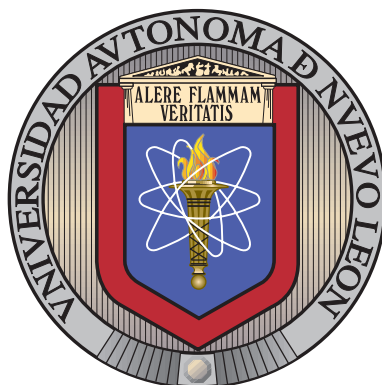


UNIVERSIDAD AUTÓNOMA DE NUEVO LEÓN

FACULTAD DE CIENCIAS FÍSICO-MATEMÁTICAS



UBICACIÓN DE INSTALACIONES CONSIDERANDO
INDICADORES DE ACCESIBILIDAD Y RADIO DE
COBERTURA VARIABLE

POR

DIANNE LARISSA VILICAÑA CERVANTES

EN OPCIÓN AL GRADO DE
MAESTRÍA EN CIENCIAS
CON ORIENTACIÓN EN MATEMÁTICAS

SAN NICOLÁS DE LOS GARZA, NUEVO LEÓN

FEBRERO 2020

UNIVERSIDAD AUTÓNOMA DE NUEVO LEÓN

FACULTAD DE CIENCIAS FÍSICO-MATEMÁTICAS



UBICACIÓN DE INSTALACIONES CONSIDERANDO
INDICADORES DE ACCESIBILIDAD Y RADIO DE
COBERTURA VARIABLE

POR

DIANNE LARISSA VILICAÑA CERVANTES

EN OPCIÓN AL GRADO DE
MAESTRÍA EN CIENCIAS
CON ORIENTACIÓN EN MATEMÁTICAS

SAN NICOLÁS DE LOS GARZA, NUEVO LEÓN

FEBRERO 2020

Universidad Autónoma de Nuevo León
Facultad de Ciencias Físico-Matemáticas

Los miembros del Comité de Tesis recomendamos que la Tesis “Ubicación de instalaciones considerando indicadores de accesibilidad y radio de cobertura variable”, realizada por el alumno Dianne Larissa Villicaña Cervantes, con número de matrícula 1931655, sea aceptada para su defensa como opción al grado de Maestría en Ciencias con Orientación en Matemáticas.

El Comité de Tesis

Dr. Omar Jorge Ibarra Rojas
Director

Dr. Jose Fernando Camacho Vallejo
Revisor

Dra. Martha Selene Casas Ramírez
Revisor

Vo. Bo.

Dr. Omar Jorge Ibarra Rojas
Coordinador de Posgrado en Ciencias con
Orientación en Matemáticas

San Nicolás de los Garza, Nuevo León, Febrero 2020

Para mi familia.

ÍNDICE GENERAL

Agradecimientos	XI
Resumen	XII
1. Introducción	1
1.1. Contexto	1
1.2. Objetivo	3
1.3. Hipótesis	3
1.4. Motivación	3
1.5. Estructura	4
2. Marco Teórico	5
2.1. Problemas de min-sum, p -mediana y máxima cobertura	5
2.2. Diferentes medidas de cobertura	6
2.3. Integración de diferentes filosofías de cobertura	8
3. Definición del problema	10
3.1. Medida de accesibilidad	10
3.2. Formulación matemática del UARV	13
4. Metodología de solución	17
4.1. Algoritmos Genéticos	17

ÍNDICE GENERAL	VI
4.2. BRKGA	18
4.3. BRKGA para UARV	21
4.3.1. Decoder	21
4.3.2. Parámetros del BRKGA	22
5. Resultados	24
5.1. Instancias y condiciones experimentales	24
5.2. BRKGA vs CPLEX con una hora tiempo límite	25
5.3. BRKGA vs CPLEX con 10 minutos de tiempo límite	29
6. Conclusiones	35
6.1. Conclusiones	35
6.2. Trabajo a futuro	36
A. Soluciones de CPLEX 1 hora	38
B. Soluciones de CPLEX 10 minutos	41
C. Soluciones de BRKGA	44

ÍNDICE DE FIGURAS

1.	Solución de una instancia con un conjunto potencial de 3 instalaciones para abrir 2 y con 6 puntos de demanda.	11
2.	Diagrama de flujo del BRKGA.	19
3.	Población en iteración k del algoritmo.	20
4.	Generación de población en iteración $k + 1$ a partir de iteración k	20
5.	Valor de la función objetivo para las soluciones obtenidas usando CPLEX con un criterio de paro de 1 hora o GAP de 1%, y BRKGA con criterio de paro de 100 iteraciones sin mejora.	26
6.	Comparación entre los valores de los diferentes indicadores de accesibilidad para las soluciones obtenidas de CPLEX y BRKGA en todas las instancias.	27
7.	Ejemplo de soluciones de instancias tipo A.	28
8.	Ejemplo de soluciones de instancias tipo B.	29
9.	Ejemplo de soluciones de instancias tipo C.	29
10.	Valor de la función objetivo para las soluciones obtenidas con CPLEX usando un criterio de paro de 10 minutos o GAP de 1%, y BRKGA con criterio de paro de 100 iteraciones sin mejora.	31
11.	Diferencia relativa en los indicadores de accesibilidad en soluciones obtenidas con BRKGA y CPLEX limitado a 10 minutos de tiempo de ejecución.	32
12.	Ejemplo de soluciones de instancias tipo A (CPLEX limitado a 10 minutos de tiempo de computo).	33
13.	Ejemplo de soluciones de instancias tipo B (CPLEX limitado a 10 minutos de tiempo de computo).	33

14. Ejemplo de soluciones de instancias tipo C (CPLEX limitado a 10 minutos de tiempo de computo).	34
--	----

ÍNDICE DE TABLAS

1.	Resumen de los estudios en literatura más relacionados con este trabajo.	9
2.	Resultados promedio de la variación de parámetros del BRKGA usando 5 instancias de cada tipo.	23
3.	Tipos de instancias.	24
4.	Resultados numéricos: CPLEX una hora.	25
5.	Mejora relativa promedio de BRKGA con respecto a CPLEX 1 hora.	27
6.	Resultados numéricos usando el solver CPLEX.	30
7.	Porcentaje de BRKGA mejor que CPLEX.	31
8.	Mejora relativa promedio de BRKGA con respecto a CPLEX 10 minutos.	31
9.	Solución de instancias tipo A usando CPLEX con condición de paro de 1 hora.	38
10.	Solución de instancias tipo B usando CPLEX con condición de paro de 1 hora.	39
11.	Solución de instancias tipo C usando CPLEX con condición de paro de 1 hora.	40
12.	Solución de instancias tipo A usando CPLEX con condición de paro de 10 minutos.	41
13.	Solución de instancias tipo B usando CPLEX con condición de paro de 10 minutos.	42

14.	Solución de instancias tipo C usando CPLEX con condición de paro de 10 minutos.	43
15.	Valores obtenidos para instancias tipo A usando metaheurística BRKGA. .	44
16.	Valores obtenidos para instancias tipo B usando metaheurística BRKGA. .	45
17.	Valores obtenidos para instancias tipo C usando metaheurística BRKGA. .	46

AGRADECIMIENTOS

Quiero dar mi más grande y sincero agradecimiento al Dr. Omar Jorge Ibarra Rojas mi asesor de tesis, quien me apoyó, guió y alentó para concluir este trabajo de investigación.

Agradezco a mis profesores quienes me dieron un gran conocimiento durante estos dos años de maestría. Gracias a mis compañeros Yadira Silva, Alison Garza y Adrian Mendoza por apoyarme en todo momento, tanto personalmente como académicamente.

También quiero agradecer a mi familia, a mi padre Rogelio Villicaña por todo el apoyo moral y económico que me brindó, a mi madre Monica Cervantes quien siempre tuvo un consejo para mi y a mi hermano Roger, quien fue mi motivación para iniciar esta etapa en mi vida.

Por último, agradezco al Consejo Nacional de Ciencia y Tecnología por la beca de manutención brindada durante la maestría.

RESUMEN

En este trabajo se propone una medida de accesibilidad en el contexto de problemas de ubicación de instalaciones donde asumimos que las instalaciones tienen un área de servicio que puede ser incrementada en la toma de decisiones, mientras que los clientes en centroides de demanda tienen un área de movilidad. En particular, nuestra medida es una suma ponderada de los siguientes indicadores: cobertura de los centroides de demanda por el área de servicio de una instalación abierta, oportunidad de los centroides de demanda dentro de su movilidad, impedimento o costo de viaje, el número de oportunidades de un centroide de demanda no cubierto, la distancia de un centroide de demanda no cubierto a la instalación abierta más cercana y segregación espacial.

Con base en lo anterior, definimos nuestro problema de “Ubicación de instalaciones considerando indicadores de accesibilidad y radio de cobertura variable”, el cual se formuló a través de un programa lineal entero-mixto. Una etapa experimental con instancias generadas aleatoriamente nos muestra que el solver comercial de CPLEX no obtiene soluciones de buena calidad en tiempo computacional razonable para instancias grandes. Más aún, para algunos casos no se obtiene solución alguna al agotar la memoria física de la computadora utilizada en la experimentación. Por lo cual, se implementa un algoritmo genético sesgado de llaves aleatorias o BRKGA por sus siglas en inglés “Biased Random-Key Genetic Algorithm”. Resultados experimentales muestran que nuestro BRKGA es una herramienta rápida para obtener soluciones de mejor calidad o comparables con las obtenidas mediante el solver CPLEX.

Palabras clave: ubicación de instalaciones, radio variable, indicadores, accesibilidad, centroides de demanda, cobertura, oportunidad.

CAPÍTULO 1

INTRODUCCIÓN

1.1 CONTEXTO

La accesibilidad es un concepto que comúnmente encontramos al referirse a habitantes de una región que puedan alcanzar un bien o servicio, sea de carácter social (acceso a educación y servicios médicos) o comercial (acceso a restaurantes, comercios, etc.). Entonces, la accesibilidad es de relevancia durante el diseño de una red de servicio (de cualquier tipo) en una región específica. Sin embargo, existen diferentes definiciones del concepto de accesibilidad, las cuales dependen del contexto donde se este desarrollando una investigación (ver revisión de este concepto en Scheurer and Curtis (2007)). Por ejemplo, mostramos las siguientes definiciones abordadas por otros estudios que nos sirven como base para proponer nuestra medida de accesibilidad y nuestro problema de toma de decisiones, a fin de diseñar una red de servicio que sea inclusiva con la población de una región.

- Bhat et al. (2000): “Una medida de la facilidad de un individuo de perseguir una actividad de un tipo deseado, en una ubicación deseada, de un modo deseado, y en un momento determinado.”
- Bertolini et al. (2005): “La cantidad y diversidad de oportunidades que pueden ser alcanzadas dentro de un tiempo y un costo establecido de recorrido”

Algunos elementos importantes de una medida de accesibilidad según Makrí and Folkesson (1999) son: cantidad de oportunidades para que clientes obtengan un bien o servicio, costo de movilidad para la obtención de la demanda y una medida de segregación espacial para tratar que el sistema de servicio sea inclusivo, al no “cortar” regiones del área potencial de servicio.

Para definir nuestra pregunta de investigación, asumimos un contexto de toma de decisiones donde es necesario definir una red de instalaciones para otorgar un bien o

servicio a un conjunto de centroides de demanda, donde los clientes en dichos centroides cuentan con un radio de movilidad conocido. Suponemos que se conoce el conjunto de ubicaciones, donde se desea abrir un subconjunto de ellas, así como determinar cual será su radio de servicio partiendo de un radio mínimo base. Las limitantes en la toma de decisiones son un radio máximo de servicio para cada instalación y un presupuesto para el incremento del radio de servicio de cada instalación con respecto al radio mínimo base. La medida de desempeño a optimizar es la accesibilidad de los clientes al servicio que provee la red de instalaciones determinada en la solución.

El reto de modelar una medida de accesibilidad es considerar indicadores de los elementos que la componen, es decir, determinar cómo definir indicadores para la cantidad de oportunidades, costo de movilidad y segregación espacial. En este trabajo, se propone una medida de accesibilidad que es una suma ponderada de seis indicadores:

- Cobertura: el número de centroides de clientes cubiertos por el área de servicio de las instalaciones (o proveedores) en la solución.
- Acceso mínimo: el número de centroides donde los clientes son capaces de satisfacer su demanda de cualquier forma, ya sea por el área de servicio de las instalaciones, o moviéndose a buscarla a instalaciones dentro de su zona de movilidad.
- Costo de viaje: para los clientes no cubiertos por el área de servicio de los proveedores, este indicador representa un “costo” (sea monetario, en tiempo, en comodidad, etc.) de trasladarse a buscar satisfacer su demanda o servicio.
- Cercanía del servicio: considerando únicamente los centroides de clientes que no pueden satisfacerse, representa las distancias desde cada centroide hasta el proveedor más cercano. La finalidad de que dichos clientes estén cerca de los proveedores es que, si el área de servicio o la zona de movilidad se extiende, puedan satisfacerse.
- Número de oportunidades: Considerando solo los centroides que no fueron cubiertos por el radio de servicio de los proveedores, las oportunidades son la cantidad de proveedores que están dentro de la zona de movilidad de los clientes en dichos centroides (a fin de que puedan satisfacerse buscando el servicio).
- Segregación geográfica: Se relaciona con el grado de dispersión que hay entre los centroides de clientes sin posibilidad de satisfacerse. Mientras mayor sea la dispersión de los centroides sin satisfacerse, es menos probable que se formen zonas marginadas de la red de proveedores.

Con base en la medida de accesibilidad propuesta, se propone un problema de optimización de ubicación de instalaciones con radio de cobertura variable.

1.2 OBJETIVO

Nuestro objetivo general es *mejorar la accesibilidad de un conjunto de clientes en una red de distribución de un bien o servicio*. Para lograr nuestra meta global, definimos los siguientes objetivos específicos:

1. Definir una medida de accesibilidad en el contexto de ubicación de instalaciones con radio de servicio variables.
2. Definir un problema de optimización para modelar la toma de decisiones en el caso de estudio, así como su formulación matemática.
3. Definir una metodología de solución capaz de obtener soluciones de buena calidad en tiempo de computo razonable para tamaños de instancias que pudieran presentarse en escenarios reales.

1.3 HIPÓTESIS

Los conceptos de investigación de operaciones son útiles para modelar el concepto de accesibilidad a fin de determinar una red de instalaciones que provee un servicio a un conjunto de clientes con cierta movilidad mediante un problema de optimización. Además, es posible definir e implementar algoritmos de solución que encuentran soluciones de buena calidad a dichos problemas de optimización.

1.4 MOTIVACIÓN

Considerar el concepto de accesibilidad en decisiones de ubicación de instalaciones puede apoyar el beneficio social y/o comercial al proveer más y mejores oportunidades para proveer un servicio. En particular, características del concepto de accesibilidad pueden modelarse mediante diferentes filosofías de cobertura en problemas de ubicación de instalaciones, tales como cobertura gradual y cobertura cooperativa. Por lo tanto, la modelación de una medida de accesibilidad, así como la solución del problema resultante que optimice dicha medida, son un área de oportunidad en la investigación de operaciones.

1.5 ESTRUCTURA

La estructura de esta tesis es la siguiente: en el Capítulo 2 se presenta una revisión de literatura acerca de ubicación de instalaciones, radio de cobertura variable e indicadores de accesibilidad. El Capítulo 3 presenta la descripción del problema de optimización y su formulación matemática. El Capítulo 4 presenta nuestro método de solución propuesto y en el Capítulo 5 se muestran los resultados obtenidos de la formulación matemática a través de un solver comercial y el método meta-heurístico. Por último, en el Capítulo 6 se presentan las conclusiones y trabajo a futuro de nuestro estudio.

CAPÍTULO 2

MARCO TEÓRICO

En este capítulo se muestra el marco teórico del problema de ubicación de instalaciones considerando indicadores de accesibilidad y radio de cobertura variable. En primer lugar se describe el problema de ubicación, después se habla de los indicadores de accesibilidad y por último, se trata el radio de cobertura variable.

2.1 PROBLEMAS DE MIN-SUM, p -MEDIANA Y MÁXIMA COBERTURA

El enfoque típico para aumentar la accesibilidad de las personas a los servicios es resolver los problemas de optimización minimizando la distancia desde los puntos de demanda a las instalaciones de servicio (p -mediana y problemas de ubicación min-sum) o maximizando el número de puntos de demanda cubiertos por las áreas de servicio de las instalaciones (problema de ubicación de máxima cobertura). Por ejemplo, el estudio de Bruno et al. (2014) presenta una implementación práctica de la solución del problema de la p -mediana en la asignación de escuelas en un caso real en Italia, para el cual se propusieron dos modelos matemáticos que se resolvieron usando el solver CPLEX, con el que se obtienen soluciones en pocos segundos para instancias de 29 escuelas con 9077 alumnos, se hacen de 2 hasta 7 clusters con las escuelas, donde la agrupación mínima de alumnos es de 1000. Stummer et al. (2004) aborda un problema de ubicación de instalaciones capacitadas para departamentos médicos al optimizar la distancia desde los puntos de demanda a las instalaciones abiertas, los costos de instalación, el número de personas no cubiertas debido a la capacidad de las instalaciones y una función de diferencia a partir de una red inicial. El problema multiobjetivo se aborda en dos fases, la primera fase es un procedimiento de búsqueda tabú multiobjetivo que determina el espacio de solución, se considera un “conjunto población” de soluciones potencialmente eficientes que se optimizan simultáneamente, cada solución lleva su propia lista tabú, además, un vector de ponderación individual se modifica antes de cada movimiento de vecindario. La segunda

fase implica explorar el espacio de solución identificado para apoyar a los tomadores de decisiones, la cual se realiza por el método de agrupamiento k-medias. Este método de solución se sobrecarga para instancias grandes, pero logra dar soluciones aceptables en 20 minutos para instancias de 4900 puntos de demanda, 5 hospitales, 3 tipos de departamentos con capacidad de hasta 500 para cada tipo. Doerner et al. (2007a) combina el enrutamiento y la ubicación de máxima cobertura para centros de salud mediante la optimización de los costos de transporte, la distancia desde el punto de demanda hasta la oportunidad más cercana y la fracción de demanda no cubierta al considerar una distancia máxima predefinida, se hace uso de tres algoritmos: optimización de colonias de hormigas, el algoritmo genético evaluado vectorial y el algoritmo genético multiobjetivo. De hecho, el estudio de Daskin and Dean (2004) declara que los problemas de ubicación son relevantes cuando mejoran el acceso de las personas a los servicios, en ambos casos, tanto en la industria como en la atención médica, ya que “*Las implicaciones de las malas decisiones de ubicación se extienden mucho más allá de las consideraciones de costo y servicio al cliente.*” Más recientemente, Boonmee et al. (2017) presenta soluciones a diferentes problemas de ubicación que se utilizan para apoyar el proceso de toma de decisiones en el contexto de la ayuda humanitaria. Los autores revisan diferentes funciones de cobertura que se pueden implementar junto con el clásico problema de ubicación de máxima cobertura, como proponemos en este estudio.

2.2 DIFERENTES MEDIDAS DE COBERTURA

La literatura relacionada con nuestra definición de accesibilidad en problemas de ubicación aborda el concepto de cobertura gradual, donde el nivel de cobertura para un cliente depende de la distancia entre el cliente y la instalación de servicio en lugar de la filosofía de todo o nada del enfoque clásico (ver Drezner et al., 2004). Por ejemplo, el estudio de Eiselt and Marianov (2009) demuestra la existencia de diferentes funciones objetivo en términos de distancia, como la cobertura lineal, la cobertura escalonada y la cobertura no lineal. Los autores extienden el clásico problema de ubicación de cobertura gradual al considerar indicadores tales como los costos de instalación, el número de clientes cubiertos (en el sentido clásico) y el nivel de servicio en la función objetivo a ser optimizada. Las aplicaciones de problemas de ubicación de cobertura gradual pueden ser orientadas a la satisfacción del cliente, problemas de entrega, diseño de red de señal y servicios médicos (vea una comparación entre la cobertura clásica y cobertura gradual en Azizan et al., 2017), entre otros. En nuestro estudio se propone un indicador de costo de movilidad, el cual depende de la distancia entre un centroide de demanda y las instalaciones que son accesibles por los clientes en dicho centroide.

Otro concepto relacionado con nuestra medida de accesibilidad es la cobertura cooperativa, en la cual un centroide de demanda puede ser satisfecho en términos de la cobertura de múltiples instalaciones (tal como se propone con nuestro indicador de oportunidades). Por ejemplo, lo anterior es relevante cuando en una red de servicio, una instalación puede no ser capaz de satisfacer a un cliente y entonces, se necesiten diversas opciones a fin de garantizar dicha satisfacción. En el contexto de cobertura doble en un problema de localización de ambulancias, Gendreau et al. (2001) optimiza la suma ponderada de la fracción de puntos de demanda satisfecha por dos vehículos y el costo de instalación. Además, los autores proponen restricciones de cobertura sencilla y doble para algunos centroides de demanda. Doerner et al. (2005) extiende el estudio de Gendreau et al. (2001) al considerar densidad de población en cada punto de demanda. Más recientemente, Erdemir et al. (2010) minimiza el costo operacional al definir ubicaciones para vehículos de emergencias aéreos y terrestres. El problema considera una restricción dura para la cobertura de todos los centroides de demanda. Un enfoque distinto es presentado por Karatasa et al. (2016), donde un punto de demanda pudiera no satisfacerse cuando una instalación que lo cubre, esta satisfaciendo temporalmente a otros puntos de demanda. Por lo anterior, los centroides de demanda deben ser cubiertos por al menos Q instalaciones. Los autores consideran el objetivo de minimizar la distancia promedio desde los puntos de demanda hasta las instalaciones.

Una característica importante de nuestro problema de optimización, es que se trata de un problema con radio de cobertura variable donde para cada instalación, se determinará el área de servicio que puede satisfacerse. Tal como propone Berman et al. (2009) al minimizar costo de instalaciones y del radio de servicio para satisfacer a un conjunto de puntos de demanda. Jabalameli et al. (2010) define un problema de ubicación de instalaciones con radios de cobertura variable con el objetivo de maximizar la cantidad de puntos de demanda cubiertos y minimizar los costos de transportación y radio de servicio. Los autores implementan solver comercial en una versión mono-objetivo del problema, donde una de las funciones objetivos es acotada en el conjunto de restricciones y se varía estratégicamente el valor de dicha cota. Se muestran resultados experimentales solo para un par de ejemplos ilustrativos. Más recientemente, Eydi and Torabi (2019) proponen una formulación entera-mixta lineal bi-objetivo para determinar la cantidad y capacidad para fábricas (en términos del radio de servicio) y centros de distribución. Además, también se determinan los modos de transporte en la red de distribución. Los autores definen un problema multiobjetivo orientado a optimizar los costos de transporte y la cantidad de puntos cubiertos. Se implementa un ε -constraint method y un NSGA II para obtener aproximaciones al Frente de Pareto, donde resultados numéricos muestran que instancias medianas y grandes solo son tratables por el meta-heurístico propuesto.

2.3 INTEGRACIÓN DE DIFERENTES FILOSOFÍAS DE COBERTURA

El estudio de Bagherinejad et al. (2018) indica que la implementación de los distintos conceptos de cobertura puede apoyar en la toma de decisiones en el contexto de ubicación de instalaciones. Por lo tanto, los autores integran cobertura gradual y cooperativa en una formulación matemática, la cual puede ser reducida a la versión clásica de cobertura gradual y cooperativa al fijar algunos de sus parámetros. Una etapa experimental implementando dos metaheurísticos (ya que el problema es intratable por solver comercial) muestra que el enfoque integral cubre más puntos de demanda en los escenarios donde las instalaciones tienen los mismos niveles de servicio. Un enfoque integral multiobjetivo es definido por Forghani Youshanlo and Sahraeian (2015), donde se maximiza cobertura simple y doble y se minimiza el costo total (en términos de ubicación, transporte y satisfacción de la demanda). Los autores resuelven la versión mono-objetivo del problema usando un orden lexicógrafo para instancias generadas aleatoriamente. Más recientemente, Karatas (2017) también presenta, un enfoque multiobjetivo que integra cobertura gradual y cooperativa, además de considerar un radio de servicio variable para las instalaciones en la solución. Los autores optimizan una suma ponderada de objetivos basada en puntos cubiertos, costo total y balance en el nivel de servicio de las instalaciones abiertas. Resultados numéricos de implementar un enfoque secuencial en dos etapas muestran que es posible obtener soluciones de gran calidad para instancias de hasta 200 instalaciones y 3000 centroides de demanda en un tiempo de computo promedio de menos de 4 horas.

En resumen, diferentes conceptos de cobertura se han definido en literatura y la integración de ellos puede conducir a mejoras en el contexto de diseñar una red de oferta sea en el contexto comercial, como para servicios sociales. Nuestro trabajo es una extensión de Ibarra-Rojas et al. (2019), donde agregamos el supuesto de radio de cobertura variable, lo cual hace que el problema sea más intratable, a diferencia del caso de dichos autores, donde la formulación matemática propuesta puede ser resuelta mediante solver comercial. Según nuestra búsqueda de literatura, no se encontró ningún estudio en problemas de ubicación de instalaciones que use los diferentes conceptos de cobertura para definir una medida de accesibilidad como la propuesta en Ibarra-Rojas et al. (2019) y en este estudio.

	p -mediana	cobertura	cooperativo	gradual	radio variable	Objetivo
Bruno et al. (2014)	✓					Distancia promedio desde los centroides de demanda a la instalación abierta asignada;
Doerner et al. (2007b)	✓	✓				Suma pondera de longitud de ruta, distancia de centroides de demanda a puntos de la ruta y la cantidad de habitantes fuera de un radio de servicio dado.
Eiselt and Marianov (2009)				✓		Costos de transporte y nivel de servicio
Berman et al. (2009)		✓			✓	Costo total
Eydi and Torabi (2019)		✓		✓	✓	Costos de transporte y cantidad de puntos cubiertos (bi-objetivo)
Ibarra-Rojas et al. (2019)		✓	✓	✓		Medida de accesibilidad usando indicadores de cobertura clásica, costo de viaje, cantidad de oportunidades, distancia a la red de servicio y segregación geográfica
este estudio		✓	✓	✓	✓	Medida de accesibilidad usando indicadores de cobertura clásica, costo de viaje, cantidad de oportunidades, distancia a la red de servicio y segregación geográfica

Tabla 1: Resumen de los estudios en literatura más relacionados con este trabajo.

CAPÍTULO 3

DEFINICIÓN DEL PROBLEMA

En este capítulo se presenta la definición del problema de ubicación de instalaciones considerando indicadores de accesibilidad y radio de cobertura variable y su formulación matemática. Primero se describen a detalle los conjuntos y parámetros de entrada así como las decisiones a fin de definir nuestra medida de desempeño a considerar en el problema de optimización, para finalmente introducir nuestra formulación matemática que consiste en un programa entero-mixto lineal.

3.1 MEDIDA DE ACCESIBILIDAD

Dado un conjunto I de potenciales ubicaciones, se abrirá un subconjunto de n instalaciones en distintas ubicaciones. Para cada instalación i se decidirá su radio de cobertura (o de servicio) partiendo de un radio base rc_i y considerando un incremento máximo ϵ_i en el radio de cobertura, así como un incremento total de ℓ para todas las instalaciones, es decir, la suma de todos los incrementos de cada instalación está limitada (como un presupuesto). Sea J el conjunto de centroides de demanda y sea ra_j el radio de movilidad de las personas en el punto $j \in J$. Considerando los conjuntos I y J definimos d_{ij} como la distancia de la instalación $i \in I$ al punto de demanda $j \in J$. Además, definimos $A(j)$ como el conjunto de todas las ubicaciones i desde donde puede satisfacerse a las personas de la zona j asumiendo el máximo radio de cobertura de dichas ubicaciones, es decir, $A(j) = \{i \in I : d_{ij} \leq rc_i + \epsilon_i + ra_j\}$. Entonces, nuestras decisiones son las ubicaciones para abrir las n instalaciones y su respectivo radio de servicio. Bajo este concepto de toma de decisiones, si tenemos una solución, es decir n instalaciones abiertas dentro de las ubicaciones I , con radio de servicio ρ_i , implícitamente definimos los conjuntos $I_1 \subset I$ e $I_2 \subset I$ de instalaciones abiertas y cerradas, respectivamente. De manera similar, existen $J_1 \subset J$ y $J_2 \subset J$ de puntos de demanda cubiertos y no cubiertos por el área de servicio de las instalaciones en I_1 , respectivamente.

La Figura 1 nos muestra un ejemplo de tres potenciales ubicaciones (denotados como rectángulos) y seis puntos de demanda (denotados por círculos). El círculo punteado de las instalaciones, representa su radio de cobertura base, el círculo azul representa el radio de cobertura de las instalaciones abiertas ($I_1 = 1, 3$), el círculo en rojo alrededor de los puntos de demanda no cubiertos ($J_2 = 2, 3, 4, 6$), representa su radio de movilidad. Note que el punto de demanda 1 está cubierto por la instalación 1, mientras que el punto de demanda 5 está cubierto por la instalación 3. Decimos que un punto de demanda j tiene una *oportunidad* en una instalación abierta $i \in I_1$ si puede satisfacerse a las personas de dicha zona considerando el radio de cobertura base y la decisión del incremento asignado a esa instalación, entonces los puntos de demanda 2 y 3 tiene oportunidad con la instalación 1, mientras que el los punto de demanda 4 y 2 tienen oportunidad con la instalación 3. Por último, el punto de demanda 6 no tiene ninguna oportunidad, ya que la instalación 2 no está abierta.

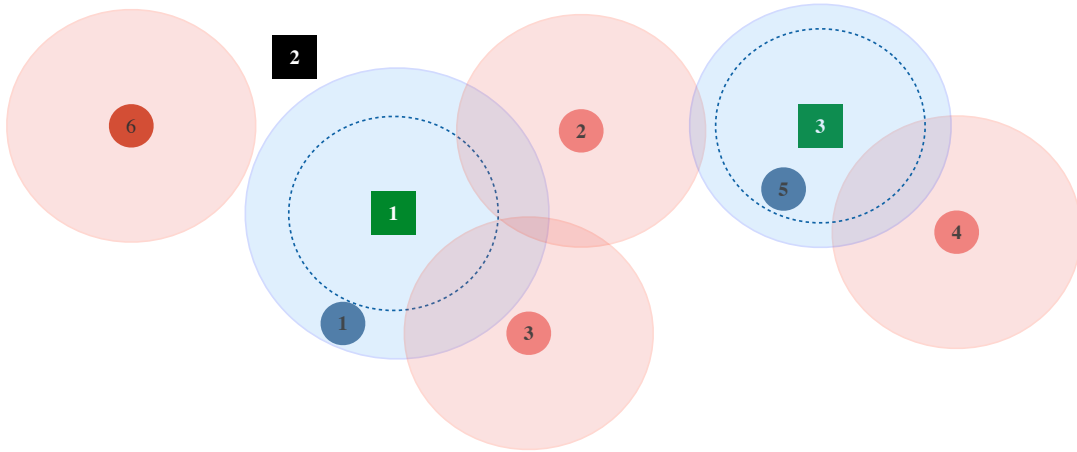


Figura 1: Solución de una instancia con un conjunto potencial de 3 instalaciones para abrir 2 y con 6 puntos de demanda.

Con base en anterior, proponemos usar una medida de accesibilidad similar a la propuesta por Ibarra-Rojas et al. (2018), la cual integra los indicadores de accesibilidad, cobertura de puntos de demanda, costo de viaje, número de oportunidades, oportunidad más cercana y segregación espacial. En particular, dados los conjuntos I_1 , I_2 , J_1 y J_2 , se definen formalmente los siguientes indicadores.

El indicador ac_j toma el valor de 1 si el punto de demanda está cubierto por el área de servicio, y 0 de otra manera. Este es el indicador clásico de problemas de cobertura

absoluta (cubierto o no cubierto).

$$ac_j = \begin{cases} 1 & \text{si } j \in J_1 \\ 0 & \text{en otro caso} \end{cases}$$

El indicador aa_j toma el valor de 0 si las personas en el punto de demanda j no tienen acceso al servicio, mientras que toma el valor de 1 si las personas pueden obtener servicio, ya sea directamente por el radio de cobertura de alguna instalación abierta, o porque su radio de movilidad se interseca con al menos un radio de servicio de las instalaciones abiertas (tiene al menos una oportunidad).

$$aa_j = \begin{cases} 1 & \text{si } j \in J_1, \text{ o } d_{ij} \leq \rho_i + ra_j \text{ para algún } i \in I_1 \\ 0 & \text{en otro caso} \end{cases}$$

También utilizamos el indicador de costo de viaje at_j para que las personas en el centroide j satisfagan su demanda. Comúnmente, el costo de viaje de un punto j a un punto i , puede representarse como una función inversa de la distancia, es decir $f(d_{ij}) = \frac{1}{d_{ij}}$ (ver Makrı and Folkesson, 1999). Entonces, nuestro indicador $at_j \in (0, 1)$ combina los inversos de las distancias considerando todas las potenciales oportunidades en el conjunto $A(j)$. Note que al maximizar el valor de aa_j , se intenta garantizar las oportunidades mas cercanas al centroide j .

$$at_j = \begin{cases} 0 & \text{si } j \in J_1 \\ \frac{\sum_{i \in A(j) \cap I_1} f(t_{ij})}{\sum_{i \in A(j)} f(t_{ij})} & \text{si } j \in J_2 \end{cases}$$

El indicador an_j representa la distancia de viaje desde el punto j no cubierto hasta la instalacion abierta mas cercana, la cual intenta minimizarse para acercar a los clientes a la red de servicio. En realidad, este es un indicador clasico en los problemas de la p -mediana.

$$an_j = \begin{cases} 0 & \text{if } j \in J_1 \\ \{\text{mın } d_{ij} : i \in I_1\} & \text{si } j \in J_2 \end{cases}$$

Con la finalidad de no solo tomar en cuenta la cercanıa de las oportunidades, tambien implementamos un indicador ao_j que representa el numero de oportunidades para las

personas en un centroide de demanda j (no cubierto por instalaciones abiertas).

$$ao_j = \begin{cases} 0 & \text{si } j \in J_1 \\ |A(j) \cap I_1| & \text{si } j \in J_2 \end{cases}$$

Finalmente, definimos un indicador as de segregación espacial, como la distancia mínima entre todos los pares de puntos de demanda sin oportunidad. Maximizamos este último indicador para evitar una situación en la que los puntos de demanda sin acceso queden cerca unos de otros, lo cual representaría una zona marginada.

$$as = \min_{j,j':aa_j=aa_{j'}=0} \{d_{jj'}\}$$

La medida de accesibilidad para los conjuntos J_1 y J_2 es definida como la siguiente suma ponderada de los indicadores propuestos anteriormente, cuyos parámetros β_k representan los pesos.

$$A(J_1, J_2) = \beta_1 \sum_{j \in J} aa_j + \beta_2 \sum_{j \in J} ac_j + \beta_3 \sum_{j \in J} at_j + \beta_4 \sum_{j \in J} an_j + \beta_5 \sum_{j \in J} ao_j + \beta_6 as.$$

Con base en lo anterior, el problema de Ubicación de instalaciones considerando indicadores de Accesibilidad y Radio de cobertura Variable (UARV) *determina las n instalaciones que se abrirán entre el conjunto I de potenciales ubicaciones, así como el incremento de sus radios de cobertura, de modo que la medida accesibilidad $A(J_1, J_2)$ basada en la cobertura, oportunidad, impedimento, oportunidad cercana, número de oportunidades y segregación espacial sea maximizada.* Ahora se detalla la de formulación matemática propuesta.

3.2 FORMULACIÓN MATEMÁTICA DEL UARV

Dada la naturaleza de nuestro problema de ubicación de instalaciones considerando indicadores de accesibilidad y radio de cobertura variable, se proponen las siguientes variables de decisión donde (3.1) y (3.2) son las variables principales y (3.3), (3.4) y (3.5)

son variables auxiliares.

$$z_i = \begin{cases} 1 & \text{si la instalación } i \text{ está abierta} \\ 0 & \text{de otra manera} \end{cases} \quad (3.1)$$

$$x_i : \text{ampliación del radio de cobertura de instalación en } i \in I. \quad (3.2)$$

$$u_{ij} = \begin{cases} 1 & \text{si } i \text{ está abierta y cubre a } j \\ 0 & \text{de otra manera} \end{cases} \quad (3.3)$$

$$w_{ij} = \begin{cases} 1 & \text{si } i \text{ está abierta y es una oportunidad para } j \\ 0 & \text{de otra manera} \end{cases} \quad (3.4)$$

$$v_{ij} = \begin{cases} 1 & \text{si el punto de demanda } j \text{ no está cubierto e } i \text{ es la instalación} \\ & \text{abierta más cercana, es decir, } z_i = 1 \\ 0 & \text{de otra manera} \end{cases} \quad (3.5)$$

Se realiza una normalización de forma que los términos en la suma ponderada de indicadores de accesibilidad tengan un valor mínimo de 0 y un valor máximo de 1. En particular, para los indicadores aa_j , ac_j , at_j , cuyos valores están entre 0 y 1, simplemente se divide entre la cantidad de centroides de demanda $|J|$. De esta forma, la suma de cada indicador también está entre 0 y 1. Respecto al indicador an_j el cual desea minimizarse, definimos un indicador auxiliar $\frac{\max_{i \in I}\{d_{ij}\} - an_j}{\max_{i \in I}\{d_{ij}\}}$, que al ser maximizado, reduce el valor de an_j . En efecto, los valores de este indicador auxiliar están entre 0 y 1, por lo que también es dividido entre la cantidad de centroides de demanda para normalizar la suma. En el caso de la suma de la cantidad de oportunidades para cada centroide j , se divide ao_j entre la cantidad máxima de oportunidades, es decir, $A(j)$. Además, el resultado se divide entre $|J|$. Finalmente, el indicador as se divide entre la máxima distancia entre ubicaciones y centroides de demanda. Entonces, nuestro programa lineal entero-mixto para el problema UARV, se define como sigue.

$$\max \sum_{j \in J} \left(\beta_1 \frac{aa_j}{|J|} + \beta_2 \frac{ac_j}{|J|} + \beta_3 \frac{at_j}{|J|} + \beta_4 \frac{\frac{\max_{i \in I}\{d_{ij}\} - an_j}{\max_{i \in I}\{d_{ij}\}}}{|J|} + \beta_5 \frac{\frac{ao_j}{|A(j)|}}{|J|} \right) + \beta_6 \frac{as}{\max_{j, j' \in J}\{d_{ij}\}} \quad (3.6)$$

sujeto a:

$$\sum_{i \in I} z_i = n \quad (3.7)$$

$$\sum_{i \in I} x_i \leq \ell \quad (3.8)$$

$$x_i \leq \varepsilon_i z_i \quad \forall i \in I \quad (3.9)$$

$$u_{ij} \leq z_i \quad \forall i \in I, j \in J \quad (3.10)$$

$$(rc_i + x_i) - d_{ij} \leq M(u_{ij} + 1 - z_i) \quad \forall i \in I, j \in J \quad (3.11)$$

$$d_{ij} - (rc_i + x_i) \leq M(1 - u_{ij}) \quad \forall i \in I, j \in J \quad (3.12)$$

$$w_{ij} \leq z_i \quad \forall i \in I, j \in J \quad (3.13)$$

$$(rc_i + x_i + ra_j) - d_{ij} \leq M(w_{ij} + 1 - z_i) \quad \forall i \in I, j \in J \quad (3.14)$$

$$d_{ij} - (rc_i + x_i + ra_j) \leq M(1 - w_{ij}) \quad \forall i \in I, j \in J \quad (3.15)$$

$$\sum_{i \in I} u_{ij} \leq M \cdot ac_j \quad \forall j \in J \quad (3.16)$$

$$ac_j \leq \sum_{i \in I} u_{ij} \quad \forall j \in J \quad (3.17)$$

$$v_{ij} \leq z_i \quad \forall i \in I, j \in J \quad (3.18)$$

$$\sum_{i \in I} v_{ij} = (1 - ac_j) \quad \forall j \in J \quad (3.19)$$

$$aa_j \leq ac_j + \sum_{i \in I} w_{ij} \quad \forall j \in J \quad (3.20)$$

$$at_j \leq (1 - ac_j) \quad \forall j \in J \quad (3.21)$$

$$at_j \leq \frac{\sum_{i \in I} \frac{w_{ij}}{d_{ij}}}{\sum_{i \in A(j)} \frac{1}{d_{ij}}} \quad \forall j \in J \quad (3.22)$$

$$ao_j \leq M(1 - ac_j) \quad \forall j \in J \quad (3.23)$$

$$ao_j \leq \sum_{i \in I} w_{ij} \quad \forall j \in J \quad (3.24)$$

$$an_j \leq \sum_{i \in I} v_{ij} d_{ij} \quad \forall j \in J \quad (3.25)$$

$$as \leq d_{jj'} + M(aa_j + aa_{j'}) \quad \forall j, j' \in J, j' \neq j \quad (3.26)$$

$$x_i \geq 0 \quad \forall i \in I \quad (3.27)$$

$$z_i, u_{ij}, w_{ij}, v_{ij} \in \{0, 1\} \quad \forall i \in I, j \in J \quad (3.28)$$

La función objetivo (3.6) representa nuestra medida de accesibilidad normalizada. La restricción (3.7) limita el número de instalaciones para ser abiertas. Restricción (3.8) acota

el incremento total de radio de las instalaciones. Restricción (3.9) limita el incremento para cada instalación abierta. Las restricciones (3.10), (3.11) y (3.12) garantizan que un punto de demanda j está cubierto si está dentro del radio de servicio de alguna instalación abierta i . Las restricciones (3.13), (3.14) y (3.15) garantizan que un punto de demanda j tiene oportunidad en una instalación abierta i si existe una intersección entre su radio de movilidad y el radio de servicio de la instalación i . Restricciones (3.16) y (3.17) fuerza al indicador ac_j a ser 1 cuando la zona j está cubierta por al menos una instalación. Las restricciones (3.18) y (3.19) garantizan que cada zona j se le asocie una instalación abierta más cercana. La restricción (3.20) garantiza que aa_j toma el valor de 1 si la zona está cubierta ($ac_j = 1$) o tiene al menos una oportunidad $i \in I_1$ dentro de su radio de movilidad $A(j)$, y 0 en caso contrario. Restricciones (3.21) y (3.22) definen el valor del indicador at_j . Observe que al maximizar nuestra medida de accesibilidad, tratamos de definir oportunidades cercanas para cada zona j no cubierta, es decir, activar las variables w_{ij} con valores pequeños de d_{ij} cuando $ac_j = 0$. Las restricciones (3.23) y (3.24) definen el número de oportunidades que tiene un punto de demanda j si no está cubierto. La restricción (3.25) define la distancia a la instalación abierta más cercana de una zona no cubierta. La restricción (3.26) define la distancia mínima entre zonas no cubiertas, Por último, las restricciones (3.27) y (3.28) indican la naturaleza numérica de las variables de decisión. Se resalta que el parámetro M es un valor positivo lo suficiente mente grande para al momento de multiplicarlo por un valor no cero (resultado de operar con las variables de decisión) no halla un número dentro de la formulación más grande.

En la sección de resultados experimentales mostramos que es posible obtener soluciones de buena calidad para instancias del UARV, de hasta 80 ubicaciones, para definir 10 instalaciones abiertas y 250 puntos de demanda mediante la implementación de solver comercial. Sin embargo, la formulación propuesta es intratable para instancias más grandes por ejemplo 200 potenciales ubicaciones con 600 puntos de demanda, abriendo 25 instalaciones, por lo que proponemos el uso de una metaheurística para obtener soluciones de buena calidad en tiempo de cómputo aceptable. Sin embargo, no se demostró formalmente la complejidad del problema.

METODOLOGÍA DE SOLUCIÓN

En esta sección se explica la metodología de solución propuesta para nuestro problema ubicaciones considerando indicadores de accesibilidad y radio de cobertura variable, al cual llamamos UARV.

Experimentación preliminar usando el solver de CPLEX 12.9 para la formulación propuesta del UARV, muestra que no es posible encontrar soluciones factibles para instancias grandes en un tiempo de computo de 3 horas. Por esta razón, se propone solucionar el problema utilizando un algoritmo metaheurístico. En particular, se implementan un algoritmo genético de llaves aleatoria sesgado (BRKGA por sus siglas en inglés) para obtener soluciones de buena calidad en un tiempo computacional razonable.

4.1 ALGORITMOS GENÉTICOS

Los algoritmos genéticos son metaheurísticas basadas en el proceso de selección natural, inspirada en el concepto de que sobrevive el que este mejor adaptado al medio, sirve para encontrar soluciones óptimas o soluciones cercanas al óptimo de problemas de optimización. El algoritmo produce una serie de generaciones donde el individuo más apto de la última generación es la solución. Cada individuo tiene un cromosoma correspondiente que codifica la solución. Un cromosoma es una cadena de genes que toman un valor. Estos algoritmos, generalmente emplean operadores iterativos de selección, cruce y mutación para formar nuevos individuos quienes serán parte de la próxima población para encontrar los mejores individuos de acuerdo con una función “fitness”. La selección se realiza de forma aleatoria dando preferencia a los individuos más aptos según la función fitness. El cruce sirve para crear nuevas soluciones combinando elementos de soluciones padres (dos individuos diferentes). Por último, la mutación es aleatoria y sirve para escapar de óptimos locales.

El Algoritmo 1 nos muestra un pseudocódigo base para los algoritmos genéticos.

Algorithm 1: Algoritmo Genético**Data:** Instancia de problema $P = \{\text{máx } f(x) : x \in \mathcal{X}\}$ **Result:** Solución $x' \in \mathcal{X}$ Genera una población pob de soluciones y calcula función “fitness”;**while** (*no se cumpla criterio de paro*) **do** Selecciona $padres \subset pob$; $hijos \leftarrow Cruza(padres)$; $hijos \leftarrow Muta(hijos)$; $pob \leftarrow Actualiza(pob, hijos, fitness)$; x' es la solución de la última población pob con mejor valor de la función fitness;

Un factor importante de los algoritmos genéticos es la representación de la solución lo cual recibe el nombre de codificación. En literatura se distinguen diversos tipos de codificación de una solución, como lo son vectores de valores reales binarios o enteros (vea Talbi, 2009). En realidad la generación aleatoria de soluciones factibles y la definición de los operadores de cruce, de mutación y de función fitness dependen de como sea la decodificación de la solución

4.2 BRKGA

A diferencia de los algoritmos genéticos convencionales, BRKGA siempre utiliza una codificación de solución como llave aleatoria (random-key), la cual se define como un vector de números reales entre 0 y 1. En este algoritmo, a cada llave aleatoria se le asocia a una solución factible del problema mediante un algoritmo determinístico llamado “decoder”. En realidad, el decoder es una función que mapea el espacio de llaves aleatorias al espacio de soluciones factibles del problema de optimización. La ventaja de estas dos características, es que la generación aleatoria de llaves aleatorias es simple y por ende, la generación aleatoria de soluciones factibles. La dificultad está precisamente, en el diseño del decoder, que depende del problema a resolver.

BRKGA es un algoritmo elitista que mantiene un pequeño grupo “elite” de las mejores soluciones de una iteración a otra. El resto de las soluciones son consideradas como población no elite. La selección de los padres para la cruce de soluciones se realiza eligiendo un individuo aleatoriamente del grupo de soluciones elite y uno de las soluciones no elite. La cruce se realiza simplemente heredando cada elemento de la llave aleatoria de alguno de los padres, y las soluciones elite tienen una mayor probabilidad de heredar sus características a la solución hijo. La mutación se realiza introduciendo un pequeño número de nuevos vectores de llaves aleatorias a la población.

La Figura 2 muestra el diagrama de flujo del algoritmo BRKGA donde se muestra que existen dos tipos de componentes del BRKGA, los componentes dependientes e independientes del problema. Los operadores de selección, cruce y mutación están pre-especificados y funcionan de la misma manera para cualquier problema de optimización. Por otro lado, el único componente dependiente del problema es el decoder, ya que la función para mapear una llave aleatoria a una solución factible del problema se basa en las características del problema a resolver (ver Gonçalves and Resende, 2011).

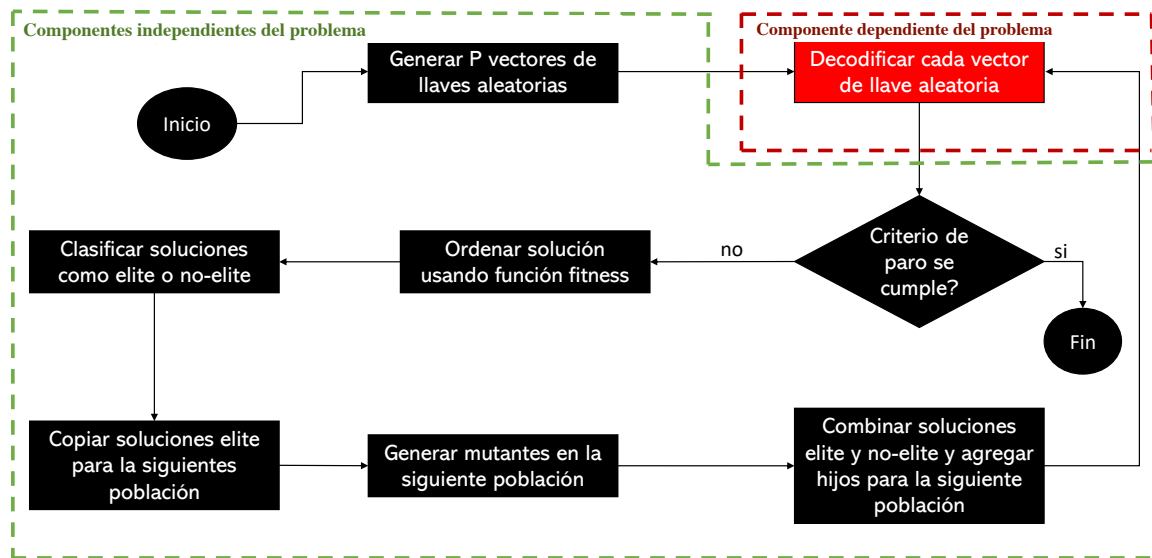


Figura 2: Diagrama de flujo del BRKGA.

En nuestro algoritmo BRKGA no hay una restricción que prohíba la existencia de individuos iguales, pero al estar representados por llaves aleatorias, la probabilidad de que existan dos iguales es muy baja.

En cada iteración k del BRKGA, se procede a obtener soluciones factibles del problema de cada una de las llaves aleatorias que forman la población, por medio del decoder. Las mejores soluciones pasaran a ser parte del grupo élite de cierto tamaño, el resto será la población no-élite, una ilustración de esto lo podemos ver en la Figura 3.

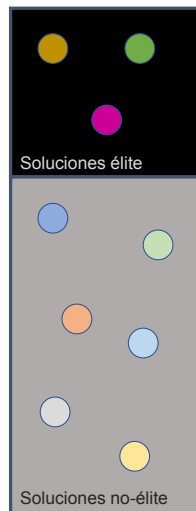


Figura 3: Población en iteración k del algoritmo.

La Figura 4 muestra el detalle para generar la población en la iteración $k + 1$ del algoritmo. Primeramente, el conjunto elite se hereda sin ninguna modificación. Después, un porcentaje pequeño de la población serán generados aleatoriamente como nuevos individuos llamados mutantes. Por último, el resto de la población se obtiene de cruzar un padre elite con un no-elite seleccionados aleatoriamente. Con esta nueva generación formada, se vuelve a hacer la clasificación del conjunto elite y no-elite para volver a repetir el proceso anterior hasta que se cumpla la condición de paro.

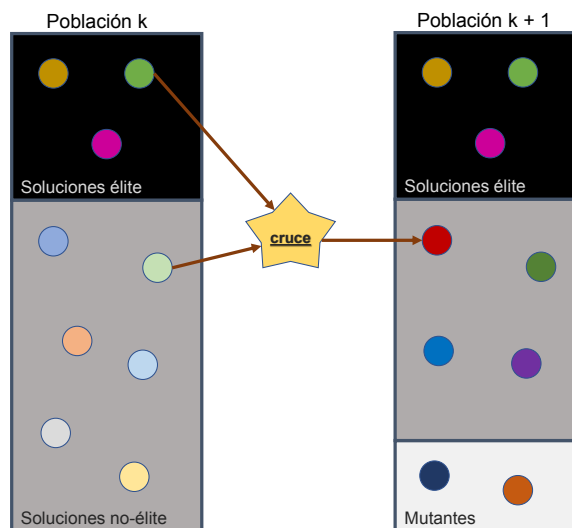


Figura 4: Generación de población en iteración $k + 1$ a partir de iteración k .

En base a lo anterior, el desempeño del algoritmo BRKGA depende de los valores de los siguientes parámetros (además del decoder).

- Tamaño de población (*pob*).
- Cantidad de soluciones elite en la población (*elite*).
- Cantidad de mutantes a generar en cada iteración (*mutant*).
- La probabilidad de heredar genes de padres elite en la crucea (*ph*).
- El criterio de paro (*iter*).

A continuación, presentamos el diseño de nuestro algoritmo decoder para el UARV.

4.3 BRKGA PARA UARV

4.3.1 DECODER

Para la implementación del BRKGA lo primero es definir el componente dependiente del problema, es decir, el decoder. Para ello necesitamos definimos llaves aleatorias del tamaño del número de potenciales ubicaciones $|I|$, más el número de instalaciones por abrir n . El Decoder descrito en Algoritmo 2 toma los elementos de la llave aleatoria y los traduce en dos decisiones importantes: cuales instalaciones abrir dentro de un conjunto potencial de ubicaciones, y cuál será el radio de cobertura que éstas tendrán.

La idea principal para la decodificación es tomar los primeros $|I|$ elementos de la llave aleatoria y obtener los n números más grandes dentro de ese conjunto. Estos elementos serán nuestras instalaciones abiertas y los elementos no seleccionados serán instalaciones cerradas. La segunda decisión será el radio de cobertura que tendrán esas instalaciones abiertas, el cual se decidirá considerando el resto de elementos de la llave aleatoria que van del elemento $|I|$ hasta el $|I| + n$, esta decisión se tomará considerando el incremento máximo para cada ubicación y el incremento total (presupuesto).

El Algoritmo 2 toma como entrada la llave aleatoria \mathbf{r} , el incremento total $\ell' = \ell$, el máximo incremento para cada instalación ε y el número de instalaciones a abrir n . Se hacen iteraciones hasta que las n instalaciones hayan sido seleccionadas. Durante las iteraciones, para la primera decisión, que es cuáles instalaciones se abrirán, $z_i = 1$, se toma el elemento de mayor valor dentro de los primeros $|I|$ elementos de la llave aleatoria, y el índice de ese valor será nuestra decisión de instalación abierta. Enseguida se descarta ese elemento de la llave aleatoria para la siguiente iteración.

Al haber decidido qué instalación abrir, se procede con la segunda decisión que es calcular el incremento de radio que tendrá dicha instalación x_i , para ello se hace uso del

resto de elementos de la llave aleatoria, que van de $|I| + 1$ hasta $|I| + n$. El cálculo se realiza tomando el mínimo entre el presupuesto disponible ℓ' y la parte entera después de multiplicar el máximo incremento ϵ_i para cada instalación por el elemento $|I| + k$ de la llave aleatoria. La decisión del incremento de radio se va asignando en orden como se van abriendo las instalaciones. Por último, se actualiza el presupuesto restando la asignación dada.

Algorithm 2: Decoder

Input : $\mathbf{r} = (r_1, r_2, \dots, r_{|I|+n})$, ℓ' , ϵ_i , n ;
for $k = 1, \dots, n$ **do**
 $i' = \operatorname{argmax}_{i=1, \dots, |I|} \{r_i\}$;
 Abrir instalación i' , es decir, $z_{i'} = 1$ y redefinir $r_{i'} = -1$;
 Calcular radio de servicio $x_{i'} = \min(\ell', \lfloor \epsilon \cdot r_{|I|+k} \rfloor)$;
 $\ell' = \ell' - x_{i'}$;
 $k = k + 1$

De esta forma se implementa el decoder de nuestro BRKGA para obtener las dos decisiones, es decir, soluciones factibles del problema de optimización.

4.3.2 PARÁMETROS DEL BRKGA

Se realizó una calibración de los parámetros para nuestro BRKGA. La Tabla 2 muestra los resultados de las distintas combinaciones de valores de los parámetros. Se usaron 5 instancias de cada tipo, y reportamos el valor promedio de la función objetivo y el tiempo computacional promedio en segundos, para cada tipo de instancias. Resaltamos en azul y rojo los mejores y peores valores, respectivamente.

<i>pob</i>	<i>elite</i>	<i>mutant</i>	<i>ph</i>	Obj_A	Tiempo	Obj_B	Tiempo	Obj_C	Tiempo
100	0.15	0.05	0.50	0.4143	1.3532	0.4490	5.6924	0.4656	51.4178
100	0.20	0.10	0.60	0.4154	1.2014	0.4532	6.331	0.4650	98.7232
200	0.15	0.05	0.50	0.4158	2.0424	0.4559	10.1174	0.4714	81.105
300	0.25	0.22	0.70	0.4162	3.1468	0.4589	25.832	0.4747	192.276
300	0.25	0.20	0.70	0.4167	2.8312	0.4613	16.4022	0.4781	153.4106
300	0.15	0.05	0.50	0.4186	2.8564	0.4620	17.0858	0.4799	137.1756
300	0.25	0.15	0.70	0.4196	6.7418	0.4609	31.9614	0.4788	282.5774
300	0.20	0.10	0.60	0.4181	6.4334	0.4544	38.1148	0.4788	326.8602
300	0.20	0.10	0.70	0.4185	12.3622	0.4639	59.1856	0.4837	252.1418
300	0.20	0.10	0.50	0.4191	3.5512	0.4619	18.4534	0.4813	172.2688
300	0.25	0.10	0.50	0.4191	3.6738	0.4618	14.8830	0.4783	119.6560
300	0.25	0.15	0.50	0.4179	3.6338	0.4599	21.0138	0.4808	183.2870
300	0.25	0.05	0.50	0.4195	3.2818	0.4594	14.3252	0.4783	136.5150
300	0.20	0.05	0.50	0.4190	3.4226	0.4605	14.1120	0.4796	146.8208
400	0.15	0.05	0.50	0.4186	9.2048	0.4587	62.0236	0.4816	399.6076
500	0.15	0.05	0.50	0.4195	4.8358	0.4650	28.2388	0.4816	153.0952
500	0.25	0.15	0.70	0.4195	8.6982	0.4630	70.4054	0.4827	215.9218
500	0.20	0.10	0.60	0.4189	11.8910	0.4634	60.3002	0.4833	488.4736
500	0.20	0.10	0.70	0.4185	6.0840	0.4586	44.3084	0.4732	274.5906
$I + n$	0.15	0.05	0.50	0.4114	0.9326	0.4494	7.0230	0.4714	100.071
$I + n$	0.25	0.20	0.70	0.4132	0.7516	0.4512	5.0686	0.4744	92.2416
$2(I + n)$	0.15	0.05	0.50	0.4104	1.0266	0.4549	11.6370	0.4819	207.6846
$2(I + n)$	0.25	0.20	0.70	0.4136	1.3216	0.4551	10.9768	0.4797	184.7594
$3(I + n)$	0.15	0.05	0.50	0.4128	1.3930	0.4580	11.1308	0.4841	242.8594
$3(I + n)$	0.25	0.20	0.70	0.4137	1.6116	0.4562	19.7482	0.4862	326.3666

Tabla 2: Resultados promedio de la variación de parámetros del BRKGA usando 5 instancias de cada tipo.

Note que no hay dominancia de ninguna combinación de parámetros con respecto a calidad y tiempo. Entonces, escogimos la siguiente combinación de valores de los parámetros por tener una calidad similar a los mejores resultados en tiempo de computo razonable. Tamaño de la población $pob = 500$; fracción de la población élite $elite = 0,20$; fracción de la población a reemplazar por mutantes $mutants = 0,10$; probabilidad de que se hereden genes de los padres élite $ph = 0,60$; y criterio de paro de 100 iteraciones sin mejora. Las iteraciones no se consideraron en la variación de parámetros porque los resultados eran similares al usar más de 100 iteraciones sin mejora.

CAPÍTULO 5

RESULTADOS

En esta sección se presentan los resultados numéricos de todos los experimentos realizados. Primero se realiza una comparativa de la eficiencia del solver comercial CPLEX contra la metaheurística BRKGA. Para el solver de CPLEX usamos una condición de parada de 1 hora o GAP de 1%. En el caso del BRKGA el criterio de paro es de 100 iteraciones sin mejora. En seguida se realiza la comparativa con un tiempo de ejecución de 10 minutos o GAP de 1% en el solver comercial CPLEX y BRKGA de 100 iteraciones sin mejora para analizar la ventaja en escenarios con tiempo de computo limitado. Por último se analizan los indicadores de accesibilidad de cada comparación.

5.1 INSTANCIAS Y CONDICIONES EXPERIMENTALES

Se diseñaron diferentes tipos de instancias basadas en el número de locaciones $|I|$, el número de instalaciones a ser abiertas n , los puntos de demanda $|J|$, un tamaño de cuadrícula para la generación aleatoria de coordenadas para todos los puntos, el radio de cobertura base rc , el radio de movilidad ra , el máximo incremento de cada instalación ϵ_i y un incremento máximo total ℓ . Las instancias se muestran en la Tabla 3, donde cada fila muestra los diferentes tipos de instancia y cada columna muestra los valores de sus parámetros. Se generaron aleatoriamente 30 instancias de cada tipo, resultando en un total de 90 instancias.

Tabla 3: Tipos de instancias.

Tipo instancia	$ I $	n	$ J $	rc_i	ϵ_i	ra_j	ℓ	Tamaño de cuadrícula
A	40	5	120	2	5	15	8	100×100
B	80	10	250	2	5	15	15	130×130
C	200	25	600	2	5	15	38	150×150

Todos los experimentos se realizaron en una MacPro con 3.5 GHz 6-core Intel Xeon E5, 16GB de RAM y usando la versión 12.9 del solver CPLEX. El criterio de paro para la metaheurística BRKGA fue de 100 iteraciones sin mejora y para el solver comercial

CPLEX fueron pruebas con criterio de paro de 1 hora de tiempo computacional o GAP de 1 %. Fijamos los parámetros de peso como $\beta_1 = 0,5$, $\beta_2 = 0,25$, $\beta_3 = 0,125$, $\beta_4 = 0,0625$, $\beta_5 = 0,03125$ y $\beta_6 = 0,015625$.

5.2 BRKGA vs CPLEX CON UNA HORA TIEMPO LÍMITE

Para comparar la calidad del BRKGA se toma de referencia el solver comercial CPLEX. Los resultados numéricos se muestran en la Tabla 4 donde se reporta el promedio del GAP relativo (GAP), la desviación estándar del GAP relativo (GAP STD), el tiempo promedio de ejecución en segundos (Tiempo) y la desviación estándar del tiempo de ejecución (Tiempo STD), de CPLEX y el BRKGA para los diferentes tipos de instancias.

Tabla 4: Resultados numéricos: CPLEX una hora.

		Instancia A	Instancia B	Instancia C
CPLEX	GAP	0.97 %	4.20 %	32.83 %
	GAP STD	0.04	4.62	4.08
	Tiempo	10.26	2557.5703	3601.03
	Tiempo STD	2.10	1125.64457	0.354
BRKGA	GAP	1.72 %	5.82 %	32.38 %
	GAP STD	0.41	4.58	2.57
	Tiempo	4.52	23.04	197.13
	Tiempo STD	0.70	4.97	43.19

Se puede observar en la Tabla 4 que las instancias tipo A se llega a un GAP de menos del 2 % en menos de 5 segundos usando BRKGA y a un GAP de menos de 1 % en más de 10 segundos usando CPLEX. Las instancias tipo B en promedio llegan a un GAP de menos del 5 % en menos de 43 minutos usando CPLEX y con BRKGA en promedio llegan a un GAP de menos del 6 % en menos de 24 segundos. Por último, las instancias tipo C en el caso de CPLEX llega a un GAP de más del 32 % usando su condición de parada de 1 hora, con BRKGA fueron menos de 4 minutos dando un GAP de más del 32 %. En promedio, nuestro BRKGA obtiene soluciones comparables con CPLEX pero en mucho menor tiempo de computo.

En la fila “Tiempo STD” para el solver CPLEX, la instancia tipo B muestra que hay grandes diferencias en los tiempos computacionales, ya que obtenemos un gran valor para la desviación estándar en comparación con el valor del tiempo promedio, para BRKGA, quien muestra mayores diferencias en el tiempo computacional es la instancia tipo C.

En la fila “GAP STD” para el solver CPLEX la instancia tipo B muestra mayor diferencia en la calidad de sus soluciones, se puede ver comparando la desviación estándar

del GAP relativo con el GAP promedio, en BRKGA al igual que CPLEX, presenta mayor diferencia en la calidad de la solución para las instancias tipo B.

En la Figura 5 el valor de la función objetivo encontrado se presentan para cada una de las instancias que se resolvieron, A, B, C, tanto en CPLEX como en BRKGA. Los triángulos negros marcan las soluciones de BRKGA que fueron mejor que las soluciones de CPLEX, se puede notar que en las instancias tipo C, BRKGA dio mejores valores en su función objetivo que el solver CPLEX.

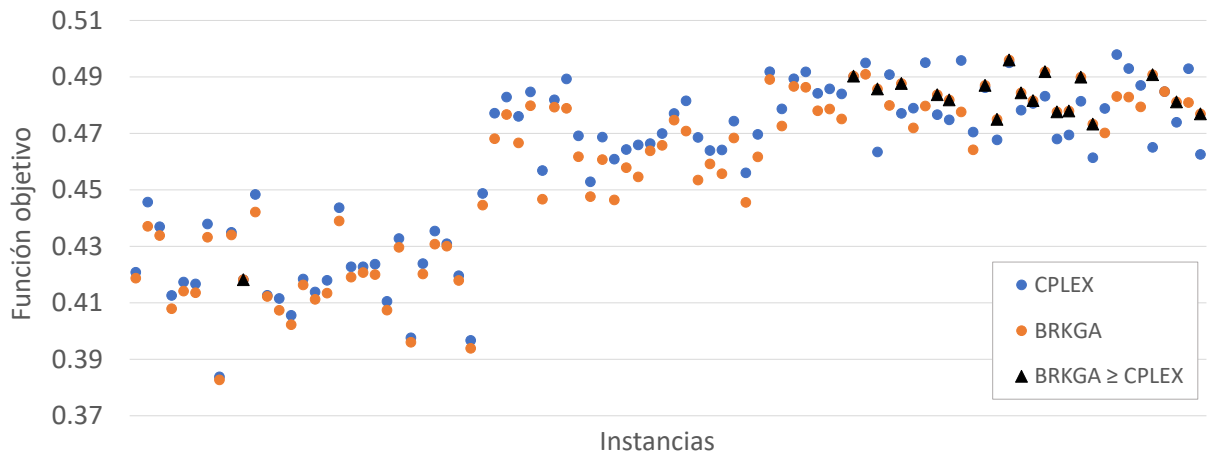


Figura 5: Valor de la función objetivo para las soluciones obtenidas usando CPLEX con un criterio de paro de 1 hora o GAP de 1%, y BRKGA con criterio de paro de 100 iteraciones sin mejora.

En la Tabla 5 se analizan las mejoras relativas que se obtuvieron con la metaheurística BRKGA con respecto al solver CPLEX para los diferentes indicadores. Se puede observar que los indicadores de impedimento *at* y número de oportunidades *ao* obtienen mejores resultados con BRKGA en todos los tipos de instancias. Mientras que en dos de los 3 tipos de instancia se obtuvieron mejores resultados con BRKGA para el indicador de oportunidad *aa*. Por otro lado, los indicadores de cobertura *ac* y el de oportunidad más cercana *an* son los que menor mejora obtuvieron. En el caso de *ac* solo las instancias tipo C obtuvieron mejores resultados, y para el indicador *an*, las soluciones de las instancias tipo B tienen mejores valores. Para las instancias tipo B fue más complicado obtener buenos valores para los indicadores de accesibilidad.

El indicador *as* no puede ser definido cuando hay 0 o 1 punto de demanda sin cubrir, ya que no es posible calcular la máxima distancia entre dos puntos de demanda sin cubrir. Entonces, no fue incluido en la comparación.

Tabla 5: Mejora relativa promedio de BRKGA con respecto a CPLEX 1 hora.

	aa	ac	at	an	ao
A	0.05 %	-5.69 %	1.78 %	-0.04 %	0.85 %
B	-0.82 %	-5.91 %	1.76 %	1.07 %	0.56 %
C	1.16 %	2.48 %	2.65 %	-0.14 %	2.61 %

En la Figura 6 se muestran las gráficas de la mejora relativa de BRKGA con respecto a CPLEX para cada uno de nuestros indicadores, a excepción de *as* donde usamos la mejora absoluta para evitar divisiones entre 0. Se puede observar que en los indicadores *aa* y *at*, BRKGA obtiene mejores valores para las instancias tipo C, en los indicadores *at* y *an*, las instancias tipo B dan mejores valores con BRKGA, en los indicadores *aa* y *ac* para las instancias tipo B BRKGA es superado por CPLEX, para el indicador *ao* no hay dominancia en por ninguno de los dos algoritmos. Finalmente, para el indicador *as* no se encontró variación significativa en la mayoría de las instancias.

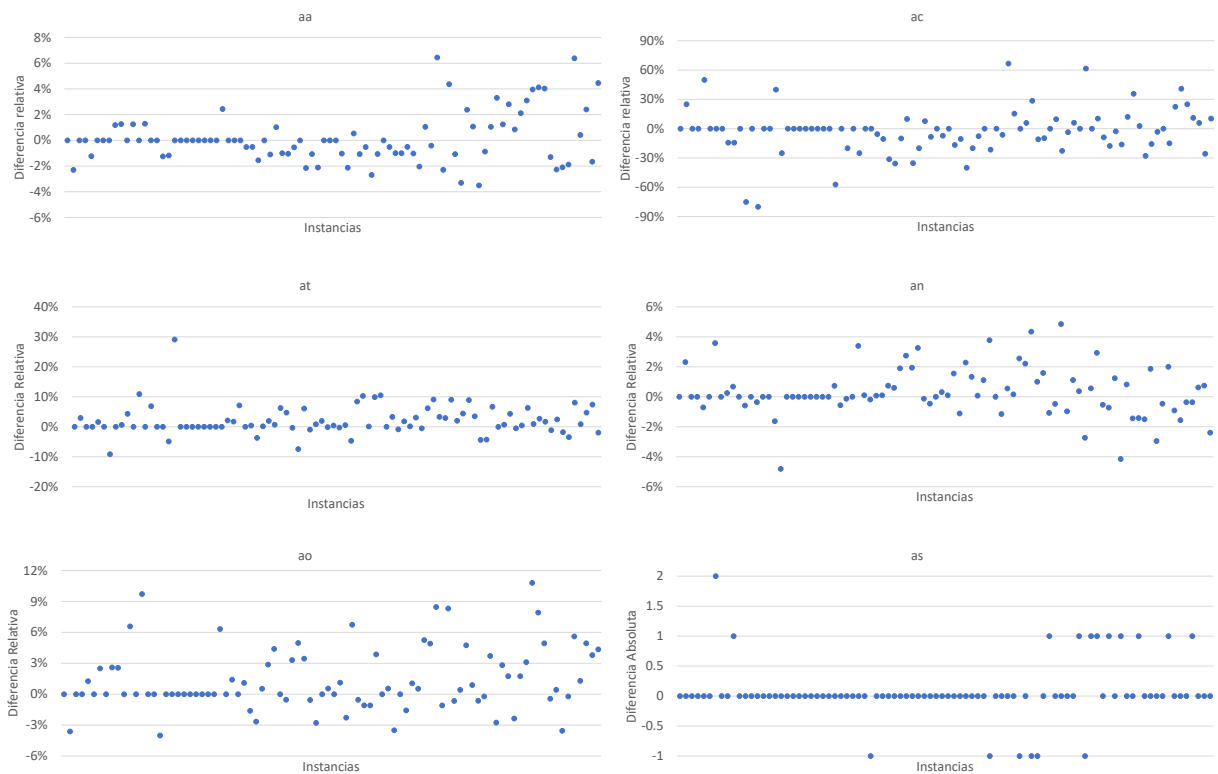


Figura 6: Comparación entre los valores de los diferentes indicadores de accesibilidad para las soluciones obtenidas de CPLEX y BRKGA en todas las instancias.

A continuación se presentan los grafos de las soluciones para cada tipo de instancia, los grafos (a) son las soluciones dadas por BRKGA y los grafos (b) son las soluciones

dadas por CPLEX. La Figura 7 muestra los grafos de una solución para una instancia tipo A, la Figura 8 muestra los grafos de una solución para una instancia tipo B y por último, la Figura 9 muestra los grafos de una solución para una instancia tipo C.

En cada una de las figuras, los cuadrados verdes representan las instalaciones abiertas I_1 , los cuadrados negros las instalaciones no abiertas I_2 , los círculos morados los puntos de demanda cubiertos ac , los círculos rosas los puntos de demanda con oportunidad aa , es decir que puede obtener el servicio moviéndose, los círculos rojos los puntos de demanda que no tienen oportunidad con ninguna instalación abierta. Por último, las líneas rosas representan hacia que instalación abierta tiene oportunidad un punto de demanda.

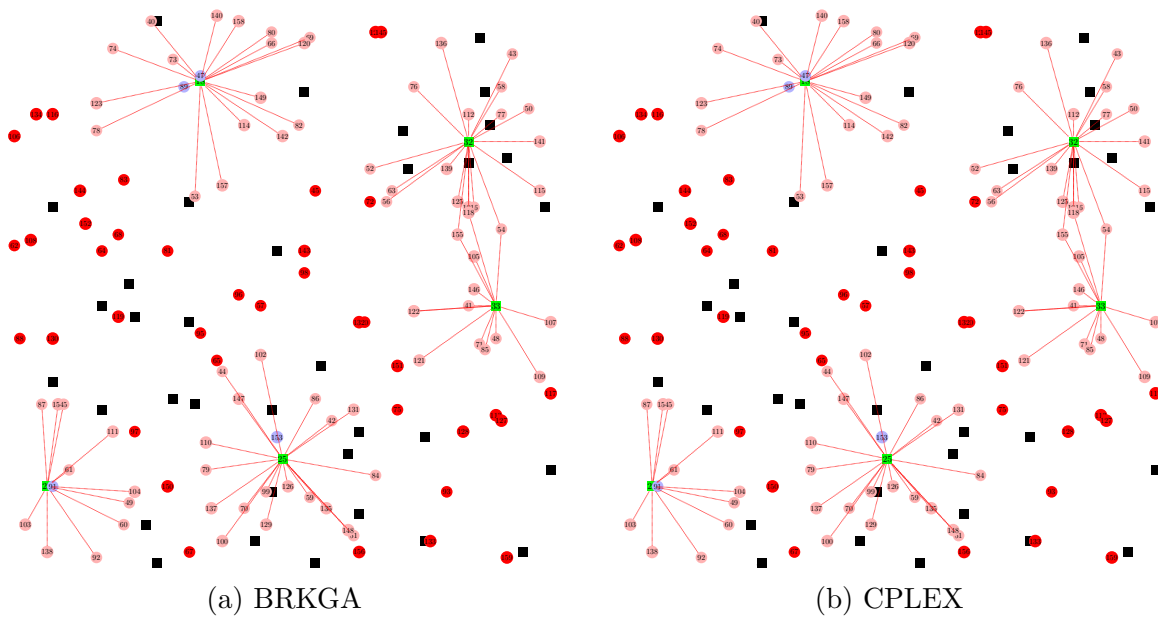


Figura 7: Ejemplo de soluciones de instancias tipo A.

Cabe destacar que se realizó una experimentación para un tipo más de instancia, la D, que tenía como parámetros, $|I| = 400$, $n = 50$, $|J| = 1200$, $rC_i = 2$, $\epsilon_i = 5$, $rA_j = 15$, $\ell = 75$. Con este tipo de instancia no se reportó ninguna solución factible en el solver CPLEX con un tiempo computacional de 1 hora. Para la metaheurística BRKGA si hubo resultados, pero no se incluyeron en el presente trabajo por falta de información para comparar.

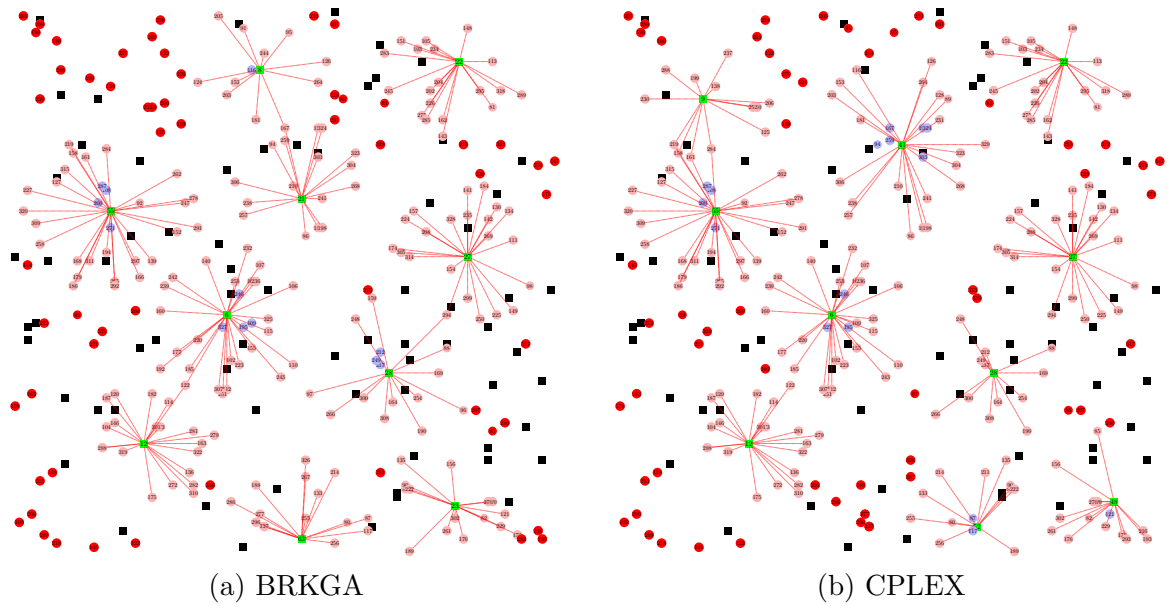


Figura 8: Ejemplo de soluciones de instancias tipo B.

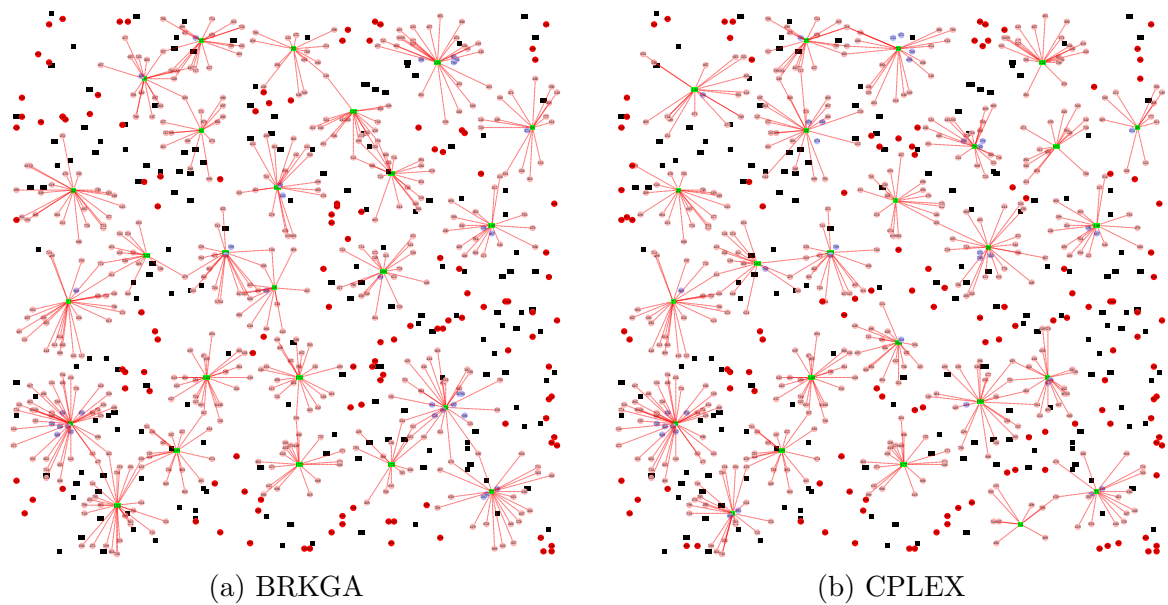


Figura 9: Ejemplo de soluciones de instancias tipo C.

5.3 BRKGA vs CPLEX CON 10 MINUTOS DE TIEMPO LÍMITE

Si el tiempo es crítico en el proceso de toma de decisiones, podemos modificar el criterio de paro, para obtener soluciones en tiempos de cómputo aceptables. En la Figura 6 se muestran resultados obtenidos usando el solver CPLEX con una condición de parada

de 10 minutos o GAP DE 1% y BRKGA con criterio de parada de 100 iteraciones sin mejora. Para obtener el GAP del solver CPLEX y metaheurística BRKGA, se usó la cota dual de los resultados obtenidos usando solver CPLEX con criterio de paro de 1 hora.

Tabla 6: Resultados numéricos usando el solver CPLEX.

		Instancia A	Instancia B	Instancia C
CPLEX	GAP	0.97 %	5.22 %	89.58 %
	GAP STD	0.04	4.81	22.51
	Tiempo	10.79	589	600.26
	Tiempo STD	2.16	49.23	0.24
BRKGA	GAP	1.72 %	5.82 %	32.38 %
	GAP STD	0.41	4.58	2.57
	Tiempo	4.52	23.04	197.13
	Tiempo STD	0.70	4.97	43.19

Podemos notar que para un tiempo corto en solver CPLEX, las soluciones de de las Instancias tipo B bajan su calidad, disminuyendo la diferencia con BRKGA. Para las Instancias tipo C, se obtiene una mala calidad usando CPLEX el cual se ve superado por nuestro BRKGA. Al igual que los resultados anteriores, las instancias del tipo B muestran una mayor diferencia en la calidad de sus soluciones, ya que se obtiene un mayor valor en el promedio de desviación estándar de su GAP relativo. Para un tiempo limitado en instancias grandes, BRKGA obtiene significativamente mejores resultados que solver CPLEX.

Se presentan las soluciones para todas las instancias tipo A, B y C generadas por solver CPLEX con la nueva condición de parada y las soluciones ya dadas por BRKGA. Los triángulos negros son las soluciones en las que BRKGA fue mejor. Podemos notar que para un corto tiempo de proceso en el solver CPLEX las soluciones en las Instancias tipo C son muy deficientes, dando mejores resultados BRKGA.

En la Tabla 7 se muestra el porcentaje de instancias en el que BRKGA obtuvo mejores soluciones que CPLEX. Podemos notar que para un tiempo corto de ejecución en CPLEX, el 100% de las soluciones para las instancias tipo C dieron mejores resultados en BRKGA, y al darle 1 hora de tiempo computacional, BRKGA fue mejor en el 60% de esas soluciones. En las instancias tipo B, para un corto tiempo de ejecución en CPLEX, más del 20% de las soluciones fueron mejores en BRKGA. Por último, más del 3% de las soluciones para las instancias tipo A dieron mejores resultados en BRKGA para ambas condiciones de paro en CPLEX.

En la Tabla 8 se muestra la mejora relativa promedio de BRKGA con respecto a solver CPLEX con un criterio de paro de 10 minutos, se puede observar que para BRKGA los indicadores de accesibilidad *aa*, impedimento *at* y número de

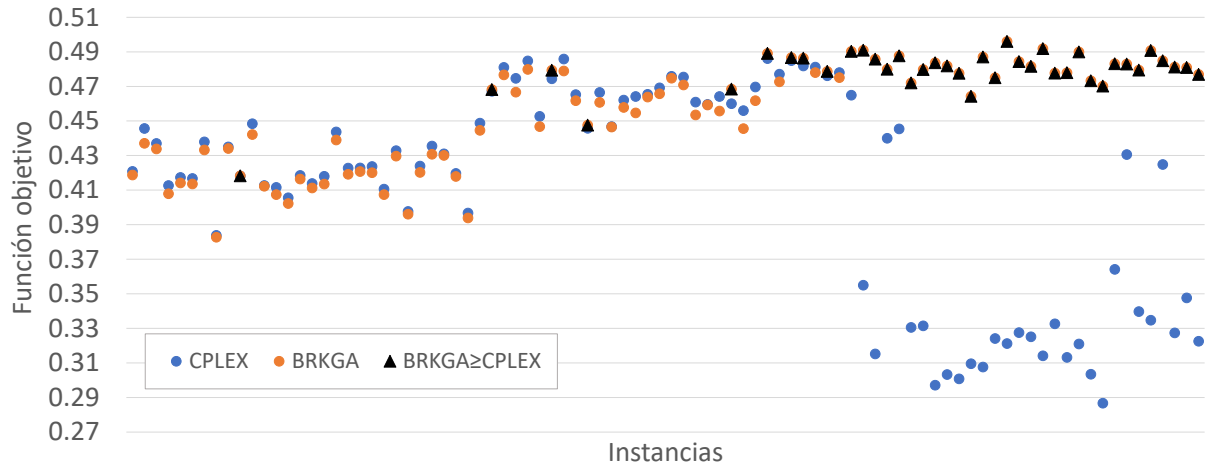


Figura 10: Valor de la función objetivo para las soluciones obtenidas con CPLEX usando un criterio de paro de 10 minutos o GAP de 1 %, y BRKGA con criterio de paro de 100 iteraciones sin mejora.

Tabla 7: Porcentaje de BRKGA mejor que CPLEX.

	A	B	C
CPLEX 1 HR	3.33 %	0 %	60 %
CPLEX 10 MIN	3.33 %	26.67 %	100 %

oportunidades *ao* en promedio brindan mejores resultados para los 3 tipos de instancia, el indicador oportunidad más cercana *an* brinda mejores resultados para las instancias tipo B en BRKGA, mientras que en las instancias tipo C las soluciones del solver CPLEX son mejores. Por último para la instancia tipo C, el indicador de cobertura *ac* en BRKGA da valores muy buenos, mientras que para las instancias tipo A y B, el solver CPLEX obtiene mejores valores.

Tabla 8: Mejora relativa promedio de BRKGA con respecto a CPLEX 10 minutos.

	aa	ac	at	an	ao
A	0.05 %	-5.69 %	1.78 %	-0.04 %	0.85 %
B	0.44 %	-8.61 %	3.30 %	0.64 %	1.99 %
C	52.66 %	205.78 %	20.97 %	-23.69 %	23.71 %

En la Figura 11 podemos observar las gráficas de los diferentes indicadores de accesibilidad, en ellas se muestra la diferencia relativa de BRKGA con respecto al Solver CPLEX, este último con una condición de paro de 10 minutos. Podemos notar que los indicadores *ac*, *at* y *ao* tienen mayores valores en las soluciones obtenidas con BRKGA para la mayoría de las instancias, comparada con las soluciones obtenidas con el solver de CPLEX. En particular, la diferencia es más significativa para instancias más grandes

(vea instancias tipo C con grandes valores de diferencia relativa). En el caso del indicador de accesibilidad *aa*, también se obtiene una mejora significativa con el algoritmo BRKGA, pero solo en instancias tipo C. Al contrario del caso anterior, en instancias tipo C, los valores obtenidos para el indicador *an* son mejores en las soluciones encontradas por CPLEX. Finalmente, para el indicador *as* se muestra la gráfica de la diferencia absoluta de las soluciones encontradas por BRKGA y CPLEX. Se puede notar que en la mayoría de las instancias tipo C, se obtuvieron mejores resultados con BRKGA. Sin embargo, no hay diferencia para la mayoría de las instancias de tipo A y B (diferencia de 0).

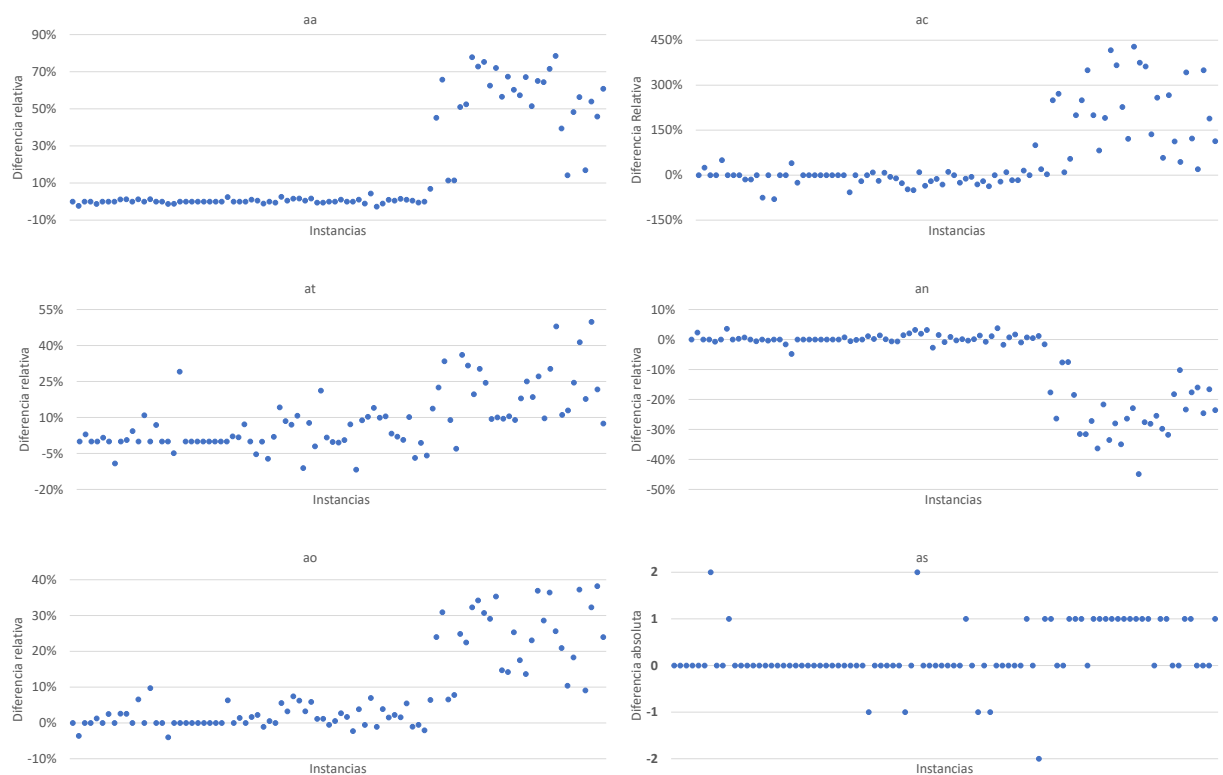


Figura 11: Diferencia relativa en los indicadores de accesibilidad en soluciones obtenidas con BRKGA y CPLEX limitado a 10 minutos de tiempo de ejecución.

La Figura 12 muestra dos grafos, ambos representan la solución de una instancia tipo A, el grafo (a) muestra la solución con BRKGA y el grafo (b) muestra la solución con CPLEX. Similarmente, las Figuras 13 y 14 representan los grafos de una solución de las instancias tipo B y C. Note que para el caso de la instancia tipo A, se obtiene la misma solución usando los dos algoritmos. Sin embargo, para las instancias de tipo B y C, la solución obtenida con CPLEX lleva a una menor cantidad de centroides de demanda con acceso a su demanda (mayor cantidad de puntos rojos en el caso (a)).

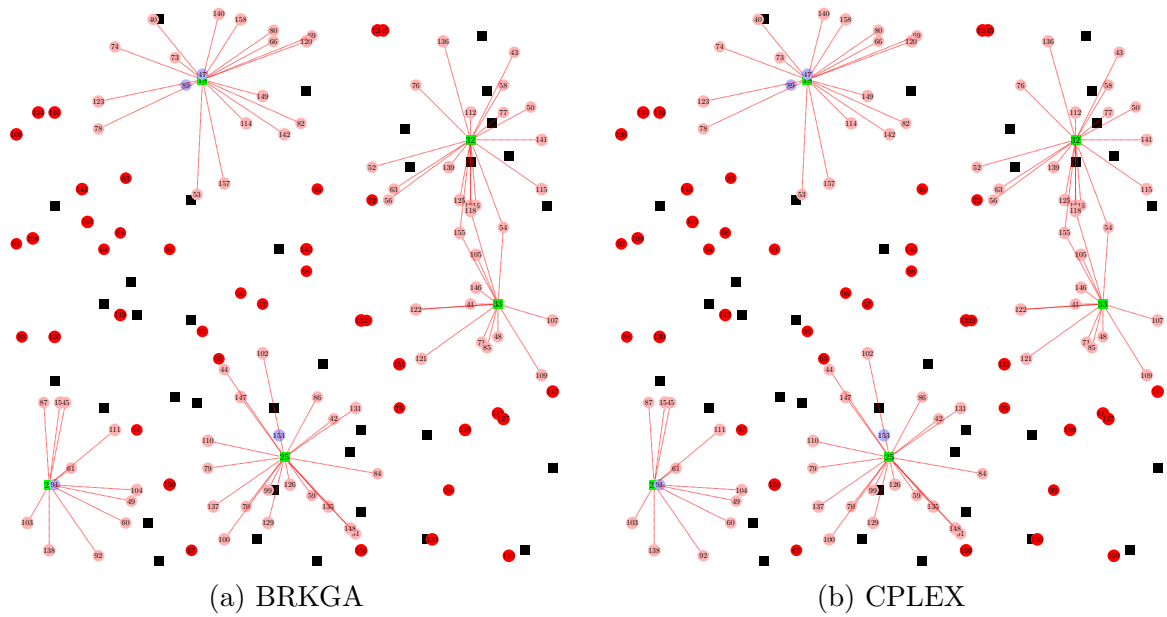


Figura 12: Ejemplo de soluciones de instancias tipo A (CPLEX limitado a 10 minutos de tiempo de computo).

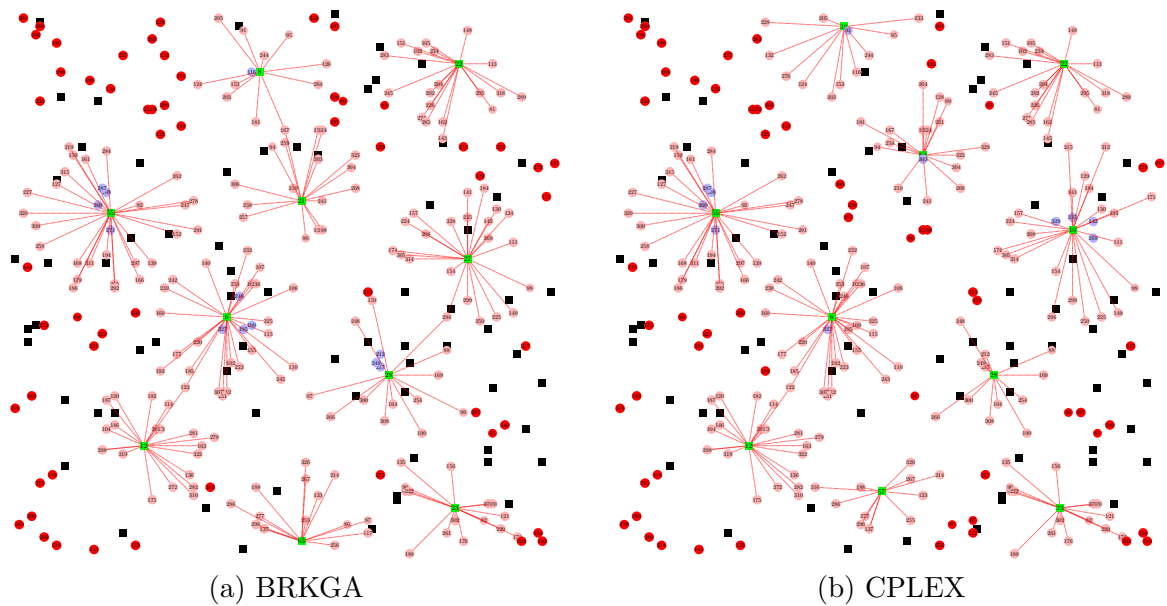


Figura 13: Ejemplo de soluciones de instancias tipo B (CPLEX limitado a 10 minutos de tiempo de computo).

Por último, se implementó un algoritmo heurístico de búsqueda local de la primer mejora y un criterio de paro de optimalidad local que toma como solución inicial, la obtenida por nuestro BRKGA. Los vecinos de una solución son obtenidos mediante la implementación de un operador de intercambio 2-opt($(i, i') : z_i = 1, z_{i'} = 0$) que cierra la instalación en i y abre la instalación en i' , es decir, se fijan $z_i = 0$ y $z_{i'} = 1$. Además, el

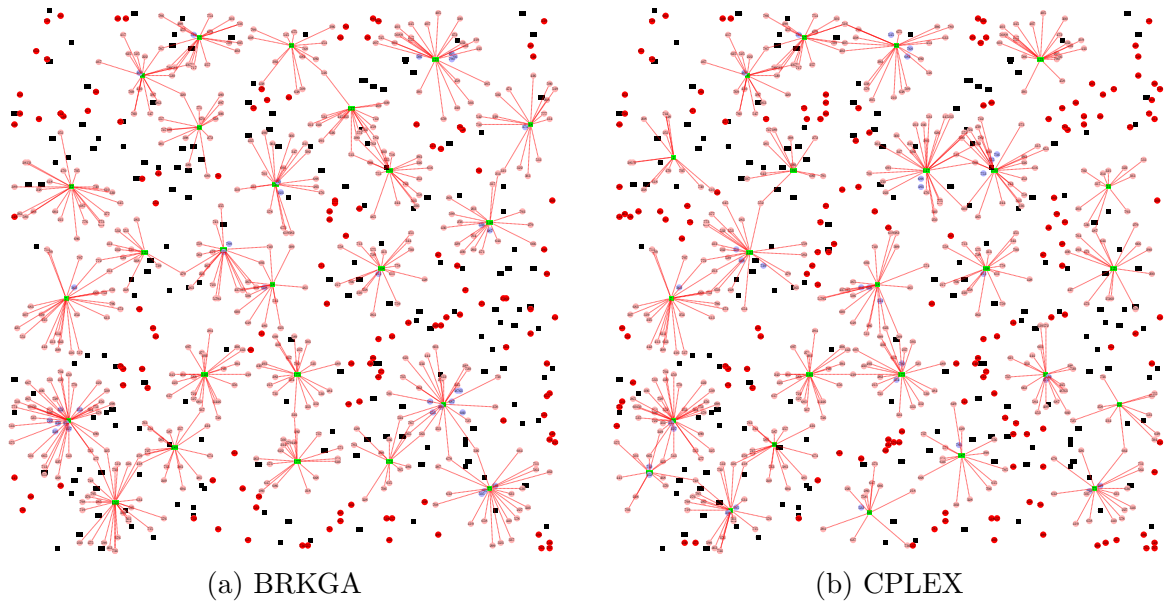


Figura 14: Ejemplo de soluciones de instancias tipo C (CPLEX limitado a 10 minutos de tiempo de computo).

radio de cobertura de la instalación i' se define con un incremento de $x_{i'} = \min \{x_i, \epsilon_{i'}\}$. Finalmente, se suma $r_i - r_{i'}$ al presupuesto remanente de incremento de radio de servicio para todas las instalaciones, y se hace $x_i = 0$. Sin embargo, no se obtuvo ninguna mejoría en la función objetivo para ninguna de las instancias, lo cual indica que solución obtenida mediante nuestro BRKGA es de buena calidad, al menos un óptimo local respecto a los vecindarios de 2-opt. Por lo anterior, decidimos omitir detalles de experimentación con las búsquedas locales implementadas.

CONCLUSIONES

En el presente capítulo se muestran las conclusiones obtenidas, para el problema de ubicación de instalaciones considerando indicadores de accesibilidad y radio de cobertura variable, luego de implementar el solver CPLEX 12.9 y la metaheurística BRKGA para los distintos tipos de instancias y comparar sus resultados.

6.1 CONCLUSIONES

El concepto de accesibilidad se basa en proporcionar más y mejores oportunidades para bienes o servicios a las personas. Incluso cuando hay múltiples definiciones de accesibilidad, hay algunos indicadores comunes.

En el presente trabajo se optimiza la medida de accesibilidad propuesta por Ibarra-Rojas et al. (2018), la cual está basada en un conjunto de seis indicadores: cobertura de los puntos de demanda ac , oportunidad que tienen los puntos de demanda aa , impedimento o costo de viaje at , número de oportunidades que tiene un punto de demanda ao , distancia a la instalación abierta mas cercana que tiene un punto de demanda no cubierto an y segregación espacial as . El problema se formuló matemáticamente a través de un programa lineal entero mixto, el cual, para instancias grandes, no es manejable por el solver comercial CPLEX.

La formulación matemática se implementó en el solver comercial CPLEX con una condición de paro de 1 hora de tiempo de ejecución o GAP de 1%, se obtuvo que para instancias de 40 instalaciones y 120 puntos de demanda se llega a la solución optima en un corto tiempo, para 80 instalaciones y 250 puntos de demanda en promedio su tiempo de ejecución es de casi una hora sin llegar al óptimo. Por último para 200 instalaciones y 600 puntos de demanda no se encuentra una solución aceptable dentro del tiempo de ejecución. Además, considerando una condición de paro de 10 minutos en tiempo computacional, las instancias de 200 instalaciones y 600 puntos de demanda reportan soluciones muy lejanas al óptimo.

Buscando mejores soluciones en tiempos de cómputo razonables, se implementó un algoritmo genético de llaves aleatorias sesgado (BRKGA), con una condición de paro de 100 iteraciones sin mejora, la cual para las instancias de 40 instalaciones y 120 puntos de demanda llega al óptimo en muy poco tiempo, las instancias de 80 instalaciones y 250 puntos de demanda obtiene una solución comparable con CPLEX en mucho menor tiempo. Por último, para instancias de 200 instalaciones y 600 puntos de demanda, BRKGA da mejores soluciones que CPLEX con un corto tiempo de cómputo.

Se resalta que para instancias con un número mayor a 200 instalaciones y mas de 600 puntos de demanda, BRKGA puede encontrar soluciones y CPLEX no puede resolver el problema.

En el Capítulo 5, al comparar los valores obtenidos para los indicadores de accesibilidad de BRKGA y CPLEX 1 hora, se pudo notar que BRKGA da mejores valores en la mayoría de los indicadores, para las instancias A los indicadores aa , at y ao dan mejores valores, para la instancia B son los indicadores at , an y ao , por último para las instancias C son aa , ac , at y ao . En los tres tipos de instancia los indicadores que obtienen mejores valores son at y ao . Para la comparación de BRKGA y CPLEX 10 minutos se pudo notar que para las instancias C los indicadores aa , ac , at y ao dan mejores valores en BRKGA, para las instancias B son los indicadores aa , at , an y ao , por último para las instancias A son aa , at y ao . Podemos notar que los indicadores que mejor se comportan para los tres tipos de instancia son aa , at y ao .

A través de la comparación realizada se logra concluir que para instancias grandes BRKGA da mejores resultados que CPLEX en un tiempo computacional aceptable. Además, BRKGA encuentra soluciones para instancias que CPLEX no. Para 40 instalaciones con 120 puntos de demanda y 80 instalaciones con 250 puntos de demanda, BRKGA obtiene soluciones comparables con CPLEX en mucho menor tiempo.

6.2 TRABAJO A FUTURO

Se podría generalizar el problema tomando en cuenta la demanda no unitaria, así como también las capacidades de las instalaciones. Sin embargo, esto implicaría diferencias significativas en el enunciado del problema.

Se podría mejorar el método heurístico BRKGA, como por ejemplo, se podría hacer mas rápido o intentar encontrar parámetros que mejoren la calidad de las soluciones dadas por este.

Aplicar el heurístico búsqueda local mejor mejora o implementar otro operador para intentar encontrar mejores soluciones del problema.

Encontrar un caso de estudio para aplicar tanto la formulación matemática como la metaheurística para obtener soluciones reales y dar un mejor análisis del comportamiento de estos.

APÉNDICE A

SOLUCIONES DE CPLEX 1 HORA

A continuación, se detalla la información de las soluciones obtenida por CPLEX en 1 hora de tiempo de computo para todas las instancias.

Tabla 9: Solución de instancias tipo A usando CPLEX con condición de paro de 1 hora.

Instancias	Cota dual	Función objetivo	GAP	Tiempo	<i>aa</i>	<i>ac</i>	<i>at</i>	<i>an</i>	<i>ao</i>	<i>as</i>
AFLP40_120_1.txt	0.424774	0.420881	0.92 %	10.2125	81	4	18.8627	2165	80	1
AFLP40_120_2.txt	0.450161	0.445725	1.00 %	9.91166	87	4	18.1029	2073	83	1
AFLP40_120_3.txt	0.441336	0.436997	0.99 %	11.2832	84	6	18.2043	2120	80	1
AFLP40_120_4.txt	0.41676	0.412661	0.99 %	11.5059	77	9	16.3967	2023	68	1
AFLP40_120_5.txt	0.421532	0.417395	0.99 %	7.30818	81	2	19.015	2260	79	1
AFLP40_120_6.txt	0.420822	0.41676	0.97 %	9.96636	80	6	14.9423	2122	74	1
AFLP40_120_7.txt	0.442299	0.437987	0.98 %	10.4971	85	5	16.981	1980	80	0
AFLP40_120_8.txt	0.387361	0.38382	0.92 %	10.7226	74	2	16.1383	2335	74	0
AFLP40_120_9.txt	0.439369	0.435027	1.00 %	17.7722	84	7	14.6976	1985	77	0
AFLP40_120_10.txt	0.422008	0.418218	0.91 %	9.65549	79	7	18.7805	2198	78	1
AFLP40_120_11.txt	0.452473	0.448462	0.89 %	8.68583	84	12	16.3197	1931	75	3
AFLP40_120_12.txt	0.416851	0.412726	1.00 %	9.95161	80	4	16.3228	2395	76	0
AFLP40_120_13.txt	0.415297	0.41159	0.90 %	8.33926	78	8	14.8268	2326	72	1
AFLP40_120_14.txt	0.40963	0.405628	0.99 %	11.4085	77	5	18.4806	2242	72	1
AFLP40_120_15.txt	0.422551	0.418488	0.97 %	9.81903	80	4	20.3016	2081	79	1
AFLP40_120_16.txt	0.41768	0.413886	0.92 %	8.89389	80	5	14.3565	2111	77	0
AFLP40_120_17.txt	0.422211	0.418041	1.00 %	8.19619	80	5	17.949	2089	75	0
AFLP40_120_18.txt	0.447594	0.443735	0.87 %	6.90794	85	8	17.1181	2163	79	1
AFLP40_120_19.txt	0.427019	0.422802	1.00 %	13.9354	81	7	15.4888	2092	77	4
AFLP40_120_20.txt	0.427039	0.422845	0.99 %	10.6283	82	4	17.026	2199	80	1
AFLP40_120_21.txt	0.427725	0.423719	0.95 %	10.2794	81	7	15.4478	2041	76	1
AFLP40_120_22.txt	0.414566	0.410567	0.97 %	10.9425	77	6	20.9151	2215	72	1
AFLP40_120_23.txt	0.437032	0.432833	0.97 %	11.851	83	6	17.4422	2048	83	1
AFLP40_120_24.txt	0.401629	0.397674	0.99 %	10.0679	77	3	15.6264	2264	76	1
AFLP40_120_25.txt	0.428158	0.423929	1.00 %	7.94315	80	7	19.2602	2088	80	1
AFLP40_120_26.txt	0.439778	0.435496	0.98 %	9.01151	83	9	14.9379	2073	75	1
AFLP40_120_27.txt	0.435304	0.431	1.00 %	13.4397	82	7	18.1977	2045	79	1
AFLP40_120_28.txt	0.423844	0.419658	1.00 %	11.0802	81	4	17.2691	2149	80	1
AFLP40_120_29.txt	0.400562	0.396754	0.96 %	8.95103	75	5	17.6229	2270	71	2
AFLP40_120_30.txt	0.453274	0.44881	0.99 %	8.6956	86	8	16.7342	2024	78	2

Tabla 10: Solución de instancias tipo B usando CPLEX con condición de paro de 1 hora.

Instancias	Cota dual	Función objetivo	GAP	Tiempo	<i>aa</i>	<i>ac</i>	<i>at</i>	<i>an</i>	<i>ao</i>	<i>as</i>
AFLP80_250_1.txt	0.481962	0.477193	1.00 %	1994.3	194	16	29.732	3829	182	1
AFLP80_250_2.txt	0.487722	0.482903	1.00 %	1778.59	195	13	40.2837	3824	186	1
AFLP80_250_3.txt	0.480829	0.476071	1.00 %	2878.42	194	13	32.3349	3959	188	2
AFLP80_250_4.txt	0.557805	0.484775	15.06 %	3600.15	196	18	31.8238	3893	189	1
AFLP80_250_5.txt	0.461464	0.4569	1.00 %	734.433	182	19	30.8089	3998	174	1
AFLP80_250_6.txt	0.486718	0.481942	0.99 %	1808.39	195	16	33.545	3772	182	1
AFLP80_250_7.txt	0.494226	0.489338	1.00 %	3571.35	200	14	32.5322	3832	191	1
AFLP80_250_8.txt	0.473919	0.469227	1.00 %	3291.94	192	10	33.169	3844	189	0
AFLP80_250_9.txt	0.493621	0.452915	8.99 %	3600.56	184	10	32.1397	4009	181	1
AFLP80_250_10.txt	0.484856	0.468721	3.44 %	3601.21	190	17	27.8433	4020	181	1
AFLP80_250_11.txt	0.494563	0.460928	7.30 %	3600.49	186	15	30.843	4047	174	2
AFLP80_250_12.txt	0.468702	0.464377	0.93 %	1808.04	188	13	32.6346	3888	178	1
AFLP80_250_13.txt	0.502574	0.465991	7.85 %	3602.16	189	12	33.9385	3948	180	1
AFLP80_250_14.txt	0.471009	0.466426	0.98 %	1854.92	190	10	35.2767	4195	189	2
AFLP80_250_15.txt	0.474708	0.470009	1.00 %	3526.53	189	14	37.216	3788	180	0
AFLP80_250_16.txt	0.534491	0.477087	12.03 %	3600.15	194	9	40.8773	3880	189	1
AFLP80_250_17.txt	0.526576	0.481618	9.33 %	3600.16	195	18	30.2192	3795	179	0
AFLP80_250_18.txt	0.473331	0.468647	1.00 %	3078.96	188	19	29.788	3966	176	1
AFLP80_250_19.txt	0.523093	0.463999	12.74 %	3600.18	187	15	32.5324	3988	178	2
AFLP80_250_20.txt	0.468818	0.464178	1.00 %	1519.1	188	15	29.4219	3974	185	0
AFLP80_250_21.txt	0.479135	0.474398	1.00 %	2244.42	193	13	33.2615	3961	186	1
AFLP80_250_22.txt	0.460613	0.456075	1.00 %	363.827	186	9	32.5243	4038	184	1
AFLP80_250_23.txt	0.474382	0.469697	1.00 %	531.629	189	14	37.2776	3946	181	2
AFLP80_250_24.txt	0.529327	0.491884	7.61 %	3600.14	202	11	35.6516	3913	201	1
AFLP80_250_25.txt	0.483526	0.478753	1.00 %	2647.95	194	16	31.4904	3827	181	1
AFLP80_250_26.txt	0.538319	0.489404	9.99 %	3600.15	201	6	43.0655	3777	200	1
AFLP80_250_27.txt	0.496772	0.491855	1.00 %	2116.94	200	13	38.0953	3697	193	1
AFLP80_250_28.txt	0.489081	0.484276	0.99 %	674.317	198	16	26.7143	3788	191	3
AFLP80_250_29.txt	0.548361	0.485818	12.87 %	3600.16	196	17	34.0517	3750	189	2
AFLP80_250_30.txt	0.488874	0.484041	1.00 %	697.543	196	14	37.0157	3887	189	2

Tabla 11: Solución de instancias tipo C usando CPLEX con condición de paro de 1 hora.

Instancias	Cota dual	Función objetivo	GAP	Tiempo	<i>aa</i>	<i>ac</i>	<i>at</i>	<i>an</i>	<i>ao</i>	<i>as</i>
AFLP200_600_1.txt	0.622126	0.490166	26.92 %	3601.72	476	37	77.4827	8960	457	1
AFLP200_600_2.txt	0.649251	0.495035	31.15 %	3600.85	484	31	80.6229	8879	468	1
AFLP200_600_3.txt	0.635212	0.463467	37.06 %	3601.32	450	26	77.3467	9159	449	0
AFLP200_600_4.txt	0.648576	0.490899	32.12 %	3601	480	31	77.2563	9005	460	1
AFLP200_600_5.txt	0.642421	0.477162	34.63 %	3601.01	458	44	74.3785	8801	433	0
AFLP200_600_6.txt	0.617758	0.479007	28.97 %	3601.07	467	28	79.2927	9387	460	1
AFLP200_600_7.txt	0.616815	0.495135	24.58 %	3601.11	484	33	78.2147	9017	472	1
AFLP200_600_8.txt	0.635426	0.476686	33.30 %	3601.05	462	36	72.763	9056	442	0
AFLP200_600_9.txt	0.636627	0.474888	34.06 %	3601.58	465	26	72.7996	9300	447	1
AFLP200_600_10.txt	0.624097	0.495881	25.86 %	3600.91	485	31	81.1828	8978	471	0
AFLP200_600_11.txt	0.633292	0.47054	34.59 %	3600.78	459	29	70.189	9231	458	0
AFLP200_600_12.txt	0.653707	0.486263	34.43 %	3600.87	475	34	70.1637	8896	458	1
AFLP200_600_13.txt	0.644479	0.467784	37.77 %	3601.03	453	34	71.1317	9179	473	0
AFLP200_600_14.txt	0.648556	0.495058	31.01 %	3600.96	481	37	80.6397	8628	461	1
AFLP200_600_15.txt	0.64148	0.478269	34.13 %	3601.11	463	37	74.0798	9310	457	0
AFLP200_600_16.txt	0.65102	0.480663	35.44 %	3600.89	468	33	73.7413	8870	467	1
AFLP200_600_17.txt	0.638304	0.483213	32.10 %	3600.91	473	28	75.1194	8896	458	1
AFLP200_600_18.txt	0.6323	0.468091	35.08 %	3600.73	451	36	75.8372	9196	450	0
AFLP200_600_19.txt	0.615107	0.469489	31.02 %	3600.93	454	36	70.9893	9172	435	1
AFLP200_600_20.txt	0.6439	0.481452	33.74 %	3600.97	461	51	69.5937	8733	429	0
AFLP200_600_21.txt	0.637969	0.461442	38.26 %	3600.76	447	31	70.4724	9365	446	1
AFLP200_600_22.txt	0.632087	0.478917	31.98 %	3600.75	463	33	83.5787	9127	458	1
AFLP200_600_23.txt	0.638232	0.497965	28.17 %	3602.51	485	40	74.4458	8896	466	0
AFLP200_600_24.txt	0.620213	0.493012	25.80 %	3601.01	477	40	81.2317	8986	451	0
AFLP200_600_25.txt	0.628174	0.48707	28.97 %	3600.76	479	22	81.3972	9310	473	1
AFLP200_600_26.txt	0.647069	0.465136	39.11 %	3600.78	454	16	87.209	9451	464	1
AFLP200_600_27.txt	0.655157	0.484869	35.12 %	3600.9	475	27	77.1091	8947	462	0
AFLP200_600_28.txt	0.663546	0.473982	39.99 %	3601.1	457	34	83.104	9100	445	0
AFLP200_600_29.txt	0.643519	0.492965	30.54 %	3600.93	482	35	72.4043	8947	474	0
AFLP200_600_30.txt	0.642806	0.462612	38.95 %	3600.74	448	29	75.6293	9482	436	1

APÉNDICE B

SOLUCIONES DE CPLEX 10 MINUTOS

A continuación, se detalla la información de las soluciones obtenida por CPLEX en 10 minutos de tiempo de computo para todas las instancias.

Tabla 12: Solución de instancias tipo A usando CPLEX con condición de paro de 10 minutos.

Instancias	Cota dual	Función objetivo	GAP	Tiempo	<i>aa</i>	<i>ac</i>	<i>at</i>	<i>an</i>	<i>ao</i>	<i>as</i>
AFLP40_120_1.txt	0.424774	0.420881	0.92 %	10.4651	81	4	18.8627	2165	80	1
AFLP40_120_2.txt	0.450161	0.445725	1.00 %	10.1348	87	4	18.1029	2073	83	1
AFLP40_120_3.txt	0.441336	0.436997	0.99 %	11.2044	84	6	18.2043	2120	80	1
AFLP40_120_4.txt	0.41676	0.412661	0.99 %	11.9995	77	9	16.3967	2023	68	1
AFLP40_120_5.txt	0.421532	0.417395	0.99 %	8.41521	81	2	19.015	2260	79	1
AFLP40_120_6.txt	0.420822	0.41676	0.97 %	12.2269	80	6	14.9423	2122	74	1
AFLP40_120_7.txt	0.442299	0.437987	0.98 %	12.548	85	5	16.981	1980	80	0
AFLP40_120_8.txt	0.387361	0.38382	0.92 %	11.2538	74	2	16.1383	2335	74	0
AFLP40_120_9.txt	0.439369	0.435027	1.00 %	18.2746	84	7	14.6976	1985	77	0
AFLP40_120_10.txt	0.422008	0.418218	0.91 %	10.0841	79	7	18.7805	2198	78	1
AFLP40_120_11.txt	0.452473	0.448462	0.89 %	8.95019	84	12	16.3197	1931	75	3
AFLP40_120_12.txt	0.416851	0.412726	1.00 %	10.2918	80	4	16.3228	2395	76	0
AFLP40_120_13.txt	0.415297	0.41159	0.90 %	8.7442	78	8	14.8268	2326	72	1
AFLP40_120_14.txt	0.40963	0.405628	0.99 %	11.8789	77	5	18.4806	2242	72	1
AFLP40_120_15.txt	0.422551	0.418488	0.97 %	10.2734	80	4	20.3016	2081	79	1
AFLP40_120_16.txt	0.41768	0.413886	0.92 %	9.2079	80	5	14.3565	2111	77	0
AFLP40_120_17.txt	0.422211	0.418041	1.00 %	8.67952	80	5	17.949	2089	75	0
AFLP40_120_18.txt	0.447594	0.443735	0.87 %	7.14922	85	8	17.1181	2163	79	1
AFLP40_120_19.txt	0.427019	0.422802	1.00 %	14.3613	81	7	15.4888	2092	77	4
AFLP40_120_20.txt	0.427039	0.422845	0.99 %	10.864	82	4	17.026	2199	80	1
AFLP40_120_21.txt	0.427725	0.423719	0.95 %	10.5674	81	7	15.4478	2041	76	1
AFLP40_120_22.txt	0.414566	0.410567	0.97 %	11.2061	77	6	20.9151	2215	72	1
AFLP40_120_23.txt	0.437032	0.432833	0.97 %	12.4259	83	6	17.4422	2048	83	1
AFLP40_120_24.txt	0.401629	0.397674	0.99 %	10.5599	77	3	15.6264	2264	76	1
AFLP40_120_25.txt	0.428158	0.423929	1.00 %	8.30067	80	7	19.2602	2088	80	1
AFLP40_120_26.txt	0.439778	0.435496	0.98 %	9.92851	83	9	14.9379	2073	75	1
AFLP40_120_27.txt	0.435304	0.431	1.00 %	14.1404	82	7	18.1977	2045	79	1
AFLP40_120_28.txt	0.423844	0.419658	1.00 %	11.4219	81	4	17.2691	2149	80	1
AFLP40_120_29.txt	0.400562	0.396754	0.96 %	9.06874	75	5	17.6229	2270	71	2
AFLP40_120_30.txt	0.453274	0.44881	0.99 %	8.98092	86	8	16.7342	2024	78	2

Tabla 13: Solución de instancias tipo B usando CPLEX con condición de paro de 10 minutos.

Instancias	Cota dual	Función objetivo	GAP	Tiempo	<i>aa</i>	<i>ac</i>	<i>at</i>	<i>an</i>	<i>ao</i>	<i>as</i>
AFLP80_250_1.txt	0.481962	0.467212	3.16 %	600.175	191	11	31.5393	3916	181	1
AFLP80_250_2.txt	0.487722	0.481025	1.39 %	600.168	193	16	38.8256	3821	179	1
AFLP80_250_3.txt	0.480829	0.474689	1.29 %	600.186	193	12	34.9069	3899	185	2
AFLP80_250_4.txt	0.557805	0.484775	15.06 %	600.197	196	18	31.8238	3893	189	1
AFLP80_250_5.txt	0.461464	0.452742	1.93 %	600.181	181	19	27.1624	4027	179	1
AFLP80_250_6.txt	0.486718	0.474445	2.59 %	600.198	192	15	32.8572	3825	180	1
AFLP80_250_7.txt	0.494226	0.485871	1.72 %	600.183	197	17	31.8463	3800	185	1
AFLP80_250_8.txt	0.473919	0.465318	1.85 %	600.183	187	18	29.8525	3838	175	0
AFLP80_250_9.txt	0.493621	0.445801	10.73 %	600.175	180	10	33.4633	3991	176	2
AFLP80_250_10.txt	0.484856	0.466531	3.93 %	600.177	189	17	27.4167	4020	184	1
AFLP80_250_11.txt	0.494563	0.446779	10.70 %	600.178	179	15	31.1813	4050	170	0
AFLP80_250_12.txt	0.468702	0.462115	1.43 %	600.18	187	16	27.1615	3991	175	1
AFLP80_250_13.txt	0.502574	0.464203	8.27 %	600.193	186	16	34.0416	3871	173	1
AFLP80_250_14.txt	0.471009	0.465425	1.20 %	600.144	190	9	35.316	4232	190	2
AFLP80_250_15.txt	0.474708	0.469213	1.17 %	600.262	189	13	37.5449	3767	180	0
AFLP80_250_16.txt	0.534491	0.475866	12.32 %	600.149	192	12	40.5344	3895	184	1
AFLP80_250_17.txt	0.526576	0.475473	10.75 %	600.188	193	17	28.3631	3850	178	0
AFLP80_250_18.txt	0.473331	0.46099	2.68 %	600.179	184	18	32.1796	3936	176	1
AFLP80_250_19.txt	0.523093	0.45962	13.81 %	600.1	186	13	32.3996	4074	183	1
AFLP80_250_20.txt	0.468818	0.464178	1.00 %	600.171	188	15	29.4219	3974	185	0
AFLP80_250_21.txt	0.479135	0.460039	4.15 %	600.177	184	19	29.2092	3994	172	2
AFLP80_250_22.txt	0.460613	0.456075	1.00 %	332.002	186	9	32.5243	4038	184	1
AFLP80_250_23.txt	0.474382	0.469697	1.00 %	532.886	189	14	37.2776	3946	181	2
AFLP80_250_24.txt	0.529327	0.486106	8.89 %	600.19	200	10	34.5133	3982	198	1
AFLP80_250_25.txt	0.483526	0.477109	1.34 %	600.182	192	18	31.8909	3756	178	1
AFLP80_250_26.txt	0.538319	0.484998	10.99 %	600.325	196	12	42.4144	3735	190	1
AFLP80_250_27.txt	0.496772	0.481906	3.08 %	600.165	196	13	35.2039	3740	183	1
AFLP80_250_28.txt	0.489081	0.48123	1.63 %	600.169	196	16	28.7199	3858	190	2
AFLP80_250_29.txt	0.548361	0.476219	15.15 %	600.182	195	9	35.2955	3815	192	1
AFLP80_250_30.txt	0.488874	0.478036	2.27 %	600.175	192	15	39.1193	4009	194	1

Tabla 14: Solución de instancias tipo C usando CPLEX con condición de paro de 10 minutos.

Instancias	Cota dual	Función objetivo	GAP	Tiempo	<i>aa</i>	<i>ac</i>	<i>at</i>	<i>an</i>	<i>ao</i>	<i>as</i>
AFLP200_600_1.txt	0.622126	0.464925	33.81 %	600.217	450	32	72.3538	9196	452	2
AFLP200_600_2.txt	0.649251	0.354978	82.90 %	600.202	332	8	71.8099	10951	396	0
AFLP200_600_3.txt	0.635212	0.315241	101.50 %	600.184	289	7	59.8882	12306	372	0
AFLP200_600_4.txt	0.648576	0.440072	47.38 %	600.211	421	31	72.9968	9704	427	1
AFLP200_600_5.txt	0.642421	0.445432	44.22 %	600.199	429	22	83.6136	9979	435	0
AFLP200_600_6.txt	0.617758	0.330552	86.89 %	600.201	306	9	59.445	11402	366	0
AFLP200_600_7.txt	0.616815	0.331496	86.07 %	600.231	307	10	62.0338	13316	387	0
AFLP200_600_8.txt	0.635426	0.297131	113.85 %	600.188	266	8	66.2236	13284	350	0
AFLP200_600_9.txt	0.636627	0.303284	109.91 %	601.514	272	14	57.8317	12420	336	0
AFLP200_600_10.txt	0.624097	0.300833	107.46 %	600.206	267	17	62.3857	14177	358	0
AFLP200_600_11.txt	0.633292	0.30956	104.58 %	600.193	280	11	61.4404	12130	354	0
AFLP200_600_12.txt	0.653707	0.307662	112.48 %	600.207	279	6	68.0316	13309	351	0
AFLP200_600_13.txt	0.644479	0.324124	98.84 %	600.191	299	6	64.9577	12652	401	0
AFLP200_600_14.txt	0.648556	0.321218	101.91 %	600.24	291	11	73.4779	13427	415	0
AFLP200_600_15.txt	0.64148	0.32758	95.82 %	600.195	297	14	70.9457	12124	371	0
AFLP200_600_16.txt	0.65102	0.325189	100.20 %	600.203	300	7	62.2096	11598	388	0
AFLP200_600_17.txt	0.638304	0.314125	103.20 %	600.2	289	8	60.349	15904	410	0
AFLP200_600_18.txt	0.6323	0.33267	90.07 %	600.213	307	8	68.0135	12515	377	0
AFLP200_600_19.txt	0.615107	0.313198	96.40 %	600.318	286	11	56.3556	12566	352	0
AFLP200_600_20.txt	0.6439	0.320984	100.60 %	600.197	292	12	65.2046	11930	360	0
AFLP200_600_21.txt	0.637969	0.303461	110.23 %	600.203	271	19	55.0104	12938	343	0
AFLP200_600_22.txt	0.632087	0.286741	120.44 %	600.191	256	9	55.8461	13315	363	0
AFLP200_600_23.txt	0.638232	0.36415	75.27 %	600.266	340	16	68.6474	11098	387	1
AFLP200_600_24.txt	0.620213	0.430649	44.02 %	600.189	409	34	70.6574	9916	394	0
AFLP200_600_25.txt	0.628174	0.33976	84.89 %	600.18	317	7	63.1194	11954	399	0
AFLP200_600_26.txt	0.647069	0.334768	93.29 %	600.253	309	9	66.6647	11431	357	0
AFLP200_600_27.txt	0.655157	0.424826	54.22 %	600.257	408	25	66.0923	10612	429	1
AFLP200_600_28.txt	0.663546	0.327436	102.65 %	600.193	304	8	58.0796	12140	353	0
AFLP200_600_29.txt	0.643519	0.347675	85.09 %	600.241	325	9	63.8914	10806	356	0
AFLP200_600_30.txt	0.642806	0.322536	99.30 %	600.182	291	15	69.0077	12105	367	0

APÉNDICE C

SOLUCIONES DE BRKGA

A continuación, se detalla la información de la cota dual obtenida por CPLEX en una hora de tiempo de computo y el detalle de las soluciones obtenidas por el BRKGA propuesto, para todas las instancias. En particular, se presentan la cota primal, tiempo de computo, GAP y valores de los indicadores.

Tabla 15: Valores obtenidos para instancias tipo A usando metaheurística BRKGA.

Instancias	Cota dual	Función objetivo	GAP	Tiempo	<i>aa</i>	<i>ac</i>	<i>at</i>	<i>an</i>	<i>ao</i>	<i>as</i>
AFLP40_120_1.txt	0.424774	0.418798	1.43 %	4.40911	81	4	18.8627	2165	80	1
AFLP40_120_2.txt	0.450161	0.437137	2.98 %	4.25777	85	5	18.6385	2121	80	1
AFLP40_120_3.txt	0.441336	0.433872	1.72 %	4.66524	84	6	18.2043	2120	80	1
AFLP40_120_4.txt	0.41676	0.407973	2.15 %	4.40184	77	9	16.3967	2023	68	1
AFLP40_120_5.txt	0.421532	0.414179	1.78 %	4.45615	80	3	19.3183	2244	80	1
AFLP40_120_6.txt	0.420822	0.413635	1.74 %	7.54091	80	6	14.9423	2122	74	1
AFLP40_120_7.txt	0.442299	0.433263	2.09 %	4.43307	85	5	15.427	2051	82	2
AFLP40_120_8.txt	0.387361	0.382778	1.20 %	4.3263	74	2	16.1383	2335	74	0
AFLP40_120_9.txt	0.439369	0.434098	1.21 %	4.1613	85	6	14.7895	1990	79	0
AFLP40_120_10.txt	0.422008	0.418254	0.90 %	4.24863	80	6	19.5948	2213	80	2
AFLP40_120_11.txt	0.452473	0.442212	2.32 %	3.90768	84	12	16.3197	1931	75	3
AFLP40_120_12.txt	0.416851	0.412291	1.11 %	4.16868	81	1	18.1053	2381	81	0
AFLP40_120_13.txt	0.415297	0.407423	1.93 %	4.19997	78	8	14.8268	2326	72	1
AFLP40_120_14.txt	0.40963	0.402305	1.82 %	4.17513	78	1	19.751	2234	79	1
AFLP40_120_15.txt	0.422551	0.416405	1.48 %	4.13097	80	4	20.3016	2081	79	1
AFLP40_120_16.txt	0.41768	0.411282	1.56 %	4.41224	80	5	14.3565	2111	77	0
AFLP40_120_17.txt	0.422211	0.413494	2.11 %	4.2781	79	7	17.0724	2055	72	0
AFLP40_120_18.txt	0.447594	0.439068	1.94 %	4.32947	84	6	22.0973	2059	79	1
AFLP40_120_19.txt	0.427019	0.419157	1.88 %	4.16591	81	7	15.4888	2092	77	4
AFLP40_120_20.txt	0.427039	0.420762	1.49 %	4.18878	82	4	17.026	2199	80	1
AFLP40_120_21.txt	0.427725	0.420073	1.82 %	4.31088	81	7	15.4478	2041	76	1
AFLP40_120_22.txt	0.414566	0.407442	1.75 %	4.37198	77	6	20.9151	2215	72	1
AFLP40_120_23.txt	0.437032	0.429708	1.70 %	5.48589	83	6	17.4422	2048	83	1
AFLP40_120_24.txt	0.401629	0.396112	1.39 %	5.01689	77	3	15.6264	2264	76	1
AFLP40_120_25.txt	0.428158	0.420284	1.87 %	6.06525	80	7	19.2602	2088	80	1
AFLP40_120_26.txt	0.439778	0.430809	2.08 %	4.48553	83	9	14.9379	2073	75	1
AFLP40_120_27.txt	0.435304	0.430105	1.21 %	4.22874	84	3	18.5847	2060	84	1
AFLP40_120_28.txt	0.423844	0.417956	1.41 %	3.90851	81	4	17.57	2137	80	1
AFLP40_120_29.txt	0.400562	0.393945	1.68 %	4.46231	75	4	18.8805	2267	72	2
AFLP40_120_30.txt	0.453274	0.444643	1.94 %	4.47455	86	8	16.7342	2024	78	2

Tabla 16: Valores obtenidos para instancias tipo B usando metaheurística BRKGA.

Instancias	Cota dual	Función objetivo	GAP	Tiempo	<i>aa</i>	<i>ac</i>	<i>at</i>	<i>an</i>	<i>ao</i>	<i>as</i>
AFLP80_250_1.txt	0.481962	0.468191	2.94 %	21.6201	193	12	29.8573	3959	184	1
AFLP80_250_2.txt	0.487722	0.476745	2.30 %	30.5928	194	13	38.7947	3828	183	1
AFLP80_250_3.txt	0.480829	0.466718	3.02 %	25.6456	191	13	32.3896	3952	183	1
AFLP80_250_4.txt	0.557805	0.479855	16.24 %	35.9508	196	17	32.4461	3896	190	1
AFLP80_250_5.txt	0.461464	0.446765	3.29 %	19.2166	180	17	31.0235	4002	179	1
AFLP80_250_6.txt	0.486718	0.479348	1.54 %	16.7489	197	11	35.6463	3800	190	1
AFLP80_250_7.txt	0.494226	0.478948	3.19 %	22.6297	198	9	34.071	3855	191	1
AFLP80_250_8.txt	0.473919	0.461809	2.62 %	27.3216	190	9	33.0569	3917	188	0
AFLP80_250_9.txt	0.493621	0.447646	10.27 %	24.3453	183	11	29.7505	4119	187	1
AFLP80_250_10.txt	0.484856	0.460759	5.23 %	23.0964	190	11	29.5414	4098	190	1
AFLP80_250_11.txt	0.494563	0.446502	10.76 %	18.3978	182	12	30.5507	4179	180	2
AFLP80_250_12.txt	0.468702	0.457904	2.36 %	19.105	186	14	32.919	3883	177	1
AFLP80_250_13.txt	0.502574	0.454683	10.53 %	21.8071	185	11	34.6037	3930	175	1
AFLP80_250_14.txt	0.471009	0.463898	1.53 %	20.6122	190	10	35.2381	4195	189	2
AFLP80_250_15.txt	0.474708	0.465824	1.91 %	32.8207	189	13	37.3761	3800	181	0
AFLP80_250_16.txt	0.534491	0.474786	12.58 %	22.417	194	9	40.7746	3884	189	1
AFLP80_250_17.txt	0.526576	0.470869	11.83 %	19.7648	193	15	30.3891	3854	181	0
AFLP80_250_18.txt	0.473331	0.453519	4.37 %	19.9341	184	17	28.3972	3922	172	1
AFLP80_250_19.txt	0.523093	0.459269	13.90 %	19.9921	188	9	35.2672	4079	190	2
AFLP80_250_20.txt	0.468818	0.455779	2.86 %	32.7027	186	12	32.4514	4027	184	0
AFLP80_250_21.txt	0.479135	0.468419	2.29 %	18.0597	192	12	33.2975	3964	184	1
AFLP80_250_22.txt	0.460613	0.445644	3.36 %	27.5464	181	9	35.7498	4083	182	1
AFLP80_250_23.txt	0.474382	0.461776	2.73 %	20.5132	187	11	41.1847	4095	188	1
AFLP80_250_24.txt	0.529327	0.489134	8.22 %	20.9975	202	11	35.6516	3913	201	1
AFLP80_250_25.txt	0.483526	0.472686	2.29 %	19.3902	193	15	32.533	3783	182	1
AFLP80_250_26.txt	0.538319	0.486694	10.61 %	28.7271	199	10	42.6994	3798	193	1
AFLP80_250_27.txt	0.496772	0.48639	2.13 %	19.6607	198	15	38.7869	3703	193	1
AFLP80_250_28.txt	0.489081	0.478069	2.30 %	25.478	197	16	26.7565	3885	188	2
AFLP80_250_29.txt	0.548361	0.478713	14.55 %	18.1821	194	18	35.0967	3833	191	2
AFLP80_250_30.txt	0.488874	0.475169	2.88 %	17.8515	192	18	36.8216	4056	190	1

Tabla 17: Valores obtenidos para instancias tipo C usando metaheurística BRKGA.

Instancias	Cota dual	Función objetivo	GAP	Tiempo	<i>aa</i>	<i>ac</i>	<i>at</i>	<i>an</i>	<i>ao</i>	<i>as</i>
AFLP200_600_1.txt	0.622126	0.490315	26.88 %	226.525	481	33	82.2795	9050	481	0
AFLP200_600_2.txt	0.649251	0.491003	32.23 %	159.821	482	28	87.9713	9020	491	1
AFLP200_600_3.txt	0.635212	0.485786	30.76 %	167.773	479	26	79.9087	9061	487	1
AFLP200_600_4.txt	0.648576	0.479983	35.12 %	154.024	469	34	79.5157	8963	455	1
AFLP200_600_5.txt	0.642421	0.487725	31.72 %	195.309	478	34	81.1054	9228	469	0
AFLP200_600_6.txt	0.617758	0.472014	30.88 %	164.272	462	27	80.8999	9296	457	1
AFLP200_600_7.txt	0.616815	0.479787	28.56 %	155.481	468	35	81.6738	9118	474	1
AFLP200_600_8.txt	0.635426	0.483741	31.36 %	190.025	473	36	79.2515	9090	463	1
AFLP200_600_9.txt	0.636627	0.481904	32.11 %	192.103	470	42	75.3391	9046	451	0
AFLP200_600_10.txt	0.624097	0.477671	30.65 %	151.088	468	31	77.6477	9029	468	1
AFLP200_600_11.txt	0.633292	0.464265	36.41 %	277.254	455	32	67.1973	9502	457	1
AFLP200_600_12.txt	0.653707	0.487068	34.21 %	234.027	480	31	74.8596	8849	475	1
AFLP200_600_13.txt	0.644479	0.475027	35.67 %	228.426	468	28	71.1262	9113	460	1
AFLP200_600_14.txt	0.648556	0.496058	30.74 %	280.399	487	36	81.197	8735	474	1
AFLP200_600_15.txt	0.64148	0.484403	32.43 %	204.789	476	31	77.2926	8924	465	1
AFLP200_600_16.txt	0.65102	0.481671	35.16 %	162.562	472	37	73.3859	8943	456	1
AFLP200_600_17.txt	0.638304	0.491916	29.76 %	164.192	483	38	75.4586	8768	466	1
AFLP200_600_18.txt	0.6323	0.477685	32.37 %	144.999	465	37	80.6121	9066	464	1
AFLP200_600_19.txt	0.615107	0.477968	28.69 %	184.171	472	26	71.6461	9035	482	1
AFLP200_600_20.txt	0.6439	0.489974	31.42 %	262.441	480	43	71.486	8896	463	0
AFLP200_600_21.txt	0.637969	0.473334	34.78 %	171.414	465	30	71.6659	9089	468	1
AFLP200_600_22.txt	0.632087	0.470205	34.43 %	153.73	457	33	82.6309	9085	456	1
AFLP200_600_23.txt	0.638232	0.483152	32.10 %	183.402	474	34	76.2664	9074	468	1
AFLP200_600_24.txt	0.620213	0.482929	28.43 %	260.203	467	49	79.789	8904	435	0
AFLP200_600_25.txt	0.628174	0.47945	31.02 %	159.663	470	31	78.5997	9165	472	1
AFLP200_600_26.txt	0.647069	0.490816	31.84 %	284.614	483	20	94.2328	9417	490	1
AFLP200_600_27.txt	0.655157	0.484855	35.12 %	188.725	477	30	77.7975	8915	468	1
AFLP200_600_28.txt	0.663546	0.481213	37.89 %	176.321	468	36	87.0302	9157	467	0
AFLP200_600_29.txt	0.643519	0.480965	33.80 %	260.711	474	26	77.7553	9014	492	0
AFLP200_600_30.txt	0.642806	0.476968	34.77 %	175.402	468	32	74.1569	9255	455	1

BIBLIOGRAFÍA

- Azizan, M., Go, T., Hatta, W., Lim, C., and Teoh, S. (2017). Comparison of Emergency Medical Services Delivery Performance Using Maximal Covering Location and Gradual Cover Location Problems. *International Journal of Electrical and Computer Engineering*, 7(5):2791–2797.
- Bagherinejad, J., Bashiri, M., and Nikzad, H. (2018). General form of a cooperative gradual maximal covering location problem. *Journal of Industrial Engineering International*, 14(2):241–253.
- Berman, O., Drezner, Z., Krass, D., and Wesolowsky, G. O. (2009). The variable radius covering problem. *European Journal of Operational Research*, 196(2):516–525.
- Bertolini, L., Le Clercq, F., and Kapoen, L. (2005). Sustainable accessibility: A conceptual framework to integrate transport and land use plan-making. the test-applications in the netherlands and a reection on the way forward. *Transport Policy*, 12:207–220.
- Bhat, C., Handy, S., Kockelman, K., Mahmassani, H., and Chen, Q. (2000). Development of an urban accessibility index: Literature review. *Research project conducted for the Texas Department of Transportation. Center for Transportation Research, University of Texas, Austin (TX), USA.*
- Boonmee, C., Arimura, M., and Asada, T. (2017). Facility location optimization model for emergency humanitarian logistics. *International Journal of Disaster Risk Reduction*, 24:485–498.
- Bruno, G., Genovese, A., Piccolo, C., and Sterle, C. (2014). A Location Model for the Reorganization of a School System: The Italian Case Study. *Procedia - Social and Behavioral Sciences*, 108:96–105.
- Daskin, M. S. and Dean, L. K. (2004). *Location of Health Care Facilities*, pages 43–76. Springer US, Boston, MA.

- Doerner, K., Focke, A., and Gutjahr, W. J. (2007a). Multicriteria tour planning for mobile healthcare facilities in a developing country. *European Journal of Operational Research*, 179(3):1078 – 1096.
- Doerner, K., Focke, A., and Gutjahr, W. J. (2007b). Multicriteria tour planning for mobile healthcare facilities in a developing country. *European Journal of Operational Research*, 179(3):1078–1096.
- Doerner, K., Gutjahr, W., Hartl, R., Karall, M., and Reimann, M. (2005). Heuristic Solution of an Extended Double-Coverage Ambulance Location Problem for Austria. *Central European Journal of Operations Research*, 13(4).
- Drezner, Z., Wesolowsky, G., and Drezner, T. (2004). The gradual covering problem. *Naval Research Logistics*, 51(6):841–855.
- Eiselt, H. and Marianov, V. (2009). Gradual location set covering with service quality. *Socio-Economic Planning Sciences*, 43(2):121–130.
- Erdemir, E. T., Batta, R., Rogerson, P. A., Blatt, A., and Flanigan, M. (2010). Joint ground and air emergency medical services coverage models: A greedy heuristic solution approach. *European Journal of Operational Research*, 207(2):736–749.
- Eydi, A. and Torabi, H. (2019). A maximal covering problem in supply chain considering variable radius and gradual coverage with the choice of transportation mode. *Arabian Journal for Science and Engineering*, 44(8):7219–7233.
- Forghani Youshanlo, M. and Sahraeian, R. (2015). *Dynamic Multi-objective Maximal Covering Location Problem with Gradual Coverage*, pages 39–47. Springer International Publishing, Cham.
- Gendreau, M., Laporte, G., and Semet, F. (2001). A dynamic model and parallel tabu search heuristic for real-time ambulance relocation. *Parallel Computing*, 27(12):1641 – 1653.
- Gonçalves, J. and Resende, M. (2011). Biased random-key genetic algorithms for combinatorial optimization. *Journal Heuristics*, 17:487–525.
- Ibarra-Rojas, O., Hernandez-Landa, L., and Ozuna, L. (2018). The accessibility vehicle routing problem. *Journal of Cleaner Production*, 172:1514–1528.
- Ibarra-Rojas, O., Ozuna, L., and López-Piñón, D. (2019). The maximal covering location problem with accessibility indicators. *Socio-Economic Planning Sciences*, page 100758.

- Jabalameh, M., Behzad, B., and Mohammad, M. (2010). Capacitated Facility Location Problem with Variable Coverage Radius in Distribution System. volume 21, pages 231–237.
- Karatas, M. (2017). A multi-objective facility location problem in the presence of variable gradual coverage performance and cooperative cover. *European Journal of Operational Research*, 262(3):1040–1051.
- Karatas, M., Razib, N., and Tozanc, H. (2016). A comparison of p-median and maximal coverage location models with Q-coverage requirement. *Procedia Engineering*, 149:169–176.
- Makrí, M. and Folkesson, C. (1999). Accessibility measures for analyses of land use and travelling with geographical information systems. In *Proceedings of the 2nd KFB-Research Conference*, pages 1–17, Lund, Sweden.
- Scheurer, J. and Curtis, C. (2007). Accessibility measures: overview and practical applications. *Department of Urban and Regional Planning, Curtin University*, 52.
- Stummer, C., Doerner, K., Focke, A., and Heidenberger, K. (2004). Determining Location and Size of Medical Departments in a Hospital Network: A Multiobjective Decision Support Approach. *Health Care Management Science*, 7(1):63–71.
- Talbi, E.-G. (2009). *Metaheuristics: From Design to Implementation*. WILEY.