

# Hybridization of NSGA-II and MILP for Optimization of the Location of Electric-Scooters Sharing-Stations

Enrique Gabriel Baquela  and Ana Carolina Olivera 

**Abstract**—A crucial aspect of the proper functioning of bikes and electric scooters' sharing systems is the correct location and dimensioning of the sharing stations. The resolution of the previous problem is carried out based on the maximization of coverage or the minimization of costs, but the two objectives are not usually treated at the same time.

In this work, we propose a method based on the hybridization of the popular Elitist Non-Domainted Sorted Genetic Algorithm (NSGA-II) with a Mixed-Integer Linear Programming (MILP) model to approximate the Pareto Frontier of the problem. This allows the decision-maker a greater understanding of the range of possible options. The NSGA-II plays the role of an outer block that deals with the selection and sizing of each of sharing stations. The MILP model is an inner block that calculates the associated coverage of that solution. The schema was compared with an adaptative-weighting algorithm, reaching the hybridization of NSGA-II and MILP a better coverage of the Pareto Frontier.

**Index Terms**—Multi-Objective Mixed-Integer Linear Programming, Meta-heuristics, Math-heuristics, E-Scooters, Bike-Sharing, Sustainable Transportation

## I. INTRODUCCIÓN

En términos de movilidad y sustentabilidad, los monopatines eléctricos ofrecen una buena combinación de velocidad, capacidad de desplazamiento y bajo consumo de fuentes de energía [1]. Si bien su uso podría ser un poco menos amistoso con el medio ambiente que las bicicletas, permiten cubrir distancias mayores manteniendo las mismas ventajas de estas en cuanto a practicidad e impacto en el sistema de tráfico [2]. Y en caso de ser alimentados por electricidad provenientes de fuentes no renovables, la emisión asociada de  $CO_2$  por km y pasajero es mucho menor al de vehículos como autos y motos [3].

Debido a la relativamente reciente irrupción de los monopatines eléctricos, no existe tanta investigación acerca del desarrollo de infraestructura y sistemas públicos de uso común para estos vehículos como existe con las bicicletas (usualmente referenciados en la literatura como *bike-sharing*). Pero, debido a sus similitudes, estos últimos son un buen punto de partida. En lo que respecta al *bike-sharing*, el diseño de infraestructura

(desde la óptica de los problemas de optimización) consiste en definir en donde instalar estaciones de intercambio, de que capacidad tienen que ser las mismas (es decir, cuantas bicicletas pueden estar estacionadas al mismo tiempo) y definir la política de reposición de bicicletas entre estaciones (es decir, la utilización de un servicio público para mover bicicletas de estación a estación a fin de cubrir desequilibrios en la demanda) [4], [5]. Cabe aclarar que, si bien el foco de este trabajo está puesto en esquemas de intercambio en los cuales los mismos se realizan en estaciones, existen experiencias que permiten el intercambio por fuera de dichas instalaciones [6], [7].

Las investigación acerca de la optimización de sistemas de *bike-sharing* ha estado focalizada o bien a maximizar la satisfacción del usuario del servicio o bien a minimizar los costos de inversión u operativos. En este trabajo, se propone una metodología multi-objetivo, aplicable tanto a sistemas públicos de monopatines eléctricos como de bicicletas, para optimizar los dos objetivos en simultaneo (en el marco de trabajo de Fronteras de Pareto).

Este trabajo se estructura como sigue: la Sección II presenta una revisión de la literatura relacionada. El problema de optimización a resolver es descrito en la Sección III. En la Sección IV se propone el algoritmo de resolución multi-objetivo para este problema, evaluándose el mismo en la Sección V. Finalmente, las conclusiones y trabajos futuros se exponen en la Sección VI.

## II. TRABAJOS RELACIONADOS

Los modelos de optimización de *bike-sharing* son modelos bastantes populares en la literatura de optimización de la última década [8], [9]. Uno de los problemas de optimización tratados es la localización de las estaciones, esto es, el diseño estructural de la red de intercambios de bicicletas [10]–[12]. La optimización en la localización de estaciones suele ser tratada en la literatura en dos formas, principalmente: métodos exactos, como Programación Lineal Entera-Mixta (MILP) [4], [13], o mediante meta-heurísticas [14], pero en general, en ambos casos, como problemas mono-objetivos.

Desde el punto de vista del nivel de servicio al usuario, ya sea usando métodos exactos o meta-heurísticas, se utilizan dos enfoques principalmente: minimizar el costo para el usuario o maximizar la cantidad de usuarios que puede gestionar el sistema [13]. En el primero, se hace uso de modelos de localización del tipo *p-medianas* [15]–[18], mientras que en los

Enrique Gabriel Baquela is with Grupo de Investigación en Simulación y Optimización Industrial (GISOI), Dpto. Ingeniería Industrial, Facultad Regional San Nicolás (FRSN), Universidad Tecnológica Nacional (UTN), Colón 332, B2900LWH, Buenos Aires, Argentina, e-mail:ebaquela@frsn.utn.edu.ar

Ana Carolina Olivera is with Instituto para las Tecnologías de la Información y las Comunicaciones, Consejo Nacional de Investigaciones Científicas y Técnicas, Universidad Nacional de Cuyo, Padre Jorge Contreras 1300, M5502JMA, Mendoza, Argentina and with Facultad de Ingeniería, Universidad Nacional de Cuyo, Centro Universitario, M5502JMA, Mendoza, Argentina e-mail:acolivera@conicet.gov.ar

segundos de modelos de cobertura máxima [19]–[24]. Desde el punto de vista de los costos de operación y/o instalación, generalmente se consolidan dichos costos en una única función objetivo [25], [26]. Cuando se tratan objetivos de costos y servicio al usuario simultáneamente, el enfoque suele estar en combinar los dos tipos de objetivos en uno solo, ponderándolos [4], [25].

Respecto del problema en si de la localización de estaciones para monopatines eléctricos, los trabajos mas recientes siguen estrategias similares a los trabajos relativos a bicicletas. Tian et. al. [27] utilizan el algoritmo *Shuffled Complex Evolution* para determinar los mejores lugares de carga para los monopatines. Yan et. al. [28] modelan el problema de la localización de estaciones de intercambio de baterías de monopatines eléctricos mediante un modelo de programación lineal entero-mixta, buscando minimizar el costo total de las operaciones. Deb et. al. [29] plantean la hibridación de meta-heurísticas para resolver el problema de localización de estaciones de carga. Y Chen et. al. [30] utilizan una meta-heurística multi-objetivo (*Enjambre de partículas multi-objetivo*, *MSPSO* particularmente) para resolver el problema de localización de distintos tipos de estaciones de carga optimizando tanto los costos como el nivel de servicio al usuario.

En este trabajo, se propone una metodología multi-objetivo, aplicable tanto a sistemas públicos de monopatines eléctricos como de bicicletas, para optimizar los dos objetivos en simultaneo (en el marco de trabajo de Fronteras de Pareto).

### III. PROBLEMA

#### A. Descripción

El problema a tratar consiste en definir en que lugares emplazar estaciones de intercambio de monopatines eléctricos (y/o bicicletas) con el objetivo de minimizar el costo total de las inversiones y maximizar la cobertura respecto a la demanda de uso de dichos monopatines.

En el tipo de sistemas analizados, los monopatines eléctricos están asignados a una determinada estación al inicio del día. En base a la demanda, los usuarios del servicios toman un monopatín, conducen hacia otra estación y los dejan cargándose, para que otro usuario pueda hacer uso de ellos. A efectos de equilibrar la oferta con la demanda, un conjunto de monopatines son reubicados desde estaciones con poca demanda a estaciones con mas demanda, siendo trasportados por personal administrador del servicio. Al final del día, se espera que los monopatines estén ubicados en las estaciones de arranque del día siguiente.

Cada estación puede tener uno o mas *racks*, esto es, espacio para depositar los monopatines. El tamaño de la estación y la capacidad de las mismas depende de esta cantidad de *racks*, así como gran parte de la inversión necesaria.

#### B. Formulación del Problema

A fin de modelar el problema, se escogió un formalismo basado en programación lineal entero-mixta multi-objetivo. El modelo (basado en [4]) permite calcular el número de estaciones necesarias, la ubicación de ellas (entre todas las posibles ubicaciones factibles), la cantidad de racks necesarios

en cada una y la política de relocalización de los vehículos. Los objetivos son, por un lado, minimizar la inversión total (la cual es estimada en base al número de racks totales) y, por otro, maximizar la cobertura respecto de la demanda total. El modelo completo es mostrado en (1) a (16).

$$\text{máx } Z_1 = \sum_{i \in J} \sum_{j \in J} \sum_{t \in T} u_{i,j,t} x_{i,j,t} \quad (1)$$

$$\text{mín } Z_2 = \sum_{i \in J} p v_i z_i + \sum_{i \in J} p f_i y_i \quad (2)$$

Sujeto a:

$$v_{i,t} = v_{i,t-1} - \sum_{j \in J} u_{i,j,t-1} x_{i,j,t-1} + \sum_{j \in J} u_{j,i,t-1} x_{j,i,t-1} - \sum_{j \in J} r_{i,j,t-1} + \sum_{j \in J} r_{j,i,t-1}, \forall i \in J, j \in J, t \in T \quad (3)$$

$$v_{i,1} = v_{i,T}, \forall i \in J \quad (4)$$

$$z_i \leq z_{max} \cdot y_i, \forall i \in J \quad (5)$$

$$z_i \geq z_{min} \cdot y_i, \forall i \in J \quad (6)$$

$$v_{i,t} \geq \sum_{j \in J} u_{i,j,t-1} x_{i,j,t-1}, \forall i \in J, t \in T \quad (7)$$

$$v_{i,t} \leq z_i, \forall i \in J, t \in T \quad (8)$$

$$v_{i,t} \geq \text{minprop} \cdot z_i, \forall i \in J, t \in T \quad (9)$$

$$v_{i,t} \geq \sum_{j \in J} r_{i,j,t-1}, \forall i \in J, t \in T \quad (10)$$

$$\sum_{j \in J} x_{i,j,t} \leq 1, \forall i \in J, t \in T \quad (11)$$

$$x_{i,j,t} \leq y_i, \forall i \in J, j \in J, t \in T \quad (12)$$

$$x_{i,j,t} \leq y_j, \forall i \in J, j \in J, t \in T \quad (13)$$

$$x_{i,j,t} \geq 0, \forall i \in J, j \in J, t \in T \quad (14)$$

$$y_i \in \{0, 1\}, \forall i \in J \quad (15)$$

$$v_{i,t}, z_i, r_{i,j,t} \in \mathbb{Z}_0^+, \forall i \in J, j \in J, t \in T \quad (16)$$

$J$  y  $T$  son los conjuntos que representan las zonas de demandas (esto es, los potenciales emplazamientos de las estaciones, indexadas por  $i$  y  $j$ ) y la partición en periodos de tiempos bajo análisis (indexados por  $t$ ).  $x_{i,j,t}$  es una variable continua que toma valores en el intervalo  $[0, 1]$  y que representa la proporción de la demanda cubierta entre las zonas  $i$  y  $j$  en el periodo  $t$ .  $y_i$  es una variable binaria que toma el valor 1 si se decide emplazar una estación de intercambio en la zona  $i$ , y 0 en el caso contrario.  $z_i$  es una variable entera que indica cuantos *racks* hay disponibles en la estación emplazada en la zona  $i$  (es decir, cual es la cantidad máxima de monopatines que puede haber en la estación en un momento determinado). La variable entera  $r_{i,j,t}$  indica cuantos monopatines tienen que ser reubicados desde la zona  $i$  a la zona  $j$  en el periodo  $t$ . Y la variable  $v_{i,t}$  indica cuantos monopatines hay, efectivamente, en la estación emplazada en la zona  $i$  al inicio del periodo  $t$ .  $z_{min}$  y  $z_{max}$  son dos parámetros que limitan la cantidad mínima y máxima de *racks*

en la estación emplazada en la zona  $i$ . El parámetro  $minprop$  indica la mínima proporción de *racks* que se desean estén ocupados (usualmente, 0 es un buen valor).  $pf_i$  es el costo fijo de instalación de una estación en la locación  $i$ , mientras que  $pv_i$  es el costo adicional por cada rack instalado. Por último, el parámetro  $u_{i,j,t}$  representa la demanda total de viajes en monopatines eléctricos desde la zona  $i$  a la  $j$  en el periodo  $t$ .

En (1) se representa el objetivo de maximizar la cobertura de la demanda (es decir, el nivel de servicio), y (2) contempla la minimización del costo total de instalación de las estaciones. Este costo total consiste en la suma, por cada potencial ubicación, del costo fijo por instalar la estación mas el costo asociado a la cantidad de racks en dicha estación (es decir, a su tamaño). (3) es una restricción de balance de flujo: la cantidad de monopatines en la estación situada en la zona  $i$  al inicio del periodo  $t$  debe ser igual a la cantidad que había en dicha estación en el periodo  $t - 1$  mas todos los arribos de monopatines menos todos los egresos. Mientras que (4) fuerza a que el fin del día sea igual al inicio. (5) y (6) definen las cotas para la cantidad de *racks* en la estación emplazada en la zona  $i$ , en caso que se decida emplazarla, o bien fuerzan a que no haya ningún *rack*, en caso que no se decida emplazar una estación en dicha zona. En (7) se vincula la cobertura de la demanda con la cantidad monopatines disponibles en cada estación. (8) y (9) restringen la cantidad de monopatines al inicio de cada periodo. (10) vincula la cantidad de monopatines al inicio del periodo con la cantidad de monopatines que pueden ser reubicados. (11) garantiza que las proporciones de cobertura de demanda sumen, como mucho, 1. (12) y (13) permiten que solo pueda haber flujos de monopatines entre dos zonas si y solo si las dos zonas tienen una estación emplazada cada una. Finalmente, (14) fuerza a  $x_{i,j,k}$  a ser no-negativa, (15) define a  $y_i$  como una variable binaria y (16) define a  $v_{i,t}$ ,  $z_i$  y  $r_{i,j,t}$  como variables enteras.

#### IV. ALGORITMO DE RESOLUCIÓN

##### A. Programación Lineal Entero-Mixta Multi-Objetivo (MO-MILP)

Dado un problema de Programación Entera-Mixta con  $k$  objetivos (siendo  $k > 1$ ), la meta es hallar la Frontera de Pareto del problema, es decir, el conjunto de todas las soluciones que presentan el *trade-off* óptimo entre objetivos. En terminología de problemas multi-objetivos, la Frontera de Pareto es el conjunto de todas las soluciones no-dominadas del problema. Asumiendo que queremos minimizar todos los objetivos, una solución  $y$  se denomina no-dominada si y solo si no hay ninguna otra solución  $x \in X$  tal que  $f_i(x) \leq f_i(y)$  para todos los objetivos  $i = 1..k$ , y  $f_i(x) < f_i(y)$  para el menos un  $i$ . Por otro lado, una solución es una solución no-dominada débil si y solo si no existe ninguna solución  $x \in X$  tal que  $f_i(x) < f_i(y)$  para todos los objetivos  $i = 1..k$  [31]

Una consideración a tener en cuenta cuando se resuelve un MOMILP es que pueden existir soluciones no-dominadas *sin soporte*. Esto es, existen soluciones no-dominadas que son dominadas por una combinación lineal de otras dos soluciones no-dominadas (no perteneciendo ninguna de estas dos soluciones a la imagen factible del problema, por ejemplo,

por no tener componentes enteras) [31]. Esto significa, en la práctica, que los métodos basados en ponderar objetivos iterativamente podrían no generar toda la Frontera de Pareto.

Dadas las dos situaciones descritas anteriormente, la combinación de métodos exactos (la programación lineal entera-mixta, por ejemplo) con métodos de aproximación (como las meta-heurísticas) ha dado buenos resultados, permitiendo escapar de las limitaciones de cada enfoque [32]–[35]. Las meta-heurísticas son muy buenos algoritmos para aproximar la Frontera de Pareto [36]–[39], pero tienen problemas para lidiar con problemas con restricciones [35], [40], [41]. Los *solvers* para MILP son muy buenos para encontrar soluciones exactas, pero pueden tener un alto costo computacional, además de no tratar en forma directa la cuestión de la optimización multi-objetivo. Este enfoque de combinar optimización matemática y meta-heurísticas es usualmente denominado *mate-heurísticas* [42], [43].

##### B. Combinando NSGA-II con MILP

El Elitist Non-Dominated Sorting Genetic Algorithm (NSGA-II) es una meta-heurística basada en Algoritmos Genéticos (GA) utilizada para resolver problemas de optimización multi-objetivo [44]. Con variaciones de su definición original, ha sido ampliamente usado para resolver problemas de optimización continua, entera y entera-mixta multi-objetivo [36], [39], [45]–[47]. En este trabajo, el NSGA-II es combinado con un algoritmo de resolución de MILP (MILP-Solver en adelante) con el fin de aproximar la Frontera de Pareto.

En el esquema propuesto en este trabajo, el algoritmo NSGA-II es el encargado de proponer en que zonas se instalará una estación de compartición de los vehículos eléctricos y que tamaño tendrán. Con las zonas y el tamaño de cada estación definidas, (es decir, con valores dados para todas las  $y_i$  y  $z_i$ ), el MILP-Solver calcula los valores óptimos para el resto de las variables. En este esquema,  $Z_2$  puede ser evaluada sin necesidad de ejecutar el MILP-Solver, quedando este con el fin de evaluar  $Z_1$ .

Las soluciones al nivel del algoritmo NSGA-II (individuos en la terminología de algoritmos genéticos) se codifican como un vector de  $J$  componentes, en el cual cada una está asociada con la variable  $z_i$ , pudiendo tomar valores entre 0 y  $z_{max} - z_{min}$ . El cómputo de  $Z_2$  se vuelve así trivial:  $y_i$  toma valor 0 si la componente asociada a  $z_i$  vale 0, y 1 en caso contrario. Y, dados estos valores de  $z_i$  e  $y_i$ , el problema de optimización entera-mixta original queda reducido a un problema mono-objetivo definido solamente en términos de las variables  $x_{i,j,t}$ ,  $v_{i,t}$  y  $r_{i,j,t}$ , sin las restricciones (5), (6) y (15).

**B1. Generación de individuos:** A fin de generar nuevas poblaciones, iteración tras iteración, las operaciones de cruzamiento y mutación son aplicadas a la población de individuos. El cruzamiento elegido es cruzamiento simple en un punto. La mutación consiste aquí en seleccionar una componente al azar de cada vector solución y obtener su nuevo valor muestreando aleatoriamente el conjunto  $\{0, 1, \dots, z_{max} - z_{min}\}$ .

La población inicial se genera mediante operaciones de mutación y cruzamiento aplicadas a las dos soluciones extremas del problema: la del problema mono-objetivo basado en  $Z_1$  y

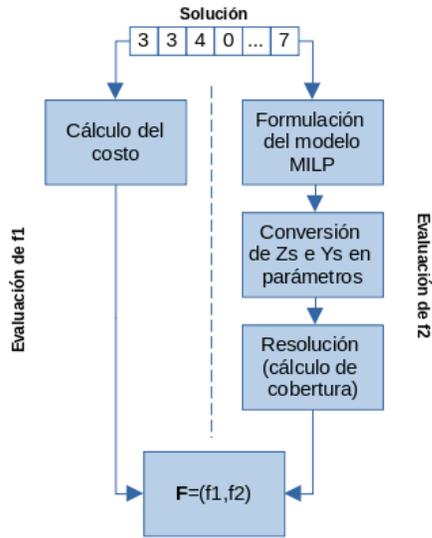


Fig. 1. Esquema de evaluación de soluciones y cálculo de funciones objetivo.

del problema mono-objetivo basado en  $Z_2$  (incluyendo estas dos en la población original). Con esto, se garantiza comenzar la búsqueda con al menos dos soluciones de la Frontera de Pareto, además de soluciones potencialmente más próximas a dicha frontera que en un esquema puramente aleatorizado.

**B2. Evaluación de individuos:** La evaluación de cada individuo tiene dos etapas diferenciadas.  $Z_2$  se evalúa aplicando la definición del objetivo en el individuo. Para evaluar  $Z_1$ , se construye un problema de optimización mono-objetivo basado en el problema original, removiendo  $Z_2$  y reemplazando todas las  $y_i$  y  $z_i$  por los valores precalculados por el NSGA-II. Bajo este esquema, las únicas variables enteras del problema son  $r_{i,j,t}$  y  $v_{i,t}$ . Este esquema se muestra en la Fig. 1. Partiendo de una solución determinada, el proceso de cálculo de cada objetivo se separa en dos ramas, el conteo de estaciones y racks, por un lado, y la formulación y resolución del MILP.

## V. EXPERIMENTOS

A fin de validar el esquema propuesto, se diseñaron 10 escenarios artificiales de testeos. Cada uno tiene su propio número de zonas y sus propios valores de parámetros. El tamaño y las características de los escenarios probados se muestran en la Tabla I. La primera columna es el nombre del escenario, la segunda la cantidad de zonas, el tercero la cota máxima del número de vehículos, el cuarto la demanda máxima por cada par  $i, j$ , la quinta la cantidad máxima de racks en cada zona y la última columna indica la cantidad de variables del modelo MILP asociado al escenario. La demanda fue generada aleatoriamente, siguiendo una distribución uniforme en los 5 primeros escenarios y una distribución clusterizada en pocas zonas en los últimos 5 (25% de las zonas con demanda máxima igual a doble de la demanda máxima en la tabla, el valor indicado en tabla para el resto). Por último, dado que el costo de la inversión es un factor fuertemente dependiente de la localidad en la cual se esté realizando el estudio, se asignó

TABLA I  
ESCENARIOS

n esce	n zonas	max veh	max dem	max $z_i$	n vars
1	4	40	10	5	680
2	10	50	10	10	4,580
3	20	50	20	10	18,760
4	20	100	40	10	18,760
5	30	200	50	10	42,540
6	10	50	10	10	4,580
7	20	50	20	10	18,760
8	20	100	30	10	18,760
9	20	200	40	10	18,760
10	30	300	50	10	42,540

el valor 1 a  $pv_i$  y el valor 0 a  $pg_i$  (para todas las  $i \in J$ ) de manera tal que (1) exprese la cantidad de racks instalados. El algoritmo NSGA-II se parametrizó con modalidad de selección por torneo, cruzamiento simple, mutación simple, un tamaño de población de 100 y 50 generaciones.

Todos los experimentos se realizaron en una computadora con CPU Intel i7 de 2.1 Ghz con 4 núcleos y 16 GB de RAM. El solver de Programación Lineal Entero Mixta utilizado fue el SCIP [48], mientras que el algoritmo NSGA-II se implementó utilizando la biblioteca Pymoo [49] de Python.

Cada escenario se optimizó utilizando el algoritmo propuesto en la Sección B y un algoritmo de pesos adaptativos basado solamente en MILP (basado en [50] y [51]) se utilizó como *benchmark*. Es decir,  $z_1$  y  $z_3$  se consolidaron en (17) (con  $\lambda \in [0, 1]$ ), muestreándose el espacio de valores de  $\lambda$  a fin de barrer la Frontera de Pareto.

$$\text{Min } (-1)\lambda Z_1 + (1 - \lambda)Z_2 \quad (17)$$

En cada escenario, se evaluó el Hipervolumen Dominado [52], [53] y la cantidad de soluciones no-dominadas halladas por cada algoritmo. La Tabla II resume los resultados encontrados. La primera columna es el escenario analizado, la segunda y la tercera muestran el hipervolumen y la cantidad de soluciones no-dominadas halladas en para el algoritmo basado en pesos, la cuarta y quinta los mismos datos para el NSGA-II + MILP, y las sexta y séptima los tiempos de computo (en minutos) del algoritmo basado en pesos y el algoritmo propuesto en este trabajo. La Fig. 2 y la Fig. 3 muestran, respectivamente, las Fronteras encontradas por el método basado en pesos y por el algoritmo propuesto en este trabajo. En ambas, el eje horizontal muestra los valores de  $-z_1$  y el eje vertical los valores de  $z_2$ . Si bien el método basado en pesos permite encontrar los puntos extremos de la Frontera de Pareto, falla en encontrar muchas de las soluciones intermedias (de hecho, en la figura mostrada, aparenta estar sesgado a buscar principalmente una buena cobertura de la demanda, no tanto bajo costo de inversión). Se observa como el algoritmo basado en NSGA-II es más exitoso en barrer la frontera completa. También se puede observar como el método de pesado generó muchas soluciones alternativas para prácticamente los mismos valores de cobertura e inversión (se observa por la gran densidad de puntos en la zona superior-izquierda de la Fig. 2), mientras que el algoritmo basado en NSGA-II aparenta no generar demasiadas soluciones alternativas para el mismo rango de valores de los objetivos. Las Fronteras encontradas

TABLA II  
COMPARACIÓN DE LOS RESULTADOS

esc	pesos dh	pesos nd	nsga-ii dh	nsga-ii nd	tpo pesos	tpo nsga-ii
1	25,519	8	32,851	107	<1	<1
2	33,600	28	44,506	170	<1	2
3	44,506	17	67,113	138	6	8
4	47,882	15	62,910	120	6	8
5	50,621	26	69,822	103	33	27
6	23,795	30	32,111	115	<1	2
7	28,603	22	35,344	101	6	8
8	37,601	19	40,997	104	6	8
9	45,876	24	57,248	110	9	8
10	63,974	16	84,173	102	36	31

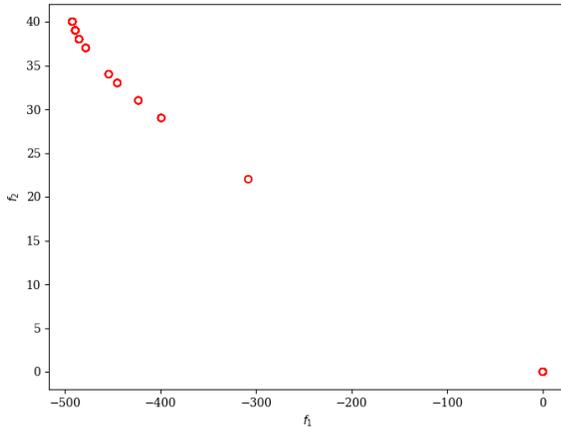


Fig. 2. Frontera de Pareto para el Escenario 03, usando el método de pesado.  $f_1$  es el negativo de la demanda cubierta total.  $f_2$  es el número total de racks instalados. El parámetro  $pv_i$  toma el valor 1 y el parámetro  $pf_i$  el valor 0.

por los algoritmos tienden a presentar un trade-off lineal entre cobertura y cantidad de instalaciones ( $z_1$  y  $z_2$  con  $pf = 0$  y  $pv = 1$  para  $i \in J$ , respectivamente) a lo largo de la mayor parte de espacio de objetivos, comenzando a perderse esta linealidad cuando la cobertura es mas alta (es decir, la cantidad de instalaciones necesarias para un determinado incremento de capacidad crece mas que proporcionalmente).

Respecto de la evolución del NSGA-II, en la Tabla III se muestra un extracto de la evolución de la Frontera de Pareto estimada en cada generación para el escenario 3 (a efectos de claridad se incluyen las primeras 21 generaciones, donde al algoritmo se estabiliza, y la última generación). La primera columna muestra la generación actual, la segunda el acumulado de soluciones evaluadas, la tercera la cantidad de soluciones no-dominadas respecto de la población actual y la cuarta el valor absoluto del Indicador Épsilon [52] entre la frontera estimada en la generación actual y la estimada hasta la generación anterior. El Indicador Épsilon mide la distancia entre dos curvas, por lo tanto, valores mas cercanos a cero indican mayores semejanzas. Se puede observar como, a partir de la generación 20, las mejoras en la calidad de la estimación de la Frontera de Pareto son marginales, ya que converge rápidamente a una buena aproximación.

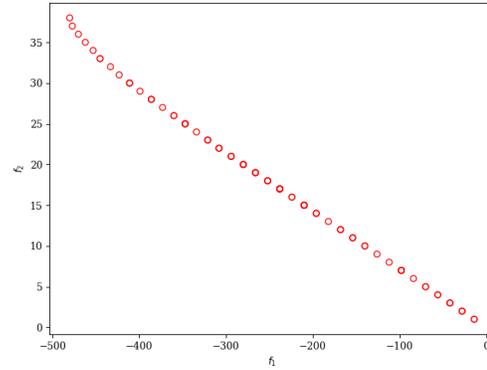


Fig. 3. Frontera de Pareto para el Escenario 03, usando NSGA-II + MILP.  $f_1$  es el negativo de la demanda cubierta total.  $f_2$  es el número total de racks instalados. El parámetro  $pv_i$  toma el valor 1 y el parámetro  $pf_i$  el valor 0.

TABLA III  
EVOLUCIÓN DEL NSGA-II + MILP PARA EL ESCENARIO 3.

n_gen	n_eval	n_nds	eps
1	100	31	-
2	200	38	0.006060260
3	300	51	0.002435675
4	400	68	0.006681514
5	500	88	0.000455558
6	600	100	0.000044543
7	700	100	0.000155902
8	800	100	0.000200445
9	900	100	0.000155902
10	1,000	100	0.000111359
11	1,100	100	0.028571429
12	1,200	100	0.000044444
13	1,300	100	0.00000E+00
14	1,400	100	0.000066667
15	1,500	100	0.00000E+00
16	1,600	100	0.000022222
17	1,700	100	0.00000E+00
18	1,800	100	0.006622517
19	1,900	100	0.00000E+00
20	2,000	100	0.000022075
21	2,100	100	0.00000E+00
...	...	...	...
50	5,000	100	0.00000E+00

A efecto de evaluar el impacto de parámetro  $pf_i$ , los escenarios anteriores se volvieron a ejecutar utilizando  $pf_i = 10$  y  $pv_i = 1$ . Es decir, por cada estación instalada, tenemos un costo de 10, y por cada rack en cada estación un costo adicional de 1. Lo Frontera hallada para el escenario 3 se puede observar en la Fig. 4) (en la cual el eje vertical es ahora el costo total). Se observa que la misma adopta otra forma, notándose el impacto de agregar una nueva estación entre salto y salto de cada grupo de puntos.

## VI. CONCLUSIONES

La combinación de una meta-heurística como el NSGA-II y un modelo/algoritmo de optimización exacta (MILP) ha sido satisfactoria para tratar el problema de optimización multi-objetivo propuesto en este trabajo. NSGA-II permite realizar una exploración mas completa del espacio de objetivos respecto del algoritmo utilizado como benchmark, mientras

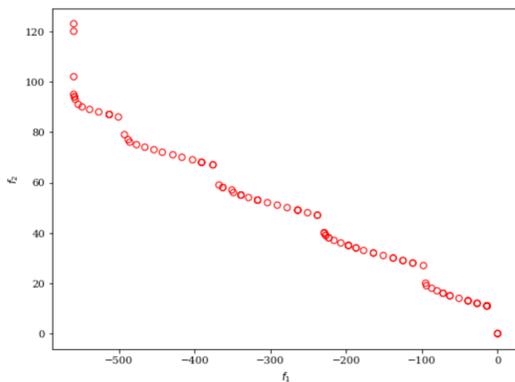


Fig. 4. Frontera de Pareto para el Escenario 03, usando NSGA-II + MILP.  $f_1$  es el negativo de la demanda cubierta total.  $f_2$  es el costo total de las instalaciones. El parámetro  $pv_i$  toma el valor 1 y el parámetro  $pf_i$  el valor 10.

que la combinación con el modelo MILP permite que, dado un valor de uno de los objetivos (generado por el NSGA-II) se puede encontrar el valor óptimo asociado del otro objetivo, descartando rápidamente regiones del espacio de objetivo lejos de la frontera óptima. Los tiempos de cómputos fueron similares al algoritmo de benchmark, pero con un nivel de cobertura mucho mayor de la Frontera de Pareto.

Como futuros trabajos, resulta necesario realizar una prueba en un contexto real (demanda y distribución de zonas potenciales), así como analizar el impacto de distintas configuraciones del algoritmo NSGA-II y como impactan en la evolución de la aproximación de la frontera.

#### AGRADECIMIENTOS

Los fondos para realizar esta investigación provienen en parte de la Universidad Tecnológica Nacional, Facultad Regional San Nicolás, a través del proyecto PIDUTN8252. Ana Carolina Olivera quiere agradecer a la Universidad Nacional de Cuyo (Proyecto 80020210100059UN) y a la Agencia Nacional de Promoción de la Investigación, el Desarrollo Tecnológico y la Innovación (AGENCIA I+D+i), a través del Fondo para la Investigación Científica y Tecnológica (FONCyT) PICT-2020-SERIEA-00743, Argentina. Ana Carolina Olivera también agradece al Consejo Nacional de Investigaciones Científicas y Técnicas, Argentina.

#### REFERENCIAS

- [1] C. Hardt and K. Bogenberger, "Usage of e-scooters in urban environments," *Transportation Research Procedia*, vol. 37, pp. 155–162, 2019, 21st EURO Working Group on Transportation Meeting, EWGT 2018, 17th – 19th September 2018, Braunschweig, Germany. [Online]. Available: <https://www.sciencedirect.com/science/article/pii/S2352146518305933>
- [2] Łukasz Nawaro, "E-scooters: competition with shared bicycles and relationship to public transport," *International Journal of Urban Sustainable Development*, vol. 13, no. 3, pp. 614–630, 2021. [Online]. Available: <https://doi.org/10.1080/19463138.2021.1981336>
- [3] A. Bozzi and A. Aguilera, "Shared e-scooters: A review of uses, health and environmental impacts, and policy implications of a new micro-mobility service," *Sustainability (Switzerland)*, vol. 13, 08 2021.
- [4] I. Frade and A. Ribeiro, "Bike-sharing stations: A maximal covering location approach," *Transportation Research Part A General*, vol. 82, p. 216–227, 12 2015.

- [5] T. Casey. (2021, Mar.) Where to dock & charge them all? Web article. CleanTechnica. <https://cleantechnica.com/2021/03/05/so-many-e-scooters-so-where-to-dock-charge-them-all/>.
- [6] H. Younes, Z. Zou, J. Wu, and G. Baiocchi, "Comparing the temporal determinants of dockless scooter-share and station-based bike-share in Washington, D.C.," *Transportation Research Part A: Policy and Practice*, vol. 134, pp. 308–320, 2020. [Online]. Available: <https://www.sciencedirect.com/science/article/pii/S0965856419311553>
- [7] P. Zhou, C. Wang, Y. Yang, and X. Wei, "E-sharing: Data-driven online optimization of parking location placement for dockless electric bike sharing," in *2020 IEEE 40th International Conference on Distributed Computing Systems (ICDCS)*, 2020, pp. 474–484.
- [8] C. Shui and W. Szeto, "A review of bicycle-sharing service planning problems," *Transportation Research Part C: Emerging Technologies*, vol. 117, p. 102648, 2020. [Online]. Available: <https://www.sciencedirect.com/science/article/pii/S0968090X20305635>
- [9] J. Zhou, Y. Guo, J. Sun, E. Yu, and R. Wang, "Review of bike-sharing system studies using bibliometrics method," *Journal of Traffic and Transportation Engineering (English Edition)*, 2022. [Online]. Available: <https://www.sciencedirect.com/science/article/pii/S2095756422000174>
- [10] J. C. García-Palomares, J. Gutiérrez, and M. Latorre, "Optimizing the location of stations in bike-sharing programs: A GIS approach," *Applied Geography*, vol. 35, no. 1, pp. 235–246, 2012. [Online]. Available: <https://www.sciencedirect.com/science/article/pii/S0143622812000744>
- [11] E. Croci and D. Rossi, "Optimizing the position of bike sharing stations. The Milan case," *SSRN Electronic Journal*, 01 2014.
- [12] J. Liu, Q. Li, M. Qu, W. Chen, J. Yang, H. Xiong, H. Zhong, and Y. Fu, "Station site optimization in bike sharing systems," in *2015 IEEE International Conference on Data Mining*, 2015, pp. 883–888.
- [13] C. Park and S. Sohn, "An optimization approach for the placement of bicycle-sharing stations to reduce short car trips: An application to the city of Seoul," *Transportation Research Part A: Policy and Practice*, vol. 105, pp. 154–166, 11 2017.
- [14] C. Cintrano, F. Chicano, and E. Alba, "Using metaheuristics for the location of bicycle stations," *Expert Systems with Applications*, vol. 161, p. 113684, 2020. [Online]. Available: <https://www.sciencedirect.com/science/article/pii/S095741742030508X>
- [15] K. E. Rosing, C. S. ReVelle, and H. Rosing-Vogelaar, "The p-median and its linear programming relaxation: An approach to large problems," *The Journal of the Operational Research Society*, vol. 30, no. 9, pp. 815–823, 1979. [Online]. Available: <http://www.jstor.org/stable/3009503>
- [16] E. Alba and E. Domínguez, "Comparative analysis of modern optimization tools for the p-median problem," *Statistics and Computing*, vol. 16, pp. 251–260, 09 2006.
- [17] T. Drezner and Z. Drezner, "The gravity p-median model," *European Journal of Operational Research*, vol. 179, pp. 1239–1251, 02 2007.
- [18] W. Mu and D. Tong, "On solving large p-median problems," *Environment and Planning B: Urban Analytics and City Science*, vol. 47, no. 6, pp. 981–996, 2020. [Online]. Available: <https://doi.org/10.1177/2399808319892598>
- [19] R. Church and C. ReVelle, "The maximal covering location problem," *Papers of the Regional Science Association*, vol. 32, no. 1, pp. 101–118, 1974. [Online]. Available: <https://doi.org/10.1007/BF01942293>
- [20] N. Megiddo, E. Zemel, and S. L. Hakimi, "The maximum coverage location problem," *Siam Journal on Algebraic and Discrete Methods*, vol. 4, pp. 253–261, 1983.
- [21] M. Fazel Zarandi, S. Davari, and S. Haddad Sisakht, "The large scale maximal covering location problem," *Scientia Iranica*, vol. 18, no. 6, pp. 1564–1570, 2011. [Online]. Available: <https://www.sciencedirect.com/science/article/pii/S1026309811002100>
- [22] V. C. Guzmán, D. A. Pelta, and J. L. Verdegay, "An approach for solving maximal covering location problems with fuzzy constraints," *International Journal of Computational Intelligence Systems*, vol. 9, pp. 734–744, 2016. [Online]. Available: <https://doi.org/10.1080/18756891.2016.1204121>
- [23] M. Bansal, "Planar maximum coverage location problem with partial coverage and general spatial representation of demand and service zones," 09 2017.
- [24] R. Alizadeh, T. Nishi, J. Bagherinejad, and M. Bashiri, "Multi-period maximal covering location problem with capacitated facilities and modules for natural disaster relief services," *Applied Sciences*, vol. 11, no. 1, 2021. [Online]. Available: <https://www.mdpi.com/2076-3417/11/1/397>
- [25] D. Ma, T. Guo, J. Yang, L. He, and K. Tang, "Layout optimization of campus bike-sharing parking spots," *Journal of Advanced Transportation*, vol. 2020, p. 8894119, 2020. [Online]. Available: <https://doi.org/10.1155/2020/8894119>

- [26] J. Yu, Y. Ji, C. Yi, C. Kuai, and D. I. Samal, "Optimization model for the supply volume of bike-sharing: Case study in nanjing, china," *Information*, vol. 12, no. 5, 2021. [Online]. Available: <https://www.mdpi.com/2078-2489/12/5/182>
- [27] Z. Tian, W. Hou, X. Gu, F. Gu, and B. Yao, "The location optimization of electric vehicle charging stations and battery charging behavior," *SIMULATION*, vol. 94, no. 7, pp. 625–636, 2018. [Online]. Available: <https://doi.org/10.1177/0037549717743807>
- [28] S. Yan, C.-K. Lin, and Z.-Q. Kuo, "Optimally locating electric scooter battery swapping stations and battery deployment," *Engineering Optimization*, vol. 53, no. 5, pp. 754–769, 2021. [Online]. Available: <https://doi.org/10.1080/0305215X.2020.1751148>
- [29] S. Deb, X.-Z. Gao, K. Tammi, K. Kalita, and P. Mahanta, "Nature-inspired optimization algorithms applied for solving charging station placement problem: Overview and comparison," *Archives of Computational Methods in Engineering*, vol. 28, pp. 91–106, 2019.
- [30] Y.-W. Chen, C.-Y. Cheng, S.-F. Li, and C.-H. Yu, "Location optimization for multiple types of charging stations for electric scooters," *Applied Soft Computing*, vol. 67, pp. 519–528, 2018. [Online]. Available: <https://www.sciencedirect.com/science/article/pii/S1568494618300978>
- [31] M. J. Alves and J. Clímaco, "A review of interactive methods for multiobjective integer and mixed-integer programming," *European Journal of Operational Research*, vol. 180, no. 1, pp. 99–115, 2007. [Online]. Available: <https://www.sciencedirect.com/science/article/pii/S0377221706002384>
- [32] M. A. Boschetti, V. Maniezzo, M. Roffilli, and A. Bolufé Röhler, "Matheuristics: Optimization, simulation and control," in *Hybrid Metaheuristics*, M. J. Blesa, C. Blum, L. Di Gaspero, A. Roli, M. Sampels, and A. Schaerf, Eds. Berlin, Heidelberg: Springer Berlin Heidelberg, 2009, pp. 171–177.
- [33] M. Ball, "Heuristics based on mathematical programming," *Surveys in Operations Research and Management Science*, vol. 16, pp. 21–38, 01 2011.
- [34] I. Herszterg, "Efficient algorithms for solving multi-objective optimization and large-scale transportation problems," Ph.D. dissertation, Georgia Institute of Technology, Aug. 2020.
- [35] M. Á. Domínguez-Ríos, F. Chicano, and E. Alba, "Improving search efficiency and diversity of solutions in multiobjective binary optimization by using metaheuristics plus integer linear programming," in *Applications of Evolutionary Computation*, P. A. Castillo and J. L. Jiménez Laredo, Eds. Cham: Springer International Publishing, 2021, pp. 242–257.
- [36] H. Xu, W. Fan, T. Wei, and L. Yu, "An or-opt nsga-ii algorithm for multi-objective vehicle routing problem with time windows," in *2008 IEEE International Conference on Automation Science and Engineering*, 2008, pp. 309–314.
- [37] J. J. Durillo, J. Garcia-Nieto, A. J. Nebro, C. A. C. Coello, F. Luna, and E. Alba, "Multi-objective particle swarm optimizers: An experimental comparison," in *EMO 2009*, 2009.
- [38] M. Elarbi, S. Bechikh, L. B. Said, and R. Datta, *Recent Advances in Evolutionary Multi-objective Optimization*. Springer, 2017, ch. Multi-objective Optimization: Classical and Evolutionary Approaches, pp. 1–30.
- [39] Q. Liu, X. Li, H. Liu, and Z. Guo, "Multi-objective metaheuristics for discrete optimization problems: A review of the state-of-the-art," *Applied Soft Computing*, vol. 93, p. 106382, 2020. [Online]. Available: <https://www.sciencedirect.com/science/article/pii/S1568494620303227>
- [40] E. Mezura-Montes and C. A. Coello Coello, "Constraint-handling in nature-inspired numerical optimization: Past, present and future," *Swarm and Evolutionary Computation*, vol. 1, no. 4, pp. 173–194, 2011. [Online]. Available: <https://www.sciencedirect.com/science/article/pii/S2210650211000538>
- [41] S. Luke. (2013) Essentials of metaheuristics. [Online]. Available: <http://cs.gmu.edu/sean/book/metaheuristics/>
- [42] T. Saber, A. Ventresque, J. Marques-Silva, J. Thorburn, and L. Murphy, "Milp for the multi-objective vm reassignment problem," *2015 IEEE 27th International Conference on Tools with Artificial Intelligence (ICTAI)*, pp. 41–48, 2015.
- [43] M. Fischetti and M. Fischetti, *Matheuristics*. Cham: Springer International Publishing, 2016, pp. 1–33. [Online]. Available: [https://doi.org/10.1007/978-3-319-07153-4\\_14-1](https://doi.org/10.1007/978-3-319-07153-4_14-1)
- [44] K. Deb, *Multi-Objective Optimization using Evolutionary Algorithms*. John Wiley and Sons, 2001.
- [45] O. L. de Weck, "Multiobjective optimization : History and promise," 2004.
- [46] N. Jozefowicz, F. Semet, and E.-G. Talbi, "Enhancements of nsga ii and its application to the vehicle routing problem with route balancing," 10 2005, pp. 131–142.
- [47] G. Srivastava, A. Singh, and R. Mallipeddi, "Nsga-ii with objective-specific variation operators for multiobjective vehicle routing problem with time windows," *Expert Systems with Applications*, vol. 176, p. 114779, 2021. [Online]. Available: <https://www.sciencedirect.com/science/article/pii/S0957417421002207>
- [48] K. Bestuzheva, M. Besançon, W.-K. Chen, A. Chmiela, T. Donkiewicz, J. van Doornmalen, L. Eifler, O. Gaul, G. Gamrath, A. Gleixner, L. Gottwald, C. Graczyk, K. Halbig, A. Hoen, C. Hojny, R. van der Hulst, T. Koch, M. Lübbecke, S. J. Maher, F. Matter, E. Mühmer, B. Müller, M. E. Pfetsch, D. Rehfeldt, S. Schlein, F. Schlösser, F. Serrano, Y. Shinano, B. Sofranac, M. Turner, S. Vigerske, F. Wegscheider, P. Wellner, D. Weninger, and J. Witzig, "The SCIP Optimization Suite 8.0," Optimization Online, Technical Report, December 2021. [Online]. Available: [http://www.optimization-online.org/DB\\_HTML/2021/12/8728.html](http://www.optimization-online.org/DB_HTML/2021/12/8728.html)
- [49] J. Blank and K. Deb, "pymoo: Multi-objective optimization in python," *IEEE Access*, vol. 8, pp. 89 497–89 509, 2020.
- [50] L. Kim and O. D. Weck, "Adaptive weighted sum method for multi-objective optimization: a new method for pareto front generation," *Structural and Multidisciplinary Optimization*, vol. 2, no. 31, pp. 105–116, 2006.
- [51] J. Ryu, S. Kim, and H. Wan, "Pareto front approximation with adaptive weighted sum method in multiobjective simulation optimization," in *Proceedings of the 2009 Winter Simulation Conference (WSC)*, Dec 2009, pp. 623–633.
- [52] E. Zitzler and L. Thiele, "Multiobjective evolutionary algorithms: a comparative case study and the strength pareto approach," *IEEE Transactions on Evolutionary Computation*, vol. 3, no. 4, pp. 257–271, Nov 1999.
- [53] E. Zitzler, D. Brockhoff, and L. Thiele, "The hypervolume indicator revisited: On the design of pareto-compliant indicators via weighted integration," in *Evolutionary Multi-Criterion Optimization*, S. Obayashi, K. Deb, C. Poloni, T. Hiroyasu, and T. Murata, Eds. Berlin, Heidelberg: Springer, 2007, pp. 862–876.



**Enrique Gabriel Baquela** is a researcher at Universidad Tecnológica Nacional, Argentina. Dr. in Engineering from Universidad Nacional de Rosario, Argentina. He is an Adjunct Professor at the Facultad Regional San Nicolás from Universidad Tecnológica Nacional. His research focuses on optimization and simulation of complex problems, mainly in the transport and supply chain areas. He has published several book chapters, articles in indexed journals and proceedings of refereed international conferences. ORCID: 0000-0002-8702-3648



**Ana Carolina Olivera** is an Adjunct Researcher at National Council of Scientifics and Technological Researches from the MINCYT, Argentine. Dr. in Computer Science from Universidad Nacional del Sur. She is an Associate Professor at the Facultad de Ingeniería from Universidad Nacional de Cuyo. Her research focuses on metaheuristics and optimization in complex problems. She has published several book chapters, articles in indexed journals and proceedings of refereed international conferences. ORCID: 0000-0001-7825-1959