Sistema Multiobjetivo para la Búsqueda de Aparcamiento Basado en Aprendizaje Profundo

Daniel J. Vargas[1] and Gabriel Luque[1][0000−0001−7909−1416]

ITIS Software, Universidad de Málaga, Malaga, España
danijvargasq@uma.es - gluque@uma.es

Resumen Este artículo aborda el desafío la movilidad urbana y la búsqueda de estacionamiento mediante un sistema inteligente de enrutamiento que proporciona rutas estocásticas para equilibrar el tiempo de viaje y la probabilidad de éxito al aparcar. El sistema integra datos dinámicos de sensores de ocupación municipales con la topología estática de Málaga. Primero existe fase predictiva que usa una red neuronal recurrente LSTM multivariante para pronosticar la disponibilidad futura de aparcamiento en un horizonte de hasta 60 minutos. Esta disponibilidad se propaga sobre el grafo vial mediante un modelo híbrido de Estimación de Densidad por Kernel en Red (NKDE). Posteriormente, el enrutamiento se aborda desde una perspectiva multiobjetivo, contrastando una heurística constructiva voraz y una heurística de búsqueda estocástica con la metaheurística evolutiva NSGA-II. La experimentación revela que, mientras el enfoque voraz es suficiente en escenarios de baja demanda, NSGA-II demuestra una superioridad integral durante las horas punta al generar un Frente de Pareto de mayor calidad, con una eficiencia computacional y estabilidad superiores a la estrategia estocástica.

Keywords: Ciudades Inteligentes · Optimización · Enrutamiento Estocástico · Multiobjetivo · NSGA-II · Aprendizaje Profundo · LSTM.

1. Introducción

La creciente congestión del tráfico en entornos urbanos y la búsqueda de aparcamiento generan un consumo innecesario de combustible, un aumento de las emisiones contaminantes y empeoran la experiencia del conductor [12]. En ciudades con alta densidad vehicular, la disponibilidad de fuentes de datos abiertos, como la API del Ayuntamiento de Málaga [1], y geográficos, como la plataforma OpenStreetMap (OSM) [8], abre la puerta al desarrollo de soluciones de enrutamiento estocástico que incrementen la probabilidad de encontrar una plaza libre en aparcamientos de superficie [5].

En la literatura científica, la planificación de rutas se ha fundamentado tradicionalmente en soluciones deterministas clásicas como el algoritmo de Dijkstra [4]. Sin embargo, debido a la alta variabilidad del tráfico, la investigación ha tendido a evolucionar hacia variantes dependientes del tiempo [14] y marcos

[1] Disponible en: https://datosabiertos.malaga.eu/api

de enrutamiento dinámico bajo condiciones estocásticas [10]. Paralelamente, la predicción de la disponibilidad de estacionamiento se ha desplazado desde los enfoques estadísticos lineales paramétricos como ARIMA [7] hacia arquitecturas de aprendizaje profundo basadas en Redes Neuronales Recurrentes, concretamente celdas LSTM (Long Short-Term Memory) [7], debido a su capacidad para procesar tensores multidimensionales y asimilar efectos de desbordamiento en redes complejas [16]. Por último, dado que el conductor se enfrenta al dilema de minimizar el tiempo de trayecto y maximizar la probabilidad de éxito, el problema se ha encuadrado bajo el marco de la Optimización Multiobjetivo, donde metaheurísticas evolutivas como el algoritmo NSGA-II (Non-dominated Sorting Genetic Algorithm II) [3] se postulan habitualmente como un estándar de resolución de referencia en el dominio del transporte urbano [6].

A pesar de la madurez de estas tres líneas de investigación por separado, la literatura actual exhibe una brecha en su uso conjunto. La mayoría de las soluciones de guiado existentes operan bajo un paradigma reactivo en tiempo real [5], evaluando el estado actual de la red sin anticipar la degradación de la ocupación durante el transcurso del viaje. Asimismo, las aplicaciones de NSGA-II en redes de transporte reales tienden a focalizarse en la planificación estática de infraestructuras o en diseños agregados a nivel macroscópico [2], omitiendo la integración de predicciones probabilísticas futuras dentro de las funciones de aptitud de la navegación en grafos viarios continuos.

En este contexto, las contribuciones principales de este trabajo se estructuran en tres frentes interconectados: 1) Un módulo ETL distribuido capaz de realizar el procesamiento integral de los datos dinámicos municipales de alta frecuencia y de los metadatos topológicos. 2) El diseño de una arquitectura Encoder-MLP con bloque recurrente LSTM multivariante que emplea una estrategia *Direct Multi-Step* para inferir simultáneamente la ocupación futura de todos los aparcamientos municipales en un horizonte de hasta 60 minutos, mitigando la acumulación del error de los enfoques recursivos tradicionales. 3) Una formulación algorítmica multiobjetivo adaptada al dominio espacial que despliega operadores genéticos específicos de orden y mutación híbrida en NSGA-II para balancear el compromiso de Pareto entre tiempo de viaje y probabilidad de éxito al aparcar, encapsulando todo el *pipeline* analítico en una aplicación web interactiva.

El resto del artículo se organiza de la siguiente manera: la Sección 2 describe el problema y la formulación espacial; la Sección 3 detalla la arquitectura propuesta y sus módulos funcionales; las Secciones 4 y 5 presentan el diseño experimental y el análisis de los módulos predictivo y de optimización, respectivamente; finalizando en la Sección 6 con las conclusiones.

2. Descripción del problema

El problema concreto a resolver en este trabajo se formula como un problema de enrutamiento estocástico multiobjetivo sobre dos grafos dirigidos independientes: la red vial $G_d = (V_d, E_d)$ y la red peatonal $G_w = (V_w, E_w)$. A cada arista de la red vial $e \in E_d$ se le asocian tres atributos: el tiempo de viaje en

coche $T_{coche}\,(e)$, el tiempo de viaje caminando al destino $T_{pie}\,(e)$ y la probabilidad de encontrar plaza de aparcamiento libre $p\,(e)$. El sistema debe determinar para el instante de viaje un conjunto de rutas que minimicen simultáneamente el tiempo total de trayecto, que incluye el tiempo de conducción y de caminata, y maximicen la probabilidad conjunta de encontrar una plaza libre de aparcamiento en superficie. Un problema al que nos enfrentamos es la falta de sensorización del aparcamiento en superficie. Para solucionarlo, se utiliza un modelo probabilístico híbrido que propaga la información de ocupación de los aparcamientos municipales por la ciudad combinando estimaciones dinámicas y estáticas.

Componente dinámica: Se asume que los aparcamientos municipales actúan como sensores de la "presión de demanda" de su entorno. La influencia que ejerce un aparcamiento i sobre una arista j se calcula mediante una Estimación de Densidad de Kernel en Red (NKDE) usando un kernel gaussiano local [7]:

$$K_\lambda(d_{ij}) = exp\left(-\frac{d_{ij}^2}{2\lambda^2}\right) \tag{1}$$

donde d_{ij} es la distancia caminable sobre el grafo peatonal (calculada como el camino más corto mediante Dijkstra) y λ, el ancho de banda, fijado en 500 metros, un valor adecuado según la literatura especializada [13]. Usando estos pesos espaciales y un estimador tipo Nadaraya-Watson [7] la probabilidad dinámica de encontrar sitio en la arista j se calcula como la media ponderada de la disponibilidad relativa de los aparcamientos municipales circundantes (O_i):

$$P_{dinamica}(j) = \frac{\sum_i K_\lambda(d_{ij}) \cdot O_i}{\sum_i K_\lambda(d_{ij})} \tag{2}$$

Componente estática: Para las calles alejadas del radio de influencia de los aparcamientos ($K_\lambda\,(d_{ij})$ tenderá a cero), el sistema mantiene una probabilidad base (P_{base}) basada en la tipología de la vía. Basándonos en estudios previos [15], se asigna una probabilidad menor a las zonas residenciales por su ocupación crónica y una mayor a las vías secundarias por su mayor rotación comercial.

Modelo probabilístico híbrido: Ambas probabilidades se integran usando un factor de confianza (α_j) que mide la "cobertura" de los sensores en esa calle:

$$\alpha_j = min\left(1, \sum_i K_\lambda(d_{ij})\right) \tag{3}$$

La ecuación de probabilidad de estacionamiento final en una arista es:

$$P_{final} = \alpha_j \cdot P_{dinamica} + (1 - \alpha_j) \cdot P_{base} \tag{4}$$

Esta fórmula da un papel predominante la predicción dinámica en calles cercanas a estacionamiento públicos ($\alpha_j \approx 1$), mientras que la probabilidad estructural de la calle rige el comportamiento en calles lejanas a ellos ($\alpha_j \to 0$).

3. Propuesta

En este trabajo se ha seguido la metodología CRISP-DM, adaptada a series temporales y optimización espacial multiobjetivo con cuatro módulos (Figura 1): ETL, Predictivo, Conversión Probabilística y Optimización.

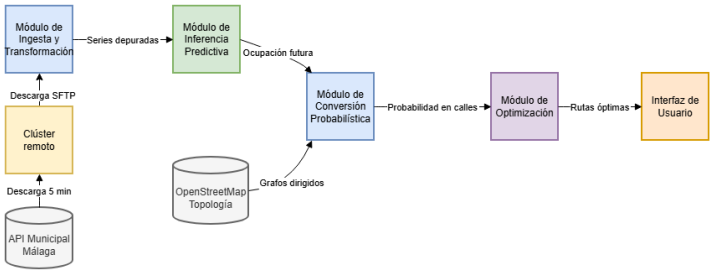


Figura 1: Arquitectura general del sistema propuesto.

3.1. Módulo ETL

La viabilidad de los módulos de predicción y de optimización depende intrínsecamente de la calidad de los datos subyacentes, lo que se resuelve mediante el diseño de un módulo ETL (*Extract, Transform, and Load*) orientado a la ingesta automatizada, saneamiento analítico y almacenamiento estructurado de la información urbana. En la fase de extracción, un orquestador remoto tolerante a fallos sondea cada 5 minutos la API de Datos Abiertos del Ayuntamiento de Málaga para capturar la microdinámica de ocupación de 10 aparcamientos municipales, mientras que de manera simultánea y única se descarga la topología urbana de OSM para generar los grafos de la Sección 2.

A continuación, la etapa de transformación está orientada a maximizar la calidad del dato. Para ello, primero se aplica una limpieza e imputación multietapa de ocupación eliminando anomalías por congelación de señal mediante umbrales de inactividad asimétricos (día/noche); los cortes breves se interpolan linealmente, las pérdidas prolongadas se reconstruyen buscando la jornada histórica homóloga que minimice la distancia euclídea entre sus franjas válidas y las observaciones circundantes al hueco, y los días enteros perdidos se sintetizan promediando las dinámicas homólogas de las semanas adyacentes. Posteriormente, la ingeniería de características aborda la ciclicidad temporal transformando trigonométricamente (seno y coseno) la hora y el día, inyectando variables lógicas para fines de semana y festividades. De manera independiente y única, la integración espacial vincula geométricamente cada recinto de OSM a su arista vial transitable más próxima, consolidando un grafo unificado. Finalmente, en la etapa de carga, la información saneada se almacena localmente en archivos GeoJSON y CSV persistentes, garantizando un acceso indexado, eficiente y reproducible para el entrenamiento predictivo y la optimización.

3.2. Módulo predictivo

El propósito de este módulo es estimar la disponibilidad conjunta de las plazas de aparcamiento municipales en el instante estimado de llegada.

El problema de predicción se formula como una regresión de series temporales multivariante bajo una estrategia *Direct Multi-Step* [7]. A diferencia de los enfoques univariantes clásicos que analizan cada ubicación de forma aislada, un modelo multivariante permite capturar dependencias cruzadas entre las distintas series temporales [9]. De este modo, el modelo puede capturar relaciones temporales conjuntas asociadas a posibles efectos de desbordamiento (*spillover effects*), en los que la saturación de un recinto se relaciona con incrementos de demanda en otros. El diseño se alinea con la familia *Encoder-MLP* (Figura 2), sustituyendo el clásico decodificador recurrente por un bloque denso global. Esta decisión reduce el coste paramétrico y agiliza la inferencia en producción.

Como núcleo del codificador se utiliza una red LSTM para mitigar el desvanecimiento del gradiente y retener dependencias secuenciales no lineales a largo plazo, superando las limitaciones estructurales de los modelos paramétricos lineales tipo ARIMA frente a las dinámicas caóticas del tráfico. Una de las principales aportaciones es el patrón de Fusión Tardía (*Late Fusion*). En lugar de forzar a la componente recurrente a inferir de forma implícita el calendario humano, esta técnica alinea dimensionalmente el contexto histórico con una proyección lineal de covariables futuras conocidas con certeza (hora, día de la semana, festivos). Esto condiciona la predicción y permite anticipar picos o descensos de demanda asociados a patrones temporales previsibles.

Finalmente, para generar el horizonte futuro de $t+1$ a $t+12$, equivalentes a 60 minutos, se descarta el enfoque recursivo clásico, en el que la predicción del instante t se realimenta como entrada para calcular $t+1$, debido a su tendencia a acumular errores. En su lugar, la estrategia *Direct Multi-Step* emite simultáneamente todo el vector de salidas para los 10 aparcamientos en una única pasada de inferencia, lo que simplifica el despliegue y reduce la latencia de predicción.

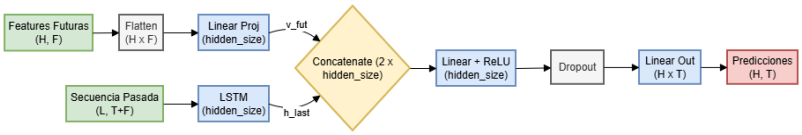


Figura 2: Arquitectura de la red neuronal.

3.3. Módulo de optimización

Para resolver el compromiso entre el tiempo total de trayecto y la probabilidad de éxito al aparcar, el espacio de búsqueda se modela dividiendo la ruta r en dos segmentos, delimitados por un "círculo de interés" alrededor del destino final cuyo radio equivale a la distancia máxima a caminar por el usuario.

El primero es el tramo exterior (r_{ext}) que comprende desde el origen hasta la intersección con el círculo de interés. Este tramo se asume determinista (no hay búsqueda de aparcamiento) y su tiempo (T_{ext}) se calcula usando la ruta más corta obtenida con el algoritmo de Dijkstra. El segundo segmento es el tramo interior (r_{int}), definido como la secuencia de calles $C = \{c_1, \ldots, c_k\}$ dentro del círculo de interés, donde opera el motor de optimización estocástico.

Para cualquier ruta candidata, se definen dos funciones objetivo a optimizar de forma simultánea. La primera es la maximización de la probabilidad de éxito para encontrar aparcamiento (P_{exito}) en la ruta interior r_{int}. Asumiendo independencia estadística entre segmentos viales, se modela como el suceso complementario de no encontrar plaza en ninguna de las calles recorridas:

$$P_{fallo}\left(r_{int}\right) = \prod_{i=1}^{k} \left(1 - p_{c_i}\right) \tag{5}$$

$$P_{exito}\left(r_{int}\right) = 1 - P_{fallo}\left(r_{int}\right) \tag{6}$$

La segunda función es la minimización del tiempo total esperado para llegar al destino ($E[T(r)]$), definido como la suma de T_{ext} y la esperanza matemática del tiempo invertido en el tramo interior ($E[T_{int}]$). Esta esperanza pondera el coste de aparcar en una calle c_i (trayecto en vehículo y a pie) por la probabilidad de haber fracasado en aparcar en las calles anteriores y tener éxito en la actual:

$$E\left[T\left(r\right)\right] = T_{ext} + E\left[T_{int}\right] \tag{7}$$

$$E\left[T_{int}\right] = \sum_{i=1}^{k} \left[\left(\prod_{j=1}^{i-1} \left(1 - p_{c_j}\right) \right) \cdot p_{c_i} \cdot \left(T_{coche}\left(c_i\right) + T_{pie}\left(c_i\right)\right) \right] \tag{8}$$

Se proponen tres estrategias de resolución. El primer enfoque es un heurístico voraz por calle (H1 - Algoritmo 1) que construye iterativamente la ruta mínima hacia cada calle con aparcamiento del interior del círculo de interés y evalúa sus métricas. Si la ruta no es dominada, se inserta en el conjunto de no dominadas y se eliminan las previas que pasen a ser dominadas por la nueva incorporación.

El segundo enfoque consiste en un heurístico de búsqueda por subconjuntos de calles (H2 - Algoritmo 2). A lo largo de un número predefinido de iteraciones N, el algoritmo selecciona estocásticamente (mediante muestreo aleatorio uniforme o ponderado por la probabilidad de aparcar) un subconjunto de K calles únicas con aparcamiento ($p(e) > 0$) pertenecientes al círculo de interés. Aplica Dijkstra secuencialmente entre cada par consecutivo para conectarlas y evalúa la ruta completa para actualizar el conjunto de soluciones no dominadas.

Con el objetivo de explorar el espacio de soluciones de forma más sofisticada, la arquitectura principal implementa un Algoritmo Genético basado en la metaheurística NSGA-II, donde cada individuo codifica una secuencia ordenada de K calles únicas con aparcamiento ($p(e) > 0$) pertenecientes al círculo de interés. Para guiar la convergencia preservando la viabilidad, se ha utilizado un operador

Algorithm 1 Heurístico 1 (H1): Evaluación por calle

Require: Conjunto de calles C dentro del círculo
Ensure: Conjunto de rutas no dominadas S
$S \leftarrow \emptyset$
for cada calle $c \in C$ **do**
  $r \leftarrow$ ruta mínima origen $\rightarrow c \rightarrow$ destino (Dijkstra)
  Calcular $P_{exito}(r)$ y $E[T(r)]$
  if r no está dominada en S **then**
    Añadir r a S
    Eliminar de S las rutas dominadas por r
  end if
end for
return S

de cruce adaptativo OX-Subset similar al *Two-Point Crossover* que mantiene el orden de visita sin duplicar calles, y un operador de mutación híbrida que alterna, al 50 %, entre alterar el orden de visita (*Swap*) y sustituir calles por nuevas opciones del mapa (*Replacement*). La supervivencia se rige por un torneo binario basado en la dominancia de Pareto y la distancia de aglomeración, garantizando diversidad y elitismo en el Frente resultante.

Algorithm 2 Heurístico 2 (H2): Búsqueda por subconjuntos de calles

Require: Conjunto de calles C, número de iteraciones N, tamaño de subconjunto K
Ensure: Conjunto de rutas no dominadas S
$S \leftarrow \emptyset$
for $i = 1$ **to** N **do**
  Seleccionar subconjunto $C' \subset C$ de tamaño K (aleatorio o ponderado por p_{ij})
  $r \leftarrow$ ruta mínima que pasa por todas las calles de C' en el orden elegido (Dijkstra)
  Calcular $P_{exito}(r)$ y $E[T(r)]$
  if r no está dominada en S **then**
    Añadir r a S
    Eliminar de S las rutas dominadas por r
  end if
end for
return S

4. Análisis del módulo predictivo

En esta sección se analizará el rendimiento del módulo predictivo, pero antes pasamos a describir el diseño experimental usado.

El conjunto de datos de ocupación de los 10 aparcamientos municipales abarca casi ocho meses de registros [2], divididos cronológicamente en particiones de

[2] Disponible en: `https://github.com/djvq02/TFM`

entrenamiento (70 %), validación (15 %) y prueba (15 %). Todas las variables objetivo se escalaron al rango $[0, 1]$ mediante una transformación *Min-Max* ajustada exclusivamente sobre el subconjunto de entrenamiento.

La optimización de la arquitectura y sus hiperparámetros se automatizó mediante el *framework* bayesiano Optuna [1]. Se ejecutaron 40 ensayos iterativos empleando un muestreador TPE (*Tree-Structured Parzen Estimator*) combinado con una estrategia de poda agresiva (*Median Pruner*). El proceso convergió de forma eficiente, localizando la parametrización óptima en el ensayo 24 con un Error Cuadrático Medio (MSE) en validación inferior a 0.0017. Los rangos explorados y la configuración óptima final resultante (incluye los valores estáticos de mecanismos de regularización como la norma de *Gradient Clipping* y la paciencia de *Early Stopping*) se detallan en la Tabla 1, mientras que la convergencia de su entrenamiento se ilustran en la curva de aprendizaje de la Figura 3.

Hiperparámetro	**Rango explorado**
seq_len	{24, 36, **48**, 60, 72}
hidden_size	{64, **128**, 256}
num_layers	{**1**, 2}
lr	$[3 \cdot 10^{-4}, 10^{-3}]$
batch_size	{**32**, 64}
dropout	{**0.0**, 0.1, 0.3}
Gradient Clipping	1.0
Paciencia ES	10 épocas

Cuadro 1: Espacio de búsqueda y configuración óptima.

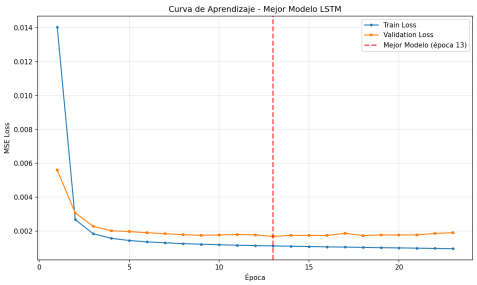


Figura 3: Curva de aprendizaje del mejor modelo entrenado.

El modelo óptimo se evaluó sobre el conjunto de prueba, arrojando un rendimiento altamente competitivo para el dominio de la movilidad urbana. Los resultados agregados demuestran la precisión del sistema, destacando un R^2 superior al 99 %. El MAE se sitúa en 6.38 plazas sobre una capacidad media de 315, mientras que el WAPE con un error del 3.03 % confirma la fiabilidad del modelo incluso en instantes de saturación (disponibilidad nula).

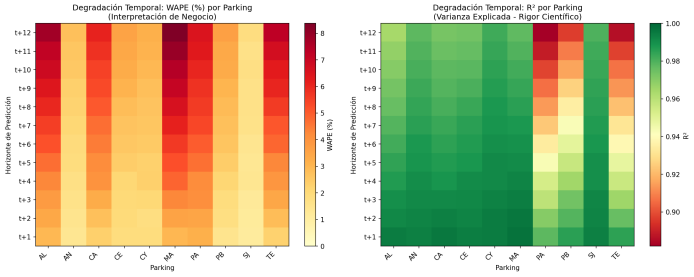


Figura 4: Mapa de calor por aparcamiento y horizonte para el WAPE y el R^2.

Al analizar el horizonte predictivo (de 5 a 60 minutos), se observa una degradación del error matemáticamente suave (Figura 4). Según las métricas agregadas por aparcamiento, para el horizonte inmediato $(t+1)$, el modelo presenta un WAPE de solo el 1.87 % y un $R^2 = 0{,}9982$. A la hora vista $(t+12)$, el error asciende de forma contenida hasta un WAPE del 4.13 %, manteniendo una capacidad explicativa superior al 0.99. Esta estabilidad garantiza que el motor de rutas trabaje con estimaciones de alta confianza técnica.

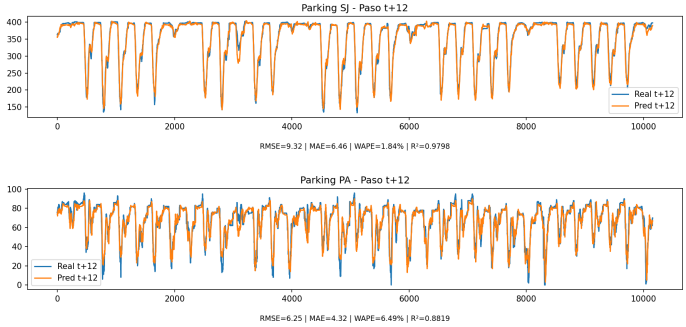


Figura 5: Gráficas de valores reales vs. predichos.

El análisis espacial revela diferencias significativas basadas en la ubicación de las infraestructuras de aparcamiento municipales. Por un lado, se aprecia un mejor desempeño para los nueve nodos situados en el centro de la ciudad y relativamente cercanos entre ellos, consolidándose el aparcamiento SJ (San Juan) como el recinto más predecible de la red. Por otro lado, el aparcamiento PA (El Palo) registra el coeficiente de determinación más bajo de la red. Su aislamiento geográfico en el extremo oriental de la ciudad limita la capacidad de la red neuronal para encontrar correlaciones cruzadas con el resto de aparcamientos, respondiendo a dinámicas residenciales más locales. En cualquiera de los casos, el modelo ofrece unos resultados robustos como se puede ver en la Figura 5.

5. Análisis del módulo de optimización

En esta sección se analiza el rendimiento del módulo de optimización tras definir su diseño experimental. Para garantizar la robustez ante la variabilidad del tráfico de Málaga, se evaluaron dos escenarios temporales opuestos: hora valle (alta disponibilidad) y hora punta (máxima congestión). Se definieron 15 instancias de estudio combinando tres orígenes (Teatinos, Parque del Oeste, El Palo) con cinco destinos céntricos (Teatro Cervantes, Vialia, El Corte Inglés, Plaza de Toros, La Catedral).

El ajuste de H2 y NSGA-II se automatizó con *irace* [11] para maximizar el Hipervolumen (HV) bajo un presupuesto fijo de evaluaciones. El espacio explorado incluyó el tamaño de subconjunto de calles ($K \in \{2, 3, 4, 5, 6, 7, 8\}$) y el método de selección (uniforme/ponderado). Para NSGA-II, basándose en la lite-

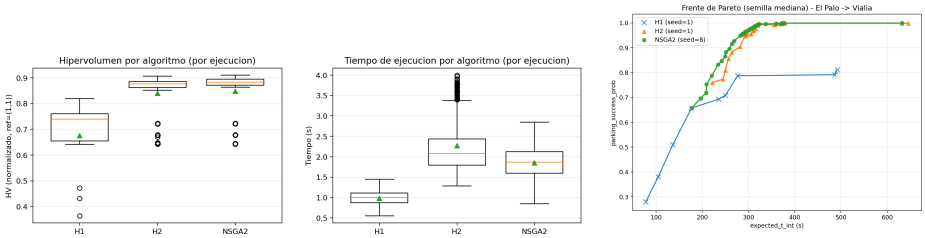
ratura [3], se añadieron la población ($\{20, 50, 80, 100, 120, 150\}$), la tasa de cruce ($\{0{,}5;\ 0{,}6;\ 0{,}7;\ 0{,}8;\ 0{,}9;\ 1{,}0\}$) y la de mutación ($\{0{,}01;\ 0{,}05;\ 0{,}10;\ 0{,}15;\ 0{,}20\}$).

La optimización reveló que la parametrización es altamente sensible a la congestión urbana. En hora punta, ante la escasez de plazas, los algoritmos requieren configuraciones más exploratorias. Tanto H2 como NSGA-II convergieron hacia un subconjunto prospectivo de $K = 3$ calles. NSGA-II demandó un gran esfuerzo exploratorio para evitar óptimos locales, fijando una población grande (80-150 individuos), tasas de cruce elevadas (0.8-0.9) y mutaciones en un rango amplio (0.05-0.2). En cambio, en hora valle, la abundancia de estacionamiento reduce la necesidad de exploración. Ambos algoritmos restringieron el horizonte a $K = 2$ calles, bastando enfoques más locales y, en el caso de NSGA-II, una población más baja (20-100 individuos), tasas de recombinación más extremas (0.8-1), y mutaciones más cercanas al límite superior del rango (0.15-0.2).

Debido al no determinismo de las técnicas, cada experimento para las configuraciones élite se ejecutó 30 veces con semillas independientes.

5.1. Análisis del horario punta

En el escenario de congestión extrema, en términos de calidad de las soluciones (Figura 6a), el enfoque determinista voraz (H1) resulta claramente insuficiente, mostrando la peor mediana de HV y estancándose en una probabilidad de éxito en el estacionamiento máxima del 85 %, por mucho que se prolongue la búsqueda (Figura 6c). Por el contrario, tanto la búsqueda aleatoria (H2) como el algoritmo evolutivo (NSGA-II) maximizan la calidad de las rutas, destacando NSGA-II con una ligera pero consistente superioridad en la franja de decisión más crítica para el conductor (entre 200 y 300 segundos), ofreciendo hasta un 10 % más de probabilidad de éxito para un mismo tiempo de viaje.



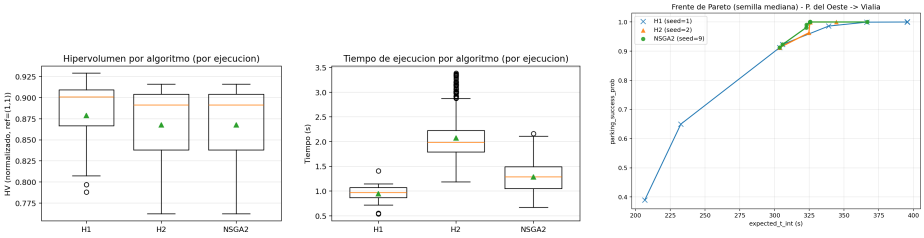
(a) Hipervolumen. (b) Tiempo de ejecución. (c) Frente de Pareto.

Figura 6: Resultados para la hora punta.

El hallazgo más destacable reside en el coste computacional (Figura 6b). Contra la intuición clásica que asume que los algoritmos evolutivos son lentos, NSGA-II resulta ser considerablemente más rápido y estable que el algoritmo estocástico H2. La inestabilidad de H2 proviene posiblemente de su selección aleatoria, que genera secuencias de calles muy lejanas entre sí, encareciendo el cálculo subyacente de la ruta con Dijkstra mientras NSGA-II orienta su búsqueda hacia rutas eficientes y cortas desde las primeras generaciones.

5.2. Análisis del horario valle

En este escenario de baja exigencia, el algoritmo determinista base (H1) resulta altamente competitivo en términos de HV (Fig. 7a), y con el menor coste computacional (Fig. 7b) al centrarse directamente en rutas cortas que, ante la abundancia de plazas, suelen ser exitosas. No obstante, al observar el Frente de Pareto resultante (Fig. 7c), se aprecian matices importantes. H1 es el único algoritmo capaz de ofrecer soluciones extremadamente rápidas (menores a 300 segundos), pero asumiendo unas probabilidades de éxito inferiores al 70 %. Por encima de ese umbral temporal, los tres algoritmos convergen rápidamente hacia garantías de aparcamiento casi absolutas, logrando NSGA-II alcanzar prácticamente el 100 % de éxito con una ligera antelación temporal frente a H1.



(a) Hipervolumen. (b) Tiempo de ejecución. (c) Frente de Pareto.

Figura 7: Resultados para la hora valle.

En conclusión, aunque en episodios de tráfico fluido la simplicidad de una heurística voraz como H1 podría ser suficiente, el algoritmo evolutivo NSGA-II mantiene una notable estabilidad computacional y logra afinar el Frente de Pareto definitivo. Estos resultados consolidan a NSGA-II como la estrategia de resolución integral y universal más robusta para el sistema, capaz de operar ininterrumpidamente sin importar la franja horaria.

6. Conclusiones y trabajos futuros

El sistema propuesto aborda de manera integral el problema del estacionamiento urbano mediante un enfoque metodológico de extremo a extremo. En el plano predictivo, la arquitectura basada en redes neuronales recurrentes LSTM multivariantes con estrategia *Direct Multi-Step* ha demostrado una robustez excepcional para mitigar la propagación del error secuencial.

Por otra parte, la integración del estimador probabilístico híbrido en red (NKDE) ha permitido propagar la presión de la demanda solventando la carencia de sensorización en superficie. Por otro lado, el algoritmo NSGA-II propuesto ha probado su eficacia al generar Frentes de Pareto representativos que resuelven de forma eficiente el conflicto inherente entre el tiempo esperado de trayecto y la probabilidad de éxito al aparcar.

Como líneas futuras de investigación, se plantean dos vertientes principales. Por un lado, a nivel metodológico, se explorará el uso de modelos avanzados de

aprendizaje profundo para capturar la topología geoespacial viaria, y el enriquecimiento del modelo con factores exógenos como las condiciones meteorológicas y el tráfico en tiempo real. Por otro lado, también se plantea la validación en otros escenarios urbanos con configuraciones topológicas y dinámicas heterogéneas.

Acknowledgments. Este trabajo ha sido parcialmente financiado por el proyecto de investigación AIM-Zero (PID2024-158752OB-I00).

Referencias

1. Akiba, T., Sano, S., Yanase, T., Ohta, T., Koyama, M.: Optuna: A next-generation hyperparameter optimization framework. In: Proceedings of the 25th ACM International conference on knowledge discovery & data mining. pp. 2623–2631 (2019)
2. Chen, Y., Chan, W.H., Su, E.L.M., Diao, Q.: Multi-objective optimization for smart cities: a systematic review of algorithms, challenges, and future directions. PeerJ Computer Science **11**, e3042 (2025)
3. Coello, C.A.C., Lamont, G.B., Veldhuizen, D.A.V.: Evolutionary algorithms for solving multi-objective problems. Springer (2007)
4. Dijkstra, E.W.: A note on two problems in connexion with graphs. Numerische Mathematik **1**(1), 269–271 (12 1959). https://doi.org/10.1007/bf01386390
5. Dogaroglu, B., Caliskanelli, S.P., Tanyel, S.: Comparison of intelligent parking guidance system and conventional system with regard to capacity utilisation. Sustainable Cities and Society **74**, 103152 (2021)
6. Faroqi, H.: Multiobjective route finding in a multimode transportation network by nsga-ii. Journal of Engineering and Applied Science **71**(1), 81 (2024)
7. Goodfellow, I., Bengio, Y., Courville, A., Bengio, Y.: Deep learning. MIT (2016)
8. Haklay, M., Weber, P.: Openstreetmap: User-generated street maps. IEEE Pervasive computing **7**(4), 12–18 (2008)
9. Lara-Benítez, P., Carranza-García, M., Riquelme, J.C.: An experimental review on deep learning architectures for time series forecasting. International journal of neural systems **31**(03), 2130001 (2021)
10. Levering, N., Boon, M., Mandjes, M., Núñez-Queija, R.: A framework for efficient dynamic routing under stochastically varying conditions. Transportation research part B: methodological **160**, 97–124 (2022)
11. López-Ibáñez, M., Dubois-Lacoste, J., Cáceres, L.P., Birattari, M., Stützle, T.: The irace package: Iterated racing for automatic algorithm configuration. Operations Research Perspectives **3**, 43–58 (2016)
12. Paidi, V., Håkansson, J., Fleyeh, H., Nyberg, R.G.: Co2 emissions induced by vehicles cruising for empty parking spaces in an open parking lot. Sustainability **14**(7), 3742 (2022)
13. Saki, S., Hagen, T.: Cruising for parking again: Measuring the ground truth and using survival analysis to reveal the determinants of the duration. Transportation Research Part A: Policy and Practice **183**, 104045 (2024)
14. Sun, Y., Yu, X., Bie, R., Song, H.: Discovering time-dependent shortest path on traffic graph for drivers towards green driving. Journal of Network and Computer Applications **83**, 204–212 (2017)
15. Westminster City Council: Westminster citywide parking occupancy survey 2022. Tech. rep., Westminster City Council, London, UK (2022)
16. Yang, S., Ma, W., Pi, X., Qian, S.: A deep learning approach to real-time parking occupancy prediction in transportation networks incorporating multiple spatio-temporal data sources. Trans. Rese. Part C: Emerging Tech. **107**, 248–265 (2019)