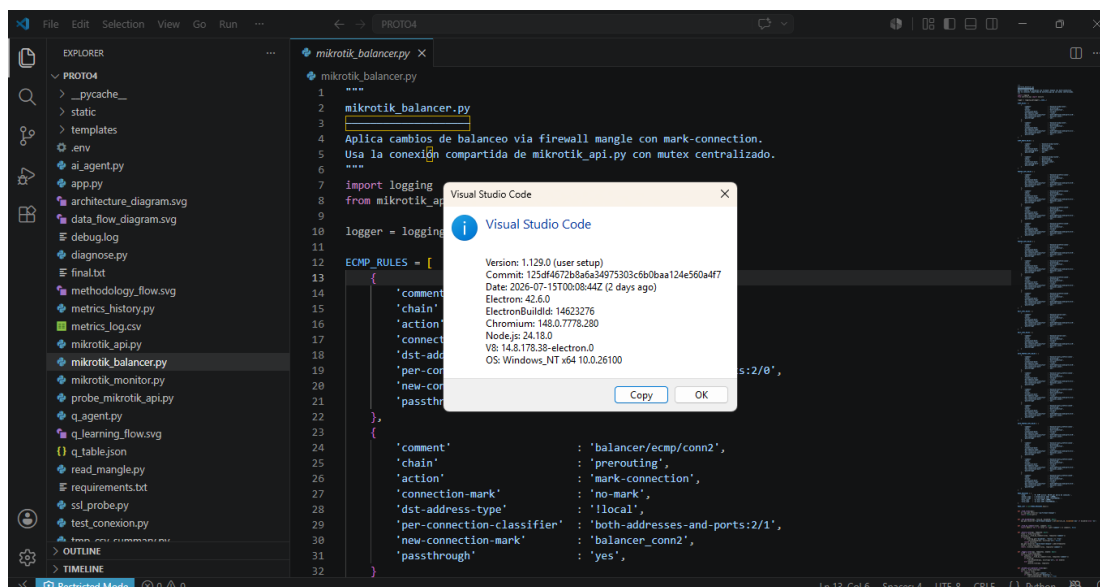
Edition	Windows Server 2019 Standard Evaluation
Version	1809
Installed on	10/3/2025
OS build	17763.3650

Nota. Elaboración propia

En la Figura 11 se muestran las especificaciones del entorno de desarrollo empleado para la implementación del sistema propuesto. Se empleó Windows Server 2019 Standard Evaluation como plataforma base para ejecutar los servicios, desarrollar la aplicación, integrar con MikroTik a través de API-SSL y ejecutar el algoritmo de aprendizaje por refuerzo, proporcionando un entorno estable para las pruebas y validación del sistema.

Figura 12

Entorno de desarrollo Visual Studio Code utilizado para la implementación del sistema



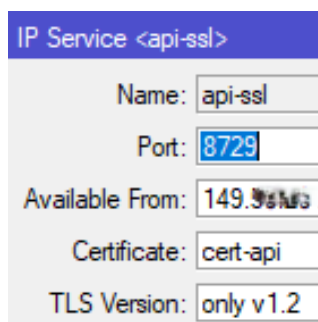
Nota. Elaboración propia

En la Figura 12 se muestra el entorno de desarrollo Visual Studio Code usado para programar el sistema de balanceo de carga basado en inteligencia artificial. La estructura del

proyecto presenta módulos para comunicación con MikroTik, agente de aprendizaje por refuerzo Q-Learning, manejo de métricas de red, una API REST desarrollada con FastAPI y los componentes del dashboard web, lo cual permite una arquitectura modular para monitorear y optimizar el tráfico en la red ISP.

Figura 13

Configuración del servicio API-SSL en MikroTik para la comunicación segura con el sistema



The screenshot shows the 'IP Service <api-ssl>' configuration window in MikroTik WinBox. The fields are as follows:

Field	Value
Name	api-ssl
Port	8729
Available From	149.36.143.143
Certificate	cert-api
TLS Version	only v1.2

Nota. Elaboración propia

En la Figura 13 se muestra como está configurado en el router MikroTik el servicio API-SSL, que es el que se utiliza para establecer una comunicación segura entre la infraestructura de red y el sistema desarrollado. Se procede a habilitar el servicio en el puerto 8729 junto con el certificado digital propio del dispositivo y mediante el uso del protocolo TLS en versión v1.2 para intercambiar de manera segura la información entre el router, la API desarrollada en FastAPI y el agente de aprendizaje por refuerzo implementado.

Figura 14

Dashboard de monitoreo en tiempo real del sistema de balanceo de carga inteligente

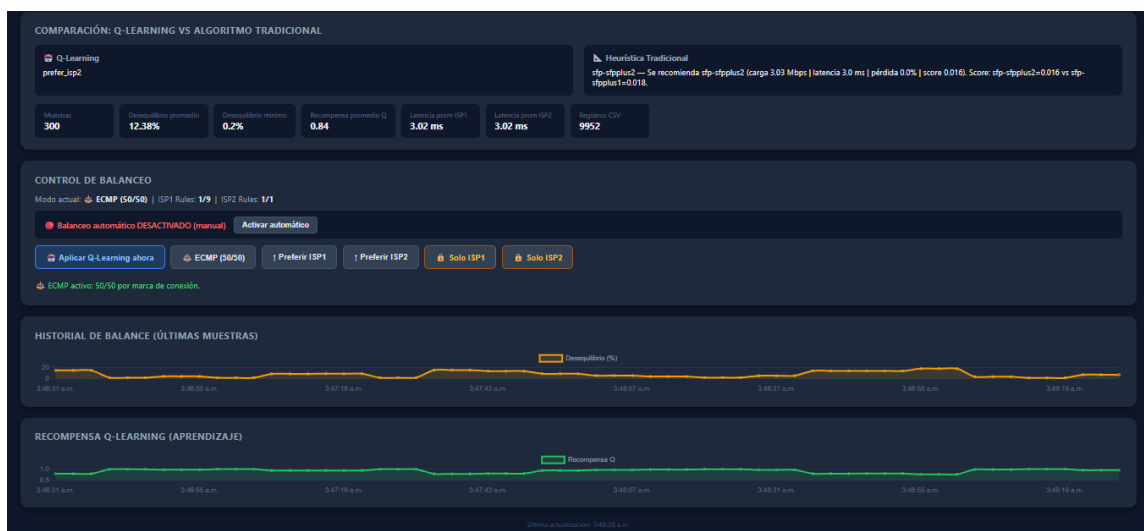


Nota. Elaboración propia

En la Figura 14, se muestra la interfaz principal del dashboard desarrollado para el monitoreo de la infraestructura de red. Se pueden ver en tiempo real las métricas de los enlaces WAN LUMEN y TELEFÓNICA, velocidad de transmisión y recepción, tráfico acumulado, latencia y pérdida de paquetes. También se muestra el porcentaje de distribución de carga entre los dos proveedores, lo cual permite valorar el estado operativo de la red y el comportamiento del sistema de balanceo puesto en marcha.

Figura 15

Panel de control del dashboard de monitoreo en tiempo real del sistema de balanceo de carga inteligente



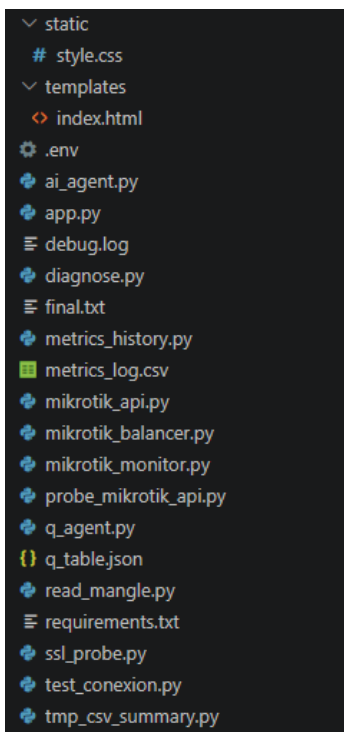
Nota. Elaboración propia

En la Figura 15 se muestra el dashboard desarrollado para el monitoreo y control del sistema de balanceo de carga. En la interfaz se muestran indicadores como el enlace escogido por el algoritmo Q-Learning, el porcentaje de desbalance, la recompensa promedio, la latencia de cada enlace WAN y el historial de las decisiones tomadas. También incluye controles para hacer el balanceo de forma manual o automática y gráficas que permiten ver la evolución del desequilibrio y la recompensa obtenida durante el proceso de aprendizaje.

Estructura del Sistema de Balanceo de Carga – Q-Learning

Figura 16

Estructura de archivos y módulos del sistema de balanceo de carga basado en inteligencia artificial



Nota. Elaboración propia

app.py

- Dashboard de FastAPI principal.
- Usa el agente Q-Learning para mostrar métricas y controlar el balanceo.

ai_agent.py

- Regla heurística tradicional.
- Elija la mejor WAN según carga, latencia y pérdida.

q_agent.py

- Agente Q-Learning.
- Aprende a seleccionar métodos de balanceo para equilibrar ambos ISP.

mikrotik_monitor.py

- Monitorización de interfaces mediante SNMP.
- Calcular el tráfico, la latencia y la pérdida para WAN1/WAN2/OLT.

mikrotik_api.py

- Conexión compartida al RouterOS de MikroTik.
- Detecta automáticamente API-SSL y soporta reconexión segura.

mikrotik_balancer.py

- Se encarga de la aplicación de reglas de balanceo en el equipo MikroTik.
- Es el encargado de definir prefer_isp1, prefer_isp2, only_isp1, only_isp2 y ecmp.

metrics_history.py

- Mantener un registro del historial de métricas y decisiones.
- Guarda datos en memoria y en el archivo csv.

ssl_probe.py

- Se encarga de hacer un testeo en el MikroTik mediante TCP/SSL.
- Se encarga de verificar el funcionamiento del handshake SSL y si el puerto 8729 responde.

test_connection.py

- Archivo de diagnóstico de conexión de MikroTik.
- Se encarga de verificar el puerto 8729 - API-SSL y la autenticación.

diagnose.py

- Guión para diagnóstico rápido.
- El archivo es el encargado de verificar el DNS, puerto API-SSL y TCP del equipo MikroTik.

probe_mikrotik_api.py

- Uso de API routers.
- Responsable de la medición de tiempos de conexión y medición de interfaces.

read_mangle.py

- Script para leer configuración actual de MikroTik .
- Visualiza reglas por defecto y reglas mangle del equipo.

tmp_csv_resumen.py

- Analiza el archivo metrics_log.csv.
- Se encarga del cálculo estadístico básico y del conteo de acciones Q-Learning.

q_table.json

- Almacenada datos de la tabla Q del agente.

metrics_log.csv

- El archivo contendrá un histórico de decisiones y de las métricas.

.env

- Archivo con variables de entorno, como usuario, host, interfaces y password.

requirements.txt**requirements.txt**

- En el archivo .txt se declaran las dependencias del proyecto, como por ejemplo routers-api, uvicorn, fastapi, jinja2 y python-dotenv.

style.css

- Contiene el diseño de un dashboard inteligente.

index.html

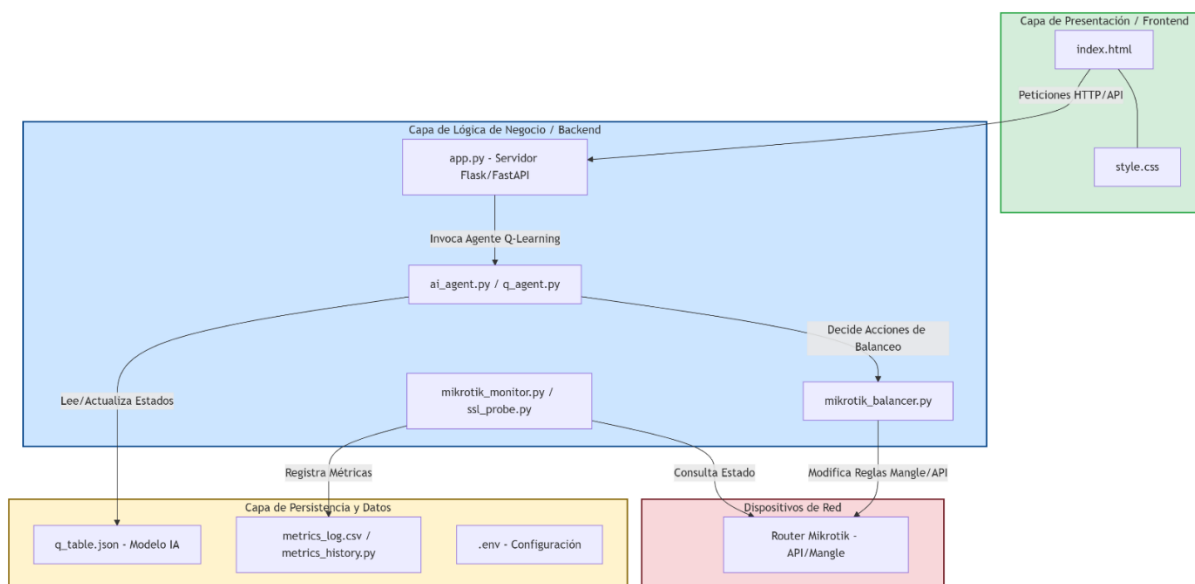
- Plantilla HTML del dashboard en general.

debug.log

- El archivo .log contiene logs de errores o actividad.

Figura 17

Arquitectura lógica del sistema de balanceo de carga basado en inteligencia artificial



Nota. Elaboración propia

El sistema funciona con un modelo de arquitectura de varias capas, orientado a la automatización de redes por inteligencia artificial. El flujo arranca en la capa de infraestructura, donde los scripts de monitoreo del backend capturan métricas en tiempo real del Router MikroTik, guardando los históricos en el archivo persistente `metrics_log.csv`. Con estos datos, el servidor central `app.py` llama a un agente de aprendizaje por refuerzo `q_agent.py` que consulta y actualiza una tabla de conocimiento `q_table.json` para determinar la acción de enrutamiento óptima, esta decisión es ejecutada de forma autónoma por un módulo de balanceo que modifica las reglas Mangle del router vía API, permitiendo al usuario supervisar todo el proceso en tiempo real desde una interfaz web dinámica frontend.

Validación y confiabilidad del instrumento

La validación de los instrumentos se hará mediante pruebas de funcionamiento en la infraestructura real del ISP, verificando que se capten las métricas, se procese la información y se generen resultados. La fiabilidad se determinará por la capacidad de los instrumentos para obtener datos consistentes y reproducibles en diferentes períodos de observación. Con el objetivo de asegurar la calidad de la información recopilada, se efectuarán pruebas piloto

previas al estudio definitivo, comprobando la integridad de los registros obtenidos desde los equipos MikroTik. También se harán comparaciones de las métricas capturadas por el sistema desarrollado con los valores reportados por las herramientas de monitoreo utilizadas por el ISP.

Escenarios de Prueba

Escenarios para la validación del sistema:

Escenario 1: Funcionamiento Normal

Se verificó el patrón del sistema en condiciones normales, comprobando la distribución de carga entre los enlaces WAN1 (Lumen) y WAN2 (Telefónica).

Figura 18

Dashboard inteligente de monitoreo y balanceo de carga basado en Q-Learning



Nota. Elaboración propia

En la Figura 18, se muestra un dashboard inteligente de balanceo de carga basado en Q-Learning en el cual se monitorean dos proveedores de internet: ISP 1 (Lumen) e ISP 2 (Telefónica). En el panel se pueden ver métricas en tiempo real como la velocidad de recepción (RX), transmisión (TX), el tráfico total, datos acumulados y el estado de cada enlace donde ambos se encuentran activos.

Figura 19

Monitoreo de tráfico de las interfaces WAN en el router MikroTik

LUMEN						
RS	sfp-sfpplus1	Ethernet	1500	1580	816 bps	480 bps
TELEFONICA						
RS	sfp-sfpplus2	Ethernet	1500	1580	785.6 kbps	37.4 kbps

Nota. Elaboración propia

En la Figura 19, se muestra la imagen de la interfaz de administración de un equipo MikroTik a través de la herramienta WinBox, donde se puede ver la lista de interfaces de red del sistema. En este panel se pueden ver dos interfaces Ethernet con los parámetros técnicos asociados como velocidad de transmisión (Tx), recepción (Rx), estos datos son fundamentales para la gestión, el monitoreo y la optimización del rendimiento de la red, ya que proporcionan una visión detallada del tráfico de datos y del estado operativo de cada conexión dentro de la infraestructura.

Escenario 2: Alta Demanda

Se estudiaron las decisiones del agente inteligente durante etapas con alto consumo de ancho de banda.

Figura 20

Dashboard de monitoreo y control del agente Q-Learning para el balanceo de carga



Nota. Elaboración propia

En la Figura 20, se muestra la interfaz de un agente de aprendizaje basado en Q-Learning aplicado al balanceo de carga de red, con los distintos parámetros del proceso de entrenamiento y toma de decisiones del algoritmo. En la parte superior se muestran métricas relevantes como el número de episodio, la tasa de exploración (ϵ), la recompensa obtenida y la acción elegida (por ejemplo, dar prioridad a un ISP específico).

Figura 21

Comparación entre las decisiones del agente Q-Learning y el algoritmo tradicional de balanceo de carga



Nota. Elaboración propia

En la Figura 21, se muestra la interfaz del agente de Q-Learning en una etapa más avanzada del proceso de aprendizaje, es el progreso del algoritmo a lo largo de los episodios, en este punto se observan los parámetros como el episodio 1593, con una tasa de exploración baja ($\epsilon = 0.05$) y una recompensa alta (0.9418) quiere decir que el modelo ha convergido a mejores decisiones.

Escenario 3: Congestión de WAN1

En este apartado se redistribuye el tráfico cuando el enlace principal se encuentra congestionado.

Figura 22

Aplicación de políticas de balanceo

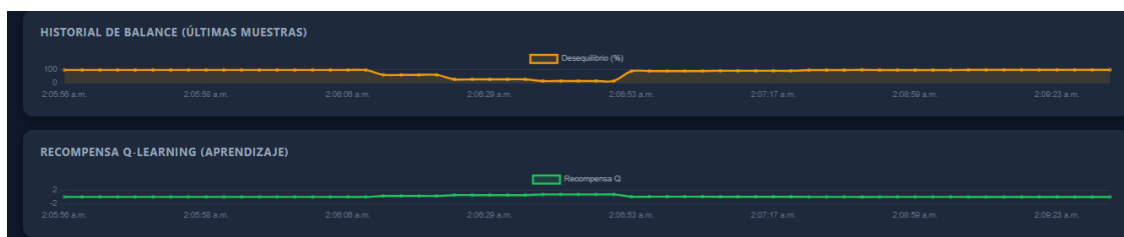
::: balancer/only1/conn		
11	X	mark connection prerouting
::: balancer/ecmp/route1		
12		mark routing prerouting
::: balancer/only2/conn		
13	X	mark connection prerouting
::: balancer/ecmp/route2		
14		mark routing prerouting
::: balancer/prefer2/conn1		
15	X	mark connection prerouting
::: balancer/prefer2/conn2		
16	X	mark connection prerouting
::: balancer/prefer2/conn3		
17	X	mark connection prerouting
::: balancer/prefer2/conn4		
18	X	mark connection prerouting
::: balancer/ecmp/conn1		
19		mark connection prerouting
::: balancer/ecmp/conn2		
20		mark connection prerouting
::: balancer/prefer1/conn1		
21	X	mark connection prerouting
::: balancer/prefer1/conn2		
22	X	mark connection prerouting
::: balancer/prefer1/conn3		
23	X	mark connection prerouting
::: balancer/prefer1/conn4		
24	X	mark connection prerouting

Nota. Elaboración propia

En la Figura 22, se presentan las reglas de Mangle configuradas en el router MikroTik para la implementación de las políticas de balanceo de carga empleadas durante la investigación. Se observan reglas de marcado de conexión (mark connection) y marcado de rutas (mark routing) asociadas a diferentes estrategias de distribución de tráfico, tales como ECMP, preferencia por ISP1, preferencia por ISP2, sólo ISP1 y sólo ISP2, estas configuraciones permiten reconocer, agrupar y enviar los flujos de tráfico a los enlaces WAN correspondientes, y constituyen la base operativa sobre la que el agente de aprendizaje por refuerzo aplica las decisiones de balanceo de carga.

Figura 23

Evolución del desbalance y la recompensa del agente Q-Learning durante el entrenamiento



Nota. Elaboración propia

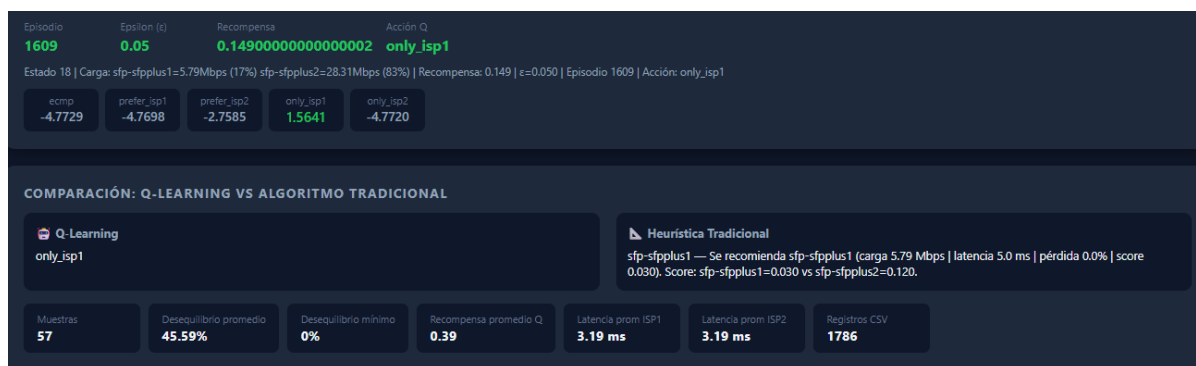
En la Figura 23, se muestran dos gráficas correspondientes al desempeño del sistema de balanceo de carga basado en Q-Learning en el tiempo, ambas gráficas permitirán analizar la evolución del sistema por una parte en el histórico de balanceo como en la parte de recompensa.

Escenario 4: Congestión de WAN2

Se comprobó la respuesta del agente inteligente ante la degradación del enlace secundario.

Figura 24

Estado de ejecución y comparación de decisiones del agente Q-Learning



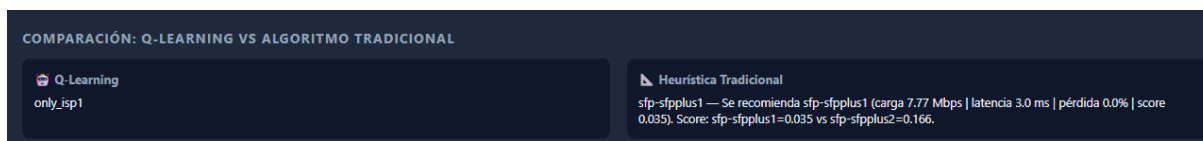
Nota. Elaboración propia

En la Figura 24, se muestra la interfaz del agente de Q-Learning aplicado al balanceo de carga de red en una fase intermedia del aprendizaje, arriba se observan parámetros como el episodio, también aparecen los valores Q asociados a cada acción posible, en la sección

inferior se incluye la comparación con un algoritmo tradicional, junto con métricas como el desbalance promedio, la latencia y la recompensa promedio.

Figura 25

Comparación de decisiones entre el agente Q-Learning y la heurística tradicional de balanceo de carga



Nota. Elaboración propia

En la Figura 25, se puede apreciar una comparación entre la acción elegida por el agente Q-Learning y la recomendación hecha por la heurística clásica de balanceo de carga. En este escenario, el agente inteligente elige la política only_isp1, mientras que la heurística hace su recomendación a partir de métricas de tráfico, latencia y pérdida de paquetes observadas en los enlaces WAN.

Figura 26

Interfaz de control y ejecución del balanceo de carga basado en Q-Learning



Nota. Elaboración propia

En la figura 26, se muestra el panel de control desarrollado para la administración de las políticas de balanceo de carga del sistema, en este escenario se ve que el modo operativo activo es Solo ISP1, con el balanceo automático apagado y el control manual encendido. La interfaz permite aplicar distintas estrategias de reparto del tráfico como Q-Learning, ECMP,

Preferir ISP1, Preferir ISP2, Sólo ISP1 y Sólo ISP2, lo que facilita la comparación de los diferentes esquemas de balanceo implementados y la validación del comportamiento del agente inteligente.

Figura 27

Ejecución del agente Q-Learning con selección de política de balanceo ECMP



Nota. Elaboración propia

En la figura 27, se muestran los parámetros del agente Q-Learning durante una ejecución del proceso de aprendizaje. Se muestra el número de episodio, el valor de exploración, la recompensa obtenida y la acción seleccionada ECMP, también, se muestran los valores Q correspondientes a las acciones disponibles como ecmp, prefer_isp1, prefer_isp2, only_isp1 y only_isp2, que emplea el agente para valorar y seleccionar políticas de balanceo de carga.

Escenario 5: Falla parcial de enlace

Se estudió la capacidad de adaptación del sistema ante fallos temporales en la conectividad.

```
# Revisar si hay un failover forzado por ISP caído
if 'error' not in data:
    failover = _detect_failover_mode(data)
    if failover:
        original = q_rec.get('recommended', 'ecmp')
        q_rec['original_recommended'] = original
        if failover != original:
            target = 'ISP1' if failover == 'only_isp1' else 'ISP2'
```

```

        q_rec['reason'] = f'El ISP {"2" if failover == "only_isp1" else "1"} registra una caída.
Se recomienda mandar todo el tráfico a {target}.'
        q_rec['recommended'] = failover
        q_rec['failover'] = True

# — Aplicar automáticamente la decisión del Q-Learning —
global _last_applied
if _auto_mode and 'error' not in data:
    action = q_rec.get('recommended', 'ecmp')

```

El siguiente fragmento de código que sigue implementa la lógica de detección de fallos en el sistema de balanceo de carga basado en Q-Learning. se comprueba primero que los datos recibidos sean correctos y, en caso de que alguno de los enlaces (ISP) esté caído, se evalúa mediante la función `_detect_failover_mode(data)`. Si se detecta un fallo, el sistema sobrescribe la recomendación original del agente guardando la decisión anterior y estableciendo una nueva acción que es enviar todo el tráfico al ISP activo. También se genera un mensaje de explicación con la razón del cambio, lo cual permite justificar la acción realizada.

Figura 28

Monitoreo en tiempo real del balance de carga entre los enlaces WAN



Nota. Elaboración propia

En la Figura 28, se muestra el monitoreo en tiempo real de dos enlaces de internet, donde se pueden ver métricas clave de rendimiento como velocidad de recepción (RX), transmisión (TX), tráfico total, latencia y pérdida de paquetes, la imagen muestra condiciones de sobrecarga en un enlace y resalta la necesidad de aplicar estrategias de balanceo dinámico para optimizar la distribución del tráfico.

Distribución de Utilización de Enlaces

Figura 29

Monitoreo de tráfico en tiempo real de las interfaces WAN del Router MikroTik

	Name	Type	MTU	Actual MTU	L2 MTU	Tx	Rx
	ether1	Ethernet	1500	1500	1600	0 bps	0 bps
...	LUMEN						
RS	sfp-sfpplus1	Ethernet	1500	1500	1580	13.6 kbps	3.7 kbps
...	TELEFONICA						
RS	sfp-sfpplus2	Ethernet	1500	1500	1580	2.7 Mbps	54.3 Mbps
...	TO OLT						
R	sfp-sfpplus3	Ethernet	1500	1500	1580	57.0 Mbps	2.1 Mbps
...	SW ... DO-BOSQUE						
RS	sfp-sfpplus4	Ethernet	1500	1500	1580	3.7 kbps	13.6 kbps
R	sfp-sfpplus5	Ethernet	1500	1500	1580	5.1 kbps	4.1 kbps
...	SW ... DO-BOSQUE						
RS	sfp-sfpplus6	Ethernet	1500	1500	1580	992 bps	0 bps
	sfp-sfpplus7	Ethernet	1500	1500	1580	0 bps	0 bps
...	to ... FP7						
R	sfp-sfpplus8	Ethernet	1500	1500	1580	47.3 kbps	2.2 Mbps

Nota. Elaboración propia

En la Figura 29, se muestra la pantalla de configuración de interfaces de red en un router MikroTik con la herramienta WinBox, donde se pueden observar las interfaces Ethernet junto con sus parámetros operativos en tiempo real. En el panel se pueden ver enlaces asociados a los distintos nodos y demás conexiones dentro de la red, con información detallada del tráfico, como velocidad de transmisión (Tx), recepción (Rx), paquetes por segundo (Tx/Rx Packet) y valores de procesamiento de paquetes (FastPath).

Figura 30

Balance de carga equilibrado obtenido mediante el agente Q-Learning



Nota. Elaboración propia

En la Figura 30, se muestra el comportamiento del sistema de balanceo de carga en tiempo real y la decisión del agente de Q-Learning en una etapa avanzada de la operación, donde se visualiza que el tráfico se encuentra balanceado.

Figura 31

Estado de aprendizaje y selección de acciones del agente Q-Learning



Nota. Elaboración propia

En la Figura 31, se muestra el estado del agente de Q-Learning en el episodio 2928, mostrando una fase avanzada del proceso de aprendizaje, de igual manera se observan parámetros obtenidos de ϵ , recompensa con la acción a ejecutar.

Tabla 4

Resultados del aprendizaje del agente Q-Learning

Episodio	Acción Seleccionada	Recompensa	Epsilon (ϵ)	Desbalance
1593	only_isp1	0.9418	0.05	8%
2288	prefer_isp1	0.9358	0.05	7.80%
2593	prefer_isp1	0.9743	0.05	3.00%
2639	only_isp1	0.9549	0.05	5.40%
2961	prefer_isp1	0.9184	0.05	10.00%
3532	prefer_isp1	0.95	0.05	6%
4309	prefer_isp1	0.7975	0.05	12%
4926	prefer_isp1	0.5634	0.05	29.40%
7269	prefer_isp1	0.7732	0.05	16%
10759	prefer_isp2	0.9221	0.05	9.60%

Nota. Elaboración propia

En la Tabla 4 se muestran los episodios que se guardan durante el proceso de aprendizaje del agente Q-Learning, se observa el número de episodio ejecutado, la acción elegida por el agente, la recompensa obtenida, el valor del parámetro de exploración ϵ (epsilon) y el porcentaje de desbalance asociado a cada decisión.

Tabla 5

Valores Q observados durante el aprendizaje del agente Q-Learning

Acción	Valor Q observado
prefer_isp1	8.1903
prefer_isp2	7.2775
ECMP	5.2135

only_isp1	4.2236
only_isp2	5.3611
<i>Nota.</i> Elaboración propia	

En la Tabla 5 se muestran los valores Q que corresponden a las distintas acciones que puede tomar el agente de aprendizaje por refuerzo en el transcurso del entrenamiento. En ella se muestran las acciones que evalúa el algoritmo Q-Learning y el valor Q que observa para cada acción.

Consideraciones éticas

Este trabajo de investigación respetará los principios éticos de uso responsable de la información y de la infraestructura tecnológica empleada. El acceso a los equipos de red sólo podrá ser realizado con autorización expresa de QuitoFibranet Cía. Ltda. y bajo supervisión de personal responsable de infraestructura. Los datos recopilados sólo se utilizarán con fines académicos manteniendo en secreto la información sensible de los clientes, configuraciones internas y operaciones del ISP. En las pruebas experimentales se procurará no incidir en la continuidad de los servicios prestados por la empresa, implantando mecanismos de supervisión y de respaldo que permitan minimizar los riesgos operativos. Por último, los resultados obtenidos serán presentados de manera objetiva, transparente y verificable, respetando los principios de integridad académica y ética profesional.

Capítulo IV

Análisis y discusión de los resultados

En este capítulo se exponen los resultados obtenidos en la fase de implementación y evaluación del sistema de balanceo de carga basado en Inteligencia Artificial mediante aprendizaje por refuerzo con el algoritmo Q-Learning. El análisis busca determinar el impacto del sistema de balanceo de carga basado en aprendizaje por refuerzo en las métricas de desempeño de la red del ISP, como la latencia, pérdida de paquetes, jitter, throughput y

tolerancia a fallos, comparando los resultados obtenidos con los mecanismos tradicionales de balanceo de carga implementados en la infraestructura MikroTik del nodo Toctiuco.

Método tradicional

Prueba de latencia y pérdida de paquetes

Tabla 6

Resultados de la prueba de latencia y pérdida de paquetes mediante Ping

sent	received	packet- loss	min-rtt	avg-rtt	max-rtt
20	20	0%	64 ms	64 ms	71 ms
40	40	0%	64 ms	64 ms	71 ms
60	60	0%	64 ms	64 ms	71 ms
80	80	0%	64 ms	64 ms	71 ms
100	100	0%	64 ms	64 ms	71 ms
120	120	0%	64 ms	64 ms	71 ms
140	140	0%	64 ms	64 ms	71 ms
160	160	0%	64 ms	64 ms	71 ms
180	180	0%	64 ms	64 ms	71 ms
200	200	0%	64 ms	64 ms	71 ms
220	220	0%	64 ms	64 ms	71 ms
240	240	0%	64 ms	64 ms	71 ms
260	260	0%	64 ms	64 ms	71 ms
280	280	0%	64 ms	64 ms	71 ms
300	300	0%	64 ms	64 ms	71 ms
320	320	0%	64 ms	64 ms	71 ms
340	340	0%	64 ms	64 ms	71 ms
360	360	0%	64 ms	64 ms	71 ms
380	380	0%	64 ms	64 ms	71 ms
400	400	0%	64 ms	64 ms	71 ms

Nota. Elaboración propia

La Tabla 6 se muestran los resultados obtenidos durante la prueba de conectividad mediante el protocolo ICMP (Ping), se comenzó con 20 paquetes y terminando en un máximo

de 400 paquetes. En todas las muestras registradas, la cantidad de paquetes recibidos fue igual a la cantidad de paquetes enviados, observándose valores de 20 a 400 paquetes, el porcentaje de pérdida de paquetes (packet-loss) fue de 0% en cada una de las mediciones realizadas. Respecto a los tiempos de respuesta, la latencia mínima (min-rtt) registrada fue de 64 ms, la latencia promedio (avg-rtt) se mantuvo constante en 64 ms y la latencia máxima (max-rtt) alcanzó un valor de 71 ms durante las pruebas realizadas.

Prueba de análisis de ruta mediante traceroute

Tabla 7

Resultados de la prueba de análisis de ruta mediante traceroute

Salto	Dirección IP	Latencia Promedio (ms)
1	186.122.132.223	0.2
2	12.113.31.49	2.1
3	10.119.234.15	2.2
4	91.123.143.45	9.7
5	91.15.34.223	63.5
6	5.51.3.124	63.8
7	190.92.121.22	65.6
8	142.245.223.72	64.8
9	74.119.34.125	65.1
10	8.8.8.8	64.3

Nota. Elaboración propia

La Tabla 7 muestra los resultados de la prueba de análisis de ruta al destino 8.8.8.8 mediante Traceroute, donde se visualizan 10 saltos que corresponden a los nodos intermedios recorridos por los paquetes en su trayectoria desde el origen hasta el destino final. Para cada salto se anota la dirección IP asociada y la latencia promedio en milisegundos. Los valores de latencia observados oscilan desde 0.2 ms en el primer salto hasta 65.6 ms en los nodos

intermedios del camino, con el destino final 8.8.8.8 mostrando una latencia promedio de 64.3 ms.

Prueba de variación de retardo (Jitter)

Tabla 8

Resultados de la prueba de variación de retardo (Jitter)

sent	received	packet- loss	min-rtt	avg-rtt	max-rtt	Jitter = Max RTT - Min RTT
20	20	0%	64 ms	64 ms	64 ms	0 ms
40	40	0%	64 ms	64 ms	64 ms	0 ms
60	60	0%	64 ms	64 ms	64 ms	0 ms
80	80	0%	64 ms	64 ms	64 ms	0 ms
100	100	0%	64 ms	64 ms	64 ms	0 ms
20	20	0%	64 ms	64 ms	64 ms	0 ms
40	40	0%	64 ms	64 ms	64 ms	0 ms
60	60	0%	64 ms	64 ms	64 ms	0 ms
80	80	0%	64 ms	64 ms	64 ms	0 ms
100	100	0%	64 ms	64 ms	64 ms	0 ms

Nota. Elaboración propia

En la Tabla 8 se muestran los resultados obtenidos en la prueba de variación de retardo (Jitter) enviando paquetes. Para cada medición se registraron cantidades de 20, 40, 60, 80 y 100 paquetes enviados, observándose el mismo número de paquetes recibidos en todos los casos, el porcentaje de pérdida de paquetes registrado fue de 0 % en cada una de las pruebas efectuadas, los tiempos de respuesta, la latencia mínima, la latencia promedio (avg-rtt) y la latencia máxima registraron valores de 64 ms en todas las mediciones realizadas. Por consiguiente, el valor de Jitter, obtenido de la diferencia entre la latencia máxima y la latencia mínima ($\text{Jitter} = \text{Max RTT} - \text{Min RTT}$), fue de 0 ms para cada uno de los grupos de paquetes analizados.

Análisis estadístico del throughput de los enlaces WAN

Tabla 9

Mediciones de throughput de los enlaces WAN bajo el método tradicional

Medición	Lumen TX	Lumen RX	Lumen Total (Mbps)	Telefónica TX	Telefónica a RX	Telefónica a Total (Mbps)
1	29.7	265.5	295.2	0.972	0.038	1.01
2	44.7	332.2	376.9	0.989	0.023	1.012
3	77.3	398	475.3	1.113	0.025	1.138
4	67.9	332.6	400.5	1.018	0.024	1.042
5	125.8	342.8	468.6	1.108	0.024	1.132
6	62	338.7	400.7	1.149	0.023	1.172
7	103.5	477.8	581.3	0.307	0.007	0.314
8	87.4	331.9	419.3	1.051	0.024	1.075
9	57.9	338.7	396.6	1.115	0.024	1.139
10	48.5	377.3	425.8	1.085	0.022	1.107
11	57.3	480.3	537.6	1.082	0.022	1.104
12	54.3	563.7	618	1.044	0.026	1.07
13	56.5	392.3	448.8	1.073	0.025	1.098
14	54	544.5	598.5	1.034	0.025	1.059
15	109.9	365.9	475.8	1.052	0.021	1.073
16	54.1	382.8	436.9	1.021	0.025	1.046
17	69.2	328.3	397.5	1.124	0.019	1.143
18	88.7	499.3	588	1.076	0.023	1.099
19	48.2	339.4	387.6	1.028	0.024	1.052
20	79.8	395	474.8	0.972	0.024	0.996

Nota. Elaboración propia

La Tabla 9 muestra los valores de throughput medidos tanto en la WAN LUMEN como en la WAN TELEFÓNICA, indicando por separado los valores de transmisión (TX), recepción (RX) y el tráfico total calculado para cada medición. Se tomaron 20 muestras de tráfico, y el enlace LUMEN tuvo valores de transmisión entre 29.7 Mbps y 125.8 Mbps, y de

recepción entre 265.5 Mbps y 563.7 Mbps. De este modo, el tráfico total medido en esa ruta osciló entre 295,2 Mbps y 618 Mbps en los distintos momentos en que se efectuaron las mediciones. Por su parte, el enlace TELEFÓNICA presentó valores de transmisión entre 0.307 Mbps y 1.149 Mbps, mientras que los valores de recepción se situaron entre 0.007 Mbps y 0.038 Mbps, registrando un tráfico total comprendido entre 0.314 Mbps y 1.172 Mbps.

Tabla 10

Análisis estadístico del throughput de los enlaces WAN bajo el método tradicional

Métrica	Lumen Total (Mbps)	Telefónica Total (Mbps)
Promedio	460.1	1.044

Nota. Elaboración propia

En la Tabla 10 se presenta un resumen estadístico del throughput total registrado en los enlaces WAN durante las pruebas realizadas. El enlace LUMEN registró un throughput promedio de 460.185 Mbps, considerando el tráfico total transmitido y recibido a lo largo de las 20 mediciones efectuadas. Por su parte, el enlace TELEFÓNICA presentó un throughput promedio de 1.04405 Mbps. Estos valores representan el comportamiento promedio del tráfico cursado por cada proveedor durante el período de evaluación y permiten sintetizar el nivel de utilización registrado en ambos enlaces WAN.

Monitoreo de recursos del router

Tabla 11

Recursos hardware y estado operativo del router MikroTik

Métrica	Valor
Uptime	1 semana 2 días 16 horas
CPU	TileGX
Núcleos CPU	72

Frecuencia CPU	1000 MHz
Carga CPU	1%
Memoria Total	15.8 GiB
Memoria Libre	14.9 GiB
Almacenamiento Total	128 MiB
Almacenamiento Libre	76.9 MiB
Equipo	CCR1072-1G-8S+

Nota. Elaboración propia

En la Tabla 11 se muestran las características de hardware y el estado del router MikroTik CCR1072-1G-8S+, con un tiempo de 1 semana, 2 días y 16 horas de tiempo de ejecución, una arquitectura TileGX de 72 núcleos a 1000 MHz, una carga de procesamiento del 1 %, 15.8 GiB de memoria total con 14.9 GiB libres y 76.9 MiB de espacio libre en disco.

Utilizando recomendaciones del agente Inteligente(IA)

Prueba de latencia y pérdida de paquetes mediante ping

Tabla 12

Resultados de la prueba de latencia y pérdida de paquetes mediante ping con inteligencia artificial

sent	received	packet- loss	min-rtt	avg-rtt	max-rtt
20	20	0%	64 ms	64 ms	64 ms
40	40	0%	64 ms	64 ms	65 ms
60	60	0%	64 ms	64 ms	66 ms
80	80	0%	64 ms	64 ms	66 ms
100	100	0%	64 ms	64 ms	66 ms
120	120	0%	64 ms	64 ms	66 ms
140	140	0%	64 ms	64 ms	66 ms
160	160	0%	64 ms	64 ms	66 ms
180	180	0%	64 ms	64 ms	66 ms
200	200	0%	64 ms	64 ms	66 ms
220	220	0%	64 ms	64 ms	66 ms

240	240	0%	64 ms	64 ms	66 ms
260	260	0%	64 ms	64 ms	66 ms
280	280	0%	64 ms	64 ms	66 ms
300	300	0%	64 ms	64 ms	66 ms
320	320	0%	64 ms	64 ms	66 ms
340	340	0%	64 ms	64 ms	66 ms
360	360	0%	64 ms	64 ms	66 ms
380	380	0%	64 ms	64 ms	66 ms
400	400	0%	64 ms	64 ms	66 ms

Nota. Elaboración propia

En la Tabla 12 se muestran los resultados obtenidos en la prueba de conectividad mediante el protocolo ICMP (Ping) con el sistema basado en inteligencia artificial. Las mediciones se hicieron con cantidades de paquetes comprendidas entre 20 y 400 paquetes enviados, obteniéndose en todos los casos la misma cantidad de paquetes recibidos que enviados. También, en cada prueba que se realizó, se registró una pérdida de paquetes del 0 %. Respecto a los tiempos de respuesta, la latencia mínima (min-rtt) y la latencia promedio (avg-rtt) fueron consistentes en 64 ms a lo largo de todas las mediciones, mientras que la latencia máxima (max-rtt) osciló entre 64 ms y 66 ms.

Prueba de análisis de ruta mediante traceroute

Tabla 13

Resultados de la prueba de análisis de ruta mediante traceroute con inteligencia artificial

Salto	Dirección IP	Latencia Promedio (ms)
1	214.29.122.225	0.3
2	10.24.11.1	1.8
3	10.24.65.13	2
4	190.26.86.97	8.9
5	200.32.15.201	62.7
6	200.16.589.34	63.1

7	142.250.15.22	64.1
8	142.250.33.14	64.5
9	142.51.70.89	64.7
10	8.8.8.8	64.4

Nota. Elaboración propia

En la Tabla 13 se muestran los resultados de la prueba de análisis de ruta mediante Traceroute al destino 8.8.8.8 utilizando el sistema basado en inteligencia artificial. Se visualiza una ruta de 10 saltos y se registraron las direcciones IP de los distintos nodos intermedios por los que circuló el tráfico. Los saltos iniciales tuvieron una latencia promedio de 0.3 ms, 1.8 ms y 2 ms, mientras que en los saltos intermedios se observó una latencia de 8.9 ms, 62.7 ms y 63.1 ms. En los últimos nodos de la ruta se obtuvieron latencias de 64,1 ms, 64,5 ms y 64,7 ms, llegando al destino 8.8.8.8 con una latencia promedio de 64,4 ms.

Prueba de variación de retardo (Jitter)

Tabla 14

Resultados de la prueba de variación de retardo (Jitter) con Inteligencia Artificial

sent	received	packet- loss	min-rtt	avg-rtt	max-rtt	Jitter = Max RTT - Min RTT
20	20	0%	64 ms	64 ms	64 ms	0 ms
40	40	0%	64 ms	64 ms	64 ms	0 ms
60	60	0%	64 ms	64 ms	64 ms	0 ms
80	80	0%	64 ms	64 ms	64 ms	0 ms
100	100	0%	64 ms	64 ms	64 ms	0 ms
20	20	0%	64 ms	64 ms	64 ms	0 ms
40	40	0%	64 ms	64 ms	64 ms	0 ms
60	60	0%	64 ms	64 ms	64 ms	0 ms
80	80	0%	64 ms	64 ms	64 ms	0 ms
100	100	0%	64 ms	64 ms	64 ms	0 ms

Nota. Elaboración propia

En la Tabla 14 se presentan los resultados de la prueba de variación de retardo (Jitter) mediante envío de paquetes ICMP. En las mediciones se enviaron 20, 40, 60, 80 y 100 paquetes, obteniéndose en cada prueba la misma cantidad de paquetes recibidos (20/20, 40/40, 60/60, 80/80 y 100/100). En todas las mediciones realizadas el porcentaje de pérdida de paquetes fue del 0 %. La latencia mínima, promedio y máxima también mostró un valor de 64 ms, obteniéndose un valor de Jitter de 0 ms para cada una de las pruebas realizadas.

Análisis estadístico del throughput de los enlaces WAN

Tabla 15

Resultados de throughput de los enlaces WAN con inteligencia artificial

Medición	Lumen TX(Mbps)	Lumen RX(Mbps)	Lumen Total(Mbps)	Telefónica TX(Mbps)	Telefónica RX(Mbps)	Telefónica Total(Mbps)
1	39.5	172.6	212.1	29.8	128	157.8
2	45.6	195.4	241	35.3	139.6	174.9
3	52.8	208.1	260.9	40.7	150.2	190.9
4	49.4	201.7	251.1	38.6	145.8	184.4
5	61.5	228.3	289.8	46.8	162.7	209.5
6	44.8	183.9	228.7	34.2	134.9	169.1
7	57.6	239.1	296.7	44.1	174.3	218.4
8	53.4	214.6	268	40.8	155.7	196.5
9	43.2	186.8	230	33.1	133.5	166.6
10	47.1	198.7	245.8	36.4	143.5	179.9
11	52.6	219.5	272.1	40.9	161.2	202.1
12	57.4	242.6	300	44.8	177.6	222.4
13	49.2	203.8	253	38.1	150.9	189
14	55.1	231.4	286.5	42.7	170.2	212.9
15	60.8	245.9	306.7	47.6	184.7	232.3
16	45.2	188.4	233.6	34.9	138.9	173.8
17	53.6	223.5	277.1	41.6	164.8	206.4
18	58.9	244.7	303.6	45.8	179.8	225.6
19	46.7	194.6	241.3	36	143.9	179.9

20	62.3	258.4	320.7	48.9	190.5	239.4
-----------	------	-------	-------	------	-------	-------

Nota. Elaboración propia

En la Tabla 15 se muestran los valores de throughput medidos en las conexiones WAN LUMEN y TELEFÓNICA tras la aplicación del sistema basado en inteligencia artificial. Se recogieron 20 mediciones de tráfico, registrando en el enlace LUMEN valores de transmisión (TX) entre 39.5 Mbps y 62.3 Mbps, y valores de recepción (RX) entre 172.6 Mbps y 258.4 Mbps. De esta forma el throughput total del enlace estuvo en un rango entre 212.1 Mbps y 320.7 Mbps. El enlace TELEFÓNICA por su parte obtuvo valores de transmisión entre 29.8 Mbps y 48.9 Mbps, valores de recepción entre 128 Mbps y 190.5 Mbps y un throughput total entre 157.8 Mbps y 239.4 Mbps.

Tabla 16

Análisis estadístico del throughput de los enlaces WAN con inteligencia artificial

Métrica	Lumen Total (Mbps)	Telefónica Total (Mbps)
Promedio	265.935	196.59

Nota. Elaboración propia

En la Tabla 16 se muestra un resumen estadístico del throughput total obtenido en los enlaces WAN bajo el sistema basado en IA. El enlace LUMEN obtuvo un promedio de 265.935 Mbps de throughput, resultado de la suma del tráfico transmitido más el recibido durante las 20 mediciones realizadas. Por su parte, el enlace TELEFÓNICA obtuvo un throughput promedio de 196.59 Mbps.

Monitoreo de recursos del router

Tabla 17

Métricas de rendimiento y utilización de recursos del router con inteligencia artificial

Métrica	Valor
----------------	--------------

Uptime	1 hora 6 minutos
CPU	TileGX
Núcleos CPU	72
Frecuencia CPU	1000 MHz
Carga CPU	0%
Memoria Total	15.8 GiB
Memoria Libre	15.2 GiB
Almacenamiento Total	128 MiB
Almacenamiento Libre	74.6 MiB
Equipo	CCR1072-1G-8S+

Nota. Elaboración propia

En la Tabla 17 se muestran las métricas de utilización de recursos del router MikroTik CCR1072-1G-8S+, también se visualiza un tiempo de funcionamiento (uptime) de 1 hora con 6 minutos, una arquitectura TileGX de procesador con 72 núcleos funcionando a 1000 MHz, una carga de CPU del 0 %, una memoria total de 15.8 GiB con 15.2 GiB de memoria libre, así como una capacidad de almacenamiento de 128 MiB, de los cuales 74.6 MiB estaban disponibles al momento de la medición.

Prueba de tolerancia a fallos (Failover)

Tabla 18

Caída de enlaces LUMEN y TELEFÓNICA

Métrica	Caída de LUMEN	Caída de TELEFÓNICA
Enlace afectado	LUMEN	TELEFÓNICA
Enlace de respaldo	TELEFÓNICA	LUMEN

Tiempo de detección (s)	1	1
Tiempo de conmutación (s)	2	1
Tiempo total de recuperación (s)	3	2

Nota. Elaboración propia

En la prueba de failover, la caída de LUMEN registró un tiempo de detección de 1 segundo, un tiempo de conmutación de 2 segundos y un tiempo total de recuperación de 3 segundos, con una pérdida de 2 paquetes durante la transición. En la prueba de caída de TELEFÓNICA, por su parte, se registró un tiempo de detección de 1 segundo, un tiempo de conmutación de 1 segundo y un tiempo total de recuperación de 2 segundos, con la pérdida de 1 paquete.

Tabla 19

Resumen comparativo de la prueba de tolerancia a fallos

Evento	Tiempo Recuperación	Paquetes Perdidos	Estado Final
Caída LUMEN	3 s	2	Operativo
Caída TELEFÓNICA	2 s	1	Operativo

Nota. Elaboración propia

La prueba de tolerancia a fallos demostró que, cuando se cayó el enlace LUMEN, el sistema tardó 3 segundos en recuperar la conectividad, y se perdieron 2 paquetes en la transición. De su parte, en la caída del enlace TELEFÓNICA, se perdió un paquete y el tiempo de recuperación fue de 2 segundos. En ambos escenarios, el estado final de la red se mantuvo con servicio activo, lo que demuestra que el sistema es capaz de realizar la conmutación

automática del tráfico al enlace de respaldo, manteniendo la continuidad del servicio durante eventos de falla en los enlaces WAN.

Resumen Comparativo

Tabla 20

Resumen de costos operativos método tradicional vs IA

Concepto	Método Tradicional (USD/mes)	Sistema con IA (USD/mes)
Monitoreo y supervisión de enlaces WAN	220	80
Diagnóstico y análisis de incidentes	180	90
Configuración y ajuste del balanceo de carga	150	50
Atención de eventos de congestión	180	80
Costo operativo mensual estimado	730	300

Nota. Elaboración propia

Los resultados demuestran que la implantación del sistema basado en inteligencia artificial redujo los costes operativos en todas las actividades evaluadas. El costo de monitoreo y supervisión de enlaces WAN pasó de USD 220 a USD 80, el diagnóstico de incidentes de USD 180 a USD 90, la configuración del balanceo de carga de USD 150 a USD 50 y la atención de eventos de congestión de USD 180 a USD 80. Como resultado, el costo operativo mensual estimado bajó de USD 730 a USD 300, lo que resultó en un ahorro mensual de USD 430, lo que equivale a una reducción de aproximadamente el 58.9 %.

Tabla general de resultados del Método tradicional y el sistema basado en Inteligencia Artificial

Tabla 21

Tabla general de resultados del método tradicional y el sistema basado en Inteligencia Artificial

Categoría	Métrica	Tradicional	IA
Ping	Paquetes enviados (máx.)	400	400
	Paquetes recibidos (máx.)	400	400
	Pérdida de paquetes (%)	0	0
	Latencia mínima (ms)	64	64
	Latencia promedio (ms)	64	64
	Latencia máxima (ms)	71	66
Traceroute	Destino final	8.8.8.8	8.8.8.8
	Número de saltos	10	10
	Latencia primer salto (ms)	0.2	0.3
	Latencia máxima observada (ms)	65.6	64.7
	Latencia al destino (ms)	64.3	64.4
Jitter	Paquetes enviados (máx.)	100	100

Categoría	Métrica	Tradicional	IA
	Paquetes recibidos (máx.)	100	100
	Pérdida de paquetes (%)	0	0
	Latencia mínima (ms)	64	64
	Latencia promedio (ms)	64	64
	Latencia máxima (ms)	64	64
	Jitter (ms)	0	0
Throughput	LUMEN Total (Mbps)	460.185	265.935
	TELEFÓNICA Total (Mbps)	1.044	196.59
Recursos Router	Equipo	CCR1072-1G-8S+	CCR1072-1G-8S+
	Arquitectura CPU	TileGX	TileGX
	Núcleos CPU	72	72
	Frecuencia CPU	1000 MHz	1000 MHz
	Carga CPU (%)	1	0
	Memoria Total (GiB)	15.8	15.8
	Memoria Libre (GiB)	14.9	15.2
	Almacenamiento Total (MiB)	128	128

Categoría	Métrica	Tradicional	IA
	Almacenamiento Libre (MiB)	76.9	74.6

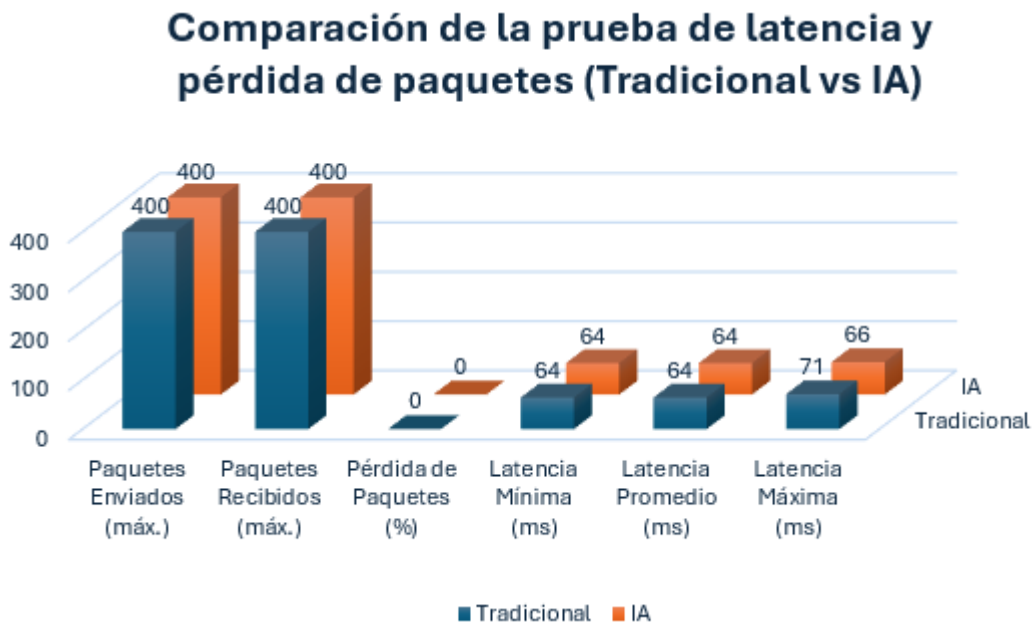
Nota. Elaboración propia

Discusión de resultados

Comparación del Método tradicional vs Inteligencia artificial

Figura 32

Comparación de la prueba de latencia y pérdida de paquetes mediante ping(Tradicional vs IA)



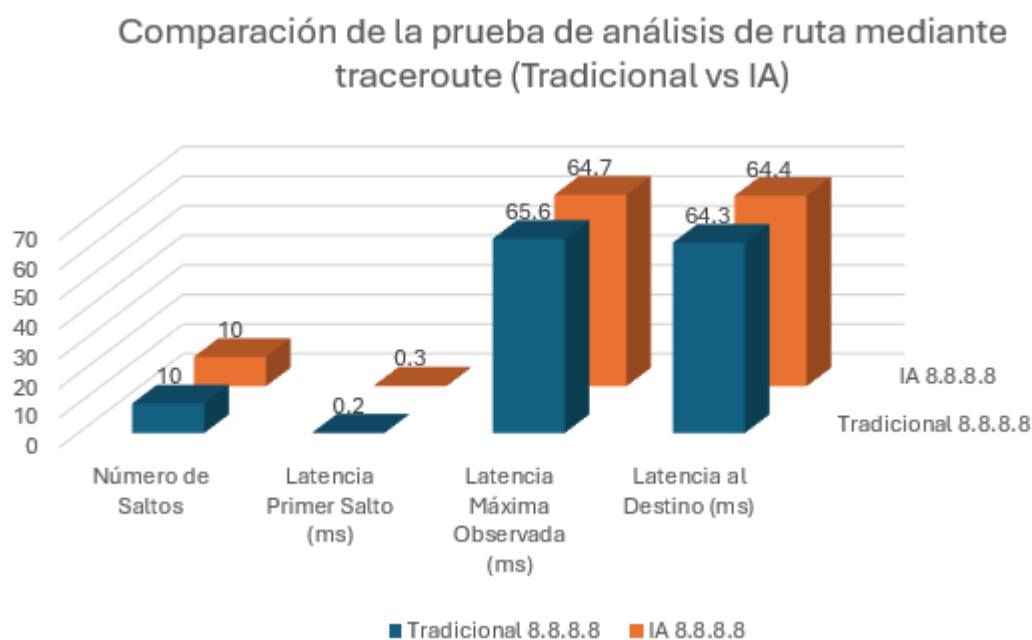
Nota. Elaboración propia

En la Figura 32, los resultados demuestran que tanto el método convencional como el sistema basado en IA llegaron a enviar y recibir un máximo de 400 paquetes, manteniendo una pérdida de paquetes del 0% en todas las mediciones realizadas. De igual manera, para ambos escenarios la latencia mínima y la latencia promedio se mantuvieron constantes en 64 ms, lo que muestra un comportamiento similar en los tiempos promedio de respuesta de la red. No obstante, la latencia máxima disminuyó de 71 ms en el método tradicional a 66 ms con la

implementación del sistema basado en inteligencia artificial, lo que representa una reducción del 7,04 %. Estos resultados muestran que la solución propuesta mantiene las condiciones de conectividad y transmisión de paquetes, a la vez que reduce los valores máximos de retardo observados durante las pruebas realizadas.

Figura 33

Comparación de la prueba de análisis de ruta mediante traceroute (Tradicional vs IA)



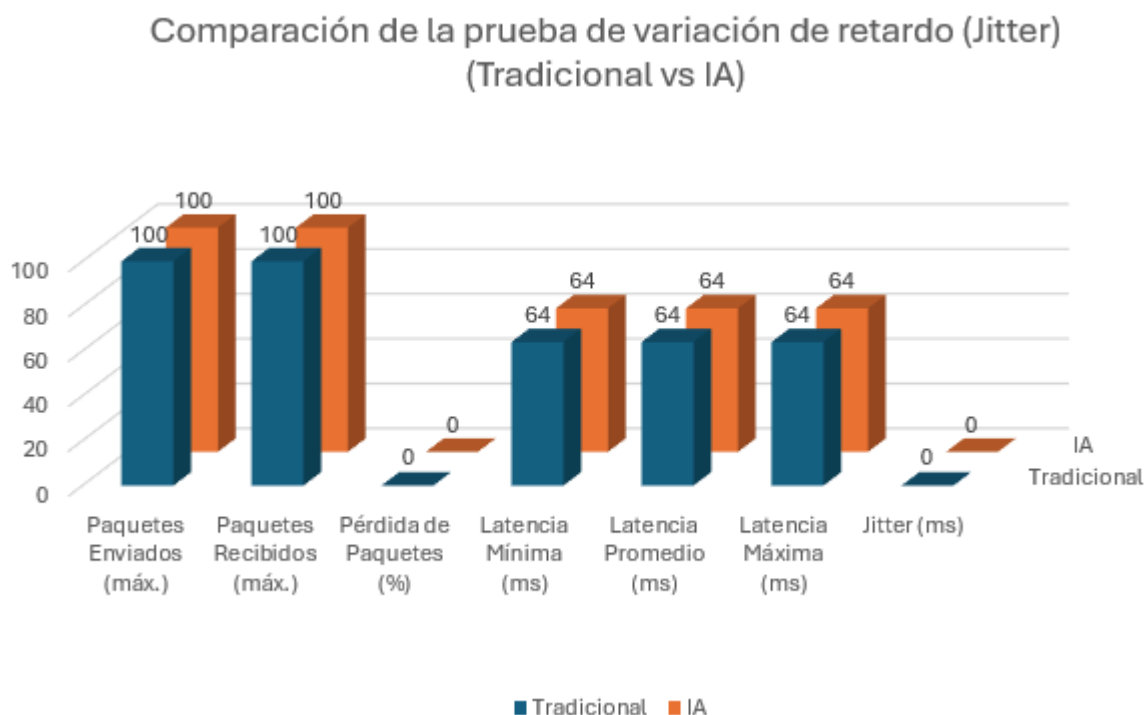
Nota. Elaboración propia

En la Figura 33, se visualizan los resultados que indican que el método tradicional como el sistema basado en IA alcanzaron el destino 8.8.8.8 en un total de 10 saltos, manteniendo la misma longitud de ruta durante la comunicación. La latencia en el primer salto se registró en 0.2 ms para el método tradicional y de 0.3 ms para el sistema basado en IA, lo que muestra una diferencia de 0.1 ms, la latencia máxima observada a lo largo del viaje disminuyó de 65.6 ms a 64.7 ms, lo que representa una reducción aproximada del 1.37 %, la latencia medida en el destino final registró valores de 64.3 ms y 64.4 ms para el método tradicional y el sistema basado en IA. Estos resultados demuestran que la implementación del sistema basado en

inteligencia artificial mantuvo un patrón de comunicación equivalente al observado en el modelo tradicional, con variaciones mínimas en los tiempos de respuesta registrados durante la prueba.

Figura 34

Comparación de la prueba de variación de retardo(Jitter) (Tradicional vs IA)



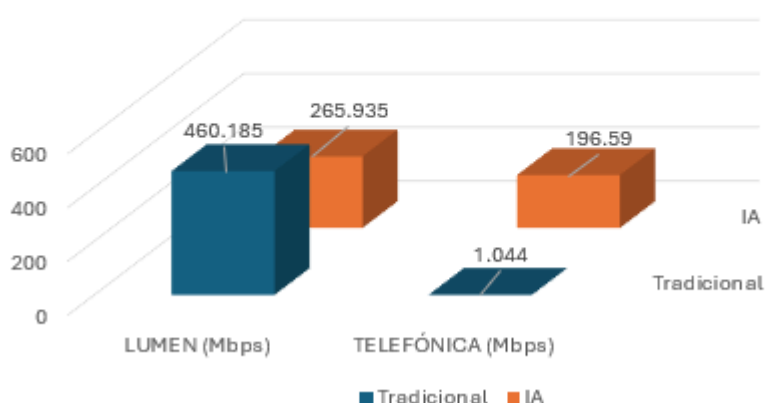
Nota. Elaboración propia

En la Figura 34, se visualizan los resultados demostrando que tanto el método tradicional como el sistema basado en inteligencia artificial registraron un máximo de 100 paquetes enviados y 100 paquetes recibidos, manteniendo una pérdida de paquetes del 0% durante todas las mediciones realizadas. También la latencia mínima, promedio y máxima se mantuvieron constantes en 64 ms en ambos escenarios evaluados. Estos valores dieron como resultado que el jitter calculado fuera de 0 ms para el método tradicional y para el sistema basado en inteligencia artificial. Estos resultados demuestran que en ambos casos se registraron condiciones de transmisión equivalentes durante la prueba, manteniéndose los mismos valores de latencia y variación de retardo durante las mediciones realizadas.

Figura 35

Comparación estadística del throughput de los enlaces WAN (Tradicional vs IA)

Comparación del throughput promedio total de los enlaces WAN (Tradicional vs IA)

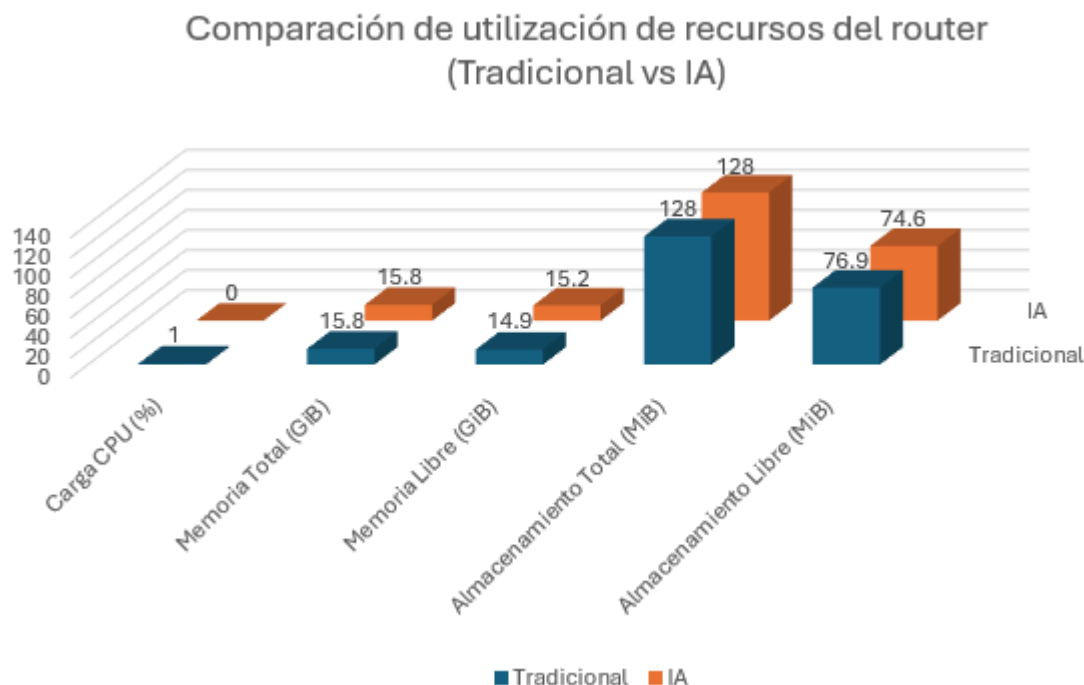


Nota. Elaboración propia

En la Figura 35, se observa que, en el método tradicional, el enlace LUMEN concentraba la mayor parte del tráfico de la red con un throughput promedio de 460.185 Mbps, mientras que TELEFÓNICA apenas llegaba a 1.044 Mbps, lo que reflejaba una utilización mínima del enlace secundario. Con la implementación del sistema basado en IA, el throughput promedio de LUMEN disminuyó a 265.935 Mbps, lo que representa una reducción del aproximadamente 42.21 %, mientras que el enlace TELEFÓNICA aumentó su throughput promedio a 196.59 Mbps, mostrando un mejor aprovechamiento de su capacidad disponible, los resultados indican una distribución más equitativa del tráfico entre los dos proveedores WAN, disminuyendo la dependencia de un único enlace y promoviendo una utilización más eficiente de los recursos de conectividad disponibles en el ISP.

Figura 36

Comparación de utilización de recursos del router (Tradicional vs IA)



Nota. Elaboración propia

En la Figura 36, se muestran los resultados donde se demuestran que las características de hardware del router se mantuvieron constantes en ambos escenarios, empleando un equipo MikroTik CCR1072-1G-8S+ con arquitectura TileGX, 72 núcleos de procesamiento y frecuencia de operación de 1000 MHz, la carga de CPU disminuyó de 1 % en el método tradicional a 0 % en el sistema basado en inteligencia artificial, la memoria total permaneció en 15.8 GiB en ambos casos, mientras que la memoria libre aumentó de 14.9 GiB a 15.2 GiB, la capacidad total permaneció en 128 MiB, con 76.9 MiB libres en el método tradicional y 74.6 MiB libres en el sistema basado en IA. Estos valores reflejan el comportamiento de los recursos del router durante la ejecución de los dos esquemas de balanceo de carga.

Capítulo V

Conclusiones

Se estudió la arquitectura de red del ISP y se encontró una alta concentración del tráfico en el enlace LUMEN, con un throughput promedio total de 460.185 Mbps, mientras que

TELEFÓNICA apenas alcanzó 1.044 Mbps. Una vez implementado el sistema basado en aprendizaje por refuerzo, el tráfico se redistribuyó entre ambos proveedores, lo que hizo que el throughput promedio de LUMEN bajara a 265.935 Mbps y el de TELEFÓNICA subiera a 196.59 Mbps. Estos resultados demuestran un uso más equilibrado de los enlaces WAN y una menor dependencia del enlace principal.

Se desarrolló un modelo de aprendizaje por refuerzo basado en Q-Learning que evalúa de manera continua el estado de la red mediante métricas de latencia, pérdida de paquetes, throughput y utilización de enlaces WAN. El agente implementado permitió seleccionar políticas de balanceo, como ECMP, `prefer_isp1`, `prefer_isp2`, `only_isp1` y `only_isp2`, mediante un mecanismo de recompensas que favoreció decisiones orientadas a reducir el desbalance del tráfico y mejorar el aprovechamiento de los enlaces disponibles.

Se implementó con éxito el sistema sobre la infraestructura MikroTik CCR1072-1G-8S+, integrando mecanismos de monitoreo, procesamiento de métricas y aplicación automática de políticas de balanceo. Los resultados obtenidos mostraron que la solución funcionó sin generar una sobrecarga significativa sobre la máquina, registrando una carga de CPU del 0 % durante la ejecución del sistema inteligente, mientras que la memoria libre alcanzó 15.2 GiB de los 15.8 GiB disponibles. Esto demostró que la implementación era viable en un entorno ISP real.

La evaluación comparativa demostró que el sistema basado en inteligencia artificial mantuvo la calidad de servicio de la red, registrando una latencia promedio de 64 ms y una pérdida de paquetes del 0 % en ambos escenarios, la latencia máxima se redujo de 71 ms a 66 ms, lo que equivale a una mejora del 7.04 % en balanceo de carga, el throughput promedio total de LUMEN disminuyó de 460.185 Mbps a 265.935 Mbps, mientras que TELEFÓNICA aumentó su throughput promedio total de 1.044 Mbps a 196.59 Mbps, evidenciando una distribución más equilibrada del tráfico entre los enlaces WAN, las pruebas de tolerancia a fallos permitieron recuperar la conectividad en 3 segundos para LUMEN y 2 segundos para TELEFÓNICA, manteniendo en ambos casos el estado operativo de la red.

La documentación y el análisis de los resultados permitieron constatar que la integración de técnicas de inteligencia artificial en redes ISP es una alternativa viable para optimizar la gestión del tráfico, automatizar actividades de monitoreo y reducir la intervención manual del personal técnico. Por otra parte, la estimación de costos operativos estimó una reducción aproximada del 57 %, de USD 500 mensuales bajo un esquema tradicional a USD 215 mensuales con la implementación del sistema inteligente, lo cual evidencia beneficios tanto técnicos como operativos para la administración de la infraestructura de red.

Recomendaciones

Implementar soluciones con IA para agilizar operaciones operativas del balanceo de carga puesto que se demostró que permiten optimizar la distribución del tráfico, mejorar la calidad de servicio y aumentar la disponibilidad de la red.

Emplear técnicas de inteligencia artificial como ayuda en la administración de redes ISP ya que facilitan la automatización de tareas operativas, reducen la necesidad de configuraciones manuales y aportan a una gestión más eficiente de los enlaces WAN.

Introducir tecnologías inteligentes en las infraestructuras de telecomunicaciones ya que permiten reducir costes operativos, optimizar el uso de los recursos tecnológicos y mejorar los tiempos de respuesta ante fallos o cambios en las condiciones del tráfico de red.

Referencias

Amézquita, G. N., Islas, J. C. G., Segundo, F. E., Gress, K. F. M., & Vázquez, O. S. (2025). Monitoreo Inteligente de Enrutadores Mikrotik® en ISPs: Detección de Anomalías para la Gestión de Redes. Boletín Científico INVESTIGIUM de la Escuela Superior de Tizayuca, 11, 295-303.

Lima Pozo, E. S. (2025). Implementación de una red de distribución de internet con un router MikroTik para balanceo de carga, segmentación por VLANs y gestión de ancho de banda.

Muñoz López, J. O. (2018). Implementación de balanceo de carga de internet con Mikrotik en la dirección de Red de Salud Conchucos Sur-Huari; 2017.

PIBAQUE VILLACRESES, M. J. (2019). RED DE DATOS CON QOS Y BALANCEO DE CARGA MEDIANTE LA TECNOLOGÍA MIKROTIK QUE COMUNIQUE LAS DEPENDENCIAS FUERA DELGAD DEL CANTÓN JIPIJAPA (Bachelor's thesis, JIPIJAPA-UNESUM).

Chancusig Maisincho, O. E., & Martínez Chiliquinga, L. S. (2012). Diseño e implementación de un sistema de control y balanceo de carga, en routers Mikrotik con calidad de servicio (Qos), para la red Wlan (Wireless local area Network) de los sistemas de comunicación Latacunga.

Chonillo Bermúdez, J. W. (2022). Diseño e implementación estructural de un sistema basado en balanceo de cargas para la optimización de tráfico de datos aplicando Failover con protocolo TCP/MSS en el laboratorio de Telecomunicaciones.

Miraval, J. D. L., Landazábal, S. A. C., & Arboleda, L. C. T. Nodo de balanceo de carga enfocado a redes de proveedores de internet.

Salguero Reinoso, D. O. (2015). Administración y distribución efectiva del internet a través de Mikrotik (Bachelor's thesis, Quito: Universidad Israel, 2015).

Ardianto, F., Alfaresi, B., & Darmadi, A. (2018). Rancang Bangun Load Balancing Dua Internet Service Provider (ISP) Berbasis Mikrotik. Jurnal Surya Energy, 3(1), 198-202.

Romero, G., & Romero, A. (2014). Análisis del balanceo óptimo de cargas en redes IP. Revista INGENIERÍA UC, 21(1), 23-35.

Espinoza Iñiguez, M. A., & Álvarez, M. A. (2025). Diseño e implementación de un plan de transición de IPv4 a IPv6 en una red implementada por equipos Mikrotik (Doctoral dissertation, ESPOL. FIEC).

Ortega Ureta, J. E. (2019). Diseño e Implementación de un sistema de control y balanceo de carga, en routers Mikrotik para mejorar la calidad de servicio (QoS) de la empresa zona vip, ubicada en el distrito de amarilis, provincia de Huánuco 2015.

Iparraguirre Moreno, T. A. (2019). Implementación de una red de datos y balanceo de carga en la Unidad de Gestión Educativa local Ugel–Cajabamba; 2019.

Calderón Rondón, J. Optimización de redes inalámbricas mediante migración OSPF con tecnología Mikrotik y Mimosa.

Rueda Hormaza, E. K., & Andrade Mora, G. R. (2017). Diseño de un sistema de alta disponibilidad para acceso a internet orientado a PYMEs (Bachelor's thesis, ESPOL. FIEC).

Suqui Carchipulla, K. M. (2010). Estudio e implementación de un radio enlace con tecnología Mikrotik para el ISP JJSistemas en el Cantón Gualaquiza, provincia Morona Santiago (Bachelor's thesis).

Rusnak, Y. (2012). Puesta en marcha de un WISP (Doctoral dissertation).

Tibaduiza Rincón, J. A. (2015). Estudio del tráfico de red por medio de un análisis estadístico de los paquetes de datos que viajan a través de los diferentes nodos y hacia cada uno de los usuarios que posee la empresa" FSD SAS".

Vílchez Espinoza, F. A., Gómez Puerto, M. R., & González Sequeira, Y. P. (2020). Desarrollo de prácticas de laboratorio utilizando tecnología Mikro Tik para el apoyo de la docencia de la asignatura redes de Computadores de la carrera Ingeniería en Telemática (Doctoral dissertation).

BAUTISTA, P. A. B., & Andrés, P. (2016). Diseño y simulación de una topología y gestión de red basadas en túneles GRE y enrutamiento dinámico OSPF Y EIGRP, caso de estudio–grupo automotriz Eljuri (Doctoral dissertation, PONTIFICIA UNIVERSIDAD CATÓLICA DEL ECUADOR).

Teixeira, A., & TRAINER, M. (2019). Balanceo de carga.

BOLAÑO, D. J. D. L., PAEZ, R. J. D., & LOPEZ, I. G. PRÁCTICAS DE CALIDAD DE SERVICIO “QoS” EN REDES IP.

León Bolaño, D. J. D., & Daza Páez, R. J. (2008). Prácticas de calidad de servicio QoS en redes IP.

Cepeda Gutiérrez, G. A. Diseño de un prototipo simulado de arquitectura de red de alta disponibilidad basada en redundancia WAN y tecnologías SDN/SD-WAN para empresas de telecomunicaciones en Yopal, Casanare.

Quiroz Pérez, G. A. (2025). Análisis del balanceo de carga en una red 4G LTE e implementación de un algoritmo de handover horizontal basado en aprendizaje por refuerzo para mejorar la calidad de servicio (QoS) (Bachelor's thesis, Riobamba).

Cidoncha Molina, P. (2025). Optimización de IRS mediante Aprendizaje por Refuerzo para Redes 6G.

Chonillo Bermúdez, J. W. (2022). Diseño e implementación estructural de un sistema basado en balanceo de cargas para la optimización de tráfico de datos aplicando Failover con protocolo TCP/MSS en el laboratorio de Telecomunicaciones.

PIBAQUE VILLACRESES, M. J. (2019). RED DE DATOS CON QOS Y BALANCEO DE CARGA MEDIANTE LA TECNOLOGÍA MIKROTIK QUE COMUNIQUE LAS DEPENDENCIAS FUERA DELGAD DEL CANTÓN JIPIJAPA (Bachelor's thesis, JIPIJAPA-UNESUM).

Muñoz López, J. O. (2018). Implementación de balanceo de carga de internet con Mikrotik en la dirección de Red de Salud Conchucos Sur-Huari; 2017.

Santiago, G., Campaña, R., & Abad Robalino, A. (2021). Descongestión de tráfico en los canales de tránsito a Internet de un ISP mediante la aplicación de algoritmos de aprendizaje por refuerzo (Doctoral dissertation).

Valle Montes, R. (2023). Implementación de un proveedor de servicios de internet inalámbrico (WISP) de banda ancha utilizando tecnologías AC Y ROUTERBOARD MIKROTIK en el Distrito de Ticlacayán.

VALENCIA, O. M. G., VALENCIA, O. M. G., & DECLARACIÓN, D. A. Y. R. (2016). ESTUDIO TÉCNICO DE FACTIBILIDAD DE UN PROVEEDOR DE.

Garibello Gonzalez, J. G., & Cadena Pichimata, P. M. Desarrollo de una herramienta software para configurar protocolos de red en por lo menos dos proveedores de router.

Uve, G. E. C., Guamán, C. A. V., Mera, A. O. A., & Bonilla, C. G. R. (2025). Semilleros de investigación: Una praxis para el desarrollo de las ciencias tecnológicas: Research incubators: A praxis for the development of technological sciences. *Boletín Científico Ideas y Voces*, 5(L2), ág-1.

Vinelli Orozco, C. E. (2015). Reingeniería de la red de área local de la Unidad Educativa Salesiana Cardenal Spellman (Bachelor's thesis, Quito, 2015.).

Lara Proaño, F. G., & Soria Pachala, M. A. (2013). Rediseño para la ampliación de la red de fibra óptica GEPON para la empresa portadora de servicios de telecomunicaciones PUNTONET SA para Tumbaco y Valle de Los Chillos (Bachelor's thesis, QUITO/EPN/2013).

de Ciencia, P. D. P. I. (2011). Tecnologías de la Información y las Comunicaciones para zonas rurales: Aplicación a la atención de salud en países en desarrollo. Á. R. Gallón, P. J. L. González, & A. M. Fernández (Eds.). CYTED.

Ortega Ureta, J. E. (2019). Diseño e Implementación de un sistema de control y balanceo de carga, en routers Mikrotik para mejorar la calidad de servicio (QoS) de la empresa zona vip, ubicada en el distrito de amarilis, provincia de Huánuco 2015.

Santiago, G., Campaña, R., & Abad Robalino, A. (2021). Descongestión de tráfico en los canales de tránsito a Internet de un ISP mediante la aplicación de algoritmos de aprendizaje por refuerzo (Doctoral dissertation).

Miraval, J. D. L., Landazábal, S. A. C., & Arboleda, L. C. T. Nodo de balanceo de carga enfocado a redes de proveedores de internet.

Llvisaca Mejia, M. D. (2025). Aplicación de aprendizaje por refuerzo en la planificación de la expansión de los sistemas de transmisión.

Lucena Alonso, E. (2025). Aprendizaje por refuerzo del balanceo de un robot de dos ruedas con microcontrolador de bajas prestaciones.

Robelly, C., Sebastian, A., Cedeño Vera, J. J., & Salazar Chacón, G. D. Optimización del rendimiento de servicios empresariales mediante balanceo de carga inteligente con redes definidas por software (SDN).

Rubio, G. M., Baeza, J. P., & Baeza, Á. B. (2024). Aprendizaje por refuerzo para robots en el espacio.

Ortiz Rojas, J. G. (2026). Desarrollo de un sistema adaptativo para el balanceo de tráfico en centros de datos mediante controladores SDN y metodologías KDN.

Payá González, A. (2025). Diseño de estrategias de refuerzo para sistemas basados en Perímetros Definidos por Software mediante mecanismos de seguridad inteligentes (Doctoral dissertation).

Fenoy Illacer, D. R. (2022). Propuesta de intervención basada en aprender haciendo para seguridad y alta disponibilidad.

Hidalgo Felipe, E. (2025). Aplicaciones de la inteligencia artificial en el diseño de redes.

Getino Petit, M. (2025). Comparativa de Técnicas de Aprendizaje Automático para la Estimación de Métricas de Calidad de Experiencia (QoE) a partir de métricas de Calidad de Servicio (QoS).

Vásquez Barrientos, W. A. (2024). Asignación automática de rutas en redes de subestaciones eléctricas virtualizadas.

Vercher Gómez, E. (2025). Detección de vulnerabilidades y comportamientos anómalos en API RESTs mediante algoritmos de Machine Learning.

Villafranca Albaladejo, A. (2025). Contribución en el estudio y desarrollo de sistemas de detección de intrusos y amenazas en IoT.

Gilces, C. E. M., & Villamar, R. P. (2019). Aplicación de balanceo de carga dinámico para servidores, basada en redes definidas por software. *Revista Ibérica de Sistemas e Tecnologías de Informação*, (32), 67-82.

Miraval, J. D. L., Landazábal, S. A. C., & Arboleda, L. C. T. Nodo de balanceo de carga enfocado a redes de proveedores de internet.

Pérez, S. C., Facchini, H. A., Dantiacq, A., Roberti, B., Césari, M., Vernier, G., & Silnik, A. (2025). Experimentación de automatización de red utilizando tecnologías de programación y equipamiento de red MikroTik. In XXVII Workshop de Investigadores en Ciencias de la Computación (Mendoza, 10 y 11 de abril de 2025).

Carcelén Tomalá, M. Á. (2026). Diseño de un entorno controlado para el análisis y monitoreo de tráfico de red en el laboratorio de redes de FACSISTEL.

Ricardo Villao, G. E. (2026). Implementación de sensores IDS para el monitoreo y la seguridad en el laboratorio de Telecomunicaciones.

Nivicela Arbito, J. F., & Vera Peláez, J. A. (2021). Diseño e implementación de un entorno de análisis, experimentación, evaluación y desarrollo de Redes Definidas por Software (SDN).

Sellán Fajardo, P. D., Urquiza Olivo, J. J., & Soto, V. (2022). Automatización de procesos de asignación de recursos en una red de proveedor de servicios de internet mediante simulación.

Ng Pante, C. K. (2025). Diseño de infraestructura de red para una empresa multisede.

Miraval, J. D. L., Landazábal, S. A. C., & Arboleda, L. C. T. Nodo de balanceo de carga enfocado a redes de proveedores de internet.

Baza Alonso, S. (2021). Gestión de dispositivos IoT con acceso a la red mediante 802.11 ax y SDN en el edge para orquestar los flujos de datos.

Castrillón Ospina, S. (2016). Modelado y validación formal de una topología SDN/OPENFLOW.

Navarrete Andrade, L. S., & Rivera Cazco, E. P. (2021). Implementación y evaluación de segment routing en el transporte de paquetes en tiempo real mediante técnicas de ingeniería de tráfico con IPV4.

More Zeña, J. J. A., & Vitón Llatas, Y. M. (2019). Implementación de un sistema de proximidad sobre plataforma wifi para mejorar el proceso de ventas en el Centro Comercial Punto Norte-Chiclayo 2016.

Rivero Luna, B. D., & García, J. C. (2020). Implementación de un sistema de balanceo de cargas en la red perimetral de una entidad financiera para ambientes de producción y de Testing (Doctoral dissertation, ESPOL. FIEC.).

Chonillo Bermúdez, J. W. (2022). Diseño e implementación estructural de un sistema basado en balanceo de cargas para la optimización de tráfico de datos aplicando Failover con protocolo TCP/MSS en el laboratorio de Telecomunicaciones.

Navarro, J. A. C. Arquitectura de balanceo de carga dinámico para la lógica de conmutación EtherChannel.

Robelly, C., Sebastian, A., Cedeño Vera, J. J., & Salazar Chacón, G. D. Optimización del rendimiento de servicios empresariales mediante balanceo de carga inteligente con redes definidas por software (SDN).

Dorado Ruiz, H. J., & Huertas Figueroa, P. L. (2008). Operación continua en las redes (Load Balancing: concepto, gestión y monitoreo).

Calderón, O. J., & Serra, X. H. (2004). Balanceo de carga en redes MPLS. Revista GTI, 3(5).

Bustos Burbano, A. G. (2007). Configuración de un cluster de alta disponibilidad y balanceo de carga en linux para satisfacer gran demanda web y servicios de resolución de nombres (Bachelor's thesis, QUITO/EPN/2007).

Lima Pozo, E. S. (2025). Implementación de una red de distribución de internet con un router MikroTik para balanceo de carga, segmentación por VLANs y gestión de ancho de banda.

Choque Castro, M. D., & Jallo Colquehuanca, T. P. Balanceo de carga y calidad de servicio sobre dispositivos MikroTik para el acceso a internet en la Municipalidad Provincial de Puno-2019.

Huane Solorzano, J. C. Propuesta de implementación de balanceo de carga con tecnología " Mikrotik" para la oficina de informática de la dirección regional de Transportes y Comunicaciones región Ancash-Huaraz; 2020.

Chonillo Bermúdez, J. W. (2022). Diseño e implementación estructural de un sistema basado en balanceo de cargas para la optimización de tráfico de datos aplicando Failover con protocolo TCP/MSS en el laboratorio de Telecomunicaciones.

Uve, G. E. C., Guamán, C. A. V., Mera, A. O. A., & Bonilla, C. G. R. (2025). Semilleros de investigación: Una praxis para el desarrollo de las ciencias tecnológicas: Research incubators: A praxis for the development of technological sciences. Boletín Científico Ideas y Voces, 5(L2), ág-1.

Vílchez Espinoza, F. A., Gómez Puerto, M. R., & González Sequeira, Y. P. (2020). Desarrollo de prácticas de laboratorio utilizando tecnología Mikro Tik para el apoyo de la docencia de la asignatura redes de Computadores de la carrera Ingeniería en Telemática (Doctoral dissertation).

Suqui Carchipulla, K. M. (2010). Estudio e implementación de un radio enlace con tecnología Mikrotik para el ISP JJSistemas en el Cantón Gualaquiza, provincia Morona Santiago (Bachelor's thesis).

Villar Zamora, L. J. D. D. (2014). Clúster de servidores Linux para alta disponibilidad de la informacion.