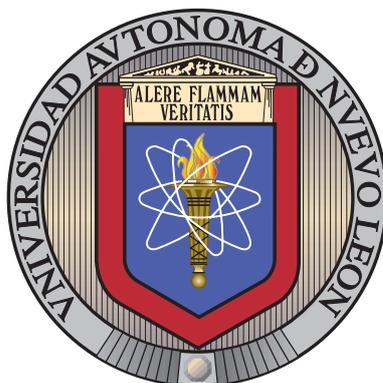


UNIVERSIDAD AUTÓNOMA DE NUEVO LEÓN

FACULTAD DE CIENCIAS FÍSICO MATEMÁTICAS



UN PROBLEMA DE LOCALIZACIÓN DE
INSTALACIONES CON PREFERENCIAS DE LOS
USUARIOS APLICADO A LA RESPONSABILIDAD
SOCIAL

POR

ÁNGELES SAHORI POLINO MARTÍNEZ

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

SAN NICOLÁS DE LOS GARZA, NUEVO LEÓN

JUNIO 2021

UNIVERSIDAD AUTÓNOMA DE NUEVO LEÓN

FACULTAD DE CIENCIAS FÍSICO MATEMÁTICAS



UN PROBLEMA DE LOCALIZACIÓN DE
INSTALACIONES CON PREFERENCIAS DE LOS
USUARIOS APLICADO A LA RESPONSABILIDAD
SOCIAL

POR

ÁNGELES SAHORI POLINO MARTÍNEZ

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

SAN NICOLÁS DE LOS GARZA, NUEVO LEÓN

JUNIO 2021

Universidad Autónoma de Nuevo León
Facultad de Ciencias Físico Matemáticas

Los miembros del Comité de Tesis recomendamos que la Tesis “Un problema de localización de instalaciones con preferencias de los usuarios aplicado a la Responsabilidad Social”, realizada por el alumno Ángeles Sahori Polino Martínez, con número de matrícula 1579047, sea aceptada para su defensa como opción al grado de Maestría en Ciencias con Orientación en Matemáticas.

El Comité de Tesis

Dr. José Fernando Camacho Vallejo
Director

Dr. Juan Guillermo Villegas Ramírez
Co Director

Dr. Omar Jorge Ibarra Rojas
Revisor

Vo. Bo.

Dr. Omar Jorge Ibarra Rojas
Coordinar del Posgrado en Ciencias con
Orientación en Matemáticas

San Nicolás de los Garza, Nuevo León, Junio 2021

ÍNDICE GENERAL

Agradecimientos	x
Resumen	xI
1. Introducción	1
1.1. Objetivo	2
1.1.1. Objetivo general	2
1.1.2. Objetivos específicos	2
1.2. Descripción del problema	2
1.3. Motivación	4
1.4. Metodología	4
1.5. Estructura	5
2. Revisión de literatura	6
3. Modelación matemática	11
4. Reformulación a un solo nivel	14
4.1. Pruebas computacionales para la reformulación	17
5. Un algoritmo basado en reencadenamiento de trayectorias	18
5.1. Preliminares	18

5.2. Algoritmo propuesto	20
5.3. Análisis de la estructura de las soluciones	22
5.3.1. Representación de la solución del nivel superior	22
5.3.2. Generación de las soluciones	23
5.3.3. Fase de reparación	24
5.4. Nivel inferior	25
5.5. Fase de ajuste	27
5.6. Búsqueda Local	30
5.7. Generar la trayectoria entre dos soluciones	32
6. Experimentación	35
6.1. Análisis de la estructura escolar en Nuevo León	35
6.2. Instancias de prueba	38
6.3. Casos de estudio	39
6.3.1. Instancia A	42
6.3.2. Instancia B	43
7. Análisis de resultados	45
7.1. Resultados para las instancias de prueba	46
7.2. Resultados para los casos de estudio	49
7.2.1. Resultados para la Instancia A	50
7.2.2. Resultados para la Instancia B	56
8. Conclusiones	61
8.1. Trabajo futuro	65

ÍNDICE DE FIGURAS

1.	Mapa del municipio de Apodaca y escuelas públicas en la zona	40
2.	Mapa del municipio de Apodaca y escuelas públicas divididas por zonas . .	41
3.	Mapa de la Zona 4 del municipio de Apodaca, escuelas públicas correspondientes y estudiantes inscritos en el programa.	43
4.	Mapa de la Zona 7 del municipio de Apodaca, escuelas públicas correspondientes y estudiantes inscritos en el programa	44
5.	Mapa de la Instancia A con la respectiva asignación de los estudiantes a la escuela 1	50
6.	Mapa de la Instancia A con la respectiva asignación de los estudiantes a la escuela 2	51
7.	Mapa de la Instancia A con la respectiva asignación de los estudiantes a la escuela 3	51
8.	Mapa de la Instancia A con la respectiva asignación de los estudiantes a la escuela 4	52
9.	Mapa de la Instancia A con la respectiva asignación de los estudiantes a la escuela 5	52
10.	Mapa de la Instancia A con la respectiva asignación de los estudiantes a la escuela 6	53
11.	Mapa de la Instancia A con la respectiva asignación de los estudiantes a la escuela 7	53
12.	Mapa de la Instancia A con la respectiva asignación de los estudiantes a la escuela 8	54

13.	Mapa de la Instancia A con la respectiva asignación de los estudiantes a la escuela 10	54
14.	Mapa de la Instancia A con la respectiva asignación de los estudiantes a la escuela 11	55
15.	Mapa de la Instancia A y ubicación de la escuela 9	55
16.	Mapa de la Instancia B con la respectiva asignación de los estudiantes a la escuela 1	56
17.	Mapa de la Instancia B con la respectiva asignación de los estudiantes a la escuela 2	57
18.	Mapa de la Instancia B con la respectiva asignación de los estudiantes a la escuela 3	57
19.	Mapa de la Instancia B con la respectiva asignación de los estudiantes a la escuela 4	58
20.	Mapa de la Instancia B con la respectiva asignación de los estudiantes a la escuela 5	58
21.	Mapa de la Instancia B con la respectiva asignación de los estudiantes a la escuela 6	59
22.	Mapa de la Instancia B con la respectiva asignación de los estudiantes a la escuela 7	59

ÍNDICE DE TABLAS

1.	Modelos de localización y variantes	10
2.	Representación de la solución para el ejemplo ilustrativo.	23
3.	Vector de números aleatorios	24
4.	Codificación de la solución generada.	24
5.	Representación de la solución reparada	25
6.	Solución inicial	30
7.	Posibles intercambios (Vecindario)	31
8.	Vecindario a explorar con la búsqueda local	31
9.	Solución inicial	33
10.	Solución guía	33
11.	Trayectoria generada entre la solución inicial y solución guía	34
12.	Supervisiones con 21 escuelas o más a su cargo	37
13.	Resultados de las instancias con 8 talleres y 10 escuelas	47
14.	Resultados de las instancias con 8 talleres y 15 escuelas	47
15.	Resultados de las instancias con 8 talleres y 20 escuelas	47
16.	Resultados de las instancias con 10 talleres y 10 escuelas	48
17.	Resultados de las instancias con 10 talleres y 15 escuelas	48
18.	Resultados de las instancias con 10 talleres y 20 escuelas	48

RESUMEN

En esta tesis se presenta un problema de localización de instalaciones con capacidad, donde la asignación de los usuarios se realiza en base a sus preferencias. El problema se desarrolla para aplicarlo en un contexto de responsabilidad social. Dicha aplicación consiste en una situación donde el gobierno desea abrir talleres extracurriculares para estudiantes de nivel primaria, de forma que los talleres ayuden al desarrollo de sus capacidades académicas, deportivas y/o sociales.

El problema se formula con un modelo de programación binivel, en el cual el gobierno está asignado al nivel superior para minimizar los costos de localización de los talleres en las diferentes escuelas y los costos de asignación de estudiantes a cada taller. El nivel inferior está asociado a la sociedad de padres de familia, quienes buscan optimizar las preferencias de los estudiantes hacia los talleres.

Para resolver el problema, se realiza una reformulación de la cual resulta un problema de un solo nivel. Sin embargo, dicha reformulación resulta efectiva únicamente para instancias pequeñas. Por esto, se justifica el uso de un algoritmo heurístico. Se propone un algoritmo basado en el reencadenamiento de trayectorias, el cual resulta ser efectivo y eficiente, ya que obtiene mejores cotas que la reformulación y mejora el tiempo consumido de manera considerable. Además, se utiliza el algoritmo propuesto para resolver un caso de estudio del municipio de Apodaca, Nuevo León.

Palabras clave: localización de instalaciones, preferencias, instalaciones no homogéneas, demanda unitaria, reencadenamiento de trayectorias.

CAPÍTULO 1

INTRODUCCIÓN

Responsabilidad social es un concepto que en la última década ha tenido un auge muy importante en las organizaciones de diversos ámbitos, por ejemplo, en empresas, organismos gubernamentales, asociaciones civiles, universidades, entre otros. Incluso se habla también de responsabilidad social individual. Esto se ha dado debido a el aumento en la conciencia del medio ambiente, los derechos humanos y la convivencia social. Surgió inicialmente por la necesidad de las empresas de mejorar su imagen ante los clientes y con esto tener una mejor posición en el mercado frente a la competencia. Realizar estas consideraciones resultó en una mejor relación con el entorno laboral y comunitario, procurando los derechos de los trabajadores, las medidas de seguridad y la aplicación de criterios no discriminatorios.

La responsabilidad social, según la Organización Internacional de Normalización en su norma ISO 26000, es la responsabilidad de una organización ante los impactos que sus decisiones y actividades ocasionan en la sociedad y el medio ambiente, mediante un comportamiento ético y transparente que contribuya al desarrollo sostenible, incluyendo la salud y el bienestar de la sociedad tomando en consideración las expectativas de sus partes interesadas. Además, debe cumplir con la legislación aplicable y debe ser coherente con la normativa internacional de comportamiento. Es importante recalcar, que dichas políticas deben estar integradas en toda la organización y deben llevarse a la práctica en sus relaciones.

Recientemente, muchas de las empresas y organizaciones se han preocupado por considerar este aspecto dentro de sus políticas y por esto han puesto en marcha programas de responsabilidad social además de incluir como parte de sus estrategias de negocio actividades sociales y ambientales para el bienestar de la población. Como ejemplo podemos mencionar la labor conjunta de la Universidad Autónoma de Nuevo León y el gobierno del estado, los cuales se unieron en un proyecto de responsabilidad social con la finalidad de mejorar el entorno social para reducir el índice de delitos en la colonia Independencia. Por

ello en el año 2013, se invirtió en un nuevo plantel de la Escuela Industrial y Preparatoria Técnica “Álvaro Obregón” ubicado en la colonia mencionada, en el cual los estudiantes cursarían sus estudios de bachillerato y carrera técnica con beca completa. Dentro del proyecto también se impartieron cursos con valor curricular de forma gratuita para que los alumnos tuvieran una mejor formación académica lo cual conlleva a tener mejores oportunidades y de esta forma alejarlos de los problemas delictivos de la comunidad. Otro ejemplo, son empresas como Walmart, Oxxo y Lala, las cuales donan productos alimenticios en buen estado, que por diversas razones no serán vendidos, a organizaciones que se encargan de distribuir estos productos a personas que lo necesitan.

1.1 OBJETIVO

1.1.1 OBJETIVO GENERAL

El objetivo de esta tesis es analizar un problema de localización de instalaciones donde la asignación se realiza considerando las preferencias de los usuarios. Además, se consideran algunas características interesantes inherentes al problema en cuestión, por ejemplo, capacidad en las instalaciones, demanda unitaria, e instalaciones no homogéneas.

1.1.2 OBJETIVOS ESPECÍFICOS

- Plantear un modelo de programación matemática para el problema bajo estudio.
- Reformular el modelo binivel en uno de un solo nivel y verificar si la reformulación es conveniente para resolver instancias medianas y grandes.
- Proponer un algoritmo heurístico para resolver el problema binivel.
- Realizar pruebas computacionales para analizar el desempeño del algoritmo propuesto.
- Aplicar el problema a un caso de estudio basado en la ciudad de Apodaca, Nuevo León.

1.2 DESCRIPCIÓN DEL PROBLEMA

El problema consiste en decidir cuales talleres ofrecer para el desarrollo académico, social y/o cultural de los estudiantes en las diferentes escuelas disponibles de la región determinada. Se considera un presupuesto limitado para ofrecer dichos talleres, hay un

cupo máximo de estudiantes que pueden inscribirse en cada taller, el cual es definido por las características de la infraestructura de cada escuela. Además, para no sesgar toda la oferta de talleres en una sola escuela, se considera que se pueden ofrecer un número máximo de talleres en cada escuela. Se debe considerar que los talleres que se decidan ofrecer tengan capacidad suficiente para atender a todos los estudiantes dentro del programa, y que exista variedad de los diferentes talleres ofrecidos.

Como este proyecto gubernamental es de responsabilidad social, se asume que el gobierno va a implementar mecanismos para trasladar gratuitamente a los estudiantes desde sus casas hasta las escuelas donde se impartirán los talleres que cursarán. Dado esto, el gobierno busca minimizar los costos totales de dicho proyecto social. Esto es, minimizar los costos de abrir los diferentes talleres en las escuelas y los costos de transportación de los estudiantes. Sin embargo, el gobierno no puede imponer a cual taller deberá asistir un estudiante, ya que si se asigna a un taller que no sea de su agrado, el estudiante no desarrollará las competencias esperadas. Más aún, el estudiante podría verse desmotivado, ocasionando que deje de asistir al taller. Es por esto, que se debe tomar en cuenta la preferencia de los estudiantes hacia los talleres. Por ejemplo, algunos de ellos preferirán estar en un taller de matemáticas en lugar de asistir a uno de pintura. Es importante resaltar que, a pesar de tener el transporte gratuito, el alumno podría considerar dejar de asistir a su taller asignado por el tiempo de traslado que este involucra, sin importar que sea asignado a su taller preferido. Entonces, los estudiantes en conjunto con sus familias calificarán las opciones de los talleres y escuelas haciendo un ordenamiento en base a sus preferencias (colocando en primer lugar el taller que sea de mayor agrado en la escuela que sea preferida para ellos, y así sucesivamente). La inscripción de estudiantes a talleres se realiza bajo la supervisión de la sociedad de padres de familia para mediar cualquier tipo de conflicto que pudiera existir.

Claramente, el problema en cuestión consta de dos decisiones, el gobierno decide cuales talleres abrir en cuales escuelas buscando minimizar los costos de dicho proyecto social, mientras que la sociedad de padres inscribe a los estudiantes en los talleres ofrecidos basado en la preferencia de dichos estudiantes. Es por esto, que el problema se abordará con un enfoque de programación binivel donde el gobierno está asociado al nivel superior y la sociedad de padres al nivel inferior, en donde se busca minimizar la suma total de las preferencias.

1.3 MOTIVACIÓN

En el ámbito de la responsabilidad social, el gobierno tiene un papel fundamental. Por un lado, debe motivar a las organizaciones a ser socialmente responsables y vigilar que cumplan con las medidas establecidas en cuanto a derechos humanos y cuidado del medio ambiente. Por otro lado, como es el encargado de velar por la seguridad de la ciudadanía en todos los aspectos (físicos, salud, educación, etc.), debe de fomentar esta práctica y preocuparse por ser un “gobierno socialmente responsable”, el cual incluya políticas y un modelo de responsabilidad social bien definido como parte de sus acciones.

Debido a esto, es muy importante que el gobierno realice proyectos que contribuyan de manera positiva a la sociedad. Sin embargo, es razonable asumir que existe un presupuesto limitado para los programas de responsabilidad social que se implementen. En particular, en esta tesis se va a investigar sobre un proyecto gubernamental enfocado a ofrecer talleres extra-escolares para mejorar el nivel académico, social y/o cultural de los estudiantes de educación primaria de una región determinada. Con la implementación de este proyecto se pretende mejorar el conocimiento de los estudiantes en áreas de su interés dentro del ámbito académico, social y cultural. Así mismo, los talleres serán un mecanismo para mantener a los estudiantes ocupados en tareas productivas y evitar así que estos se encuentren en actividades delictivas.

1.4 METODOLOGÍA

- Revisión de literatura de programación binivel
- Revisión de literatura de localización de instalaciones y variantes del mismo
- Reformulación en base a la teoría de dualidad
- Realizar pruebas para la eficiencia de la reformulación
- Diseño del algoritmo basado en el reencadenamiento de trayectorias
- Implementación del algoritmo en FICO Xpress para instancias aleatorias
- Comparación de resultados de la reformulación y los resultados del algoritmo propuesto
- Caso de estudio

1.5 ESTRUCTURA

El resto de esta tesis está organizado como se describe a continuación. El Capítulo 2 presenta una revisión de literatura de problemas de localización de instalaciones y algunas de sus variantes, las cuales serán utilizadas para la formulación del problema. En el Capítulo 3 se presenta el modelo matemático planteado para el problema, el cual se formula como un modelo de programación binivel. El problema se reformula como un problema de un solo nivel en el Capítulo 4. En el Capítulo 5, se describe el algoritmo heurístico basado en el reencadenamiento de trayectorias diseñado para resolver el problema bajo estudio.

En el Capítulo 6, se describe la estructura escolar presente en el estado de Nuevo León para construir el caso de estudio. Además, se describe la experimentación computacional realizada y se define formalmente el caso de estudio. En el Capítulo 7, se presenta el análisis de los resultados. Finalmente, en el Capítulo 8 se presentan algunas de las conclusiones obtenidas en este trabajo y se mencionan algunas líneas de investigación futuras.

CAPÍTULO 2

REVISIÓN DE LITERATURA

En el área de investigación de operaciones, uno de los problemas clásicos y que han sido base para muchas investigaciones y aplicaciones son los problemas de localización de instalaciones, este problema consiste en lo siguiente: dado un conjunto de sitios potenciales y un conjunto de clientes, tenemos dos decisiones a considerar, primero en cuales sitios abrir las instalaciones y después a cuales clientes asignar en cada instalación abierta, de forma que se optimice una función objetivo cumpliendo con las demandas de los clientes sin rebasar la capacidad de las plantas abiertas (si lo hubiese).

Este problema ha sido muy estudiado en la literatura ya que tiene diversas aplicaciones, decidir donde abrir una escuela, hospitales, donde localizar ambulancias para que acudan en un tiempo mínimo a los llamados de emergencia, etc., por esta razón han surgido muchas variantes, algunas de ellas, problemas sin capacidades, con capacidades, demanda unitaria, cuando se abre un número determinado de instalaciones, entre otras variantes. En [?] y [?] podemos encontrar algunas variantes de los problemas de localización y ejemplos de aplicación, así como su modelo matemático y algunos métodos para resolver estos problemas.

Tal y como se mencionó, en algunas de estas variantes se consideran clientes con demanda unitaria, esto es, la demanda de todos los clientes es igual a uno. Esto se aplica cuando las instalaciones brindan algún tipo de servicio a los clientes, por ejemplo, alumnos que se asignan a una escuela, cuando la población se debe atender en centros de salud, cuando se conectan PC's a puertos de una estación o servidor, etc. En estos casos la demanda de clientes se satisface al ser asignados a una instalación con capacidad para atenderlo.

Por ejemplo en [34] se trata un problema de demanda unitaria en el cual se busca ubicar puntos de agregación de datos universal (UDAP) los cuales permiten el transpor-

te de información proveniente de medidores inteligentes. El modelo busca minimizar la cantidad de UDAP activos, con el objetivo de cubrir el mayor porcentaje de los usuarios usando la menor cantidad de UDAP garantizando un cubrimiento mínimo. Cada UDAP tiene cierta capacidad de usuarios que pueden estar conectados a él, el cliente se considera cubierto si es atendido por el UDAP al que está conectado, esto es, tiene una demanda unitaria.

Otra variante que aparece en problemas de localización es cuando las instalaciones no son homogéneas, es decir, cuando se consideran diferentes tipos de instalaciones por abrir. En este caso las variables de localización son de la forma y_{ij} donde $y_{ij} = 1$ si se abre el tipo de planta j en el sitio potencial i y $y_{ij} = 0$ si no.

Un ejemplo de aplicación de este tipo de problemas es [32], en donde se busca localizar instalaciones bancarias. Los bancos ofrecen diversos servicios y para ello cuenta con diferentes tipos de instalaciones, cajeros automáticos, sucursales bancarias de oficinas y bancos principales. El objetivo es localizar estos tres tipos de instalaciones de acuerdo a tres objetivos, (1) Maximización de la rentabilidad del mercado; (2) Maximización del poder de atracción de clientes; (3) Minimización de todos los riesgos asociados con los compromisos de recursos hechos para abrir bancos. Para esto se crea una variable de decisión $y_{ik} = 1$ si la instalación bancaria de tipo k se abre en el sitio potencial i y se propone un modelo estocástico de tres ubicaciones jerárquicas para organizar sucesivamente los patrones espaciales de las instalaciones bancarias.

Otro ejemplo de instalaciones no homogéneas es [16], en el cual presenta el desarrollo de un sistema para el control de incendios forestales. Este sistema emplea algunos modelos básicos de cobertura de demanda para desplegar los recursos disponibles de lucha contra incendios en las posiciones adecuadas para que cualquier incendio forestal sea atacado dentro de un límite de tiempo específico. La variable dentro de la formulación matemática es $x_{ik} = 1$ si se asigna un vehículo del tipo k al sitio i . En donde los vehículos están clasificados de acuerdo a características como capacidad, velocidad, tamaño, etc. El modelo utilizado se divide en dos, primero se determina la cantidad mínima de vehículos de cada tipo que son necesarios para cubrir el área del incendio, luego se busca dónde colocar estos vehículos para lograr la cobertura máxima posible del área.

Las variantes se presentan también en la forma de asignación de los clientes, en muchos trabajos de localización de instalaciones la asignación de los clientes a las plantas se

realiza aplicando la regla de la planta más cercana, pero en algunas aplicaciones conviene utilizar otros criterios. Por ejemplo, las reglas de proporcionalidad, a la segunda más cercana, dentro de un radio de distancia, etc. En [43] podemos ver un ejemplo de aplicación en el área de salud, donde se busca localizar ambulancias en sitios potenciales de forma que se maximice la cobertura, a los usuarios se les asigna la ambulancia disponible que pueda atenderlo, esto es, que la distancia de la base de la ambulancia a la localización del usuario sea menor que una distancia máxima establecida.

Otro ejemplo con diferentes formas de asignación de los usuarios es [39]. El problema de captura máxima que, supone, al igual que la mayoría de los problemas de localización, que los consumidores siempre son condescendientes con la instalación más cercana, independientemente de la propiedad o la proximidad a las instalaciones alternativas. En ese trabajo se proponen dos formas diferentes de asignación de los clientes. También en [?] se analiza un problema de asignación de aguas en el regadío en el cual se comparan tres formas diferentes de la asignación, sistema proporcional, sistema de mercado y la regla de reparto uniforme.

También es lógico considerar las preferencias de los clientes, en muchos casos los clientes tienen instalaciones preferidas, ya sea por la distancia, por la calidad, precios etc., por lo cual, para ciertas aplicaciones, es conveniente tomar en cuenta estas preferencias. Este problema fue planteado en [?], aquí los clientes tienen un orden de preferencia en una lista de sitios potenciales mediante el cual se determina su asignación. Este orden de preferencia depende de las características del cliente, de los sitios y del viaje entre la ubicación del cliente y los sitios. Estas preferencias se traducen en un conjunto de restricciones que se agregarán al problema de localización.

En [10, 15, 8, 9, 11, 7, 20, 41, 42, 29, 5] podemos encontrar literatura acerca del problema de localización de instalaciones considerando las preferencias de los usuarios. En [4] se formula el problema de la p -mediana tomando en cuenta las preferencias de los clientes como un modelo de optimización binivel donde el objetivo del líder es minimizar los costos de distribución y ubicación, y el seguidor optimiza las preferencias del cliente.

Un ejemplo de aplicación de este tipo de problemas se encuentra en [?], en ese trabajo se resuelve un problema de optimización de precios como un problema binivel, donde el problema de optimización de nivel superior consiste en maximizar el beneficio de la empresa, mientras que en el problema de nivel inferior busca satisfacer la regla de

elección de los clientes. En este caso, las empresas producen una variedad de productos y el cliente está interesado únicamente en adquirir como máximo un producto. El cliente tiene preferencias respecto a los productos ofrecidos por la empresa, es por esto que el nivel inferior busca maximizar la suma de preferencias de los usuarios.

Como podemos ver la localización de instalaciones es un área muy explorada y aplicada en diferentes contextos, todo esto genera la gran diversidad de variantes. Por eso, a continuación se agrupan mediante una tabla la revisión de literatura algunos artículos donde se muestran las variantes de problemas de localización de instalaciones mencionadas anteriormente con la finalidad de resumir y resaltar las aportaciones de cada uno.

En la última fila de la Tabla 1 se pueden ver las características del presente trabajo. Además de las variantes del problema de localización de instalaciones que se incluyeron. Es importante notar que ninguno de los artículos encontrados en la literatura considera al mismo tiempo las variantes capacidad, demanda unitaria, instalaciones no homogéneas mientras se consideran las preferencias de los usuarios. Vale la pena destacar que la aplicación de este tipo de problemas en un contexto de responsabilidad social es muy novedoso.

Como ya se mencionó, en este trabajo se formula un modelo de programación binivel, donde el nivel superior decide cuales talleres abrir y en cuales escuelas para minimizar los costos de apertura y de asignación de los alumnos. Mientras que en el nivel inferior, se decide la asignación de los alumnos a los diferentes talleres abiertos de forma que se optimice la suma de preferencias de los estudiantes. Una de las principales aportaciones de este trabajo es el hecho de considerar localización de instalaciones no homogéneas. Además, debido al contexto de la aplicación propuesta, se agregan restricciones para dos tipos de capacidades diferentes.

Para resolver el problema planteado, primero se utiliza una reformulación que reduce el problema binivel en un problema de un solo nivel, la cual resulta ser eficiente únicamente para instancias pequeñas. Para resolver instancias más grandes, se propone un algoritmo heurístico basado en el reencadenamiento de trayectorias.

Referencia	Capacidad	Demanda unitaria	Instalaciones no homogéneas	Asignación	Modelo	Método de solución	Aplicación
Pérez et al. [34]	✓	✓		Radio de cobertura	Programación lineal entera-mixta	Heurístico	Ubicar puntos de agregación de datos universal
Min and Melachrinoudis [32]	✓		✓		Programación biobjetivo con restricciones de azar	Suma ponderada	Ubicar tipos de servicios bancarios
Dimopoulou and Giannikos [16]	✓		✓	Radio de cobertura	Máxima Cobertura	Exacto	Control de incendios forestales
Villegas et al. [43]	✓			Radio de cobertura	Máxima Cobertura	Exacto	Localización de ambulancias de atención hospitalaria
Serra et al. [39]		✓		Diferentes reglas de decisión del consumidor	Programación entera	Búsqueda Tabú	
Limfati et al. [27]	✓		✓	Instalación más cercana	Programación lineal entera mixta	Búsqueda Tabú	
Medina Tapia and Cerda Troncoso [30]	✓		✓	Instalación más cercana	Programación biobjetivo	Branch and Bounds	Localización de estaciones de transferencia y rellenos sanitarios
Camacho-Vallejo et al. [4]		✓		Preferencias de los clientes	programación binivel	Reformulación a un solo nivel	
Díaz et al. [15]		✓		Preferencias de los clientes	programación binivel	GRASP - Tabú	
Camacho-Vallejo et al. [5]		✓		Preferencias de los clientes	programación binivel	Algoritmo Stackelberg-Evolutivo	
Maldonado-Pinto et al. [28]		✓		Preferencias de los clientes	programación binivel	Algoritmo evolutivo - PR	
Casas-Ramírez and Camacho-Vallejo [9]		✓		Preferencias de los clientes	programación binivel	Búsqueda dispersa - GRASP	
Casas-Ramírez et al. [11]	✓	✓		Preferencias de los clientes	programación binivel	Entropía cruzada	
Calvete et al. [3]	✓	✓		Preferencias de los clientes	programación binivel	Algoritmo evolutivo - PR	
	✓	✓	✓	Preferencias de los clientes	programación binivel	PR	Responsabilidad social gubernamental

Tabla 1: Modelos de localización y variantes

CAPÍTULO 3

MODELACIÓN MATEMÁTICA

Para analizar el problema bajo estudio, se recurre a un modelo de programación binivel, donde la decisión del nivel superior está dada por el gobierno, el cual busca minimizar los costos de localización y asignación para la implementación de este proyecto. Mientras que el nivel inferior está asociado con la sociedad de padres de familia, los cuales buscan la asignación de los alumnos de modo que se maximice la suma de las preferencias hacia los talleres y escuelas que les serán asignados a cada uno de ellos.

A continuación se presenta la formulación matemática del problema. Sea I el conjunto de talleres que se pueden ofrecer, J el conjunto de escuelas potenciales donde se puede abrir un taller y K el conjunto de estudiantes que asistirán a los talleres. Consideremos las variables de decisión de la siguiente forma: en el nivel superior se tiene a la variable binaria y_{ij} , la cual será 1 si se impartirá el taller i en la escuela j y 0 si no; en el nivel inferior tendremos la variable binaria x_{ijk} , que será 1 si se asigna al estudiante k al taller i en la escuela j y 0 en caso contrario.

Los parámetros de este problema son presentados a continuación en la siguiente tabla:

Parámetro	Descripción
m	Cantidad de espacios necesaria para asignar a todos los estudiantes
f_{ij}	Costo de abrir el taller i en la escuela j
c_{jk}	Costo de asignar al estudiante k en la escuela j
q_{ij}	Capacidad del taller i en la escuela j
g_{ijk}	Preferencia del estudiante k por el taller i en la escuela j
b_i	Cantidad máxima de talleres i que se pueden abrir
q_j	Cantidad máxima de talleres que se pueden abrir en la escuela j

Entonces el problema queda descrito en el siguiente modelo matemático:

$$\text{mín} \sum_{i \in I} \sum_{j \in J} f_{ij} y_{ij} + \sum_{i \in I} \sum_{j \in J} \sum_{k \in K} c_{jk} x_{ijk} \quad (3.1)$$

s.t.

$$\sum_{i \in I} \sum_{j \in J} q_{ij} y_{ij} \geq m \quad (3.2)$$

$$\sum_{j \in J} y_{ij} \leq b_i \quad \forall i \in I \quad (3.3)$$

$$\sum_{i \in I} y_{ij} \leq q_j \quad \forall j \in J \quad (3.4)$$

$$\sum_{j \in J} y_{ij} \geq 1 \quad \forall i \in I \quad (3.5)$$

$$y_{ij} \in \{0, 1\} \quad (3.6)$$

$$\text{mín} \sum_{i \in I} \sum_{j \in J} \sum_{k \in K} g_{ijk} x_{ijk} \quad (3.7)$$

s.t.

$$\sum_{i \in I} \sum_{j \in J} x_{ijk} = 1 \quad \forall k \in K \quad (3.8)$$

$$\sum_{k \in K} x_{ijk} \leq q_{ij} y_{ij} \quad \forall i \in I, \forall j \in J \quad (3.9)$$

$$x_{ijk} \in \{0, 1\} \quad \forall i \in I, \forall j \in J, \forall k \in K \quad (3.10)$$

El modelo de programación binivel está definido por (3.1) - (3.10), donde (3.1) es la función objetivo del nivel superior en la cual se busca minimizar los costos de abrir los talleres en cada escuela y además los costos de asignación de los estudiantes. La restricción (3.2) asegura que los talleres abiertos en todas las escuelas tengan capacidad para todos los estudiantes que participarán en dicho proyecto. En (3.3) aseguramos que en todas las escuelas j se abran como máximo b_i talleres i y (3.4) nos permite abrir a lo mucho q_i talleres i en la escuela j . Para que exista variedad de talleres, en (3.5) hacemos que el taller i se abra en al menos una escuela. En (3.6) se indica la binariedad de la variable y_{ij} .

Además, se tiene que x_{ijk} resuelve el problema de nivel inferior, cuya función objetivo está definida en (3.7) donde se busca minimizar la suma total de las preferencias de los estudiantes. Se garantiza en (3.8) que se asigne a cada estudiante a un único taller.

La restricción (3.9) nos dice que el estudiante k puede ser asignado al taller i en la escuela j , siempre y cuando este taller se ofrezca y aún tenga capacidad.

Las preferencias de los estudiantes g_{ijk} tomarán valores desde 1 hasta $|I| * |J|$, el estudiante ordenará todos los pares posibles de taller - escuela de forma que el par más preferido tomará el valor 1, el segundo par más preferido tendrá el valor de 2, y así sucesivamente, para todas las combinaciones posibles de la forma taller - escuela.

Cabe resaltar que las preferencias pueden no ser únicas, esto es, algunos estudiantes tendrán las mismas preferencias respecto a talleres y escuelas. Este caso se puede dar debido a diferentes aspectos, como por ejemplo, la localización geográfica de los estudiantes. Considere el caso de dos vecinos, es lógico que ambos tengan la misma preferencias respecto a las escuelas por la cercanía a ellas y al mismo tiempo pueden tener las mismas preferencias respecto a los talleres. Otro caso en que se pudiera presentar esta situación es con estudiantes que quieran ser asignados al mismo taller que otra persona (por ejemplo entre hermanos o amigos), pondrán las mismas preferencias para tratar de ser asignados juntos. Debido a esto, el problema de nivel inferior puede contar con múltiples soluciones óptimas, así que para tener bien definido al problema binivel, consideraremos el enfoque optimista durante su análisis, es decir, si existen múltiples soluciones óptimas para el nivel inferior, aquella que proporcione un mejor valor objetivo para el líder será seleccionada.

Una vez definido el problema y modelado mediante programación binivel, se procede a plantear una reformulación de este problema mediante condiciones de holgura complementaria y realizar pruebas sobre la efectividad de la misma.

CAPÍTULO 4

REFORMULACIÓN A UN SOLO NIVEL

En esta sección reformularemos el problema binivel recién descrito como un problema entero-mixto de un solo nivel, el cual puede ser resuelto mediante optimizadores comerciales.

Considere una solución dada del nivel superior \tilde{y}_{ij} , esto implica que ya se conocen los talleres que se impartirán en las escuelas, por lo que el problema del nivel inferior queda de la siguiente manera:

$$\text{mín} \sum_{i \in \tilde{I}} \sum_{j \in \tilde{J}} \sum_{k \in K} g_{ijk} x_{ijk} \quad (4.1)$$

s.t.

$$\sum_{i \in I} \sum_{j \in J} x_{ijk} = 1 \quad \forall k \in K \quad (4.2)$$

$$\sum_{k \in K} x_{ijk} \leq q_{ij} \quad \forall (i, j) \in A \quad (4.3)$$

$$x_{ijk} \in \{0, 1\} \quad \forall (i, j) \in A, \forall k \in K \quad (4.4)$$

donde $A = \{(i, j) / \tilde{y}_{ij} = 1\}$.

Entonces podemos describir el problema de nivel inferior como

$$\text{mín} \sum_{a \in A} \sum_{k \in K} g_{ak} x_{ak} \quad (4.5)$$

s.t.

$$\sum_{a \in A} x_{ak} = 1 \quad \forall k \in K \quad (4.6)$$

$$\sum_{k \in K} x_{ak} \leq q_a \quad \forall a \in A \quad (4.7)$$

$$x_{ak} \in \{0, 1\} \quad \forall a \in A, \forall k \in K \quad (4.8)$$

Podemos observar que este problema se reduce a un problema de transporte donde cada destino tiene demanda unitaria, lo cual nos permite relajar la variable binaria x_{ak} , obteniendo así el siguiente problema.

$$\text{mín} \sum_{a \in A} \sum_{k \in K} g_{ak} x_{ak} \quad (4.9)$$

s.t.

$$\sum_{a \in A} x_{ak} = 1 \quad \forall k \in K \quad (4.10)$$

$$\sum_{k \in K} x_{ak} \leq q_a \quad \forall a \in A \quad (4.11)$$

$$x_{ak} \geq 0 \quad \forall a \in A, \forall k \in K \quad (4.12)$$

Ahora si, se puede obtener el problema dual correspondiente al problema definido por 4.9 - 4.12. Sean u_k y v_a las variables duales, de modo que el problema dual queda dado como sigue:

$$\text{máx} \sum_{k \in K} u_k - \sum_{a \in A} q_a v_a \quad (4.13)$$

s.t.

$$u_k - v_a \leq g_{ak} \quad \forall a \in A, \forall k \in K \quad (4.14)$$

$$u_k \in \mathbb{R} \quad \forall k \in K \quad (4.15)$$

$$v_a \geq 0 \quad \forall a \in A \quad (4.16)$$

Entonces, el problema binivel se puede reformular como un problema mixto-entero de un solo nivel, el cual se presenta a continuación:

$$\text{mín} \sum_{i \in I} \sum_{j \in J} f_i y_{ij} + \sum_{i \in I} \sum_{j \in J} \sum_{k \in K} c_{jk} x_{ijk} \quad (4.17)$$

s.t.

$$\sum_{i \in I} \sum_{j \in J} q_{ij} y_{ij} \geq m \quad (4.18)$$

$$\sum_{j \in J} y_{ij} \leq b_i \quad \forall i \in I \quad (4.19)$$

$$\sum_{i \in I} y_{ij} \leq q_j \quad \forall j \in J \quad (4.20)$$

$$\sum_{j \in J} y_{ij} \geq 1 \quad \forall i \in I \quad (4.21)$$

$$\sum_{i \in I} \sum_{j \in J} x_{ijk} = 1 \quad \forall k \in K \quad (4.22)$$

$$\sum_{k \in K} x_{ijk} \leq q_{ij} y_{ij} \quad \forall i \in I, \forall j \in J \quad (4.23)$$

$$u_k - v_{ij} \leq g_{ijk} + M(1 - y_{ij}) \quad \forall i \in I, \forall j \in J, \forall k \in K \quad (4.24)$$

$$v_{ij} \leq M y_{ij} \quad \forall i \in I, \forall j \in J \quad (4.25)$$

$$\sum_{i \in I} \sum_{j \in J} \sum_{k \in K} g_{ijk} x_{ijk} = \sum_{k \in K} u_k - \sum_{a \in A} q_{ij} v_{ij} \quad \forall i \in I, \forall j \in J, \forall k \in K \quad (4.26)$$

$$u_k \in \mathbb{R} \quad \forall k \in K \quad (4.27)$$

$$v_{ij} \geq 0 \quad \forall i \in I, \forall j \in J \quad (4.28)$$

$$x_{ijk} \geq 0 \quad \forall i \in I, \forall j \in J, \forall k \in K \quad (4.29)$$

$$y_{ij} \in \{0, 1\} \quad (4.30)$$

Donde Eq. (4.17) es la función objetivo del nivel superior, en las Ec. (4.18) - (4.21) se indican las restricciones correspondientes al nivel superior. Las Eq. (4.22) y (4.23) corresponden a las restricciones del nivel inferior. Mientras que de las Ec. (4.24) a (4.28) se encuentran las restricciones dadas por la teoría de dualidad. Esas restricciones aseguran la solución óptima tanto para el problema primal como para el problema dual.

Es importante resaltar la diferencia entre y_{ij} y \tilde{y}_{ij} , se utiliza \tilde{y}_{ij} para una solución ya dada del nivel superior. Además la reformulación agrega $|I \times J| + |K|$ variables y $2|I \times J \times K| + |I \times J|$ restricciones.

4.1 PRUEBAS COMPUTACIONALES PARA LA REFORMULACIÓN

Para probar la eficiencia de la reformulación descrita en la sección anterior, se realizó experimentación computacional utilizando una computadora con un procesador Intel Core i7-4790 a 3.60 GHz con 32.00 GB de RAM funcionando bajo Windows 8.1. El modelo fue resuelto con el optimizador FICO Xpress 8.10.

En primer lugar se generó una instancia pequeña, en donde se consideraban tres talleres impartidos en 4 escuelas para 20 alumnos participantes. La reformulación obtuvo el óptimo en un tiempo de 1.848 segundos, lo cual nos indica que para instancias chicas la reformulación funciona de manera correcta obteniendo el óptimo en un tiempo corto.

Sin embargo, como el problema planteado busca ser un proyecto presentado a las autoridades de gobierno regional, se creó una instancia de tamaño real, la cual consta de 8 talleres que se busca implementar en 10 escuelas para un total de 600 alumnos. Para esta instancia la reformulación no fue capaz de obtener el óptimo en un tiempo límite de 12 horas. Primero se corrió la instancia durante 3 horas, en este tiempo el software encontró al menos una solución factible y se obtuvo una holgura de optimalidad del 63.410%. Luego, se corrió la misma instancia con un tiempo límite de 6, 9 y 12 horas. Cuando el software tardó las 12 horas, la holgura de optimalidad mejoró menos de un 0.05%, es decir, la holgura de optimalidad era mayor al 63%. Lo cual puede ser despreciable y no se justifica un incremento en el tiempo límite ni el esfuerzo computacional invertido.

Debido a esto, a continuación proponemos el uso de una metaheurística para resolver instancias de tamaños medianos y grandes.

CAPÍTULO 5

UN ALGORITMO BASADO EN REENCADENAMIENTO DE TRAYECTORIAS

En esta sección se describe el algoritmo utilizado para resolver el problema bajo estudio. Como se ha mencionado anteriormente, se propone un algoritmo basado en el reencadenamiento de trayectorias. Este método busca generar una trayectoria entre dos soluciones obtenidas previamente con la finalidad de agregar los elementos de intensificación y diversificación a la búsqueda de mejores soluciones. A continuación se presentan los preliminares del algoritmo y cada una de sus fases.

5.1 PRELIMINARES

El reencadenamiento de trayectorias (PR, por las siglas en inglés para path relinking) fue propuesto por primera vez en [18] como una forma de incorporar estrategias de intensificación y diversificación en la búsqueda tabú o búsqueda dispersa. El algoritmo consiste en explorar la trayectoria trazada mediante soluciones intermedias entre dos soluciones seleccionadas previamente las cuales son llamadas solución inicial (S_i) y solución guía (S_g).

La idea principal del PR es generar una trayectoria iniciando de la solución S_i y mediante movimientos definidos introducir elementos a esta solución de forma que incorpore atributos presentes en la solución guía. Se basa en la hipótesis de que algunas de las soluciones generadas en esta trayectoria pueden ser mejores que S_i y S_g (intensificación) pero suficientemente diferentes (diversificación).

Un aspecto importante del PR es la dirección en que se realiza la trayectoria. Sean S_1 y S_2 dos soluciones generadas previamente, entonces podemos seleccionar alguna de las siguientes direcciones para crear la trayectoria entre S_1 y S_2 :

- **Aleatoria** En este caso se seleccionan la solución inicial y solución guía de forma aleatoria.
- **Hacia arriba** Se considera a la mejor solución entre S_1 y S_2 como la solución guía. De este modo la trayectoria será generada iniciando con la peor solución.
- **Hacia abajo** La trayectoria comienza con la mejor solución y va agregando atributos hasta llegar a la que resultó ser la peor solución entre S_1 y S_2 .
- **Nueva** Esta dirección considera siempre a la nueva solución generada como la solución inicial y a la solución existente como la solución guía.

Estas direcciones se describen en [36], donde también se mencionan brevemente las ventajas de utilizar cada una de estas. En [38], se muestra un pseudocódigo del PR para soluciones con representación binaria, el cual puede ser útil para entender la forma en que se genera una trayectoria.

El reencadenamiento de trayectorias ha sido muy utilizado en el área de investigación de operaciones con la finalidad de mejorar soluciones obtenidas anteriormente con alguna otra metaheurística. Algunos ejemplos se pueden encontrar en [2], donde se resuelve un problema de optimización aplicando PR a soluciones creadas mediante un GRASP. Los resultados fueron muy competitivos comparados con otros algoritmos. En [31], se estudia un problema de diseño territorial con un método GRASP-PR mostrando que el PR ayuda a mejorar aún más la calidad de las soluciones obtenidas con el procedimiento constructivo y la búsqueda local. Otros trabajos en donde podemos encontrar el método de solución GRASP-PR son [35, 26, 37, 1, 6]. El PR se ha utilizado también con otras metaheurísticas como búsqueda tabú y búsqueda dispersa, como en [21, 38, 19].

En [36] podemos encontrar un GRASP-PR para resolver el problema de la p-mediana. En [33] se evalúa el procedimiento GRASP integrado con el PR para resolver un problema de la p-mediana sin capacidades donde se pudo observar que el PR condujo a una mejora de al menos un 5% en el costo total. En [28], se estudia un problema de localización de instalaciones sin capacidades considerando preferencias de los usuarios, el cual es formulado como un problema binivel. Los autores proponen la hibridación entre un algoritmo evolutivo y el reencadenamiento de trayectorias para sustituir el cruce aleatorio común, el cual muestra mejores resultados que el algoritmo evolutivo.

El algoritmo aquí propuesto genera soluciones factibles (de forma saturada, es decir con capacidad en exceso para atender la demanda de talleres) para el nivel superior, a las cuales se les aplica tres fases diferentes: la fase de reparación de la solución para obtener soluciones factibles, la fase de ajuste para cerrar talleres innecesarios, y la fase de búsqueda local para mejorar la calidad de la solución generada. A partir de las soluciones creadas se crea una trayectoria para realizar la exploración. Al mejor elemento en esta trayectoria se le aplica un procedimiento de búsqueda local y la fase de ajuste. De esta forma se brinda como resultado la mejor solución encontrada después de un cierto número de iteraciones. En la siguiente sección se darán los detalles de todas las fases del procedimiento propuesto.

5.2 ALGORITMO PROPUESTO

Para poder implementar el reencadenamiento de trayectorias en el problema aquí estudiado, primero se generan dos soluciones aleatorias saturadas. Después se les aplica una fase de reparación y otra fase de ajuste a dichas soluciones. La forma en que se construyen las soluciones saturadas y las dos fases mencionadas se mostrarán en secciones posteriores.

Una vez obtenidas las soluciones resultantes después de aplicar ambas fases se genera la trayectoria entre estas dos soluciones. Para generar esta trayectoria se consideran todos los elementos que diferencian a estas dos soluciones, es decir, en cada taller se analiza cuales escuelas de la solución inicial S_i son diferentes a las que se encuentran en la solución guía S_g . Las posiciones de estas escuelas son almacenadas en una lista correspondiente a cada solución y aleatoriamente se selecciona el elemento que se cambiará de la solución inicial por un elemento de la solución guía. De esta forma, después de cada iteración, la solución inicial tendrá más atributos de los que se encuentran en la solución guía .

Una vez generada la trayectoria entre las primeras dos soluciones, guardamos el mejor elemento encontrado al cual se le aplicará la fase de ajuste seguida de una búsqueda local con un vecindario de intercambios. La solución obtenida después de la búsqueda local es comparada con la mejor solución hasta el momento para actualizar la solución incumbente. Se genera una nueva solución a la cual se le aplica la fase de ajuste, luego la búsqueda local y ahora la nueva trayectoria se realizará entre la solución incumbente y la nueva solución generada. El mismo procedimiento es aplicado hasta completar un determinado número de iteraciones.

Para definir cual será la solución inicial y la solución guía, se realizaron experimentaciones computacionales preliminares con las cuatro formas descritas en 5.1. Se observó que generar la trayectoria desde la solución actual da resultados menos prometedores que las otras formas. Por otro lado, elegir como solución inicial a la mejor solución entre la solución incumbente y la nueva solución arroja mejores resultados, además de requerir un menor tiempo de cómputo. Por este motivo se decide utilizar este enfoque para realizar la elección de la solución inicial y solución guía lo cual podemos ver en el Pseudocódigo 1 en las líneas 13 a 19.

Algorithm 1 Pseudocódigo del reencadenamiento de trayectorias propuesto.

```

1:  $i \leftarrow 1$ 
2:  $y^{(0)} \leftarrow$  Solución inicial aleatorizada
3:  $y^{(0)} \leftarrow$  Reparación( $y^{(0)}$ )
4:  $y^{(0)} \leftarrow$  Ajuste( $y^{(0)}$ )
5:  $y^{(0)} \leftarrow$  BúsquedaLocal( $y^{(0)}$ )
6:  $y^* \leftarrow y^{(0)}$ 
7:  $F^* \leftarrow F(y^{(0)})$ 
8: while  $i \leq \text{Maxiter}$  do
9:    $y^{(i)} \leftarrow$  Solución aleatorizada
10:   $y^{(i)} \leftarrow$  Reparación( $y^{(i)}$ )
11:   $y^{(i)} \leftarrow$  Ajuste( $y^{(i)}$ )
12:   $y^{(i)} \leftarrow$  BúsquedaLocal( $y^{(i)}$ )
13:  if  $F(y_i) < F(y^*)$  then
14:     $S_i \leftarrow y^{(i)}$ 
15:     $S_g \leftarrow y^*$ 
16:  else
17:     $S_i \leftarrow y^*$ 
18:     $S_g \leftarrow y^{(i)}$ 
19:  end if
20:   $P \leftarrow$  Trayectoria( $S_i, S_j$ )
21:   $y' \leftarrow \arg \min_{y \in P} F(y)$ 
22:   $y' \leftarrow$  Ajuste( $y'$ )
23:   $y' \leftarrow$  BúsquedaLocal( $y'$ )
24:   $y^* \leftarrow \arg \min_{y \in \{y', y^{(i)}, y^*\}} F(y)$ 
25:   $F^* \leftarrow F(y^*)$ 
26:   $i \leftarrow i+1$ 
27: end while

```

En el Pseudocódigo 1 podemos observar el procedimiento del PR aquí propuesto. Más adelante se presentan los detalles de las fases de reparación, ajuste y la búsqueda local aplicadas durante el PR. Pero primero, se describe la codificación de las soluciones.

5.3 ANÁLISIS DE LA ESTRUCTURA DE LAS SOLUCIONES

Para calcular el valor objetivo de cada solución correspondiente al problema modelado en 3 se necesita contar con las variables y_{ij} correspondientes al nivel superior y con las variables x_{ijk} del nivel inferior. Como se mencionó en 5.1, el algoritmo propuesto trabaja explícitamente con soluciones parciales y_{ij} (asociadas al problema del nivel superior). Una vez obtenida una solución parcial y_{ij} , se resuelve el problema del nivel inferior parametrizado en esos valores, es decir, considera implícitamente los valores de las variables del nivel inferior. Después de hacer esto, se puede evaluar la función objetivo del problema (la del nivel superior) para encontrar el valor objetivo y después aplicar las fases de mejora correspondientes.

Para esto, se propone una forma eficiente para representar a las soluciones del problema del nivel superior con las cuales va a trabajar el algoritmo propuesto.

5.3.1 REPRESENTACIÓN DE LA SOLUCIÓN DEL NIVEL SUPERIOR

Es conveniente recordar que las variables del nivel superior se definieron como

$$y_{ij} = \begin{cases} 1 & \text{si se abre el taller } i \text{ en la escuela } j \\ 0 & \text{si no} \end{cases}$$

Podemos notar que de esta forma, la solución del nivel superior naturalmente se representa como una matriz cuyas entradas son unos o ceros. Esta representación conduce a trabajar con una matriz de tamaño $|I \times J|$ donde la mayoría de las posiciones serán cero, lo cual complica de forma significativa el trabajo computacional. Con la finalidad de reducir el tamaño de la matriz binaria se plantea una forma distinta para representar la solución y del nivel superior. Lo conveniente de la representación propuesta es que nos permite considerar automáticamente algunas de las restricciones del problema en la codificación debido a la manera en que se construyen las soluciones.

La solución y correspondiente al nivel superior del problema será representada como un vector de tamaño $\sum_{i \in I} b_i$, donde en las primeras b_1 posiciones se colocan los números

correspondientes a las escuelas en donde se ofrecerá el taller 1, de igual forma, de la posición $b_1 + 1$ hasta la $b_1 + b_2$ corresponden al taller 2, y así sucesivamente. En caso de colocar un "0" en alguna posición, este nos indica que el taller correspondiente tiene oportunidad para ser abierto en otra escuela.

Por ejemplo, consideremos una instancia para este problema con 4 escuelas disponibles para ofrecer 3 tipos diferentes de talleres. Además, consideremos el vector de capacidad de cada taller $b_i = [3 \ 2 \ 2]$. Para esta instancia $\sum_{i \in I} b_i = 7$ por lo cual, el vector solución constará de 7 elementos. Como $b_1 = 3$ entonces las primeras 3 posiciones corresponden al taller 1. Los talleres 2 y 3 pueden ser abiertos en dos escuelas, lo cual nos indica que las posiciones 4 y 5 le corresponden al taller 2 mientras que las posiciones 6 y 7 le corresponden al taller 3. Entonces, una posible solución para este problema estaría dada de la siguiente forma.

$$[0 \ 1 \ 3 \ 4 \ 0 \ 2 \ 0]$$

Tabla 2: Representación de la solución para el ejemplo ilustrativo.

La solución presentada en 2 nos indica que el taller 1 se abrirá en la escuela 1 y en la escuela 3, como en la posición uno se colocó un cero sabemos que este taller podría ser abierto en alguna otra escuela. Los talleres 2 y 3 se abren en las escuelas 4 y 2 respectivamente, ambos tienen oportunidad de ser ofrecidos en otra escuela que cuente con capacidad disponible.

En la solución descrita anteriormente podemos notar que la restricción 3.3 se cumple implícitamente, ya que la representación solo permite abrir el taller i en a lo mucho b_i escuelas.

5.3.2 GENERACIÓN DE LAS SOLUCIONES

Como se mencionó anteriormente, para el reencadenamiento de trayectorias necesitamos dos soluciones representadas de la forma descrita en la Sección 5.3.1. Es conveniente recordar que de las dos soluciones generadas una será la solución inicial mientras que la otra será la solución guía. La principal problemática es cómo generar dichas soluciones. Estas pueden ser generadas de dos formas distintas, la primera es de forma constructiva, sin embargo no se tiene mucha información para realizarlo de manera voraz. La segunda opción es realizar la construcción de forma destructiva, es decir, generar una solución

aleatoria *saturada*, a la cual se le aplicará una fase de cierre de instalaciones. Por solución *saturada* nos referimos a una solución con la mayor cantidad posible de talleres abiertos.

Para la implementación del algoritmo propuesto se sigue un criterio destructivo para generar las soluciones. Se generan lexicográficamente componente a componente, cada una de las posiciones tiene la probabilidad de $\frac{1}{|K|}$ para colocar el número de la escuela correspondiente en la posición actual del taller i . A continuación se muestra un ejemplo de este procedimiento.

Consideremos una instancia para este problema con 4 escuelas disponibles para ofrecer 3 tipos diferentes de talleres. Además consideremos el vector de capacidad de cada taller $b_i = [3 \ 2 \ 2]$.

$$[0.2 \ 0.5 \ 0.8 \ 0.6 \ 0.3 \ 0.2 \ 0.1]$$

Tabla 3: Vector de números aleatorios

Como la instancia cuenta con 4 escuelas, entonces tendremos 4 intervalos de probabilidad $(0, 0,25]$, $(0,25, 0,5]$, $(0,5, 0,75]$ y $(0,75, 1]$ los cuales pertenecen a cada una de las escuelas, respectivamente. La primera posición del vector aleatorio es 0,2, la cual cae en el intervalo de la escuela 1, entonces la solución tendrá en la primera posición a la escuela 1. Luego tenemos 0,5, por lo cual se colocará la escuela 2, y el mismo razonamiento se aplica para las demás posiciones. Obteniendo así la solución:

$$[1 \ 2 \ 4 \ 3 \ 2 \ 1 \ 1]$$

Tabla 4: Codificación de la solución generada.

Ahora, se puede notar que en la solución generada la sexta y séptima posición pertenecen al taller 3 y en ambas posiciones se indica que dicho taller se abrió en la escuela 1. Es evidente que esto genera una solución infactible ya que solo se puede abrir cada taller a lo mucho una vez en cada escuela. Por esta razón se implementa una fase de reparación.

5.3.3 FASE DE REPARACIÓN

Como se mencionó en el proceso de generación de soluciones, es posible que se obtengan soluciones infactibles para el problema aquí considerado. Del modelo 3 se puede

ver que para obtener una solución factible se deben cumplir las cuatro restricciones del nivel inferior. Tenemos que asegurar que los talleres abiertos tengan capacidad para todos los alumnos (3.2), además de las capacidades correspondientes a la cantidad máxima de talleres en cada escuela (3.4) y la cantidad máxima de talleres i que se pueden abrir (3.3). Más aun, debemos asegurar que se abrirá cada taller mínimo en una escuela (3.5). Por la forma en que se genera una solución, no se asegura que se generen soluciones factibles. Debido a esto, es importante un proceso de reparación para las soluciones generadas.

En esta fase se verifica la solución generada para identificar si se encuentran posiciones que no cumplan con las restricciones del problema. En caso de que se abra el mismo taller varias veces en la misma escuela, se cierran aleatoriamente hasta que la solución sea factible. Para ejemplificar, considere la solución generada en la Sección 5.3.2, y si se le aplica la fase de reparación se obtendría por ejemplo, la siguiente solución:

$$[1 \quad 2 \quad 4 \quad 3 \quad 2 \quad 1 \quad 0]$$

Tabla 5: Representación de la solución reparada

Podemos notar que en la sexta y séptima posición donde antes ambas tenían la escuela 1, después de aplicar la fase de reparación se cierra el taller 3 en la escuela 1 que corresponde a la séptima posición. Con esto se obtiene una solución factible con la cual ahora podemos encontrar la solución del nivel inferior.

5.4 NIVEL INFERIOR

Una vez obtenida una solución factible del nivel superior con el procedimiento descrito en la sección anterior, ya podemos calcular la solución del nivel inferior. Es conveniente recordar que las variables del nivel inferior están dadas por:

$$x_{ijk} = \begin{cases} 1 & \text{si se asigna al estudiante } k \text{ al club } i \text{ en la escuela } j \\ 0 & \text{si no} \end{cases}$$

Entonces ya se tiene el problema del nivel inferior parametrizado por las variables del nivel superior, y dicho problema se resuelve mediante un optimizador comercial. Una característica importante del problema del nivel inferior es que cuenta con múltiples soluciones óptimas. Como ya se mencionó anteriormente, esto puede darse por la naturaleza

de las preferencias de los estudiantes, por el hecho de que dos o más estudiantes puedan tener las mismas preferencias. Para que el problema binivel se encuentre bien definido, se decide asumir un enfoque optimista para seleccionar una única solución de todas las múltiples soluciones óptimas para el nivel inferior. El cual significa que en caso de tener múltiples soluciones óptimas para el problema del nivel inferior dado un valor conocido de las variables del líder, entonces se va a tomar la solución que más favorezca a la función objetivo del nivel superior.

Para poder garantizar que se asume el enfoque optimista debemos realizar un procedimiento que obtenga todas las soluciones óptimas del nivel inferior. Para lograr esto, agregaremos cortes al problema del nivel inferior de tal manera que la solución óptima obtenida deje de ser factible y así se forzará a cambiar de solución (en caso de haber soluciones factibles).

Tomando en cuenta que el problema tiene la restricción de asignar a todos los estudiantes, entonces para que la solución obtenida se vuelva infactible tenemos que forzar que la asignación seleccionada de estudiantes cambie, es decir, que al menos un estudiante no pueda volverse a asignar al taller que fue asignado. Para conseguir esto, hacemos que al menos una de las $x_{ijk} = 1$ en la solución óptima, ahora tenga que tomar el valor de cero. Esto es, se agrega al problema como una restricción adicional en el nivel inferior.

Definamos T como el conjunto de tercias (i, j, k) tales que $x_{ijk} = 1$ en las soluciones obtenidas anteriormente, entonces la restricción que se agregaría queda de la siguiente manera:

$$\sum_{(i,j,k) \in T} x_{ijk} \leq |K| - 1 \quad (5.1)$$

Si el problema resultante tiene solución y el valor objetivo se mantiene, entonces obtendremos un óptimo alternativo. En caso contrario, es decir, que la función objetivo del nivel inferior empeore indica que la solución ya no está en la región inducible. Para aclarar, estamos obteniendo una solución que es peor que las previamente generadas, lo cual nos indicaría que esta última solución cambia y no es un óptimo del nivel inferior. Es claro que los cortes se van a ir agregando hasta que el valor objetivo del problema aumentado cambie.

En el Pseudocódigo 2 se puede observar el proceso descrito anteriormente para agregar los cortes al nivel inferior y así generar todas las soluciones óptimas múltiples para

una solución conocida del nivel superior y . F representa el valor óptimo del nivel superior, mientras que f es el valor óptimo para el problema del nivel inferior. El algoritmo arroja x_{ijk}^* , lo cual indica la solución óptima del nivel inferior que es mejor para la función objetivo del nivel superior y será la solución seleccionada del nivel inferior.

Algorithm 2 Algoritmo para generar las múltiples soluciones óptimas.

```

1: Dada una solución  $y$ 
2:  $x^{(0)} \leftarrow \text{Nivel\_Inferior}(y)$ 
3: Evaluar  $f_0(x^{(0)})$ 
4: Evaluar  $F_0(x^{(0)}, y)$ 
5:  $F^* \leftarrow F_0$ 
6:  $i \leftarrow 1$ 
7: while  $f_i = f_0$  do
8:    $T \leftarrow (i, j, k) : x_{ijk}^{(i-1)} = 1$ 
9:   Nivel_Inferior  $\leftarrow \text{Nivel\_Inferior} \cup \sum_{(i,j,k) \in T} x_{ijk}^{(i-1)} \leq |K| - 1$ 
10:   $x^i \leftarrow \text{Nivel\_Inferior}(y)$ 
11:  Evaluar  $f_i(x^i)$ 
12:  Evaluar  $F_i(x^i, y)$ 
13:  if  $F^* > F_i$  then
14:     $F^* \leftarrow F_i$ 
15:     $x^* \leftarrow x^i$ 
16:  end if
17:   $i \leftarrow i + 1$ 
18: end while

```

De esta forma ya se cuenta con una manera eficiente de representar y generar las soluciones optimistas del nivel superior. Con esto, ya se puede evaluar la función objetivo del problema (nivel superior) para luego aplicar las fases siguientes del algoritmo.

5.5 FASE DE AJUSTE

Dada la naturaleza de las soluciones, esto es, la forma saturada con la cual son generadas es lógico que la capacidad para los alumnos de estas soluciones supere a la cantidad de estudiantes en el programa. Esto provoca la subutilización de los talleres abiertos. Por este motivo se agrega una fase de ajuste (o proceso destructivo) a las soluciones, la cual busca cerrar talleres uno a uno de forma voraz hasta llegar a una solución factible con la menor cantidad posible de talleres abiertos.

Primero, se crea un vecindario cerrando uno a uno todos los talleres posibles que se encuentren abiertos en la solución inicial. El taller se cierra colocando un cero en esa posición, cuidando las restricciones de capacidad de los estudiantes y al mismo tiempo, se cuida que ese taller se encuentre abierto en al menos una escuela. De todos los posibles elementos que se encuentran en este vecindario se selecciona el que brinde un mejor valor de la función objetivo.

Para esta fase se analizaron dos enfoques diferentes, la diferencia principal de estos enfoques es la forma en que se resuelve en nivel inferior para cada uno de los elementos en el vecindario. Ambos enfoques son descritos a continuación.

- Enfoque 1

En este enfoque una vez cerrado el taller se resuelve el problema del nivel inferior mediante un software comercial y obtenemos la función objetivo del nivel superior correspondiente para compararla con la función objetivo dada anteriormente. Si se presenta una mejora en el valor objetivo, entonces la solución será guardada como la incumbente. El procedimiento se realiza con todos los vecinos, guardando el valor y la solución incumbentes. Posteriormente, se actualiza la solución y como la solución incumbente y se repite el procedimiento anterior, así hasta que ya no se pueda cerrar ningún taller o en caso de cerrarlo no exista ninguna mejora en el valor objetivo. Este es el enfoque clásico anidado para algoritmos de búsqueda para resolver problemas de programación binivel [40].

Algorithm 3 Fase de Ajuste Enfoque 1

```

1: Solución inicial  $y$ 
2:  $y^* \leftarrow y$ 
3: for  $d$  en  $1:\sum_{i=1}^{|I|} b_i$  do
4:   contador=0
5:   for  $i$  en  $1:|I|$  do
6:     if  $y_d = i$  then
7:        $contador(i) = contador(i) + 1$            ▷ Contar talleres abiertos del tipo  $i$ 
8:     end if
9:   end for
10: end for
11: for  $d$  en  $1:\sum_{i=1}^{|I|} b_i$  do
12:   if  $y_d \neq 0$  then
13:      $D = D \cup d$                                ▷ Posiciones con talleres abiertos
14:   end if
15: end for
16: while  $D \neq \emptyset$  do
17:   for  $d$  en  $1:\sum_{i=1}^{|I|} b_i$  do
18:      $y_d = 0$ 
19:     if  $capacidad(y) > m$  then
20:       if  $contador(d) - 1 > 0$  then           ▷ Si no es el único taller  $i$  abierto
21:          $x \leftarrow \text{Nivel\_Inferior}(y)$ 
22:         if  $F(y) < F(y^*)$  then
23:            $y^* \leftarrow y$ 
24:         else
25:            $D = D \setminus d$ 
26:         end if
27:       else
28:          $D = D \setminus d$ 
29:       end if
30:     end if
31:   end for
32:    $D = D \setminus d$ 
33:    $y \leftarrow y^*$ 
34: end while

```

- Enfoque 2

Con el objetivo de reducir los tiempos de cómputo de la búsqueda local se presenta una alternativa en la cual no se resuelve el nivel inferior para cada vecino. Partiendo del punto en que es computacionalmente más costoso resolver un problema de programación lineal que realizar operaciones básicas para decidir la reasignación de los estudiantes, es la razón por la que se propone este esquema.

En este enfoque se obtienen los vecinos de la solución inicial de igual forma que en el enfoque 1, ahora la única diferencia es en la forma de calcular el valor objetivo. Para no resolver el nivel inferior al cerrar un taller se analiza cual es la mejor manera de reasignar a los estudiantes que estaban dentro del taller que ahora estará cerrado. De este conjunto de estudiantes, uno a uno se obtiene cual es ahora el taller abierto que tiene como más preferido. Si es posible asignarlo, esto es, si el taller tiene capacidad para otro estudiante, entonces el estudiante será asignado a ese taller. En dado caso que el taller abierto más preferido no cuente con cupo para este estudiante, se decide cual es el mejor estudiante para cambiar de taller, para esto se considera la diferencia que habría en caso de mover a cada estudiante dentro de este taller a su segundo taller abierto más preferido. Se guarda el mejor vecino obtenido. Dado que este proceso de reasignación no asegura el óptimo, se resuelve el nivel inferior únicamente para el mejor vecino obtenido con la reasignación de estudiantes.

Después, se actualiza la solución y como la solución incumbente y se realiza el procedimiento anterior hasta que ya no se pueda cerrar ningún taller o en caso de cerrarlo, que no exista mejoría en el valor objetivo. Después de este proceso se obtiene una solución mejorada y .

5.6 BÚSQUEDA LOCAL

Después de aplicar la fase de ajuste para las soluciones, en donde se busca cerrar escuelas de forma que se obtenga la mejor solución posible, se va a implementar un procedimiento de intensificación. Se diseña una búsqueda local de intercambio para obtener la solución que tenga la mejor localización de las escuelas que se encuentran abiertas. Para esta búsqueda local se genera el vecindario con todos los posibles intercambios entre las diferentes escuelas abiertas para los diferentes talleres (un cambio a la vez).

Se comienza con la posición 1 correspondiente al taller 1. Después, se verifica con cuales posiciones del taller 2 en adelante se puede intercambiar cuidando que esta nueva solución siga siendo factible.

[1 3 4 3 2 1 0]

Tabla 6: Solución inicial

En la Figura 6 se muestra una solución factible del nivel superior. Siguiendo el procedimiento descrito anteriormente, se muestra el vecindario posible.

$$\begin{array}{l}
 [3 \ 3 \ 4 \ 1 \ 2 \ 1 \ 0] \\
 [2 \ 3 \ 4 \ 3 \ 1 \ 1 \ 0] \\
 [1 \ 3 \ 4 \ 3 \ 2 \ 1 \ 0] \\
 [0 \ 3 \ 4 \ 3 \ 2 \ 1 \ 1] \\
 [1 \ 3 \ 4 \ 3 \ 2 \ 1 \ 0] \\
 [1 \ 2 \ 4 \ 3 \ 3 \ 1 \ 0] \\
 [1 \ 1 \ 4 \ 3 \ 2 \ 3 \ 0] \\
 [1 \ 0 \ 4 \ 3 \ 2 \ 1 \ 3] \\
 [1 \ 3 \ 3 \ 4 \ 2 \ 1 \ 0] \\
 [1 \ 3 \ 2 \ 3 \ 4 \ 1 \ 0] \\
 [1 \ 3 \ 1 \ 3 \ 2 \ 4 \ 0] \\
 [1 \ 3 \ 0 \ 3 \ 2 \ 1 \ 4] \\
 [1 \ 3 \ 4 \ 1 \ 2 \ 3 \ 0] \\
 [1 \ 3 \ 4 \ 0 \ 2 \ 1 \ 3] \\
 [1 \ 3 \ 4 \ 3 \ 1 \ 2 \ 0] \\
 [1 \ 3 \ 4 \ 3 \ 0 \ 1 \ 2]
 \end{array}$$

Tabla 7: Posibles intercambios (Vecindario)

En la Figura 7 se pueden encontrar todos los intercambios posibles para generar los posibles vecinos obtenidos con la búsqueda local de la solución dada en la Figura 6. Sin embargo, se puede notar que algunos de esos posibles intercambios no generan soluciones factibles para el problema en cuestión, por lo cual se eliminan estos intercambios para explorar únicamente los vecinos factibles. Entonces, se obtiene el siguiente vecindario de búsqueda:

$$\begin{array}{l}
 [2 \ 3 \ 4 \ 3 \ 1 \ 1 \ 0] \\
 [1 \ 0 \ 4 \ 3 \ 2 \ 1 \ 3] \\
 [1 \ 3 \ 2 \ 3 \ 4 \ 1 \ 0] \\
 [1 \ 3 \ 0 \ 3 \ 2 \ 1 \ 4] \\
 [1 \ 3 \ 4 \ 1 \ 2 \ 3 \ 0] \\
 [1 \ 3 \ 4 \ 0 \ 2 \ 1 \ 3] \\
 [1 \ 3 \ 4 \ 3 \ 1 \ 2 \ 0] \\
 [1 \ 3 \ 4 \ 3 \ 0 \ 1 \ 2]
 \end{array}$$

Tabla 8: Vecindario a explorar con la búsqueda local

En la Figura 8 se muestra el vecindario generado con la búsqueda local. Para cada uno de los vecinos se evalúa su función objetivo y se selecciona al mejor de todos. Una vez encontrada la mejor solución en el vecindario, esta se convierte en la solución inicial y con esto se inicia el procedimiento nuevamente. El proceso se realiza iterativamente hasta que no se encuentra una solución mejor a la incumbente.

Para la búsqueda local, al igual que en la fase de ajuste, se generan dos enfoques distintos. El primer enfoque resuelve el nivel inferior para cada vecino, y en base a esas funciones objetivo se selecciona el mejor elemento en el vecindario. El segundo enfoque realiza un proceso de reasignación similar al descrito en el enfoque dos de la Sección 5.5. Esto con la finalidad de disminuir los tiempos de cómputo del algoritmo.

5.7 GENERAR LA TRAYECTORIA ENTRE DOS SOLUCIONES

Una vez generadas dos soluciones, la parte fundamental del reencadenamiento de trayectorias consiste en la forma en que se genera una trayectoria entre estas dos soluciones conocidas. El procedimiento para generar dicha trayectoria en nuestro problema se muestra en el siguiente algoritmo.

Algorithm 4 Algoritmo para el generar el camino entre solución inicial y solución guía

```
1: Elementos_Inicial(i) ← Posiciones del taller i que son diferentes a las de la solución
   guía.
2: Elementos_Guía(i) ← Posiciones del taller i con elementos que no contiene la solución
   inicial
3: while |Elementos_Inicial| ≠ 0 do
4:    $d_{EI} \leftarrow$  Posición aleatoria de Elementos_Inicial
5:    $d_{EG} \leftarrow$  Posición aleatoria de Elementos_Guía
6:    $y(d_{EI}) = y(d_{EG})$ 
7:    $v_i \leftarrow y$ 
8:   Elementos_Inicial ← Elementos_Inicial  $\setminus \{d_{EI}\}$ 
9:   Elementos_Guía ← Elementos_Guía  $\setminus \{d_{EG}\}$ 
10: end while
```

En el Pseudocódigo 4 se muestra el algoritmo utilizado para generar la trayectoria entre dos soluciones. Los conjuntos Elementos_Inicial y Elementos_Guía sirven para guardar las posiciones de cada taller que serán necesarias cambiar de la solución inicial. El cambio a realizar se selecciona de forma aleatoria, en el caso en que el elemento generado no sea factible, se procede a realizar el cambio correspondiente al siguiente elemento de

la trayectoria sin resolver el nivel inferior.

Para comprender mejor el algoritmo mostrado en el Pseudocódigo 4, se presenta un ejemplo con una instancia conformada por 4 escuelas disponibles para ofrecer 3 diferentes tipos de talleres. El vector de capacidad de cada taller es $b_i = [3 \ 2 \ 2]$.

Considere las siguientes soluciones:

[2 0 1 3 2 0 1]

Tabla 9: Solución inicial

[2 0 3 0 3 2 4]

Tabla 10: Solución guía

Se puede ver que en las primeras 3 posiciones (correspondientes al taller 1) ambas soluciones abren la escuela 2 y la "escuela 0", entonces no se necesita hacer un cambio entre las posiciones correspondientes a estas escuelas. En la tercera posición la solución inicial tiene la escuela 1 y la solución guía tiene la escuela 3, por lo que tendremos que cambiar esta posición en la solución inicial. Para lograr esto se coloca en `Elementos_Inicial(1)` el número 3 y en `Elementos_Guía(1)` también el número 3, ambos correspondientes a la posición en cuestión.

Para las posiciones que corresponden al taller 2 se puede apreciar que en ambas soluciones se abre ese taller en la escuela 3; la solución inicial lo abre en la 4ta posición y la solución guía en la 5ta posición. Entonces, estas posiciones no tendrán que ser modificadas. La 5ta posición de la solución inicial y la 4ta posición de la solución guía entrarán a los conjuntos `Elementos_Inicial(2)` y `Elementos_Guía(2)`, respectivamente. Para el taller 3 es necesario cambiar ambas posiciones por lo cual en `Elementos_Inicial(3)` y `Elementos_Guía(3)` se agregan los elementos 6 y 7.

De esta forma se tendrían guardadas las posiciones que son necesarias cambiar en cada taller en `Elementos_Inicial(i)` y que serán reemplazadas con un elemento de `Elementos_Guía(i)`. El orden de estos cambios se selecciona de forma aleatoria. Se fija un taller y

CAPÍTULO 6

EXPERIMENTACIÓN

Como se mencionó en el inicio de este documento, esta investigación está orientada a la responsabilidad social gubernamental. En particular, nos enfocaremos en el gobierno del Estado de Nuevo León. Por esta razón, primero se analiza la estructura escolar del Estado. Después, en base a la estructura identificada se podrá definir el tamaño de las instancias que serán generadas de forma pseudo-aleatoria pero buscando que sean representativas para el caso de estudio.

Esta sección se divide en tres partes: en la primera se analiza la estructura escolar presente en el país, y en particular, en el Estado de Nuevo León, luego se muestra la experimentación realizada para probar el algoritmo propuesto, y por último, se presenta la aplicación del algoritmo a un caso de estudio.

6.1 ANÁLISIS DE LA ESTRUCTURA ESCOLAR EN NUEVO LEÓN

La educación en México es percibida como un derecho de todo individuo, lo cual está establecido en el artículo 3ro de la Constitución Política de los Estados Unidos Mexicanos. El criterio para orientar a esa educación debe ser basado en los resultados del progreso científico y debe tender a desarrollar todas las facultades del ser humano e inculcar los valores necesarios para una mejor convivencia social y con el medio ambiente. Aunado a esto, el artículo 31 de la Constitución Política de los Estados Unidos Mexicanos establece la obligación de los padres al hacer que sus hijos o pupilos menores de edad cursen la educación preescolar, primaria y secundaria.

Por otro lado, el artículo 3ro de la Constitución Política de los Estados Unidos Mexicanos también establece que el Estado Mexicano está obligado a garantizar educación básica (inicial, preescolar, primaria y secundaria) de calidad y de forma gratuita para

todos los niños del Estado. Además, debe buscar eliminar las carencias educativas que afectan a las diferentes regiones y entidades federativas. Actualmente, los lineamientos generales que rigen en materia educativa son: la Constitución Política de los Estados Unidos Mexicanos [17], en especial sus artículos 3 y 31; la Ley General de Educación [12]; la Ley Orgánica de la Administración Pública Federal [13], en particular su artículo 38; y el Reglamento Interior de la Secretaría de Educación Pública.

Desde el año 2003 el Instituto Nacional para la Evaluación de la Educación (INEE) es un organismo público el cual es el encargado de evaluar la calidad, el desempeño y los resultados del sistema educativo nacional en la educación de los niveles preescolar, primaria, secundaria y media superior. Cada año presenta un análisis de diversos indicadores los cuales ayudan a medir la calidad educativa en México desde diversos factores. Estos indicadores miden aspectos desde el tamaño de la población objetivo de la educación básica y como se alcanzan los objetivos básicos hasta el cumplimiento de las políticas implementadas en materia de educación, en [23] se puede encontrar el panorama educativo de México de cada año desde 2003.

Según el INEE, en el año 2018 el Estado contaba con un total de 87,756 escuelas primarias públicas en todo el país atendiendo a un total de 12,678,241 alumnos. De este total 2,396 están ubicadas en el Estado de Nuevo León las cuales atienden a 496,726 alumnos del Estado. Con estos datos, los cuales se encuentran en [22], es claro que se debe buscar una buena estructura administrativa para garantizar el buen funcionamiento de todas las escuelas.

Para mejorar el funcionamiento se considera un organigrama, el cual tiene como escalón más bajo al maestro. Se forma un grupo de máximo 45 alumnos, los cuales cursan el mismo grado escolar, y dicho grupo es asignado a un único maestro. Éste a su vez, es asignado a un único grupo. En un nivel más alto se encuentra el director, el cual es el encargado de coordinar a los maestros de toda la escuela asignando las actividades correspondientes a cada uno de ellos. También vigila que los maestros cumplan con los requisitos y obligaciones estipulados, y al mismo tiempo hace la gestión necesaria para cubrir las necesidades de la escuela bajo su cargo para asegurar un correcto funcionamiento.

El responsable de dar un seguimiento continuo a los directores y al cumplimiento de su trabajo es el supervisor escolar, el cual debe asegurar que las escuelas de la zona escolar bajo su cargo cumplan con los requisitos establecidos por la Secretaria de Educación

Pública (SEP) y por las autoridades competentes. Además, debe realizar las funciones de vigilancia de las actividades técnico-pedagógicas y administrativas, y debe fungir como mediador entre las autoridades educativas y las escolares. Los supervisores deben conocer las necesidades educativas de la comunidad con la finalidad de orientar la educación de sus escuelas mejorando así la formación de sus alumnos en diferentes aspectos.

Estas zonas escolares están conformadas por escuelas ubicadas en la misma región geográfica, cada supervisor de zona tiene a su cargo un máximo de 20 escuelas. Que un supervisor cuente con más de 20 escuelas, complica su labor, debido a que tiene que atender las necesidades específicas de cada una de estas escuelas, las cuales presentan diferentes condiciones y dificultades que puede facilitar o hacer más complejo su trabajo. Este corte de 20 escuelas se establece considerando que si los supervisores tuvieran que dedicar al menos un día al mes para atender a cada escuela bajo su responsabilidad, se verían con limitaciones de tiempo para realizar sus labores contemplando que las funciones de los supervisores no solo incluyen realizar visitas si no también recopilar y organizar las estadísticas educativas de su zona; identificar las necesidades de actualización de los docentes y organizar cursos de capacitación; trabajar con los directores sobre las necesidades de cada escuela; coadyuvar en la conformación de los consejos técnicos de escuela y de zona; impulsar la participación de los padres de familia y solucionar conflictos, entre otras actividades.

El INEE cuenta con un indicador para medir el porcentaje de zonas con más de 21 escuelas a su cargo, esto con la finalidad de identificar la cantidad de zonas donde los supervisores podrían estar teniendo sobrecarga de trabajo. Esta información sirve a las autoridades para diseñar estrategias que posibiliten una operación adecuada de las escuelas y poder distribuirlas de mejor manera. A continuación se presenta este indicador para el Estado de Nuevo León.

Porcentaje	Cantidad	Máximo de escuelas
0.7	2	23

Tabla 12: Supervisiones con 21 escuelas o más a su cargo

En la Tabla 12 se puede ver que el porcentaje de zonas escolares en Nuevo León con más de 20 escuelas es muy pequeño, 0.7%. Además de que la supervisión escolar con mayor carga tiene a su cargo 23 escuelas.

A fines de cumplir con los objetivos establecidos al poner en marcha este proyecto y al mismo tiempo permitir que se dé un buen seguimiento en torno a las competencias adquiridas por los estudiantes inscritos, se plantea que se aplique con las mismas bases de funcionamiento que las zonas escolares planteadas por la SEP, esto es, la localización de talleres en las diferentes escuelas y la asignación de estudiantes se realizará por grupos con un máximo de 20 escuelas.

Otro aspecto importante es la capacidad que tendrán los talleres, si se crean grupos muy grandes es claro que la persona encargada de impartir el taller no tendrá la oportunidad de “personalizar” el taller con cada uno de los estudiantes ni vigilar que se obtengan los aprendizajes esperados. Además, es importante considerar que de acuerdo al tipo de taller ofrecido se puede tener alguna cantidad máxima de estudiantes que puedan estar inscritos en él. Por esta razón se recomienda que los talleres sean considerados con capacidades entre 15 y 25 estudiantes para tener un mejor control sobre los grupos y asignar el tiempo necesario para cada estudiante inscrito.

Dicho esto, para probar el algoritmo propuesto se construyeron instancias aleatorias con un máximo de 20 escuelas y con capacidades para cada taller generadas de forma aleatoria entre 15 y 25 estudiantes. Con esto último se asegura que el algoritmo funcione también para los casos de estudio que tendrán las mismas dimensiones.

6.2 INSTANCIAS DE PRUEBA

Como ya se mencionó, para probar el algoritmo propuesto se creó un conjunto de instancias de forma pseudo-aleatoria con un máximo de 20 escuelas. Esto debido al supuesto de que el proyecto se aplicará de tal manera en que se dividan las escuelas correspondientes al área de interés en diferentes zonas para mejorar el seguimiento de los estudiantes.

Para esta experimentación se generaron un total 96 instancias con 24 tamaños diferentes, los cuales fueron creados considerando todas las combinaciones de los siguientes aspectos:

1. Talleres
 - 8
 - 10

2. Escuelas

- 10
- 15
- 20

3. Alumnos

- 600
- 800
- 1000
- 1500

Para cada uno de estos tamaños se generaron tres instancias donde los parámetros del problema se crearon de forma aleatoria pero asegurando que se cumplan las restricciones del problema, y así poder obtener soluciones factibles. El generador de instancias se realizó mediante un algoritmo programado en FICO Xpress 8.10, el cual genera todos los parámetros correspondientes dentro del rango señalado. Vale la pena resaltar que la capacidad de cada taller q_{ij} se genera aleatoriamente entre 15 y 25 estudiantes buscando mantener la atención debida a cada uno de los estudiantes.

Cada una de estas instancias se intentó resolver mediante la reformulación del problema dada en las Ec. (4.17)-(4.30), y se le fijó un tiempo máximo de 6 horas de corrida. Los resultados obtenidos con el algoritmo propuesto se comparan con los obtenidos con la reformulación del problema. Debido a la aleatoriedad existente en el algoritmo propuesto, cada instancia fue resuelta 10 veces y se calcula el promedio del tiempo de cómputo y el promedio del valor de la función objetivo.

Como se mencionó anteriormente, las instancias de prueba se generan con estos tamaños con la finalidad de que tengan características similares a las instancias generadas en base a datos reales. Con esto se asegura que el algoritmo sea capaz de resolver de manera eficiente los casos de estudio que mencionaremos a continuación.

6.3 CASOS DE ESTUDIO

Como se argumentó en secciones anteriores, el proyecto propuesto tiene como finalidad fortalecer la responsabilidad social por parte del gobierno de Nuevo León implemen-

tando talleres académicos como medida para mitigar el rezago educativo.

En esta sección, se analiza la estructura escolar de un caso de estudio que tiene lugar en el municipio de Apodaca, Nuevo León. Para ello, se utilizó el software de libre acceso QGIS como sistema de información geográfica. Los mapas del municipio se obtuvieron de la biblioteca digital de mapas del INEGI [24] como archivos tipo SHAPE [25] los cuales son fáciles de manejar en el QGIS.

Se consideran como posibles ubicaciones de los talleres las escuelas públicas pertenecientes al municipio de Apodaca. Para ello, se investigó en fuentes oficiales de la SEP la cantidad de escuelas ubicadas en este municipio. En [14] se encuentra una base de datos, la cual contiene los nombres de todas de las escuelas pertenecientes al municipio de interés, la ubicación geográfica de cada una de ellas y el tipo de sostenimiento (público o privado). Además, se obtuvieron datos relevantes como la cantidad de alumnos inscritos en las respectivas escuelas. Mediante esta base de datos se encuentra que el municipio de Apodaca cuenta con un total de 70 escuelas públicas y se extraen las coordenadas geográficas de cada una de ellas.

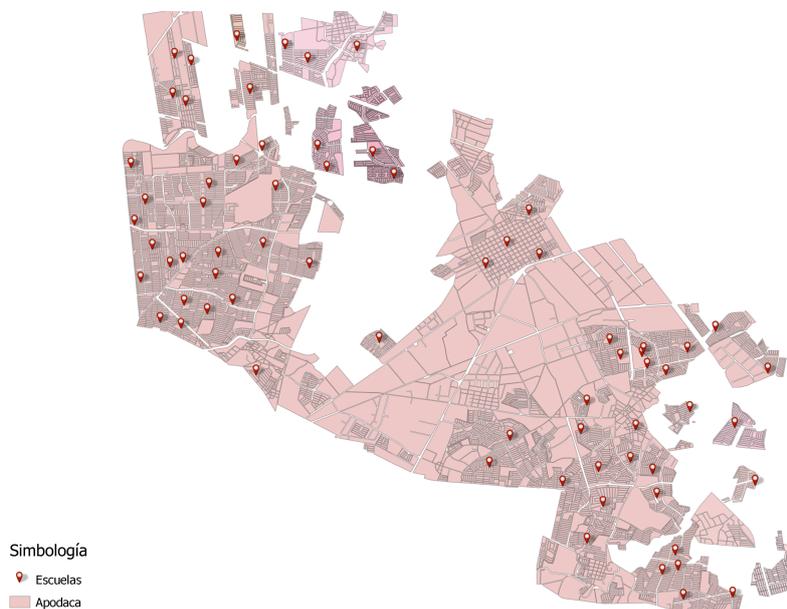


Figura 1: Mapa del municipio de Apodaca y escuelas públicas en la zona

Las 70 escuelas públicas en el municipio de Apodaca se dividieron en 7 zonas diferentes, tal que estas no se traslapen entre si. Además, se consideró que cada zona tenga como máximo 20 escuelas (descrito en la sección 6.1) para asegurar un mejor funciona-

miento de los talleres ofrecidos y al mismo tiempo limitar la distancia de traslado de los estudiantes. A continuación podemos ver la división territorial que se realizó.

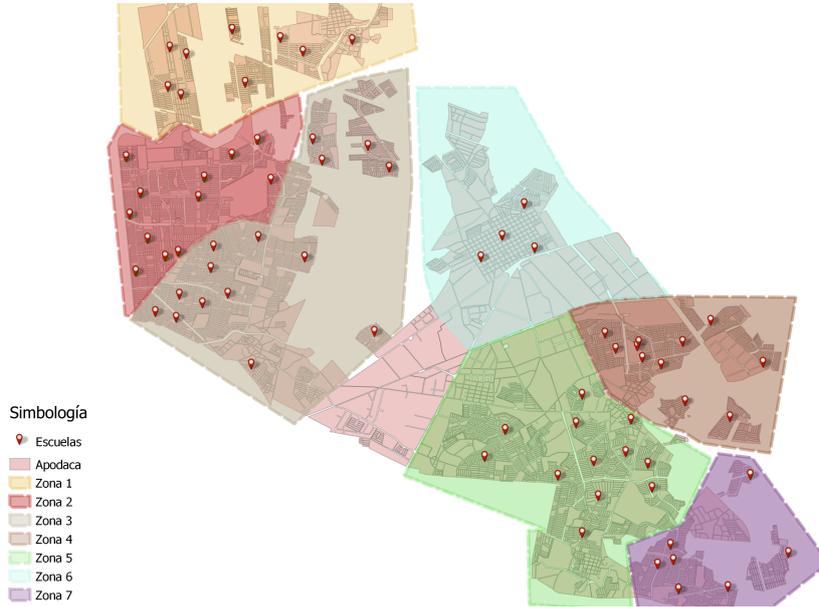


Figura 2: Mapa del municipio de Apodaca y escuelas públicas divididas por zonas

Para identificar a los estudiantes que participarán en los talleres ofrecidos se seleccionaron puntos aleatorios dentro del área considerada por la zona perteneciente a esta instancia. Se toma en cuenta que existen algunos lugares donde no hay población (centros comerciales, parques, industrias), en los cuales no se seleccionan estudiantes con la finalidad de hacer un caso de estudio lo más real posible. La cantidad de estudiantes se consideró tomando el 15 % del total de alumnos inscritos en las escuelas correspondientes a la instancia, para obtener estos datos se utilizó la misma base de datos de la SEP mencionada anteriormente.

Por otro lado, los costos de asignación (c_{jk}) para cada una de las instancias se aproximaron calculando la distancia entre los estudiantes inscritos en el programa y las escuelas correspondientes a la instancia mediante la fórmula de Haversine [?]. Esta fórmula nos ayuda a calcular la distancia entre dos puntos en coordenadas geográficas considerando la aproximación esférica de la superficie terrestre. Sea ϕ_i la latitud de la coordenada geográficas i y λ_i la longitud de la coordenada geográficas i podemos calcular entonces:

$$\begin{aligned}\Delta\phi &= \phi_1 - \phi_2 \\ \Delta\lambda &= \lambda_1 - \lambda_2 \\ a &= \sin^2\left(\frac{\Delta\phi}{2}\right) + \cos(\phi_1) * \cos(\phi_2) * \sin^2\left(\frac{\Delta\lambda}{2}\right)\end{aligned}$$

De esta forma, la distancia haversiana esta dada por:

$$d = 2 * r * \arcsin(\sqrt{a})$$

donde $r = 6371$ km es el radio medio de la Tierra.

Los demás parámetros del problema se generan de forma pseudo-aleatoria. Para las preferencias del estudiante k se generan $|I| * |J|$ números aleatorios los cuales son ordenados de forma descendente y se asigna como preferencia del taller i en la escuela j a la posición correspondiente de este número aleatorio. Una vez generados estos parámetros se han completado las instancias para el problema en cuestión.

Cabe recordar que el proyecto presentado tiene como propósito ayudar a fortalecer el rendimiento académico, social y cultural de los estudiantes. Además, puede utilizarse como estrategia para alejar a los estudiantes de actividades delictivas y mejorar con esto la convivencia de las comunidades. Por este motivo y para fines de este estudio, se consideró la implementación del proyecto únicamente en dos de las siete zonas mencionadas anteriormente, las cuales son seleccionadas estratégicamente por considerarse zonas conflictivas.

6.3.1 INSTANCIA A

Para esta instancia se escoge en particular la zona de la colonia **Pueblo Nuevo** y alrededores (Zona 4) dado que es una de las zonas más conflictivas del municipio de Apodaca. La instancia cuenta con un total de 11 escuelas, en las cuales se ofertan 10 talleres para un total de 600 estudiantes. Los talleres ofrecidos se mencionan a continuación:

No.	Taller	No.	Taller
1	Matemáticas	6	Fútbol
2	Literatura	7	Teatro
3	Inglés	8	Pintura
4	Ajedrez	9	Dibujo
5	Voleibol	10	Danza

La instancia queda dada sobre el mapa territorial de la siguiente manera:

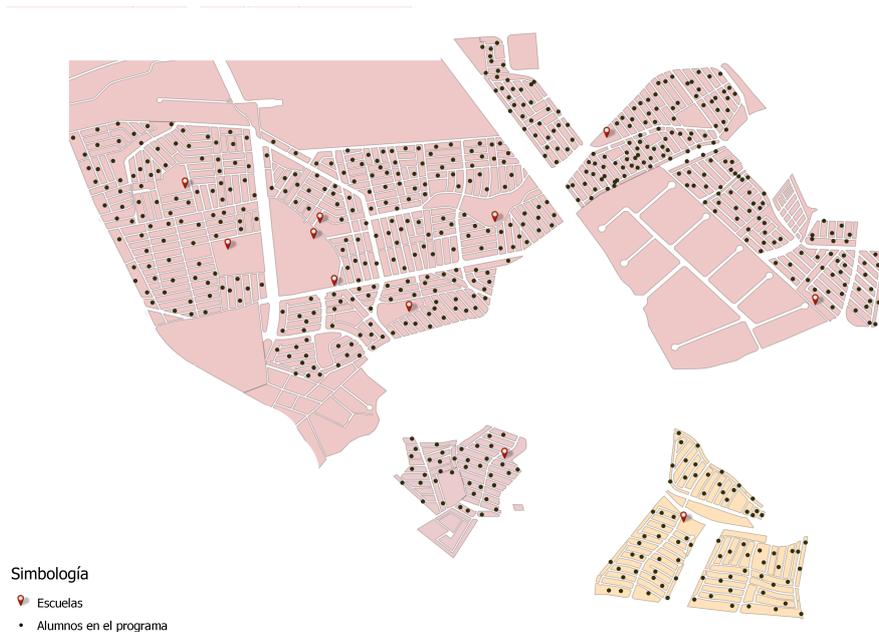


Figura 3: Mapa de la Zona 4 del municipio de Apodaca, escuelas públicas correspondientes y estudiantes inscritos en el programa.

6.3.2 INSTANCIA B

Para esta instancia se seleccionó la colonia **Hacienda las Margaritas** y alrededores, la cual podemos identificar en 2 como la Zona 7, dado que colinda con colonias conflictivas de otros municipios. La instancia cuenta con un total de 7 escuelas, en las cuales se ofertan 8 talleres para un total de 500 estudiantes. Los talleres ofrecidos se mencionan a continuación:

No.	Taller
1	Matemáticas
2	Literatura
3	Inglés
4	Voleibol
5	Fútbol
6	Pintura
7	Dibujo
8	Danza

A continuación podemos ver la instancia sobre el mapa territorial:

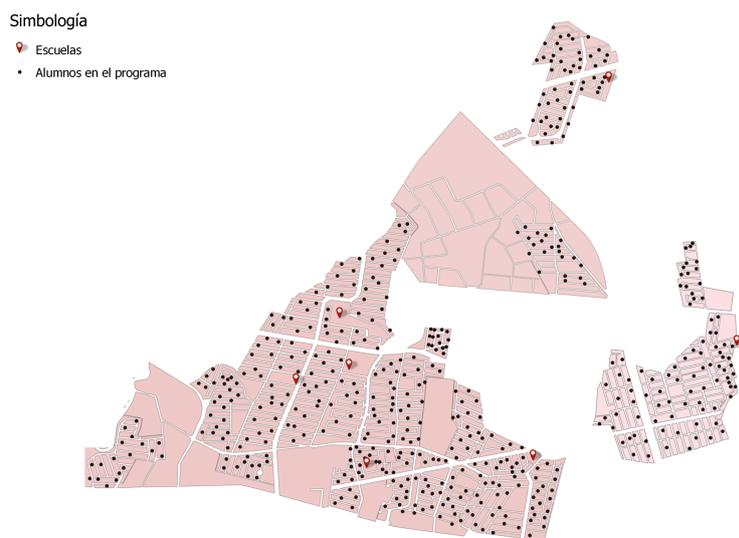


Figura 4: Mapa de la Zona 7 del municipio de Apodaca, escuelas públicas correspondientes y estudiantes inscritos en el programa

CAPÍTULO 7

ANÁLISIS DE RESULTADOS

En este capítulo se muestran los resultados obtenidos de la experimentación computacional realizada con las instancias pseudo-aleatorias y con el caso de estudio. Toda la experimentación computacional se realizó en una computadora con un procesador Intel Core i7-4790 a 3.60 GHz con 32.00 GB de RAM funcionando bajo Windows 8.1. El modelo fue resuelto con el optimizador FICO Xpress 8.10.

Para mejorar el desempeño del algoritmo propuesto en esta tesis se realizaron algunas experimentaciones previas. Esto con la finalidad de elegir uno de los dos enfoques propuestos para la fase de ajuste (ver Sección (5.5)) y uno de los dos enfoques presentados para la búsqueda local (ver Sección (5.6)). En esta experimentación preliminar se observó que en ambas fases el enfoque 2 (cuando se hace una reasignación de estudiantes en lugar de resolver el nivel inferior en cada paso) no muestra una diferencia significativa con respecto al valor de la función objetivo obtenida por el enfoque 1 (cuando se resolvía el nivel inferior en cada movimiento). Sin embargo, en términos del tiempo computacional requerido, el enfoque 2 reduce de manera significativa los tiempos del enfoque 1. De esta forma se decidió aplicar el esquema descrito en el Algoritmo 5.2 donde para la fase de ajuste y para la búsqueda local se utiliza el enfoque 2.

El número de iteraciones también se seleccionó en base a las experimentaciones preliminares. Se aplicó el algoritmo a una muestra aleatoria de instancias con un número predeterminado de iteraciones que se fijó en 5, 8 y 10, respectivamente. Estas experimentaciones arrojaron mejores resultados cuando se realizaban 10 iteraciones del reencadenamiento de trayectorias. Para los otros dos valores considerados, el algoritmo presentaba inestabilidad en cuanto a los resultados obtenidos para el mejor valor encontrado de la función objetivo del líder.

Por estas razones se muestran únicamente los resultados del algoritmo donde la Fase de Ajuste se realiza mediante el enfoque 2, la búsqueda local aplica el enfoque 2 y el número de iteraciones es 10.

7.1 RESULTADOS PARA LAS INSTANCIAS DE PRUEBA

Para cada una de las instancias de prueba descritas en el capítulo anterior se realizaron 10 corridas del algoritmo, luego se obtiene el valor objetivo del líder promedio y el tiempo promedio por corrida. Estos resultados son comparados con los resultados obtenidos al resolver la reformulación mediante un optimizador comercial. Es importante mencionar que a la reformulación se le puso un tiempo máximo de corrida de 6 horas. Recordemos que los tamaños de las instancias se realizaron con combinaciones entre 8 y 10 talleres, 10, 15 y 20 escuelas, y para los estudiantes se consideraron 600, 800, 1000 y 1500. En los casos cuando se consideraban 1500 estudiantes, las instancias con 10 escuelas no eran factibles. Es por esto que esa configuración de parámetros no se utilizó en la experimentación. La infactibilidad se da debido a la capacidad de los talleres. Lo mismo sucede cuando tenemos 8 talleres en 10 escuelas, por eso se consideraron un total de 63 instancias analizadas.

A continuación se presentan las tablas correspondientes a los resultados obtenidos para las 63 instancias aplicando la reformulación del problema, y por otro lado, el algoritmo propuesto. En las siguientes tablas se muestran los resultados obtenidos para las instancias de prueba y las holguras obtenidas con la reformulación y con el algoritmo propuesto. La columna “ $|I|$ ” indica el número de talleres considerando en la instancia, en “ $|J|$ ” y “ $|K|$ ” se establece el número de escuelas y la cantidad de estudiantes, respectivamente. En las columnas “Reformulación” y “Algoritmo” se muestra el valor objetivo del líder encontrado por la reformulación y el valor objetivo promedio obtenido por el algoritmo, respectivamente. La columna “Gap R” indica el porcentaje de holgura obtenida por la reformulación del problema y “Gap A” es porcentaje de holgura obtenido al aplicar el algoritmo, ambos se calcularon de la siguiente manera:

$$Gap = \left(\frac{\text{Valor obtenido} - \text{Mejor solución encontrada}}{\text{Mejor solución encontrada}} \right) * 100$$

En “Tiempo ALG” encontramos el tiempo promedio en segundos que tarda el algoritmo en resolver la instancia correspondiente. Por último, en “Ahorro tiempo” podemos ver el porcentaje de tiempo que se ahorra utilizando el algoritmo en lugar de utilizar la reformulación de un solo nivel. Es importante recordar que todas las reformulaciones se

corrieron por un tiempo máximo de 6 horas (21600 segundos).

$ I $	$ J $	$ K $	Rep.	Reformulación	Algoritmo	Tiempo(seg)	Mejor	GAP A(%)	GAP R(%)	Ahorro tiempo(%)
8	10	600	1	7915	7670.3	855.802	7603	0.885	4.104	96.038
			2	2740	2718.8	772.580	2681	1.410	2.201	96.424
			3	8035	7802.9	782.152	7707	1.244	4.256	96.379
		800	1	10888	10616.3	1140.580	10537	0.753	3.331	94.720
			2	8223	7884.8	756.968	11714	0.670	1.246	94.076
			3	8167	7841	879.882	11631	0.469	0.722	94.001
		1000	1	13654	13576.8	1369.227	13402	1.304	1.880	93.661
			2	13735	13545.7	1581.488	13484	0.458	1.861	92.679
			3	13809	13778.9	1451.675	13566	1.569	1.791	93.279

Tabla 13: Resultados de las instancias con 8 talleres y 10 escuelas

$ I $	$ J $	$ K $	Rep.	Reformulación	Algoritmo	Tiempo(seg)	Mejor	GAP A(%)	GAP R(%)	Ahorro tiempo(%)
8	15	600	1	7940	7860.3	1749.751	7835	0.323	1.340	91.9
			2	7906	7832	1935.009	7802	0.385	1.333	91.042
			3	7947	7927.4	1395.452	7903	0.309	0.557	93.54
		800	1	10634	10575.7	2735.705	10557	0.177	0.729	87.335
			2	11925	11655.9	2810.953	11552	0.899	3.229	86.987
			3	12383	12087.2	2645.932	11910	1.488	3.971	87.751
		1000	1	15413	15374.9	2732.606	15262	0.740	0.989	87.35
			2	13881	13101	3435.266	13005	0.738	6.736	84.096
			3	13727	13146	3000.809	13005	1.084	5.552	86.108

Tabla 14: Resultados de las instancias con 8 talleres y 15 escuelas

$ I $	$ J $	$ K $	Rep.	Reformulación	Algoritmo	Tiempo(seg)	Mejor	GAP A(%)	GAP R(%)	Ahorro tiempo(%)
8	20	600	1	8189	7494.2	3085.658	7410	1.136	10.513	85.715
			2	6410	6216.3	3133.671	6161	0.898	4.042	85.493
			3	8929	8351.6	3487.115	8237	1.391	8.401	83.856
		800	1	10887	10474.9	4631.901	10338	1.324	5.311	78.557
			2	11797	11290.3	4301.597	11152	1.240	5.784	80.086
			3	12028	11135.7	4085.138	11082	0.485	8.536	81.088
		1000	1	4553	4491.8	5497.338	4460	0.713	2.085	74.55
			2	14879	14624.9	6134.314	14512	0.778	2.529	71.601
			3	14967	14631.9	6242.751	14437	1.350	3.671	71.099
		1500	1	22132	21848	10048.835	21729	0.548	1.855	53.478
			2	22127	21786	9333.190	21658	0.591	2.165	56.791
			3	22010	21487.8	9903.840	21413	0.349	2.788	54.149

Tabla 15: Resultados de las instancias con 8 talleres y 20 escuelas

$ I $	$ J $	$ K $	Rep.	Reformulación	Algoritmo	Tiempo(seg)	Mejor	GAP A(%)	GAP R(%)	Ahorro tiempo(%)
10	10	600	1	8259	7818.3	1097.335	7710	1.405	7.121	94.92
			2	8168	7936.6	1038.816	7656	3.666	6.688	95.191
			3	8848	8754.6	1195.157	8587	1.952	3.04	94.467
		800	1	3682	3663.8	1801.829	3634	0.821	1.321	91.659
			2	11497	11362	2814.057	11242	1.068	2.269	86.972
			3	10610	10542.2	1846.404	10411	1.261	1.912	91.452
		1000	1	13742	13295.7	2455.947	13216	0.604	3.981	88.63
			2	13760	13744.1	1912.85	13668	0.557	0.674	91.145
			3	14513	14300.2	2349.628	14236	0.451	1.946	89.123

Tabla 16: Resultados de las instancias con 10 talleres y 10 escuelas

$ I $	$ J $	$ K $	Rep.	Reformulación	Algoritmo	Tiempo(seg)	Mejor	GAP A(%)	GAP R(%)	Ahorro tiempo(%)
10	15	600	1	9164	8701.2	2963.653	8625	0.884	6.25	86.28
			2	8298	7475.4	2793.613	7394	1.101	12.227	87.067
			3	8228	7518.7	3344.917	7424	1.276	10.83	84.515
		800	1	12256	11929.4	3960.739	11794	1.149	3.918	81.664
			2	12269	11795	4242.625	11673	1.046	5.106	80.359
			3	8404	8324.3	3998.369	8259	0.791	1.756	81.49
		1000	1	14758	14574.8	5171.312	14340	1.638	2.915	76.059
			2	14837	14656.1	4828.481	14552	0.716	1.959	77.646
			3	14726	14435.1	5042.612	14322	0.79	2.821	76.655
		1500	1	22755	22379.3	9573.607	22271	0.487	2.174	55.678
			2	22245	21998.2	8960.109	21835	0.748	1.878	58.519
			3	22548	22409.2	8094.294	22323	0.387	1.008	62.527

Tabla 17: Resultados de las instancias con 10 talleres y 15 escuelas

$ I $	$ J $	$ K $	Rep.	Reformulación	Algoritmo	Tiempo(seg)	Mejor	GAP A(%)	GAP R(%)	Ahorro tiempo(%)
10	15	600	1	8834	8159.2	5442.493	8067	1.143	9.508	74.804
			2	8959	8339.3	4509.697	8204	1.65	9.203	79.122
			3	8968	8352.2	4287.208	8297	0.666	8.088	80.152
		800	1	11894	11208	6492.49	11121	0.783	6.951	69.943
			2	11904	11371.2	6262.821	11269	0.907	5.635	71.005
			3	11794	11265.3	6417.735	11184	0.727	5.455	70.289
		1000	1	14877	14305.5	7865.674	14090	1.53	5.586	63.585
			2	14952	14147.9	7498.203	14055	0.661	6.383	65.287
			3	14249	13158.6	9197.731	12985	1.337	9.735	57.418
		1500	1	22289	21664.8	12851.972	21491	0.809	3.714	40.501
			2	22163	21668.5	12768.938	21541	0.592	2.888	40.885
			3	22334	21760.4	13991.48	21693	0.311	2.955	35.225

Tabla 18: Resultados de las instancias con 10 talleres y 20 escuelas

En las Tablas 13- 18 se puede ver que en ninguna de las instancias la reformulación llegó al óptimo en el tiempo limite de 6 horas. Recuerde que en 4.1 se presenta evidencia para justificar el no invertir más tiempo computacional dado que el cambio en la holgura de optimalidad es insignificante.

Al comparar las columnas “ Reformulación ” y “ Algoritmo ” es fácil ver que el valor objetivo promedio obtenido al aplicar el algoritmo propuesto es mejor que el valor obtenido por la reformulación en el tiempo límite. Esto indica que el algoritmo cumple con el objetivo de encontrar mejores resultados respecto a la reformulación del problema.

En la Tabla 13 y la primera parte (el caso que considera 600 alumnos) de la Tabla 14 en la columna “ Ahorro tiempo ” se puede notar que para las instancias con 8 talleres y 10 escuelas, el algoritmo resuelve el problema al menos con un tiempo 90 % menor al tiempo de corrida de la reformulación. Además, obtiene una solución hasta 3 % mejor. Para instancias con 10 talleres, 10 escuelas y con 600 u 800 alumnos se aprecia de la Tabla 16 que el ahorro en tiempo sigue siendo superior al 90 %. Cuando las instancias son más grandes, el ahorro en tiempo empieza a disminuir, sin embargo, se puede ver que el algoritmo ahorra al menos un 50 % de tiempo y obtiene una mejor solución que la reformulación.

Como resultado de esta experimentación se puede prever que el algoritmo funciona de manera correcta para los casos de estudio. Esto es debido a que se definieron las zonas escolares con tamaños inferiores a 15 escuelas, tal y como se mencionó en la Sección 6.3. En las instancias de prueba de tamaño similar el ahorro en tiempo del algoritmo fue superior al 80 %. Además, el algoritmo obtiene un mejor valor objetivo respecto a la reformulación, en todas las instancias.

7.2 RESULTADOS PARA LOS CASOS DE ESTUDIO

Para resolver los casos de estudio también se aplicó el algoritmo propuesto. En la sección anterior se mostró que obtuvo buenos resultados para resolver las instancias de prueba. En este caso el algoritmo se aplicó 5 veces a cada una de las dos instancias. En los resultados se presenta la mejor solución encontrada.

Para mostrar la solución se presentan las imágenes de cada instancia con la asignación correspondiente de los estudiantes. Por motivos de visualización se muestra una imagen por cada escuela donde se abrió un taller en la solución, podemos ubicar cada estudiante asignado a esta escuela mediante una línea que va desde el estudiante hacia la escuela correspondiente. Para diferenciar a cual taller se asigna cada estudiante se presentan las líneas de asignación con diferentes colores, donde cada uno representa un taller abierto en la escuela en cuestión.

7.2.1 RESULTADOS PARA LA INSTANCIA A

Este caso de estudio cuenta con un total de 11 escuelas en las cuales se impartirán 10 talleres a 600 estudiantes. El tiempo promedio que tardó el algoritmo en resolver esta instancia fue inferior a 1600 segundos (26.5 minutos), lo cual es un tiempo razonable para resolver un problema de este tipo. Recuerde que son decisiones estratégicas y no se toman tan a menudo.

A continuación se muestra la mejor solución encontrada de manera gráfica:

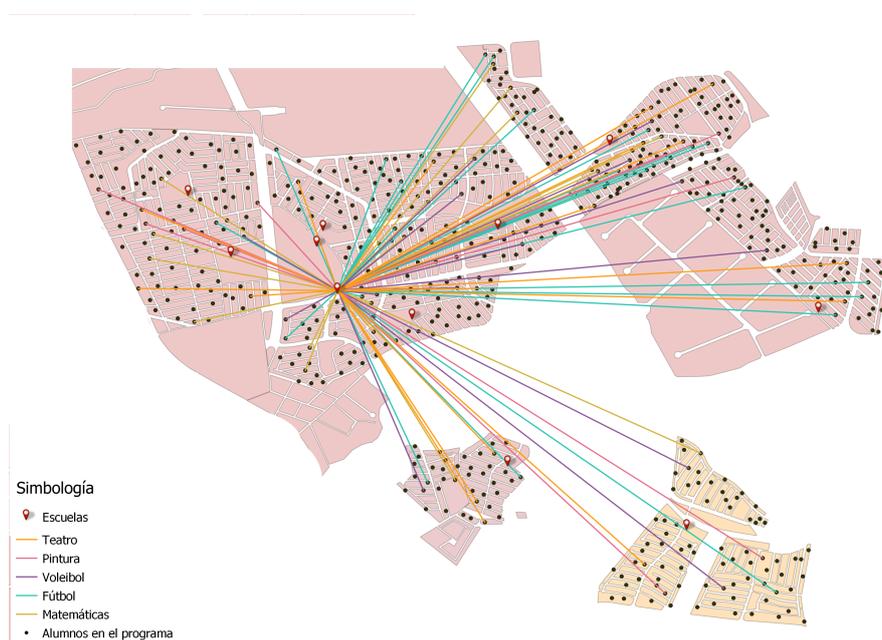


Figura 5: Mapa de la Instancia A con la respectiva asignación de los estudiantes a la escuela 1

En la Figura 5 se puede ver la asignación de los estudiantes a la escuela “NINFA ESTHER FUENTES DE GARCIA” en la cual se ofrecen los talleres de teatro, pintura, voleibol, fútbol y matemáticas. Podemos identificar los estudiantes que fueron asignados a esta escuela mediante las líneas presentadas en el mapa. Cada color está identificado en la simbología de tal forma que si la línea es de ese color, entonces el alumno correspondiente fue asignado a ese taller. De forma similar, las demás figuras muestran la situación obtenida para las otras escuelas.

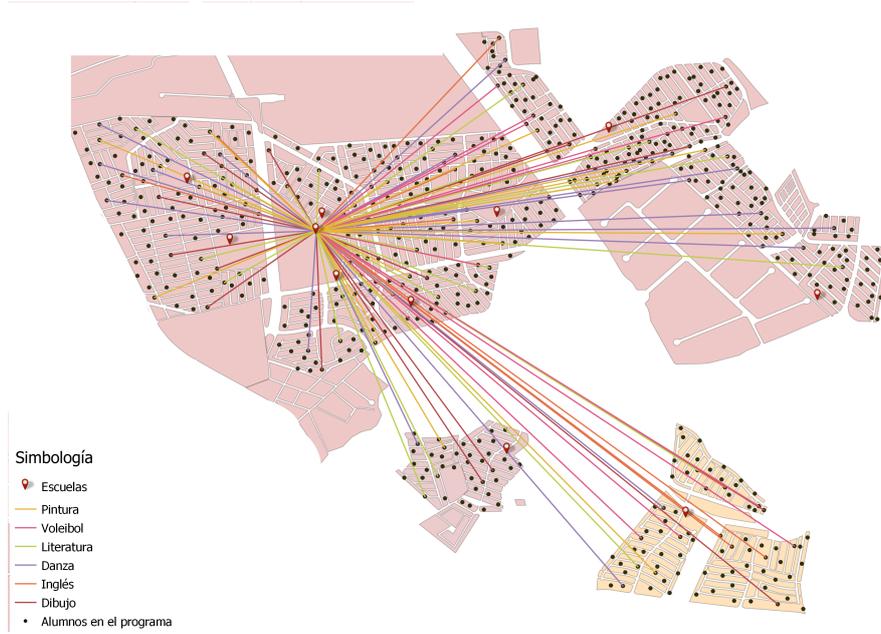


Figura 6: Mapa de la Instancia A con la respectiva asignación de los estudiantes a la escuela 2

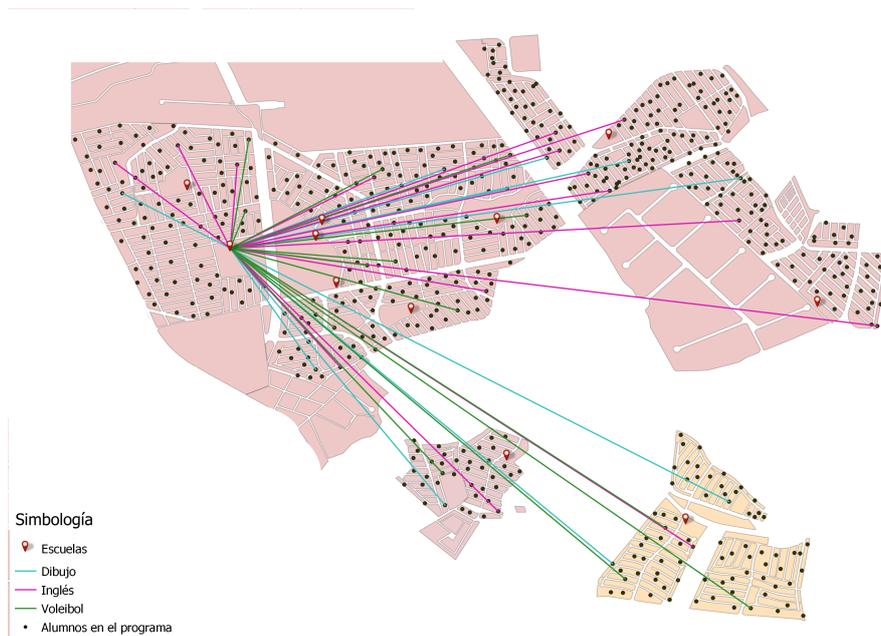


Figura 7: Mapa de la Instancia A con la respectiva asignación de los estudiantes a la escuela 3

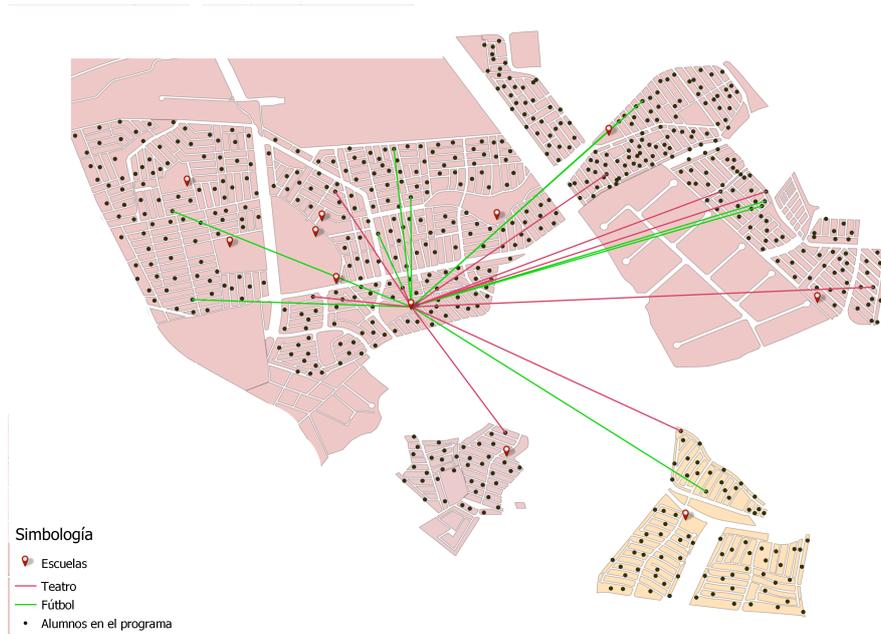


Figura 8: Mapa de la Instancia A con la respectiva asignación de los estudiantes a la escuela 4

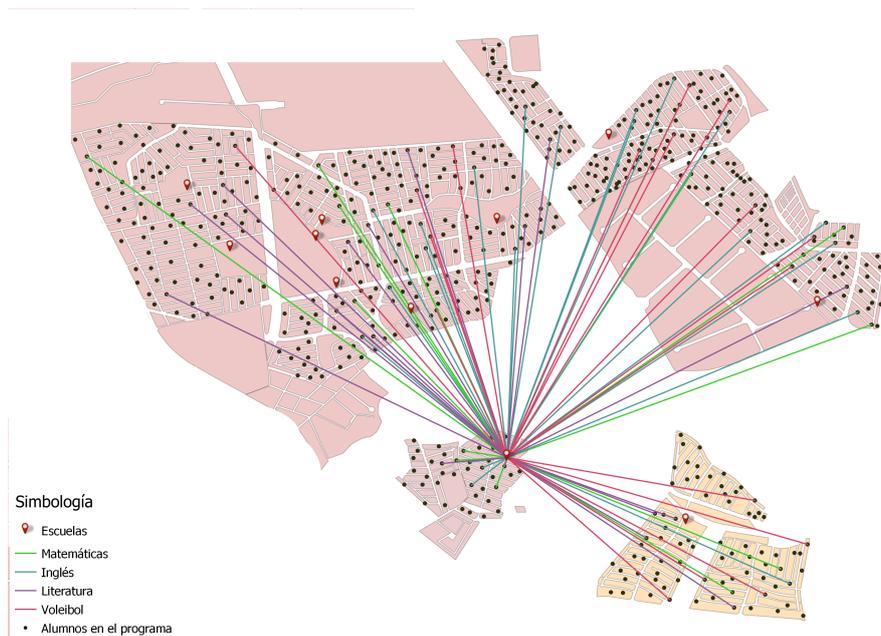


Figura 9: Mapa de la Instancia A con la respectiva asignación de los estudiantes a la escuela 5

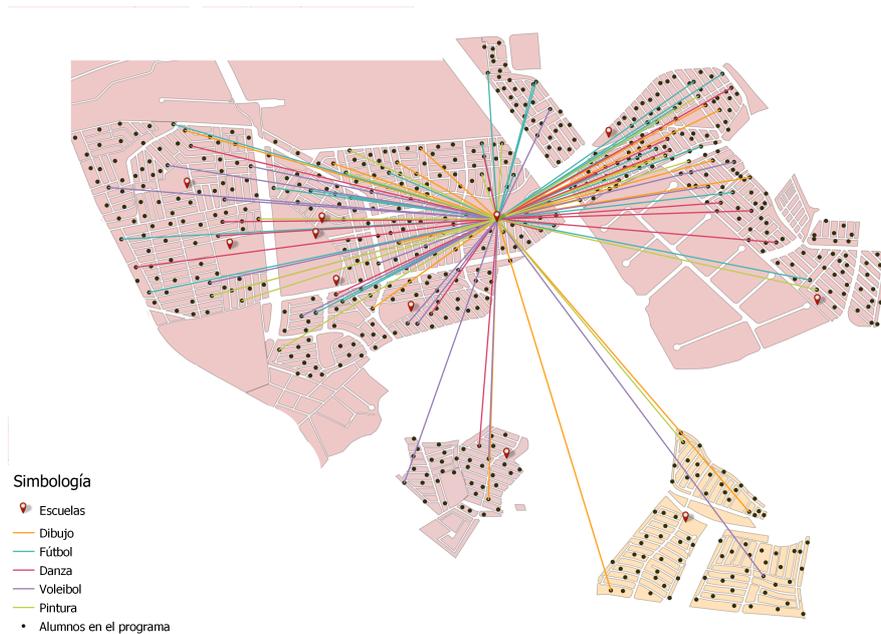


Figura 10: Mapa de la Instancia A con la respectiva asignación de los estudiantes a la escuela 6

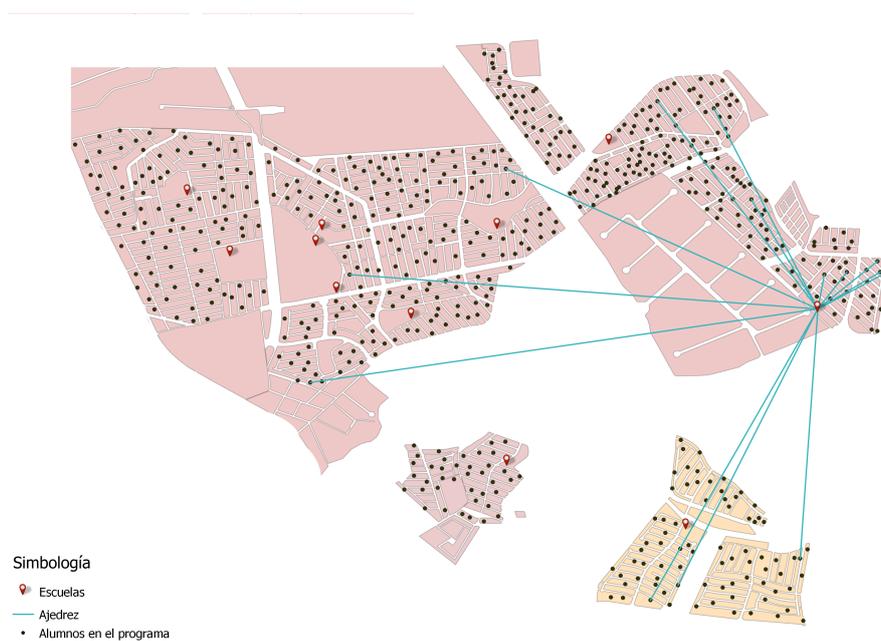


Figura 11: Mapa de la Instancia A con la respectiva asignación de los estudiantes a la escuela 7

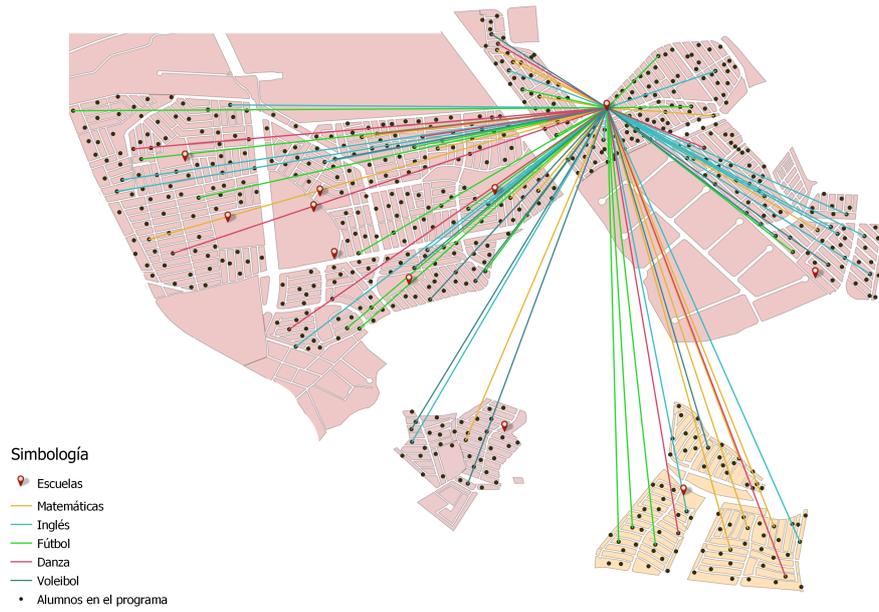


Figura 12: Mapa de la Instancia A con la respectiva asignación de los estudiantes a la escuela 8

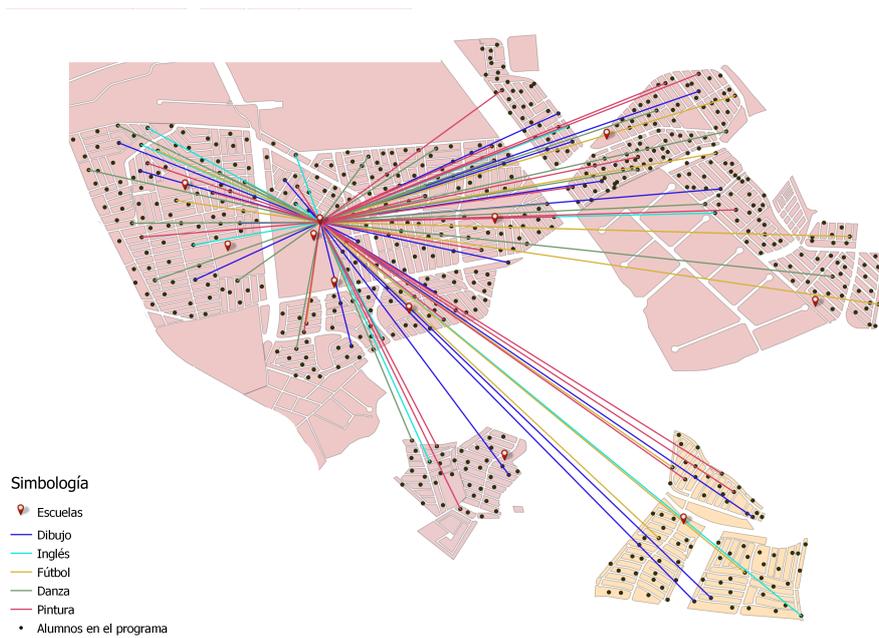


Figura 13: Mapa de la Instancia A con la respectiva asignación de los estudiantes a la escuela 10

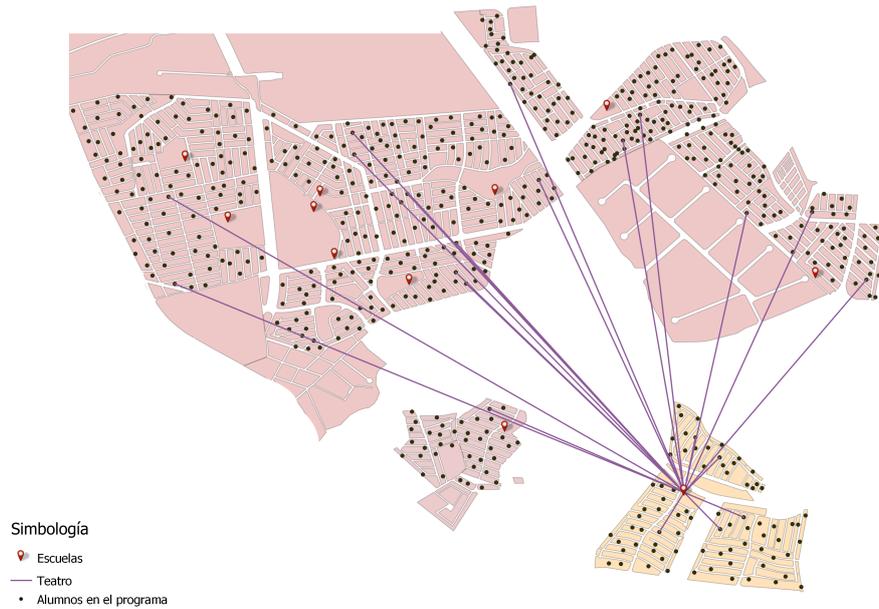


Figura 14: Mapa de la Instancia A con la respectiva asignación de los estudiantes a la escuela 11

En las Figuras 5 - 15 se ilustra gráficamente la solución encontrada por el algoritmo para la Instancia A. Se puede apreciar que en la escuela 9 no se abrió ningún taller. A continuación se muestra esa situación de la escuela 9:

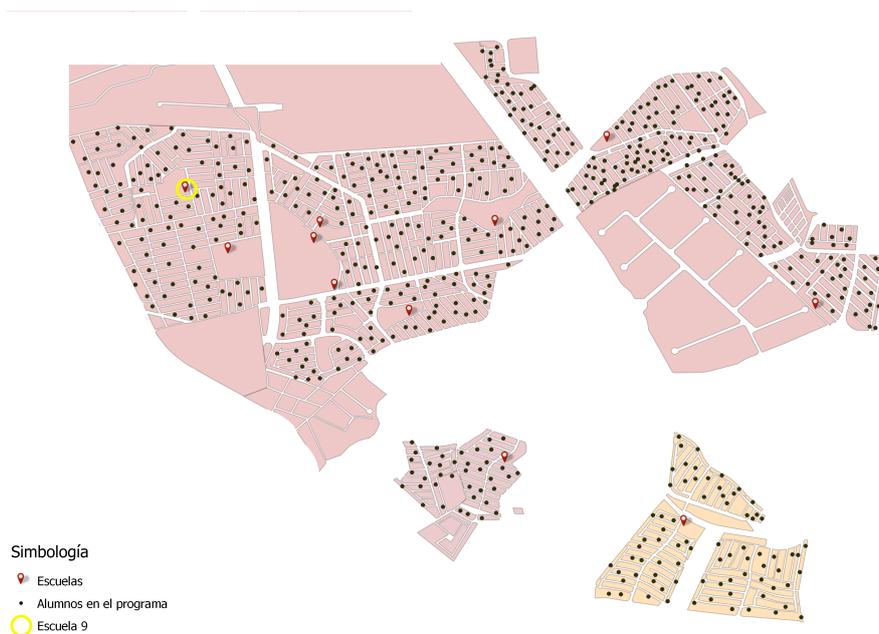


Figura 15: Mapa de la Instancia A y ubicación de la escuela 9

En la Instancia A puede darse el caso en donde una escuela no se necesita para impartir talleres. Esto puede deberse a los costos de la misma y a las preferencias que tienen los estudiantes sobre esa escuela. Por ejemplo, en un caso real puede suceder que haya pocos estudiantes cerca de alguna escuela. Esto podría llevar a que los estudiantes no prefirieran a esa escuela, por quedar muy lejana.

7.2.2 RESULTADOS PARA LA INSTANCIA B

Este caso de estudio cuenta con un total de 7 escuelas, en las cuales se impartirán 8 talleres a un total de 500 estudiantes. El tiempo promedio que tardó el algoritmo en resolver esta instancia fue menor a 480 segundos (8 minutos).

De manera análoga que para la Instancia A, en las siguientes ilustraciones se puede ver la asignación obtenida para los estudiantes a cada una de las escuelas. Además, se ilustra cuales talleres se abren en las escuelas correspondientes.

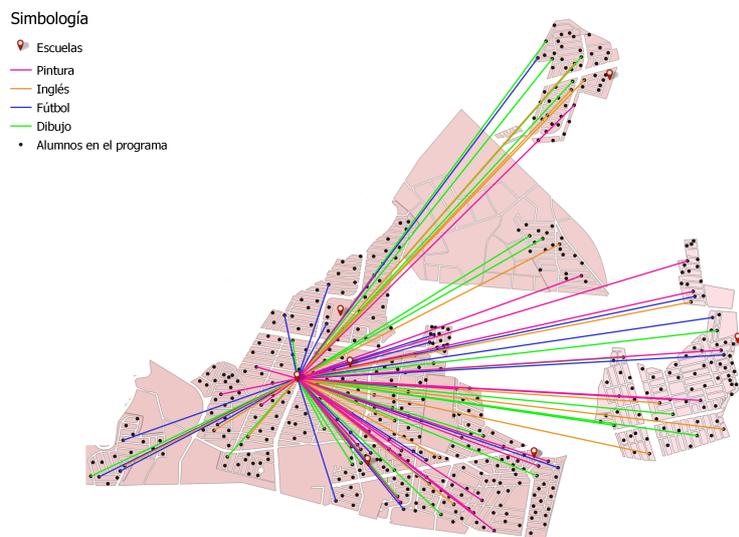


Figura 16: Mapa de la Instancia B con la respectiva asignación de los estudiantes a la escuela 1

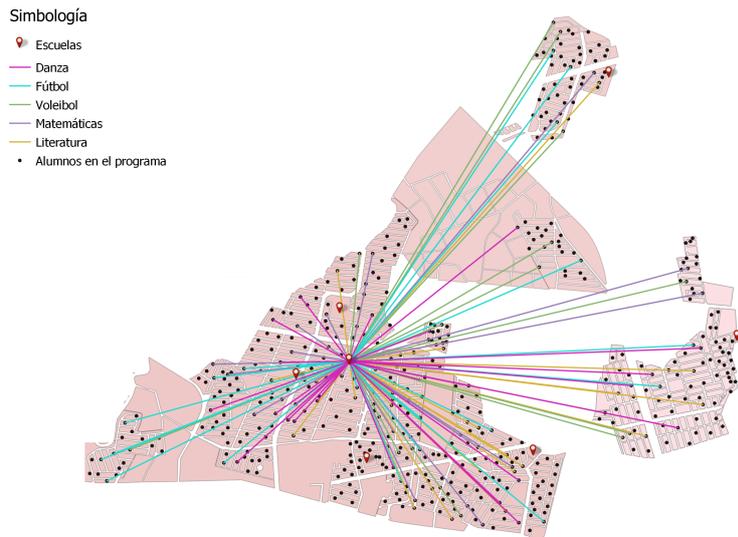


Figura 17: Mapa de la Instancia B con la respectiva asignación de los estudiantes a la escuela 2

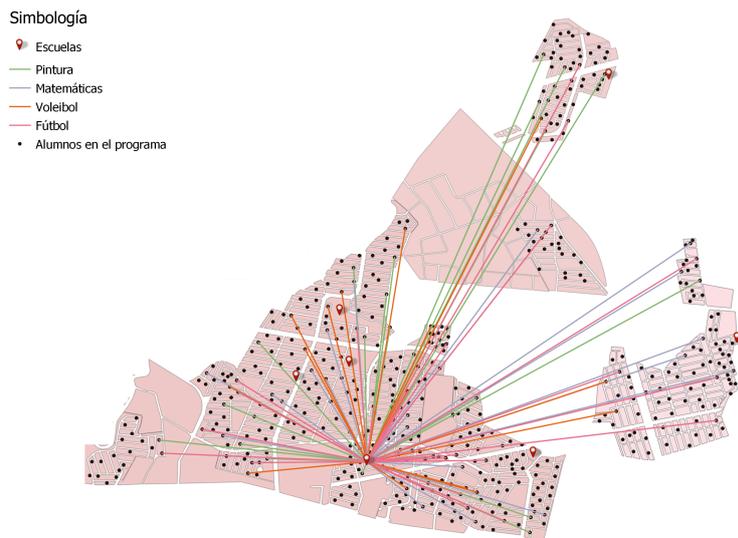


Figura 18: Mapa de la Instancia B con la respectiva asignación de los estudiantes a la escuela 3

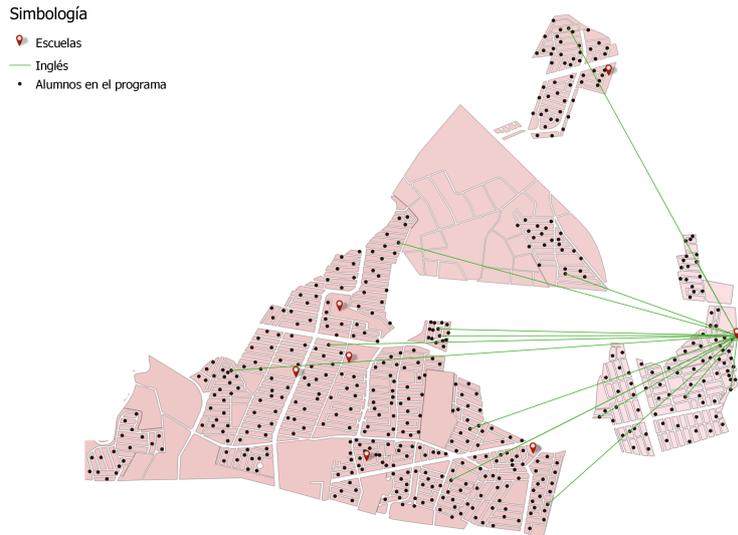


Figura 19: Mapa de la Instancia B con la respectiva asignación de los estudiantes a la escuela 4

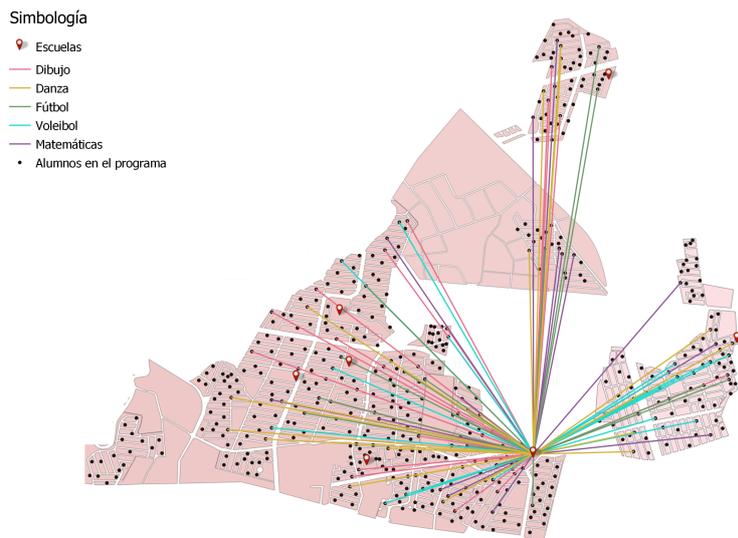


Figura 20: Mapa de la Instancia B con la respectiva asignación de los estudiantes a la escuela 5

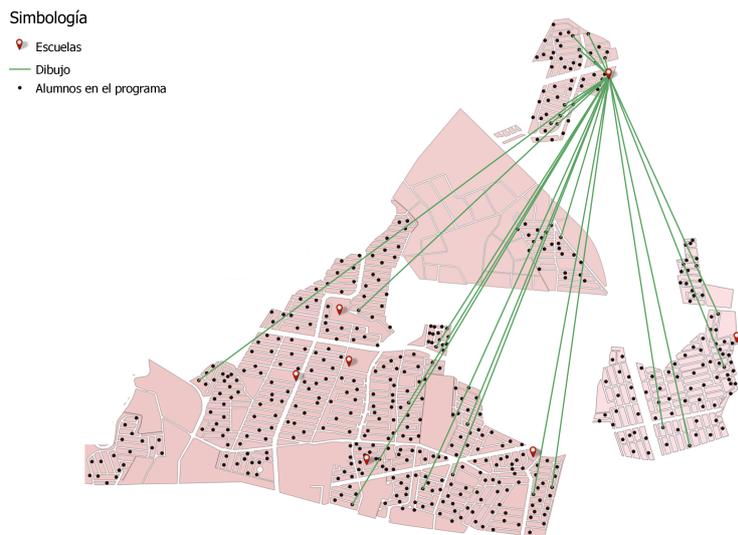


Figura 21: Mapa de la Instancia B con la respectiva asignación de los estudiantes a la escuela 6

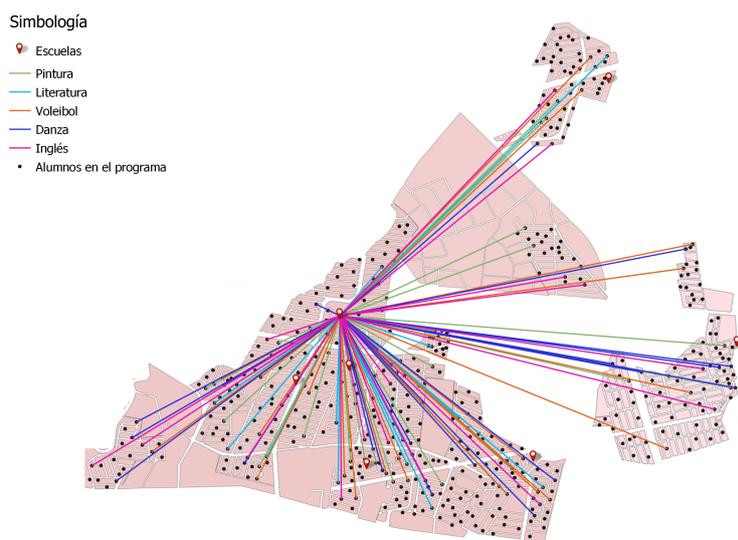


Figura 22: Mapa de la Instancia B con la respectiva asignación de los estudiantes a la escuela 7

En las imágenes recién presentadas se puede observar la localización y asignación que se obtuvo mediante el algoritmo propuesto para esta instancia particular. De estas imágenes se identifica que los estudiantes, en algunas ocasiones, son asignados a escuelas que están más alejadas, es decir, no son las más cercanas. Sin embargo, eso es de esperarse ya que se consideran las preferencias que tienen los estudiantes hacia cada uno de talleres y escuelas. Por eso, no es sorprendente que los estudiantes no se asignen a las escuelas más

cercanas.

Podemos notar la diferencia en los tiempos de cómputo de esta instancia en comparación con la Instancia A, los cuales fueron de 8 a 30 minutos. Estas dos instancias tienen 2 talleres, 4 escuelas y 100 alumnos de diferencia, podemos notar como el tiempo aumenta de forma considerable conforme aumentamos el tamaño de la instancia. Por esto, consideramos importante resaltar el aspecto del poco tiempo de cómputo requerido para ambos casos de estudio. Para realizar otro caso de aplicación real se recomienda tomar en cuenta estos resultados y los de las Tablas 13 - 18, con la intención de realizar las futuras instancias con un tamaño considerable de acuerdo al tiempo que se este dispuesto a invertir en la solución del caso.

CAPÍTULO 8

CONCLUSIONES

Como se ha mencionado en el desarrollo de este trabajo, es de suma importancia que se apliquen proyectos de responsabilidad social en sus diferentes aspectos con la finalidad de mejorar la calidad de vida de la población en general. Por esta misma razón, es aún más importante que el gobierno, tanto a nivel estatal como federal, se preocupe por invertir en proyectos que contribuyan a la sociedad y convertirse en un gobierno socialmente responsable.

La responsabilidad social gubernamental es aquella que tiene el gobierno de un área geográfica ante el hecho de mejorar las condiciones de vida de la población gobernada. Esta responsabilidad se da en diferentes aspectos. Por ejemplo, socialmente se da mediante la creación de políticas públicas y generando proyectos sociales que contribuyan al desarrollo de las comunidades. También incluye ámbitos como la preservación del medio ambiente, velar por los derechos de los trabajadores, entre otros.

Bajo este contexto es que surge este trabajo, donde se abordó un problema de localización de instalaciones con preferencias de los usuarios. Este problema se aplica a la responsabilidad social gubernamental. La finalidad de este proyecto es contribuir de forma positiva al rezago escolar impartiendo talleres de forma gratuita que mejoren las capacidades académicas, sociales y culturales de los estudiantes pero minimizando los costos de operación del gobierno.

La situación aquí estudiada se modela como un problema de programación binivel. Es por esto que se asocia al gobierno con el nivel superior, y busca minimizar los costos de la aplicación del proyecto. Por otro lado, los estudiantes se asocian al nivel inferior con la finalidad de tomar en cuenta sus preferencias para que el proyecto genere en ellos el impacto deseado. Se consideran restricciones relacionadas con las capacidades de cada grupo de acuerdo al taller y escuela en la cual se ofrece. Además, existen otras capacidades

respecto a cuantos talleres puede tener cada escuela y cuantos grupos de cada taller se pueden abrir. El proyecto debe asignar a todos los estudiantes inscritos a talleres abiertos que tengan suficiente capacidad. Todas estas restricciones complican de manera significativa el problema planteado. Es por ello que se requiere diseñar un algoritmo que obtenga una buena solución en un tiempo considerable.

Para resolver este problema, primero, se utiliza una reformulación del problema bi-nivel a un problema de un solo nivel. La reformulación se logra mediante las condiciones de holgura complementaria. Dicha reformulación se resuelve directamente utilizando un software comercial (Xpress). Realizando pruebas computacionales se observó que la reformulación solo llega al óptimo para instancias muy pequeñas (3 escuelas). Para tamaños más grandes se corrió hasta por 24 horas sin que el optimizador lograra terminar. Además, se observó que corriendo el algoritmo durante 6 horas y después por un periodo de 12 horas la holgura de optimalidad mejoró solamente menos de un 0.05 %, por lo cual se concluyó que no se justifica el incremento en tiempo computacional.

Como propuesta de solución alternativa a la reformulación, se presentó un algoritmo basado en el reencadenamiento de trayectorias. En este algoritmo, las soluciones se generan mediante un procedimiento tipo GRASP, el cual en lugar de construir una solución, comienza con una solución aleatoria y la va destruyendo (cerrando talleres) aplicando un procedimiento voraz. El algoritmo está compuesto por tres fases principales: (i) generación de las soluciones, la cual se realiza de forma aleatoria; (ii) aplicación de un procedimiento para reparar la solución; y (iii) el procedimiento destructivo, el cual tiene como salida la solución que entrará en la siguiente fase.

Una vez generada esta solución se procede a aplicar una fase de búsqueda local. Aquí es importante resaltar el hecho de que en esta fase no se resuelve en cada paso el nivel inferior, sino que se calcula el valor de la función objetivo mediante una reasignación de estudiantes para cada vecino y al final solo se resuelve el nivel inferior para el mejor vecino en base a la reasignación. Con esto se redujeron considerablemente los tiempos del algoritmo. La fase final del algoritmo es el reencadenamiento de trayectorias, en el cual se realizan cambios a una solución hasta convertirla en otra solución objetivo. La finalidad de explorar todas las soluciones generadas en este camino es para aportar diversidad al algoritmo.

Como se vió en el Capítulo 7, el algoritmo propuesto arroja buenos resultados. En todas las instancias de prueba se obtuvo una mejor solución en comparación a la encontrada por la reformulación en un tiempo de corrida de 6 horas. Además, el tiempo del algoritmo mejora de manera considerable aún en las instancias más grandes, y para las instancias más pequeñas el tiempo se reduce hasta en un 95 %.

En los casos de estudio podemos observar que el algoritmo propuesto obtiene una buena solución en un tiempo considerable. La Instancia A tenía un tamaño de 11 escuelas donde se impartieron 10 talleres para 600 alumnos, y la solución se obtuvo en un tiempo aproximado de 30 minutos. Ese tiempo es mucho menor al compararlo con los 360 minutos (6 horas) que se usaron como criterio de paro. Más aún, con la reformulación se obtenía una solución de peor calidad. Por otro lado, la Instancia B, que tenía 7 escuelas donde se impartieron 8 talleres para 500 alumnos, fue resuelta en 8 minutos.

Dada toda esta evidencia, se puede concluir que la efectividad del algoritmo propuesto frente a la reformulación del problema binivel como un problema de un solo nivel ha quedado demostrada. Se redujo la holgura de optimalidad obtenida en 6 horas, además de mejorar en un porcentaje alto el tiempo de cómputo. Como observación adicional, se puede mencionar que el algoritmo propuesto funciona mejor para instancias de más de 5 escuelas donde la reformulación comienza a tener problemas para encontrar soluciones factibles.

Otro aspecto importante que se puede resaltar es que para instancias muy grandes (ver la Tabla 18 para 1500 estudiantes), los tiempos de cómputo comienzan a ser mayores a 3 horas. Por lo tanto, se recomienda utilizar las instancias de menor tamaño en caso de requerir resultados en un tiempo menor. En ocasiones puede ser interesante el hecho de considerar como máximo ubicar 20 instalaciones. En la literatura se pueden encontrar algoritmos que funcionan perfectamente para instancias de 50 plantas, como en [5]. Sin embargo, hay que considerar el hecho que en este trabajo no solo se decide cuales escuelas abrir, sino también los tipos de talleres se ofrecerán en esa escuela. Así, las variables de decisión de localización forman una matriz de dimensión $20 * |J|$. Por ejemplo, considerando 8 talleres (para que exista variedad en el problema real), entonces tendremos una matriz de 160 entradas. Esa matriz es evidentemente de mayor tamaño que la representación de las variables de localización que consideran únicamente a las plantas. Esta característica es conocida como ubicación de instalaciones no homogéneas.

Una de las principales aportaciones de este trabajo es el hecho de considerar instalaciones no homogéneas. Sin embargo, es natural hacerlo al considerar la apertura de diferentes tipos de talleres en las escuelas. En la literatura no existen trabajos donde se presenten problemas binivel considerando preferencias de los usuarios con instalaciones no homogéneas. Por este motivo, el algoritmo propuesto se compara únicamente con la reformulación del problema ya que no existe otro algoritmo hasta el momento probado en problemas de este tipo.

En la literatura se pueden encontrar diferentes artículos en los cuales se han trabajado problemas binivel de localización de instalaciones con preferencias de los usuarios con o sin capacidad. Algunos ejemplos se pueden encontrar en [10, 15, 8, 9, 11, 7, 20, 41, 42, 29, 5, 4]. Es importante enfatizar que esos artículos trabajan con instalaciones homogéneas. Además de esta diferencia, no es común encontrar en estos artículos aplicaciones prácticas del problema. Por lo que resulta en otra motivación principal de esta investigación.

En los artículos recién citados se pueden encontrar diferentes métodos propuestos para solucionar el problema de localización de instalaciones con preferencias de los usuarios. Esos métodos van desde una reformulación conseguida por diferentes técnicas hasta algoritmos metaheurísticos. En esta investigación también se puede considerar una contribución a la forma en que se generan las soluciones con un enfoque tipo GRASP, el cual sabemos es un procedimiento que construye una solución agregando elementos uno a uno de forma que al agregar este elemento la solución sea mejor al estado anterior. En este caso se aplica de forma contraria, a partir de una solución aleatoria se van eliminando elementos uno a uno buscando mejorar esta solución. Se puede señalar que este es un enfoque diferente, el cual puede ser considerado novedoso.

Otra contribución importante es la propuesta de un algoritmo basado en el reencadenamiento de trayectorias. A pesar de ser un algoritmo que ha mostrado buenos resultados en diferentes problemas y aplicaciones, es un algoritmo poco común para problemas de localización de instalaciones, y más aún, en problemas considerando la preferencia de los usuarios.

Recapitulando, se pueden resumir las contribuciones en la siguiente lista:

- Proyecto social enfocado en la responsabilidad social gubernamental.

- Variante de un problema de localización de instalaciones capacitadas con preferencias de los usuarios, en la cual ahora se consideran instalaciones no homogéneas.
- Aplicación a un caso de estudio del problema de localización de instalaciones capacitadas con preferencias de los usuarios considerando instalaciones no homogéneas.
- Variante de un algoritmo tipo GRASP que destruye una solución.
- Diseño de un algoritmo de reencadenamiento de trayectorias para resolver el problema en cuestión.

8.1 TRABAJO FUTURO

Antes de llegar a proponer el algoritmo basado en el reencadenamiento de trayectorias se realizaron pruebas con diversos componentes. Por ejemplo, con un algoritmo de búsqueda local iterada que tenía un criterio de paro de cierto número iteraciones sin mejora. Sin embargo, este algoritmo no obtuvo buenos resultados, a pesar de consumir un menor tiempo no se obtuvieron mejores soluciones que las obtenidas por la reformulación en un tiempo de 2 horas. Por esta razón no se recomienda aplicar ese algoritmo en futuras líneas de investigación.

Sobre esta misma línea, otra observación importante para proponer el reencadenamiento de trayectorias fue que se observó que una solución generada que parecía buena en cuanto a valor objetivo, al entrar a la búsqueda local no necesariamente mejoraba de forma significativa. Mientras que para soluciones que en un inicio parecían ser muy malas, al salir de búsqueda local se encontraba que se mejoraban incluso más que las consideradas como buenas soluciones. Esto llevó a la idea de que había que considerar un algoritmo con alto grado de diversificación. Otros métodos con ese esquema se pudieran considerar para futuras investigaciones.

Una futura dirección de este trabajo puede ser proponer un algoritmo híbrido utilizando el reencadenamiento de trayectorias propuesto en este trabajo pero creando las soluciones mediante un método de búsqueda local iterada. Esto con la finalidad de mejorar las soluciones iniciales que, en este caso, fueron creadas de forma aleatoria.

Se debe resaltar la importancia que brinda este trabajo a la aplicación social del mismo. Sin embargo, la situación actual de pandemia por el virus COVID-19 y por las

restricciones sociales existentes que imperan en la actualidad, fue imposible aplicar este proyecto a un caso real. En un futuro cercano es un proyecto que se puede plantear al gobierno de algún municipio para su implementación.

Así como en este trabajo fue muy importante generar una aplicación para este tipo de problemas, se considera como trabajo futuro buscar más aplicaciones en las cuales se pueda utilizar el modelo propuesto aquí, o bien, hacerle ciertas modificaciones. Sería interesante buscar otras áreas en las cuales se consideren instalaciones no homogéneas y preferencias de los clientes para poder generar más aplicaciones.

Para terminar, con la finalidad de incentivar la investigación y facilitar a las personas que consideren investigar dentro esta misma línea se pretende escribir un artículo de investigación en donde se incluyan resultados, hallazgos, consejos y observaciones presentados en este trabajo. Así mismo, esto ayudará a promover la divulgación científica y permitirá conocer los avances que se tienen hasta el momento en problemas de localización de instalaciones capacitadas con preferencias de los usuarios considerando instalaciones no homogéneas.

BIBLIOGRAFÍA

- [1] R. M. Aiex, M. G. Resende, P. M. Pardalos, and G. Toraldo. Grasp with path relinking for three-index assignment. *INFORMS Journal on Computing*, 17(2):224–247, 2005.
- [2] S. Bakhtar, H. Jazayeriy, and M. Valinataj. A multi-start path-relinking algorithm for the flexible job-shop scheduling problem. In *2015 7th Conference on Information and Knowledge Technology (IKT)*, pages 1–6. IEEE, 2015.
- [3] H. I. Calvete, C. Galé, J. A. Iranzo, J.-F. Camacho-Vallejo, and M.-S. Casas-Ramírez. A matheuristic for solving the bilevel approach of the facility location problem with cardinality constraints and preferences. *Computers & Operations Research*, 124: 105066, 2020.
- [4] J.-F. Camacho-Vallejo, M. Casas-Ramírez, and P. Miranda. The p-median bilevel problem under preferences of the customers. *Recent Advances in Theory, Methods and Practice of Operations Research*, pages 121–127, 2014.
- [5] J.-F. Camacho-Vallejo, Á. E. Cordero-Franco, and R. G. González-Ramírez. Solving the bilevel facility location problem under preferences by a stackelberg-evolutionary algorithm. *Mathematical Problems in Engineering*, 2014, 2014.
- [6] V. Campos, R. Martí, J. Sánchez-Oro, and A. Duarte. Grasp with path relinking for the orienteering problem. *Journal of the Operational Research Society*, 65(12): 1800–1813, 2014.
- [7] L. Cánovas, S. García, M. Labbé, and A. Marín. A strengthened formulation for the simple plant location problem with order. *Operations Research Letters*, 35(2):141 – 150, 2007. ISSN 0167-6377. doi: <https://doi.org/10.1016/j.orl.2006.01.012>. URL <http://www.sciencedirect.com/science/article/pii/S0167637706000320>.
- [8] M.-S. Casas-Ramírez and J.-F. Camacho-Vallejo. Considerando preferencias de los clientes en problemas de localización de instalaciones. *Investigación de Operaciones y Ciencias Administrativas*.

- [9] M.-S. Casas-Ramírez and J.-F. Camacho-Vallejo. Solving the p-median bilevel problem with order through a hybrid heuristic. *Applied Soft Computing*, 60:73 – 86, 2017. ISSN 1568-4946. doi: <https://doi.org/10.1016/j.asoc.2017.06.026>. URL <http://www.sciencedirect.com/science/article/pii/S1568494617303691>.
- [10] M.-S. Casas-Ramírez and J.-F. Camacho-Vallejo. Analyzing valid bounds for a facility location bilevel problem with capacities. *International Journal of Combinatorial Optimization Problems and Informatics*, 10(2):8–16, Oct. 2018. URL <https://www.ijcopi.org/index.php/ojs/article/view/101>.
- [11] M.-S. Casas-Ramírez, J.-F. Camacho-Vallejo, and I.-A. Martínez-Salazar. Approximating solutions to a bilevel capacitated facility location problem with customer’s patronization toward a list of preferences. *Applied Mathematics and Computation*, 319:369 – 386, 2018. ISSN 0096-3003. doi: <https://doi.org/10.1016/j.amc.2017.03.051>. URL <http://www.sciencedirect.com/science/article/pii/S0096300317302357>. Recent Advances in Computing.
- [12] S. DE. Ley general de educación. 1994.
- [13] C. de Diputados. Ley orgánica de la administración pública federal. *Recuperado de http://www.diputados.gob.mx/LeyesBiblio/pdf/153_090819.pdf*, pages 30–33, 1979.
- [14] S. de Información y Gestión Educativa. Consulta de esucelas. <https://www.siged.sep.gob.mx/SIGED/escuelas.html>, 2017.
- [15] J. A. Díaz, D. E. Luna, J.-F. Camacho-Vallejo, and M.-S. Casas-Ramírez. Grasp and hybrid grasp-tabu heuristics to solve a maximal covering location problem with customer preference ordering. *Expert Systems with Applications*, 82:67 – 76, 2017. ISSN 0957-4174. doi: <https://doi.org/10.1016/j.eswa.2017.04.002>. URL <http://www.sciencedirect.com/science/article/pii/S0957417417302361>.
- [16] M. Dimopoulou and I. Giannikos. Towards an integrated framework for forest fire control. *European Journal of Operational Research*, 152(2):476 – 486, 2004. ISSN 0377-2217. doi: [https://doi.org/10.1016/S0377-2217\(03\)00038-9](https://doi.org/10.1016/S0377-2217(03)00038-9). URL <http://www.sciencedirect.com/science/article/pii/S0377221703000389>. New Technologies in Transportation Systems.
- [17] L. Federal. Constitución política de los estados unidos mexicanos. Última Reforma DOF 24-12-2020:5–9, 41.
- [18] F. Glover. Tabu search and adaptive memory programming advances, applications and challenges. In *Interfaces in computer science and operations research*, pages 1–75. Springer, 1997.

- [19] F. Glover, M. Laguna, and R. Marti. Scatter search and path relinking: Advances and applications. In *Handbook of metaheuristics*, pages 1–35. Springer, 2003.
- [20] P. Hansen, Y. Kochetov, and N. Mladenovi. *Lower bounds for the uncapacitated facility location problem with user preferences*. Groupe d'études et de recherche en analyse des décisions, HEC Montréal, 2004.
- [21] H. J. F. Huacuja and M. C. Rodríguez. Búsqueda local mejorada para la solución del problema robusto de abastecimiento internacional con capacidad finita (rocis).
- [22] INEE. México cifras básicas. <https://historico.mejoredu.gob.mx/evaluaciones/panorama-educativo-de-mexico-isen/pub-cifras-basicas/>, 2018.
- [23] INEE. Panorama educativo de México. <https://historico.mejoredu.gob.mx/evaluaciones/panorama-educativo-de-mexico-isen/pub-pe/>, 2018.
- [24] INEGI. Biblioteca digital de mapas. <https://www.inegi.org.mx/app/mapas/default.html?t=0710000000000000&ag=19>, 2010.
- [25] INEGI. Glosario. <https://www.inegi.org.mx/app/glosario/default.html?p=cngspspe2017#letraGloS>, 2017.
- [26] M. Laguna and R. Marti. Grasp and path relinking for 2-layer straight line crossing minimization. *INFORMS Journal on Computing*, 11(1):44–52, 1999.
- [27] R. Linfati, J. W. Escobar, and G. Gatica. Un algoritmo metaheurístico para el problema de localización y ruteo con flota heterogénea. *Ingeniería y Ciencia*, 10(19): 55–76, 2014.
- [28] S. Maldonado-Pinto, M.-S. Casas-Ramírez, and J.-F. Camacho-Vallejo. Analyzing the performance of a hybrid heuristic for solving a bilevel location problem under different approaches to tackle the lower level. *Mathematical Problems in Engineering*, 2016, 2016.
- [29] M. Marić, Z. Stanimirović, and N. Milenković. Metaheuristic methods for solving the bilevel uncapacitated facility location problem with clients' preferences. *Electronic Notes in Discrete Mathematics*, 39:43–50, 2012.
- [30] M. Medina Tapia and J. Cerda Troncoso. Modelo de localización óptima de actividades no deseadas aplicado a los residuos sólidos en la región metropolitana. *Ingeniare. Revista chilena de ingeniería*, 16(1):211–219, 2008.

- [31] R. Z. R. Mercado and H. J. E. Balderas. Una metaheurística con reencadenamiento de trayectorias para optimizar planes territoriales.
- [32] H. Min and E. Melachrinoudis. The three-hierarchical location-allocation of banking facilities with risk and uncertainty. *International Transactions in Operational Research*, 8(4):381–401, 2001. doi: 10.1111/1475-3995.00272. URL <https://onlinelibrary.wiley.com/doi/abs/10.1111/1475-3995.00272>.
- [33] C. N. Parajara, G. B. Libotte, G. G. de Paula, and G. M. Platt. A study on the impact of balancing the number of elements of the clusters on the uncapacited p-medians location problem using grasp with path-relinking. In *International Conference on Engineering Optimization*, pages 317–328. Springer, 2018.
- [34] D. Pérez, E. Inga, and R. Hincapié. Optimal sizing of a network for smart metering. *IEEE Latin America Transactions*, 14(5):2114–2119, May 2016. ISSN 1548-0992. doi: 10.1109/TLA.2016.7530403.
- [35] M. G. Resende and J. L. G. Velarde. Grasp: Procedimientos de búsqueda miopes aleatorizados y adaptativos. *Inteligencia Artificial. Revista Iberoamericana de Inteligencia Artificial*, 7(19):0, 2003.
- [36] M. G. Resende and R. F. Werneck. A grasp with path-relinking for the p-median problem. *Journal of Heuristics*, 2002.
- [37] M. G. Resende, R. Martí, M. Gallego, and A. Duarte. Grasp and path relinking for the max-min diversity problem. *Computers & Operations Research*, 37(3):498–508, 2010.
- [38] M. G. Resende, C. C. Ribeiro, F. Glover, and R. Martí. Scatter search and path-relinking: Fundamentals, advances, and applications. In *Handbook of metaheuristics*, pages 87–107. Springer, 2010.
- [39] D. Serra, H. A. Eiselt, G. Laporte, and C. S. ReVelle. Market capture models under various customer-choice rules. *Environment and Planning B: Planning and Design*, 26(5):741–750, 1999. doi: 10.1068/b260741. URL <https://doi.org/10.1068/b260741>.
- [40] E.-G. Talbi. A taxonomy of metaheuristics for bi-level optimization. In *Metaheuristics for bi-level optimization*, pages 1–39. Springer, 2013.
- [41] I. Vasilyev, K. Klimentova, and Y. A. Kochetov. New lower bounds for the facility location problem with clients’ preferences. *Computational Mathematics and Mathematical Physics*, 49(6):1010–1020, 2009.

-
- [42] I. L. Vasilyev and K. B. Klimentova. The branch and cut method for the facility location problem with clients' preferences. *Journal of Applied and Industrial Mathematics*, 4(3):441–454, 2010.
- [43] J. G. Villegas, C. Castañeda, and K. A. Blandón. Mejoramiento de la localización de ambulancias de atención prehospitalaria en medellín (colombia) con modelos de optimización. *CLAIO/SBPO2012*, 123:12, 2012.