

Una Metaheurística basada en el Algoritmo Genético de Ordenamiento No-Dominado II, aplicado al Problema de Ruteo de Vehículos de Productos Perecederos

Paula A. Nisperuza, Jorge M. López y Helman E. Hernández

Facultad de Ingeniería, Dpto. de Ingeniería Industrial, Universidad de Córdoba, Montería-Colombia
(e-mail: hhernandez@correo.unicordoba.edu.co)

Recibido May. 6, 2019; Aceptado Jul. 1, 2019; Versión final Ago. 20, 2019, Publicado Dic. 2019

Resumen

Este artículo presenta un nuevo algoritmo basado en el algoritmo genético de ordenamiento no-dominado II (NSGA II) para la distribución de productos perecederos, que minimiza el número de productos dañados y la distancia total recorrida en la entrega de estos. Ambos algoritmos fueron probados en instancias de la literatura de 50 y 100 clientes, para evaluar la calidad y eficiencia de los mismos. Del análisis se obtuvo que el algoritmo propuesto tiene la capacidad de generar mejores resultados comparándolo con el NSGA II para instancias de 50 clientes. Sin embargo, sus soluciones pierden calidad cuando las instancias son de 100 clientes y la complejidad del problema aumenta. Así, los tomadores de decisiones tienen ahora, como herramienta, esta comparación para la planeación de rutas de productos perecederos según el número de clientes.

Palabras claves: algoritmo; ruteo de vehículos; productos perecederos; NSGA II; frontera de Pareto

A Metaheuristic based on Non-Dominated Sorting Genetic Algorithm II, applied to the Vehicle Routing Problem of Perishable Products

Abstract

This article presents a new algorithm based on non-dominated sorting genetic algorithm II (NSGA II) for the distribution of perishable products, which minimizes the number of damaged products and the total distance traveled in the delivery of these. Both algorithms were tested in instances of the literature of 50 and 100 clients, to evaluate the quality and efficiency of the same. From the analysis, it was obtained that the proposed algorithm has the capacity to generate better results comparing it with the NSGA II for instances of 50 clients. However, their solutions lose quality when the instances are 100 clients and the complexity of the problem increases. Therefore, the decision makers have now, as a tool, this comparison for the planning of routes of perishable products according to the number of clients.

Keywords: algorithm; vehicle routing; perishable products; NSGA II; Pareto front

INTRODUCCIÓN

Uno de los más importantes problemas de optimización combinatoria es el problema de transporte conocido como ruteo de vehículos (VRP), (Vigo y Toth, 2002), en los últimos años este problema ha sido bastante estudiado debido a su aplicabilidad en la vida real, por lo que es visto como problema multiobjetivo para estudiar y resolver casos de tipo real donde diversos objetivos se han identificado (Jozefowicz et al., 2008). Existen diversas extensiones y variantes para este problema, dependiendo de la flota y capacidad de los vehículos, los tiempos de entrega permitidos, número de depósitos, entre otros, razón por la que su complejidad aumenta directamente con las variables trabajadas, así por ejemplo, la extensión del VRP conocida como ruteo de vehículos con ventanas de tiempo (VRPTW) en la cual el servicio de cada cliente se da solo dentro de un intervalo de tiempo definido, llamado ventanas de tiempo, posee mayor complejidad que el VRP clásico (Solomon, 1987), el VRPTW se caracteriza por incluir tanto los aspectos espaciales del VRP como los aspectos de tiempo de la programación y así garantizar el cumplimiento de las restricciones de ventanas de tiempo (Solomon y Desrosiers, 1988), el ruteo de vehículos con ventanas de tiempo es uno de los problemas que más se presenta en las empresas de distribución (Brysy y Gendreau, 2005).

La aplicabilidad del VRPTW multiobjetivo trasciende a diversas organizaciones tales como entrega y recogida de productos de supermercados, de basuras, rutas de autobuses, periódicos urbanos, servicios de seguridad, bancos, entre otros, por lo que ha sido necesario desarrollar diversos métodos para su resolución, estos van desde métodos exactos hasta métodos de tipo aproximado, que si bien no arrojan una solución óptima brindan al tomador de decisiones un conjunto de soluciones lo suficientemente buenas (Kallehauge y Solomon, 2005). Generalmente en los problemas multiobjetivo de ruteo de vehículos con ventanas de tiempo el objetivo principal de minimizar los costos es mantenido y luego otros objetivos son agregados, entre esos están los que buscan mejorar la satisfacción de los requerimientos de los clientes, la correcta distribución de bienes y servicios, entre otros (Jozefowicz et al., 2008). En este artículo se propone un nuevo algoritmo multiobjetivo el cual genere soluciones competitivas a los puntos de referencia conocidos para el problema de ruteo de vehículos con ventanas de tiempo.

El problema de ruteo de vehículos juega un papel central en la distribución física y logística de productos (Laporte, 1992), por lo que en la literatura existe un amplio estudio de este problema, sus variedades y los métodos desarrollados para su solución. La extensión de este problema que incluye ventanas de tiempo conocida como VRPTW, es un problema importante que ocurre en muchos sistemas de distribución (Brysy y Gendreau, 2005). La mayoría de los estudios y trabajos tratados como problemas de ruteo de vehículos, son de alta complejidad, debido a que no es posible resolverlos en un tiempo polinomial, están en el grupo de problemas denominados Np- duros (Lenstra y Rinnooy-Kan, 1981). Numerosos métodos han sido propuestos para su solución, existen métodos de tipo exacto donde es posible encontrar una solución óptima, pero requieren grandes tiempos para su ejecución, los cuales aumentan directamente con el número de clientes, por lo que generalmente estos métodos son usados para los problemas de pocos clientes, cuando el número de clientes aumenta la complejidad del problema aumenta, en este caso se utilizan métodos aproximados, estos no garantizan soluciones óptimas, sin embargo, arrojan soluciones muy cercanas a las óptimas en un razonable tiempo computacional. Para una descripción más profunda de los métodos de solución exactos se puede referir a (Laporte y Norbert, 1987; Toth y Vigo, 1998). Así mismo dentro de los métodos de solución aproximados, se encuentran las heurísticas y las metaheurísticas, estas últimas con mejores resultados debido a que pueden escapar de soluciones óptimas locales, realizando una mejor exploración y explotación en el espacio de búsqueda.

Algunos algoritmos de tipo evolutivo han sido muy apropiados para los problemas con múltiples objetivos, toman varias soluciones en una sola ejecución, tal es el caso de Zitzler y Thiele (1998), quienes desarrollaron un algoritmo evolutivo de Pareto fuerte (SPEA), este algoritmo además de la población, mantiene un conjunto de individuos que contiene las soluciones óptimas de Pareto que hayan sido generadas hasta el momento, este conjunto es usado para evaluar la aptitud de un individuo de acuerdo a la relación de dominancia de Pareto, la diversidad se mantiene con el criterio de dominancia, y mediante un método de agrupamiento este conjunto es reducido sin perder sus características de optimalidad. Este algoritmo es mejorado años después por Zitzler et al. (2001), y es llamado SPEA II, en contraste este algoritmo usa una asignación mejorada de la aptitud, donde para cada individuo es conocido el número de individuos dominantes y dominados, se incorpora una técnica para una mejor orientación de búsqueda y finalmente se aplica un nuevo método de truncamiento para limitar el número de soluciones óptimas de Pareto. Buscando solucionar algunas de las dificultades que presentaban las metaheurísticas multiobjetivo, Deb et al. (2000), desarrollan un algoritmo de optimización multiobjetivo conocido como algoritmo genético de ordenamiento no-dominado II (NSGA II), por lo que este algoritmo es menos costoso computacionalmente, utiliza un enfoque elitista para acelerar el rendimiento del algoritmo genético y un enfoque de intercambio sin parámetros.

Por su parte Ombuki et al. (2006), para tratar el problema de VRPTW con enfoque multiobjetivo, plantean un algoritmo genético en el que los objetivos del problema son el número de vehículos y la distancia, utilizando la técnica de clasificación de Pareto, Garcia-Najera y Bullinaria (2011), analizan y proponen un algoritmo multiobjetivo evolutivo para resolver el problema de VRPTW, el cual incluye métodos que miden la semejanza entre las soluciones y de esta forma proporcionan una mejor aproximación al frente de Pareto completo. Dentro de los autores que han tratado el problema de ruteo de vehículos con diversos objetivos y con diferentes métodos, se encuentra Zhang et al., (2012), quienes desarrollaron el algoritmo evolutivo cuántico (MOQEA), este soluciona el problema de ruteo de vehículos considerando la satisfacción del cliente y los costos de viaje, el grado de satisfacción del cliente es representado por una ventana de tiempo difusa, y modelan el problema como un programa lineal entero mixto; Suarez-Chilma et al., (2018) propusieron un modelo multiobjetivo para el VRP, que estudia un balance entre el costo del transporte y el impacto ambiental; Gómez et al., (2019) trabajaron en el desarrollo de metaheurísticos para dar solución al problema de ruteo en la recuperación de medicamentos sobrantes y suministros en hospitales y Sivaram et al. (2014), desarrollaron un algoritmo genético con enfoque agregativo para el problema VRPTW multiobjetivo denominado como algoritmo genético de aptitud agregada (FAGA), entre otros.

Recientemente Podlaski y Wiatrowski (2017), propusieron un nuevo algoritmo evolutivo que usa una versión modificada del operador de cruzamiento de la ruta con mejor costo (BCRC) para encontrar mejores soluciones para el VRPTW. Con relación al VRPTW con productos perecederos, Hsu et al., (2007) estudiaron el problema considerando la aleatoriedad de los procesos en la entrega de productos perecederos; Gonzalez-L. et al., (2015), desarrollaron un modelo para el problema de ruteo de vehículos para productos perecederos, teniendo en cuenta componentes estocásticos tales como la demanda y los tiempos de entrega y servicios; Rabbani et al. (2015), en su estudio propusieron un modelo multiobjetivo no lineal que maximiza tanto los beneficios del distribuidor como la frescura total de los diversos productos que se entregarán a los clientes, esto mediante el método TH, basado en las preferencias del tomador de decisiones; Sirait et al., (2017), analizaron y desarrollaron para el problema de CVRPTW con múltiples depósitos, minimizando los costos totales y a la vez maximizando la satisfacción del cliente, mientras se mantiene la frescura de los productos en su proceso de entrega y Galarcio et al., (2017), desarrollaron una nueva metaheurística aplicada al problema de ruteo de vehículos capacitados (CVRP) para la distribución de productos perecederos. Así mismo, Buelvas et al., (2018), desarrollan una mejora de la metaheurística NSGA II para darle solución al problema de ruteo de vehículos con ventana de tiempos para productos perecederos, donde buscan minimizar la cantidad de producto dañado considerando las condiciones físicas de los caminos por donde estos son transportados, dado que para este problema no han sido propuestos muchos algoritmos y no es posible determinar cuál es la mejor metaheurística para su resolución, la finalidad de este artículo es proponer un algoritmo metaheurístico modificado basado en el NSGA II, el cual pueda obtener compensaciones para los objetivos evaluados en dicho problema, las cuales sean competitivas con las ya mostradas.

EL PROBLEMA Y EL MODELO

Esta sección muestra el modelo del problema de ruteo multiobjetivo con ventanas de tiempo considerando las condiciones físicas de los caminos expuesto por Buelvas et al. (2018). El problema está representado por un grafo direccionado $G=(V,A)$, donde $V=\{0, \dots, N+1\}$ denota el conjunto de vértices, N el conjunto de clientes y $A=\{(i,j) \mid i, j \in V, i \neq j\}$ el conjunto de arcos. Los vértices 0 y $N+1$ representan el depósito, donde una flota homogénea de vehículos de tamaño K con capacidad C son estacionados. Existe una demanda q_i ($q_0=q_{N+1}=0$) no negativa asociada a cada vértice $i \in V$, una ventana de tiempo $[e_i, l_i]$ que no es flexible, la cual representa el tiempo más temprano y más tardío de inicio de servicio en el cliente, y un tiempo de servicio S_{ik} ($S_0=S_{N+1k}=0$). A cada arco se le asocia una distancia d_{ij} y un tiempo de viaje t_{ijk} que indica el tiempo que le toma a un vehículo k ir desde i hasta j . Una tasa de daño P_{ij} que indica el radio de daño producido por unidad de carga en una unidad de tiempo se le asigna a cada arco. La variable decisión x_{ijk} de tipo binario toma el valor de 1 si el conductor k visita el vértice j después del i y 0 de otro modo. Finalmente, el problema cuenta con la variable adicional H_j la cual acumula el porcentaje de productos dañados entregados al cliente j , el daño inicial es considerado cero ($H_0=0$). Con la anterior notación el modelo propuesto es el siguiente:

$$\text{Min } Z = \sum_{j \in N} H_j q_j \quad (1)$$

$$\text{Min } Z = \sum_{i \in V} \sum_{j \in V} \sum_{k \in K} d_{ij} x_{ijk} \quad (2)$$

Sujeto a:

$$\sum_{k \in K} \sum_{\substack{j \in V \setminus \{0\} \\ i \neq j}} x_{ijk} = 1, \quad \forall i \in V \quad (3)$$

$$\sum_{j \in V \setminus \{0\}} x_{0jk} = 1, \quad \forall k \in K \quad (4)$$

$$\sum_{\substack{i \in V \setminus \{N+1\} \\ i \neq j}} x_{ijk} - \sum_{\substack{i \in V \setminus \{0\} \\ i \neq j}} x_{jik} = 0, \quad \forall j \in V, \quad \forall k \in K \quad (5)$$

$$(\tau_{ik} + S_{ik} + t_{ijk} - \tau_{jk}) \leq (1 - x_{ijk}) * M, \quad \forall (i, j) \in A, \forall k \in K \quad (6)$$

$$e_i \leq \tau_{ik} \leq l_i, \quad \forall i \in V, \forall k \in K \quad (7)$$

$$\sum_{i \in V} \sum_{j \in V, i \neq j} q_i x_{ijk} \leq C, \quad \forall k \in K \quad (8)$$

$$H_j \geq H_i + P_{ij} t_{ijk} + (x_{ijk} - 1)M, \quad \forall (i, j) \in A, \forall k \in K \quad (9)$$

$$H_j \geq 0 \quad \forall j \in N \quad (10)$$

$$H_j \leq 1 \quad \forall j \in N \quad (11)$$

$$x_{ijk} \in \{0, 1\}, \quad \forall (i, j) \in A, k \in K \quad (12)$$

$$\tau_{ik} \geq 0, \quad \forall i \in V, \forall k \in K \quad (13)$$

La función objetivo (1) minimiza el número de productos dañados acumulados a lo largo de las rutas y la función objetivo (2) minimiza la distancia recorrida por el vehículo. Las restricciones (3) establecen que cada cliente debe ser atendido exactamente por un solo vehículo. El conjunto de restricciones (4) indican que cada vehículo debe partir desde el depósito. Las restricciones (5) establecen la conservación de flujo en cada uno de los nodos. Los conjuntos de restricciones (6) y (7) representan el cumplimiento de las ventanas de tiempo en cada uno de los nodos. Seguidamente la restricción (8) considera la capacidad cada uno de los vehículos de la flota que se considera homogénea. El grupo de restricciones en (9) acumulan el daño de los productos que llegan a cada cliente en base a los arcos previamente transitados. Finalmente, las restricciones de la (10) a la (13) muestran los dominios de las variables de decisión.

METODOLOGÍA

El modelo es probado para instancias de tamaño de 50 y 100 clientes empleando un método aproximado, con el cual obtenemos un conjunto de soluciones que son lo suficientemente buenas para cada objetivo del problema presentado. El método utilizado es un algoritmo basado en el (NSGAII), el cual al ser aplicado a diversos problemas de optimización ha demostrado ser bastante competitivo generando soluciones bastante eficientes (Deb et al., 2000). El algoritmo presentado en este artículo tiene una estructura similar a la del NSGA II, sin embargo, este difiere en algunos procesos. Tal como se muestra a continuación.

Los primeros procesos de la estructura del algoritmo presentado son: 1) población inicial y codificación: La población inicial es de tamaño N y generada aleatoriamente, cada cromosoma que la conforma está codificado de manera permutada como una cadena de números del 1 hasta el número de clientes, indicando de esta manera el orden en el cual estos son asignados a cada ruta; 2) evaluación y decodificación: Los clientes son asignados a las rutas en el orden permutado de cada solución, si el cliente no cumple alguna restricción de capacidad de carga o ventanas de tiempo, es saltado, es decir no se incluye en la ruta actual, sino que es asignado a una ruta diferente, esta asignación es hecha de manera que todas las rutas sean factibles; y 3) selección y elitismo: Con el fin de guardar siempre los mejores elementos de la población sin hacerle ningún cambio, los individuos son clasificados de acuerdo a su aptitud, se les asigna una posición por su nivel de no dominancia y se realiza el proceso de selección por torneo binario, donde dos individuos son elegidos aleatoriamente y sus posiciones son comparadas, el individuo ganador es aquel que tenga mejor posición, en caso de ambos individuos tengan la misma posición, entonces el ganador es aquel individuo que tenga mayor distancia de apilamiento. Esto mediante dos operadores conocidos como clasificación rápida no dominada y distancia de apilamiento.

Con el objetivo de obtener resultados competitivos a los generados por el NSGA II desarrollado en el trabajo de Buelvas et al. (2018), en este trabajo se toma como estrategia usar de manera consecutiva dos operadores de cruzamiento, cruzamiento de la ruta con mejor costo (BCRC) y cruzamiento del orden de 2 puntos de bloques similares (SB2OX) respectivamente, esto para lograr una mayor explotación dentro del algoritmo y así aumentar la probabilidad de generar mejores soluciones. Así, tiene lugar el proceso de: 4) cruzamiento: el primer operador de cruzamiento utilizado es el mismo usado en el trabajo de referencia (Buelvas et al., 2018), BCRC, este es un operador de cruzamiento especializado (Ombuki et al., 2006; Sivaram et al., 2014), muy bien conocido por sus buenos resultados en el cual: De cada padre se selecciona una ruta aleatoria, luego los clientes de cada ruta elegida son removidos del padre opuesto para así crear los hijos, finalmente, cada uno de los clientes de la ruta eliminada es reinsertado en la mejor posición posible, es decir donde la mejor aptitud sea obtenida, y así crear cada hijo. Si una inserción es no factible entonces una nueva ruta es creada, el segundo operador de cruzamiento es el conocido como SB2OX, en el cual todas las posiciones de cada padre son examinadas, se consideran los bloques con al menos dos cromosomas consecutivos idénticos en ambos padres y solo estos son copiados directamente a los hijos. Luego dos puntos de corte aleatorios son seleccionados en cada padre, la sección que esta entre estos dos puntos de corte es directamente copiada a los hijos. Finalmente, los elementos faltantes en cada hijo son copiados en el orden del padre opuesto (Ruiz et al., 2005).

Los últimos procesos de la estructura del algoritmo son: 5) mutación: la técnica de mutación aplicada es conocida como mutación de intercambio donde dos posiciones de un cromosoma se seleccionan aleatoriamente y sus valores son intercambiados. La probabilidad de que la mutación sea realizada es conocida; y 6) evaluación, ordenamiento y selección de la descendencia generada por la población inicial N : Posteriormente a los pasos mencionados la población de hijos es evaluada. Así, en la iteración los padres P_t y los hijos Q_t son combinados generando una población de tamaño $2N$ llamada R_t , siguiente a esto un procedimiento de clasificación rápida no dominada es aplicado para crear una nueva población P_{t+1} , en orden de menor a mayor posición los frentes son adicionados a la población, en caso de que un frente no pueda ser insertado totalmente en la nueva población, se lleva a cabo un ordenamiento de apilamiento.

EXPERIMENTACIÓN

Con el fin de analizar el comportamiento del algoritmo NSGA II, en conjunto con el algoritmo propuesto en este trabajo, y mostrar que este es capaz de generar soluciones que sean competitivas con las ya obtenidas en el trabajo desarrollado por Buelvas et al., (2018), ambos algoritmos fueron evaluados y probados en algunas de las instancias de Solomon, estas consisten en 3 grupos llamados C, R y RC, cada grupo a su vez, está conformado por 2 subgrupos. Estas instancias distan de acuerdo a la ubicación geográfica de los clientes y a las ventanas de tiempo. De acuerdo a la ubicación, en el conjunto R1 y R2, los datos geográficos fueron generados aleatoriamente, en C1 y C2, los datos fueron agrupados y en los conjuntos RC1 y RC2 son una mezcla de datos aleatorios y agrupados y en relación a las ventanas de tiempo, los grupos R1, C1 y RC1, tienen un horizonte de programación pequeño, mientras que R2, C2 y RC2, tienen un horizonte de programación largo (Errico et al., 2018). De esta manera existe un total de seis conjuntos de problemas formados por los grupos de instancias de Solomon, R1, C1 y RC1, R2, C2 y RC2, para este experimento fue seleccionada aleatoriamente una instancia de cada grupo anteriormente mencionado, esto con el fin de probar ambos algoritmos bajo seis escenarios distintos, es decir bajo las distintas ubicaciones geográficas y horizontes de tiempo que poseen los clientes en las instancias de Solomon, así, para el experimento se tomaron de manera aleatoria seis instancias de 50 clientes y seis instancias de 100 clientes, para un total de 12 instancias. El criterio de corte para los algoritmos aplicados fue el tiempo de ejecución, fueron evaluados en cada una de las instancias seleccionadas por un periodo de ejecución de treinta minutos con 15 réplicas cada uno, así 15 (replicas) $\times 2$ (algoritmos) $\times 12$ (instancias), lo que permitió obtener un total de 360 corridas en la experimentación.

Debido a que el algoritmo con el cual estamos comparando es el NSGA II desarrollado en el trabajo de Buelvas et al. (2018), las probabilidades de cruzamiento, mutación y tamaño de población empleados son los mismos usados en dicho trabajo según el estudio realizado por sus autores, de tal manera que el tamaño de población fue de 100, la probabilidad de cruzamiento fue de 0,95 y la probabilidad de mutación de 0,05. Esto basándose en trabajos donde altas probabilidades de cruzamiento y bajas probabilidades de mutación arrojaban buenos resultados, es decir, resultados competitivos a las soluciones de referencia para las instancias de Solomon para este problema, tal como los obtenidos en los trabajos de (Sivaram et al., 2014; Ombuki et al., 2006). Los métodos utilizados para realizar las comparaciones entre los dos algoritmos, fueron los generalmente usados para comparaciones entre algoritmos multiobjetivo, la comparación entre los gráficos de los frentes de pareto y el uso de métricas de comparación que evalúan aspectos tales como la tasa de error, la cobertura y el espaciado entre las soluciones generadas por cada algoritmo. La experimentación se llevó a cabo utilizando el software Matlab, en un procesador Intel Core i3 y sistema operativo Windows 10.

RESULTADOS

Inicialmente se usó un criterio gráfico con el que fuera posible comparar la calidad de resultados del algoritmo NSGA II y del algoritmo propuesto, mediante este método pudieron observarse las diferencias visuales entre los resultados generados por ambos algoritmos para las instancias escogidas de 50 y 100 clientes. Para las instancias de 50 clientes, se obtuvieron los resultados que se pueden apreciar en la fig. 1.

La figura 1 nos permite observar un frente real aproximado para el problema estudiado, un frente generado por el algoritmo NSGA II y un frente generado por el algoritmo propuesto (NSGA II modificado). Los resultados mostrados en la figura 1 indican que en cada una de las instancias evaluadas el algoritmo propuesto obtiene la mayoría de las mejores compensaciones entre los objetivos estudiados en el problema. En este problema cada objetivo debe ser minimizado por lo que los mejores resultados serán aquellos que muestren los menores valores posibles tanto para la distancia recorrida como para el daño generado en los productos. Por otra parte, para las instancias de 100 clientes, se obtuvieron los resultados mostrados en la figura 2. En las figuras 1 y 2 los símbolos \blacktriangle hacia la izquierda son las soluciones del algoritmo NSGAII, \bullet negro son las soluciones del algoritmo propuesto; y el círculo blanco son las soluciones que conforman el frente real aproximado.

La figura 2 muestra que, para las instancias de 100 clientes los resultados del algoritmo propuesto no fueron muy competitivos, al ser el ruteo de vehículos un problema de complejidad NP-Duro, como se dijo anteriormente, implica que, a mayor tamaño de las instancias, el tiempo computacional necesario para obtener soluciones óptimas es más grande, por lo que cuando el número de clientes aumenta la complejidad del problema aumenta. Solo en la instancia R202 el algoritmo propuesto muestra mejor calidad que el NSGA II, en esta instancia las soluciones pertenecientes al algoritmo propuesto están más cercanas al frente real aproximado del problema.

Como segundo criterio de comparación, evaluación y medición de la calidad de respuesta de ambos algoritmos, se utilizaron tres métricas conocidas para explorar los frentes óptimos de Pareto en problemas de optimización multiobjetivo, y conocer así cual mostraba mayor eficiencia, para mayor información de las métricas se puede consultar en (Yen y He 2014). Las métricas usadas en este trabajo fueron: tasa de error, cobertura y espaciado. La métrica de tasa de error permite conocer el porcentaje de soluciones no dominadas que el algoritmo encontró que no pertenecen al frente de Pareto real aproximado del problema, es cero si todos los valores del frente del algoritmo evaluado pertenecen al frente real aproximado. La métrica de cobertura se aplica para medir qué tan bien está cubierto el frente real verdadero por el frente generado por cada algoritmo, mientras mayor sea el valor de esta métrica mayor es la cobertura que tiene el frente real aproximado por los valores generados por el algoritmo estudiado. Finalmente, la métrica de espaciado evalúa la distribución de los puntos no dominados hallados como solución del problema, si el valor de esta métrica es cero entonces todos los puntos no-dominados obtenidos de cada algoritmo están equiespaciados.

En la tabla 1 la métrica de tasa de error muestra que, en la mayoría de las instancias de 50 clientes, las soluciones no dominadas del frente del algoritmo propuesto pertenecen al frente de Pareto real aproximado del problema, por lo que en todas las instancias a excepción de última (RC201), el algoritmo propuesto posee menor tasa de error, es decir, mayor calidad en las soluciones que el algoritmo NSGA II. La métrica de cobertura muestra que en todas las instancias de 50 clientes los valores generados por el algoritmo propuesto presentan mayor cobertura del frente real aproximado, finalmente para la tercera métrica, espaciado, cuatro de las seis instancias de 50 clientes poseen mejor distribución de los puntos de su frente de Pareto generado. Así también fueron calculadas estas métricas para las instancias de 100 clientes, para cada uno de los algoritmos estudiados, pueden ser observados en la tabla 2.

Tabla 1. Métricas para instancias de 50 clientes: tasa de error, cobertura, espaciado

Instancia	Tasa de error		Cobertura		Espaciado	
	NSGA II	Algoritmo propuesto	NSGA II	Algoritmo propuesto	NSGA II	Algoritmo propuesto
C101	0,9833	0,0000	0,8315	0,8846	173,5285	57,2398
C201	1,0000	0,0000	0,8202	1,0000	81,9125	88,0447
R101	0,9873	0,0097	0,7751	0,9999	101,8868	95,0199
R201	1,0000	0,0000	0,7409	1,0000	126,5363	41,2930
RC101	0,8307	0,1727	0,7995	1,0000	408,5771	205,7775
RC201	0,2352	0,7500	0,8458	0,9258	454,3777	588,4840

En la tabla 2 la métrica de tasa de error muestra que, en la mayoría de las instancias de 100 clientes, las soluciones no dominadas del frente del algoritmo propuesto no pertenecen al frente de Pareto real del problema, por lo que, en todas las instancias a excepción de la cuarta (R202), el algoritmo propuesto posee menor calidad que el NSGA II. La métrica de cobertura muestra que en todas las instancias de 100 clientes

que fueron evaluadas los valores generados por el NSGA II presentan mayor cobertura del frente real aproximado, en ningún caso el algoritmo propuesto tuvo mejor cobertura del frente real aproximado para las instancias de 100 clientes, finalmente para la métrica de espaciado todas las instancias en el NSGA II a excepción de la última (RC202), poseen mejor distribución de los puntos de su frente de pareto generado.

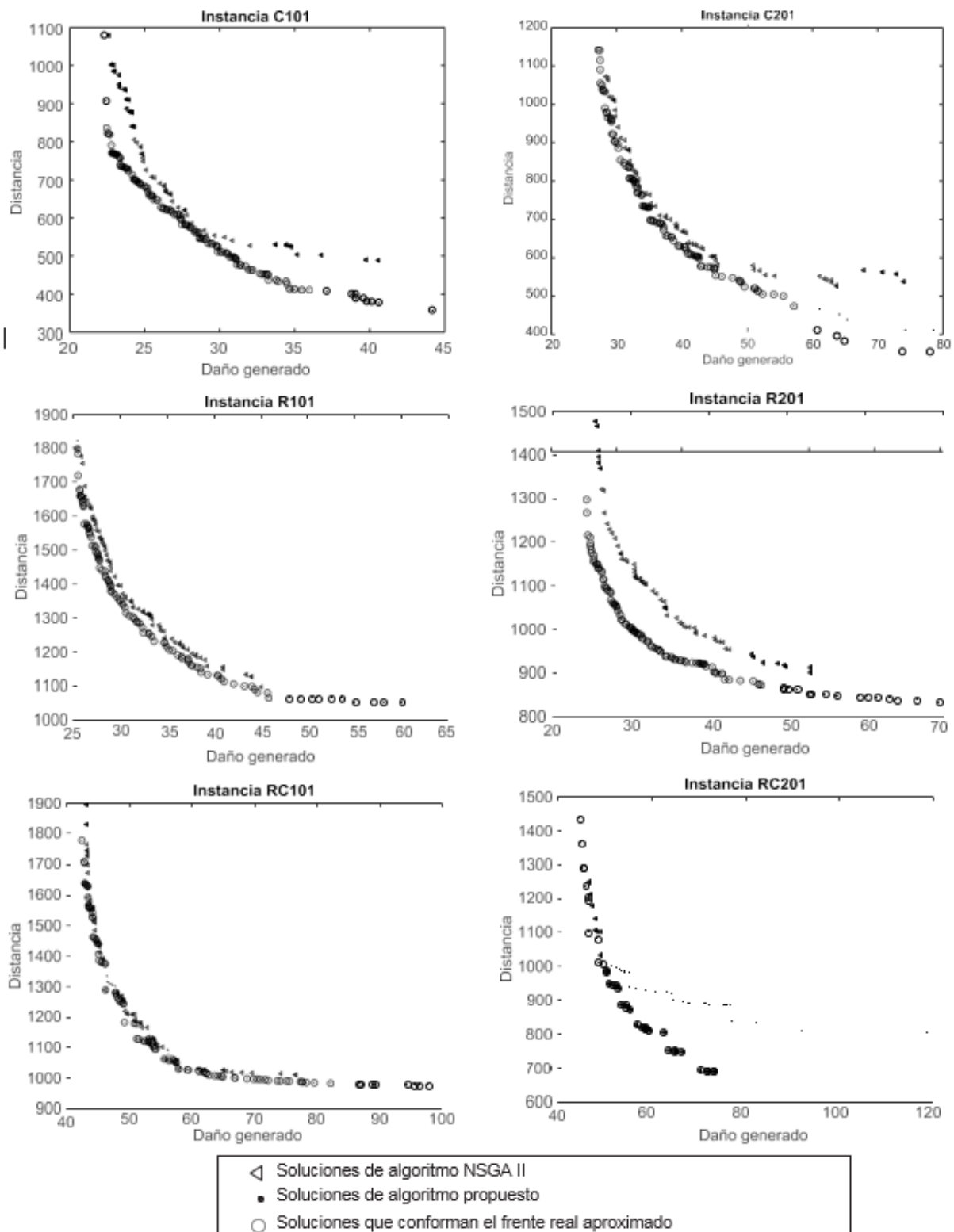


Fig. 1. Fronteras de pareto para instancias de 50 clientes: frente real aproximado, NSGA II, algoritmo propuesto

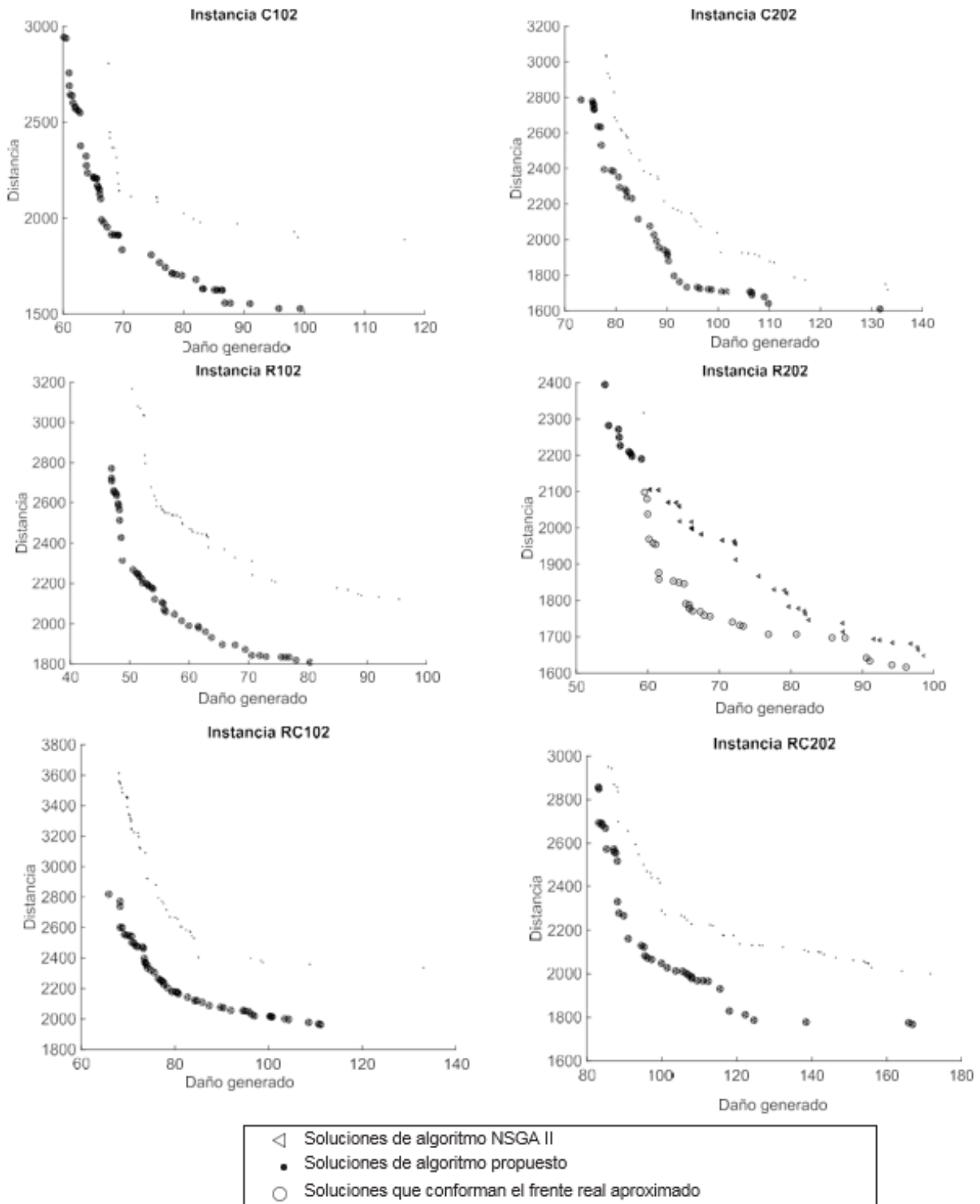


Fig. 2. Fronteras de Pareto para instancias de 100 clientes: frente real aproximado, NSGA II, algoritmo propuesto

Tabla 2. Métricas para instancias de 100 clientes: tasa de error, cobertura, espaciado

Instancia	Tasa de error		Cobertura		Espaciado	
	NSGA II	Algoritmo propuesto	NSGA II	Algoritmo propuesto	NSGA II	Algoritmo propuesto
C102	0,00	1,00	1,0000	0,7345	1473,0685	5665,2710
C202	0,00	1,00	1,0000	0,9116	1093,8064	1270,9423
R102	0,00	1,00	1,0000	0,7929	518,2175	1278,6890
R202	0,70	0,03	0,9798	0,8860	423,8683	1747,3132
RC102	0,00	1,00	1,0000	0,7821	480,3520	1068,6036
RC202	0,00	1,00	1,0000	0,8828	1848,1601	868,7190

CONCLUSIONES

En general, de acuerdo a los resultados, el algoritmo propuesto presentó mejor calidad que el NSGA II para la mayoría de las instancias de tamaño de 50 clientes, pero al aumentar la complejidad del problema (número de clientes), la respuesta del algoritmo perdió eficiencia. Para las instancias con tamaño de 100 clientes el NSGA II mostró mejores resultados que el algoritmo propuesto, indicando que a medida que el problema aumenta de complejidad el algoritmo propuesto pierde calidad y no puede superar los resultados del NSGA II. El NSGA II es un algoritmo que mantiene la calidad, efectividad y eficiencia de sus soluciones al compararse con el algoritmo propuesto, a pesar de que la complejidad aumenta, los resultados de este trabajo comprueban la eficiencia del NSGA II en este tipo de problemas.

Los resultados arrojados por las métricas usadas, permiten afirmar que el algoritmo propuesto presentó menor tasa de error, mayor cobertura y mejor espaciado para la mayoría de instancias de 50 clientes, pero contrariamente, obtuvo mayor tasa de error, menor cobertura y peor espaciado para la mayoría de las instancias de 100 clientes, confirmando que, para problemas de mayor complejidad, es decir, problemas con más de 50 clientes el algoritmo propuesto posee menor calidad que el NSGA II. Fue notable que a pesar de que el NSGA II generó mejor calidad en los resultados en las instancias de 100 clientes, para algunas excepciones como la instancia RC202 Y R202, el algoritmo propuesto obtuvo menor tasa de error y mayor cobertura. Finalmente, aunque el algoritmo propuesto no logró superar los resultados del algoritmo NSGA II para instancias de 100 clientes, mostró ser altamente competitivo para las instancias de 50 clientes.

REFERENCIAS

- Brysy, O. y M. Gendreau, Vehicle Routing Problem with Time Windows, Part I: Route Construction and Local Search Algorithms, doi:10.1287/trsc.1030.0056, Transportation Science, 39(1), 104-118 (2005)
- Buelvas, M., P. Nisperuza, H. Hernandez y J. Lopez, Vehicle Routing Problem for the Minimization of Perishable Food Damage considering Road Conditions, doi:10.23773/2018_2, Logistics Research, 11(2), 1-18 (2018)
- Deb, K., S. Agrawal, A. Pratap y T. Meyarivan, A Fast Elitist Non-dominated Sorting Genetic Algorithm for Mult-objective Optimization: NSGA-II, doi: 10.1007/3-540-45356-3_83, Lecture Notes in Computer Science, vol. 1917. Springer, Berlin, Heidelberg (2000)
- Errico, F., G. Desaulniers y otros tres autores, The Vehicle Routing Problem with Hard Time Windows and Stochastic Service Times, doi:10.1007/s13676-016-0101-4, EURO Journal on Transportation and Logistics, 7(3):223-251 (2018)
- Galarcio, J. D., M. P. Buelvas y otros tres autores, Una Nueva Metaheurística Aplicada Al Problema De Ruteo De Vehículos Capacitados (Cvrp) Para La Distribución De Productos Perecederos, Revista Ingeniería e Innovación, 5(1), 60-72 (2017)
- Garcia-Najera, A. y J. Bullinaria, An Improved Multi-objective Evolutionary Algorithm for the Vehicle Routing problem with time windows, doi: 10.1016/j.cor.2010.05.004, Computers and Operations Research, 38(1), 287-300 (2011)
- Gómez, R. A., F. Salazar y N. Rincón, Metaheurísticos para Solución del Problema de Ruteo en la Recuperación de Medicamentos Sobrantes y Suministros en Hospitales, doi:10.4067/s0718-07642019000200303, Información Tecnológica, 30(2), 303-314 (2019)
- Gonzalez-L., E. C., W. Adarme-Jaimes y J. A. Orjuela-Castro, Stochastic Mathematical Model for Vehicle Routing Problem in Collecting Perishable Products, doi:10.15446/dyna.v82n189.48549, Dyna, 82(189), 199-206 (2015)
- Hsu, C. I., S. F. Hung y H. C. Li, Vehicle Routing Problem with Time-Windows for Perishable Food Delivery, doi:10.1016/j.jfoodeng.2006.05.029, Journal of Food Engineering, 80(2), 465-475 (2007)
- Jozefowiez, N., F. Semet y E. Talbi, Multi-objective Vehicle Routing Problems, doi: 10.1016/j.ejor.2007.05.055, European Journal of Operational Research, 189(2), 293-309 (2008)
- Kallehauge, B., J. Larsen, O. Madsen y M. Solomon, Vehicle Routing Problem with Time Windows, doi: 10.1007 / 0-387-25486-2_3, Column Generation. Springer, 1-33 (2005)
- Laporte, G., The Vehicle Routing Problem: An Overview of Exact and Approximate Algorithms, doi: 10.1016/0377-2217(92)90192-C, European Journal of Operational Research, 59(3), 345-358 (1992)
- Laporte, G. e Y. Norbert, Exact Algorithms for the Vehicle Routing Problem, doi: 10.1016/S0304-0208(08)73235-3. North-Holland Mathematics Studies, 132, 147-184 (1987)
- Lenstra, J. y A. Rinnooy-Kan, Complexity of Vehicle Routing and Scheduling Problems, doi:10.1002/net.3230110211, Networks, 11(2), 221-227 (1981)
- Ombuki, B., B. Ross y F. Hanshar, Multi-objective Genetic Algorithms for Vehicle Routing Problem with Time Windows, doi: 10.1007/s10489-006-6926-z, Applied Intelligence, 24(1), 17-30 (2006)
- Podlaski, K. y G. Wiatrowski, Multi-Objective Optimization of Vehicle Routing Problem Using Evolutionary Algorithm with Memory, doi:10.7494/csci.2017.18.3.1809, Computer Science, 18(3), 271 (2017)
- Rabbani, M., M. Ramezankhani, H. Farrokhi-Asl y Farshbaf-Geranmayeh, Vehicle Routing With Time Windows and Customer Selection for Perishable Goods, ISSN-Online: 2383-2525, International Journals of Supply and Operations

Management, 2(2), 700–719 (2015)

Ruiz, R., C. Maroto y J. Alcaraz, Solving the Flowshop Scheduling Problem with Sequence Dependent Setup Times using Advanced Metaheuristics, doi: 10.1016/j.ejor.2004.01.022, European Journal of Operational Research, 165(1), 34-54 (2005)

Sirait, J. K., S. Suwilo y Mardiningsih, Vehicle Routing Problem for delivery perishable food. Bulletin of Mathematics, 9(1), 9–25 (2017)

Sivaram, V., M. Thansekhar, R. Saravanan y S. Miruna, Solving Multi-objective Vehicle Routing Problem with Time Windows by FAGA, doi: 10.1016/j.proeng.2014.12.461. Procedia Engineering, 97, 2176-2185 (2014)

Solomon, M., Algorithms for the Vehicle Routing and Scheduling Problems with Time Window Constraints, doi:10.1287/opre.35.2.254. Operations Research, 35(2), 254-265 (1987)

Solomon, M. y J. Desrosiers, Survey Paper-Time Window Constrained Routing and Scheduling Problems, doi:10.1287/trsc.22.1.1, Transportation Science, 22(1), 1-81 (1988)

Suarez-Chilma, V. F., W. A. Sarache e Y. J. Costa, Una Solución al Enrutamiento de Vehículos en Ciudades Montañosas Considerando Aspectos Ambientales y Económicos, doi:10.4067/s0718-07642018000300003, Información Tecnológica, 29(3), 3–14 (2018)

Toth, P. y D. Vigo, Exact Solution of the Vehicle Routing Problem, doi: 10.1007/978-1-4615-5755-5_1, Centre for Research on Transportation, Springer, 1-31 (1998)

Vigo, P. y D. Toth, Vehicle Routing Problems, Methods, and Applications, 2ª edición, 1-463, MOS-SIAM Series on Optimization; No. 18, Philadelphia: SIAM (2014)

Yen, G. y Z. He, Performance Metric Ensemble for Multiobjective Evolutionary Algorithms, doi:10.1109/TEVC.2013.2240687, IEEE Transactions on Evolutionary Computation, 18(1), 131–144 (2014)

Zhang, J., W. Wang, Y. Zhao y C. Cattani, Multiobjective Quantum Evolutionary Algorithm for the Vehicle Routing Problem with Customer Satisfaction, doi:10.1155/2012/879614, Mathematical Problems in Engineering, 2012, 1-19 (2012)

Zitzler, E., M. Laumanns y T. Lothar, SPEA2: Improving the Strength Pareto Evolutionary Algorithm, doi: 10.3929/ethz-a-010782581, TIK-Report 103, 1-19 (2001)

Zitzler E. y L. Thiele, An Evolutionary Algorithm for Multiobjective Optimization: The strength Pareto Approach, doi: 10.1.1.40.7696, TIK-Report, 43(43), 1-43 (1998)