

UNIVERSIDAD AUTÓNOMA DE NUEVO LEÓN
FACULTAD DE INGENIERÍA MECÁNICA Y ELÉCTRICA



TESIS

**ANÁLISIS DEL DESEMPEÑO DE ALGORITMOS BASADOS EN LA
TEORÍA DE CAMPO MEDIO PARA PROBLEMAS TIPO MOCHILA**

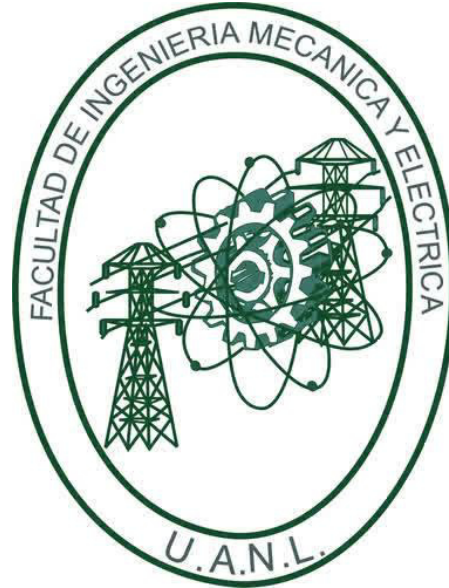
POR

JUAN ANTONIO BANDA MORENO

**COMO REQUISITO PARCIAL PARA OBTENER EL GRADO
DE DOCTOR EN INGENIERÍA CON ESPECIALIDAD EN
INGENIERÍA DE SISTEMAS**

JUNIO, 2018

UNIVERSIDAD AUTÓNOMA DE NUEVO LEÓN
FACULTAD DE INGENIERÍA MECÁNICA Y ELÉCTRICA
SUBDIRECCIÓN DE ESTUDIOS DE POSGRADO



TESIS

**ANÁLISIS DEL DESEMPEÑO DE ALGORITMOS BASADOS EN LA
TEORÍA DE CAMPO MEDIO PARA PROBLEMAS TIPO MOCHILA**

POR

JUAN ANTONIO BANDA MORENO

**COMO REQUISITO PARCIAL PARA OBTENER EL GRADO
DE DOCTOR EN INGENIERÍA CON ESPECIALIDAD EN
INGENIERÍA DE SISTEMAS**

JUNIO, 2018

Universidad Autónoma de Nuevo León
Facultad de Ingeniería Mecánica y Eléctrica
Subdirección de Estudios de Posgrado

Los miembros del Comité de Tesis recomendamos que la Tesis "Análisis del desempeño de algoritmos basados en la teoría de campo medio para problemas tipo mochila", realizada por el alumno Juan Antonio Banda Moreno, con número de matrícula 1407510, sea aceptada para su defensa como requisito parcial para obtener el grado de Doctor en Ingeniería con especialidad en Ingeniería de Sistemas.

El Comité de Tesis


Dr. José Arturo Berrones Santos


Asesor


Dr. Jonás Velasco Álvarez


Co-asesor


Dra. Satu Elisa Schaeffer

Revisora


Dr. Romeo Sánchez Nigenda

Revisor


Dr. Francisco Javier Almaguer Martínez

Revisor

V. Bo.


Dr. Simón Martínez Martínez

Subdirección de Estudios de Posgrado



San Nicolás de los Garza, Nuevo León, junio 2018

A mi familia:

Cruz Banda Acosta (Padre),

Teresa Moreno Almaguer (Madre),

Juan Manuel Banda Moreno (Hermano),

Juana Araceli Banda Moreno (Hermana).

ÍNDICE GENERAL

Agradecimientos	x
Resumen	xi
1 Introducción	1
1.1 Hipótesis	3
1.2 Objetivos	3
1.3 Estructura de la tesis	3
2 Marco teórico	4
2.1 Panorama general de un problema de optimización	4
2.2 Dualidad	8
2.3 Relajación Lagrangiana	9
2.4 Algoritmos de búsqueda local	10
3 Teoría de campo medio	12
3.1 Conceptos básicos	12

<i>ÍNDICE GENERAL</i>	vi
3.2 Distribución de Maxwell-Boltzmann	14
4 Metodología	17
4.1 Campo medio variacional	17
4.2 Casos de prueba	19
4.2.1 Problema de la mochila	20
4.2.2 Problema de la mochila cuadrática	20
4.2.3 Problema de la mochila multidimensional	21
4.3 Ecuaciones de campo medio para los casos de prueba	22
5 Experimentos computacionales	24
5.1 Algoritmos de campo medio para los casos de prueba	24
5.2 Implementación	28
5.3 Resultados	29
5.4 Discusión	37
6 Conclusiones	39
6.1 Contribuciones	39
6.2 Trabajo futuro	40
A Resultados computacionales detallados	41

ÍNDICE DE FIGURAS

5.1	Tiempo promedio obtenido por BB para el KP.	30
5.2	Tiempo promedio obtenido por CM para el KP.	31
5.3	Diagramas de cajas para el KP.	31
5.4	Comparativo del CB entre CM y BB para el KP con $R=10^6$	32
5.5	Tiempos promedio y diagramas de cajas de las DPs para el QKP con tamaños de 100, 200 y 300.	33
5.6	Tiempos promedio y diagramas de cajas de las DPs para el QKP con tamaños de 1000, 2000, 5000 y 6000.	34
5.7	Comparativo del CB de cuatro heurísticos para el QKP en tamaños de 1000 y 2000.	34
5.8	Comparativo del CB de dos heurísticos para el QKP en tamaños de 5000 y 6000.	35
5.9	Tiempos promedio y diagramas de cajas de las DPs para el MKP con tamaños de 100, 250 y 500.	36
5.10	DP promedio de grupo de casos separados por tipos de correlación.	36
5.11	Comparativo del CB de dos heurísticos para el MKP.	37

ÍNDICE DE CUADROS

A.1	Resultados comparativos de CM y BB para el KP con valores de R desde 10^2 hasta 10^5 para los diferentes valores de N	42
A.2	Resultados comparativos de CM y BB para el KP con valores de R desde 10^6 hasta 10^7 para los diferentes valores de N	43
A.3	Resultados de CM para valores de $N \in \{100, 200, 300\}$	44
A.4	Resultados comparativos de CM con tres algoritmos del estado del arte para $N=1000$	45
A.5	Resultados comparativos de CM con tres algoritmos del estado del arte para $N=2000$	46
A.6	Resultados comparativos de CM con algoritmo del estado del arte para $N=5000$	47
A.7	Resultados comparativos de CM con algoritmo del estado del arte para $N=6000$	47
A.8	Resultados comparativos de CM con un algoritmo del estado del arte para problemas de 100 variables con 5 y 10 restricciones.	48
A.9	Resultados comparativos de CM con un algoritmo del estado del arte para problemas de 100 y 250 variables con 30 y 5 restricciones, respectivamente.	49

A.10 Resultados comparativos de CM con un algoritmo del estado del arte para problemas de 250 variables con 10 y 30 restricciones.	50
A.11 Resultados comparativos de CM con un algoritmo del estado del arte para problemas de 500 variables con 5 y 10 restricciones.	51
A.12 Resultados comparativos de CM con un algoritmo del estado del arte para problemas de 500 variables con 30 restricciones.	52

AGRADECIMIENTOS

A la Universidad Autónoma de Nuevo León y a la Facultad de Ingeniería Mecánica y Eléctrica por permitirme realizar mis estudios de posgrado en sus instalaciones.

Al Consejo Nacional de Ciencia y Tecnología por el apoyo económico brindado durante el doctorado.

A todos mis amigos que han creído en mí: chiapanecos, colombianos, vallenses, regiomontanos, coahuilenses, oaxaqueños, queretanos, nicaragüenses y otros.

A los profesores del posgrado por su paciencia y dedicación, gracias por compartir sus conocimientos.

A mis revisores de tesis, por el tiempo que dedicaron en leerla y por sus aportes para mejorarla.

A la Dra. Elisa que siempre me ofreció su ayuda incondicional. Es una de las personas que más admiro. En lo personal me da mucho gusto el haberla conocido, gracias por sus enseñanzas y consejos.

A los doctores Berrones y Jonás por la oportunidad de colaborar con ustedes, siempre guiándome por el camino correcto para sacar a flote el presente trabajo.

RESUMEN

Juan Antonio Banda Moreno.

Candidato para obtener el grado de Doctor en Ingeniería con especialidad en Ingeniería de Sistemas.

Universidad Autónoma de Nuevo León.

Facultad de Ingeniería Mecánica y Eléctrica.

Título del estudio: ANÁLISIS DEL DESEMPEÑO DE ALGORITMOS BASADOS EN LA TEORÍA DE CAMPO MEDIO PARA PROBLEMAS TIPO MOCHILA.

Número de páginas: 58.

Se propone una metodología basada en teorías de campo medio para resolver problemas tipo mochila con funciones objetivo lineales y cuadráticas a gran escala. Además, se consideran problemas desde una hasta treinta restricciones lineales. Estos problemas son conocidos en la literatura como el problema de la mochila, el problema de la mochila cuadrática y el problema de la mochila multidimensional. Fueron seleccionados por su sencilla interpretación y múltiples aplicaciones en la vida real. Asimismo, en los dos primeros problemas, se toman casos en los que se sabe que dado el algoritmo exacto no es conveniente su implementación. Para el tercer problema simplemente se toman los casos más usados para validar la eficiencia de algoritmos, casos en los que el valor óptimo es desconocido para algunos tipos.

La esencia de la metodología propuesta es encontrar una función de distribución

de probabilidad asociada a un problema de optimización. Una de las más usadas es la distribución de Boltzmann que involucra la función objetivo y sus restricciones, mediante la relajación Lagrangiana, transformando un problema discreto en uno continuo. Sin embargo, la distribución por sí sola es compleja y difícil de tratar, por lo que se realiza una aproximación de campo medio que resulta de elegir de un conjunto de distribuciones sencillas, aquella que ofrezca la menor diferencia entre la distribución de Boltzmann y ésta.

Los problemas de optimización usados para validar la eficiencia de la metodología propuesta son binarios por lo que la distribución general de campo medio que se plantea es adecuada para este tipo. En dado caso en el que se quiera utilizar esta metodología en otro tipo de problemas, es necesario presentar otra distribución de campo medio que se ajuste a ellos.


El enfoque de campo medio usado en el presente trabajo permite encontrar ecuaciones independientes que estiman la probabilidad de ocurrencia de cada una de las variables a través del espacio dual; es decir, dando valores a los multiplicadores de Lagrange, es posible construir un vector de probabilidades en el que cada elemento representa la probabilidad de activar una determinada variable de una solución del problema binario.

El algoritmo propuesto es determinista y capaz de encontrar soluciones de alta calidad en los problemas de prueba, con tiempos de ejecución cuyos órdenes de magnitud son inferiores a algoritmos recientemente estudiados.

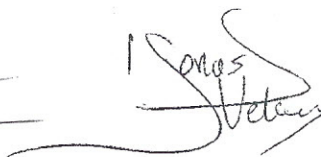
OBJETIVOS Y MÉTODO DE ESTUDIO: Distinguir e identificar las bondades de utilizar un modelo probabilístico de campo medio, en problemas tipo mochila, para la construcción de soluciones factibles. Para ello, se parte de que cualquier problema de optimización está relacionado con la distribución de probabilidad de Boltzmann la cual es aproximada por una distribución mucho más sencilla. Teniendo la distribución aproximada es posible construir una solución binaria mediante técnicas de redondeo.

CONTRIBUCIONES Y CONCLUSIONES: Se logra obtener una metodología rápida y eficaz para construir soluciones factibles en problemas de gran escala de tipo mochila. Se abordan problemas con restricciones lineales, funciones objetivo cuadráticas y lineales, e inclusive problemas con múltiples restricciones. En todos estos casos se encuentran soluciones de calidad en poco tiempo, en promedio conforme crece su tamaño la diferencia entre lo mejor conocido y la solución de la metodología propuesta tiende a disminuir. Esto último es debido a que la teoría de campo medio, como su nombre lo indica, trabaja con un esquema de promedios por lo que a medida que crece el número de variables las soluciones tienden a ser más precisas.

Firmas de asesores:



Dr. José Arturo Berrones Santos



Dr. Jonás Velasco Álvarez

CAPÍTULO 1

INTRODUCCIÓN

Diversos problemas de optimización reales tienen estructuras de modelos combinatorios, que en su mayoría no existe metodología para resolverlos, o bien, existe pero a un costo computacional muy alto. Algunos ejemplos son los problemas tipo mochila, los cuales tienen varias aplicaciones en múltiples disciplinas y son usados para examinar la robustez de nuevos algoritmos. Para estos problemas no existe (al menos por ahora) un algoritmo que sea capaz de encontrar la solución óptima en tiempo polinomial [26]. Sin embargo, hay algoritmos que los resuelve en tiempo pseudo-polinomial, cuya complejidad depende de los valores de entrada dados por el caso [39].

Los métodos usados para resolver los problemas de optimización combinatoria son los exactos y heurísticos. Los métodos exactos obtienen la mejor solución (*solución óptima*), los más conocidos son ramificación y acotamiento [48]; y ramificación y corte [34]. Cuando un problema es de naturaleza tal que no se conoce ningún método exacto para su solución, o aunque exista, su uso es computacionalmente muy costoso se recurre a un método heurístico, el cual genera soluciones de buena calidad. Dentro de los heurísticos básicos se encuentran los constructivos y los de búsqueda local [1, 9, 29] (de estos últimos se explica a más detalle en la sección 2.4), que a su vez constituyen los cimientos de los métodos metaheurísticos. Un metaheurístico es una estrategia que permite escapar de óptimos locales e incrementar su desempeño en el

proceso de búsqueda [47].

Los métodos constructivos generan soluciones, generalmente factibles, mediante una cierta estrategia iterativa que permita incorporar componentes hasta ser completada. Por otra parte, los métodos de búsqueda local parten de una solución inicial e iterativamente reemplazan la solución actual por una mejor solución, la cual se encuentra dentro de un vecindario definido a partir de la solución actual y una serie de movimientos. Un método de este tipo finaliza cuando ya no existe ninguna solución accesible que la mejore.

Los métodos heurísticos y metaheurísticos más usados para resolver problemas combinatorios son búsqueda tabú, recocido simulado, enjambre de partículas, GRASP (Greedy Randomized Adaptive Search Procedure), algoritmos genéticos, colonia de hormigas, entre otros [16, 26, 30, 31, 35, 51]. Estos heurísticos dependen de una distribución de probabilidad arbitraria, que usualmente tiene poca o ninguna relación con el problema, por ejemplo, la variabilidad de un vector de solución (denominado individuo en los algoritmos genéticos) está dada por un operador llamado mutación. La mutación hace ligeros cambios en la solución usando una distribución uniforme que no tiene ninguna relación con el problema.

Otro tipo de metodologías que han sido recientemente estudiadas son las basadas en la construcción de modelos probabilísticos [41], las cuales han logrado obtener buenas soluciones para una amplia variedad de problemas de optimización. Dentro de este contexto de los modelos probabilísticos se introduce la teoría de campo medio, en específico campo medio variacional que reemplaza una distribución difícil por una más sencilla. Una función de distribución de probabilidad que se le puede asociar a cualquier problema de optimización es la distribución de Maxwell-Boltzmann, la cual ofrece la mayor cantidad de información, sin embargo, es muy difícil de manejar por lo que se aproxima por una distribución fácil de tratar. La distribución sencilla es elegida de tal forma que la diferencia entre ambas distribuciones sea mínima.

1.1 HIPÓTESIS

Una buena solución factible puede construirse mediante un modelo probabilista de campo medio en problemas tipo mochila con un tiempo de cómputo reducido.

1.2 OBJETIVOS

Distinguir e identificar las bondades de utilizar un modelo probabilístico de campo medio, en problemas tipo mochila, para la construcción de soluciones factibles.

1.3 ESTRUCTURA DE LA TESIS

En el capítulo 2 se describen conceptos generales de optimización para quienes no estén familiarizados con el área. De la misma forma en el capítulo 3 se exponen ideas y conceptos generales sobre la teoría de campo medio que permitirán comprender mejor la metodología propuesta en el capítulo 4; además, en este último, se introducen los problemas utilizados de casos prueba para validar la eficiencia de los algoritmos propuestos. En el capítulo 5 se definen las características de los casos prueba y del equipo utilizado. También se describen los algoritmos implementados y los resultados obtenidos. Finalmente en el capítulo 6 se evidencian las conclusiones obtenidas del presente trabajo y se puntualizan algunas líneas para trabajo futuro.

CAPÍTULO 2

MARCO TEÓRICO

Un problema de optimización tiene ciertas características y formas de representarlas matemáticamente, de las que se desprenden los modelos de programación lineal. Estos últimos tienen asociados un problema denominado dual que genera el mismo valor objetivo que el problema original llamado primal. Otra forma de representarlos es mediante relajaciones Lagrangianas, de las cuales existen condiciones que garantizan el haber encontrado la mejor solución. Además, hay diferentes formas de clasificar un problema de optimización en base a que tan difícil es resolverlo. En dichos problemas se puede ofrecer la mejor solución o una buena solución, para los que se encuentran muchas técnicas conocidas que lo realizan. En el presente capítulo se muestran conceptos generales de lo anteriormente mencionado, que permiten familiarizar al lector para una mejor comprensión del presente trabajo.

2.1 PANORAMA GENERAL DE UN PROBLEMA DE OPTIMIZACIÓN

Diversos problemas pueden modelarse como problemas de optimización en áreas como matemáticas, ciencias aplicadas, ingeniería, economía, medicina, estadística, entre otras. Un *problema de optimización* está compuesto por un conjunto de variables

independientes x_1, x_2, x_3, \dots , las cuales representaremos por un vector \vec{x} (este tipo de notación se usará en otros casos con otras letras); las cuales producen un beneficio que surge de una función objetivo; además el problema tiene ciertas limitantes o restricciones [23]. Dicho problema puede expresarse mediante un modelo matemático el cual contiene expresiones algebraicas que permiten comprender y analizar el comportamiento de un sistema.

Las restricciones a satisfacer de un modelo matemático pueden ser de igualdad y/o de desigualdad, formando una región de factibilidad entre ellas, es decir, se crea una región en la que se encuentra el conjunto de todas las soluciones que satisfacen las restricciones, las cuales son conocidas como *soluciones factibles*. La función objetivo puede ser maximizada o minimizada, según sea el caso. La solución de la región factible que brinde el mejor valor de la función objetivo es llamada *solución óptima* con valor óptimo.

Un modelo matemático general de un problema de optimización puede representarse, para un caso de minimización, de la siguiente forma:

$$\begin{aligned} \min \quad & f(\vec{x}) & (2.1) \\ \text{sujeto a: } \quad & g_i(\vec{x}) \geq 0, & i \in I; \\ & h_j(\vec{x}) = 0, & j \in J; \\ & \vec{x} \in \mathbb{R}^n, \end{aligned}$$

donde $f(\vec{x})$, $g(\vec{x})$ y $h(\vec{x})$ son funciones continuas diferenciables. La expresión $f(\vec{x})$ representa la función objetivo, $g(\vec{x})$ y $h(\vec{x})$ representan los conjuntos de restricciones de desigualdad e igualdad, respectivamente. Para el caso de maximización se tiene una representación similar, únicamente es necesario realizar la siguiente transformación:

$$\max f(\vec{x}) \equiv \min -f(\vec{x}). \quad (2.2)$$

Un submodelo de la ecuación (2.1) se muestra a continuación, el cual representa al problema de optimización conocido como modelo de *programación lineal*, en el que

los valores de \vec{x} son no negativos, la función objetivo y restricciones son lineales.

$$\begin{aligned} \min \quad & f(\vec{x}) \\ \text{sujeto a:} \quad & g_r(\vec{x}) \geq 0, \quad r \in R; \\ & \vec{x} \geq 0. \end{aligned} \tag{2.3}$$

Otro submodelo de la ecuación (2.1) es aquel en el que las variables de \vec{x} deben tomar valores enteros, con función objetivo y restricciones lineales. A este tipo de casos se les denomina problemas de *programación entera*. Note que al restringir el problema de optimización de tomar valores enteros, la función objetivo y sus restricciones dejan de ser funciones continuas y diferenciables. Generalmente, los métodos conocidos para resolver un problema de programación lineal, no pueden ser usados directamente para encontrar la solución óptima pero sí son usados para aproximar la solución. Un caso especial de programación entera se da cuando todas las variables se restringen a tomar valores de 0 y 1. Conocido en la literatura como modelos de *programación entera binaria 0 – 1*, teniendo una representación matemática como sigue:

$$\begin{aligned} \min \quad & f(\vec{x}) \\ \text{sujeto a:} \quad & g_r(\vec{x}) \geq 0, \quad r \in R; \\ & \vec{x} \in \{0, 1\}^N. \end{aligned} \tag{2.4}$$

En complejidad computacional los problemas de optimización se clasifican en clases de complejidad [21, 38, 39], que se definen como una colección infinita de lenguajes. A continuación se mencionan las más relevantes pero antes es necesario definir los conceptos *máquina de Turing* y *reducción*. Una máquina de Turing es un dispositivo que opera símbolos, en base a ciertas reglas definidas, sobre una cinta y es utilizada para simular la lógica de cualquier algoritmo. Se dice que B reduce a A si existe una transformación R , la cual produce una entrada equivalente $R(x)$ de A para cada entrada x de B .

1. La clase P es una clase de complejidad que contiene todos los problemas que son decididos por máquinas de Turing deterministas en tiempo polinomial.

2. La clase NP es una clase de complejidad que contiene todos los problemas que son decididos por máquinas de Turing no deterministas en tiempo polinomial.
3. Sea L un lenguaje en NP, se dice que L es NP-completo si cualquier lenguaje $L' \in \text{NP}$ se puede reducir a L .
4. Un lenguaje L es llamado NP-difícil si todos los lenguajes en NP se reducen a L pero no se sabe si L está en NP.

Los problemas de programación entera lineal pueden resolverse usando un método exacto o un método heurístico. Los métodos exactos logran encontrar la solución óptima del problema, sin embargo, en diversos casos el tiempo de respuesta crece exponencialmente conforme incrementa el tamaño de la instancia, ello depende de la complejidad del problema. Los métodos exactos más usados para resolver este tipo de problemas son el algoritmo de ramificación y acotamiento y, el algoritmo de ramificación y corte. Cuando se conoce que la complejidad de un problema es alta, como consecuencia desarrollar un método exacto para un conjunto de instancias grandes no es viable debido a que no podrá resolverlas; un método heurístico es una buena opción para encontrar una buena solución factible. Estos métodos encuentran en un tiempo de respuesta razonable soluciones factibles; sin embargo, no garantizan encontrar la solución óptima. Los métodos heurísticos son algoritmos basados en la experiencia para resolver problemas e inspirados muchas de las veces en comportamientos de la naturaleza, los más usados son la búsqueda tabú, GRASP, recocido simulado, algoritmos genéticos, entre otros [39].

Existen otro tipo de problemas binarios 0 – 1 en los que la función objetivo es cuadrática y sus restricciones lineales, por lo que los métodos exactos para resolver problemas de programación entera no garantizan encontrar la solución óptima. Un ejemplo de este tipo es el problema de la mochila cuadrática el cual se describe a detalle en el capítulo 4 dado que es uno de los casos de prueba para validar la eficiencia del algoritmo propuesto.

2.2 DUALIDAD

Un problema de programación lineal, llamado *primal*, de la forma de la ecuación (2.3), puede representarse también en forma matricial como

$$\begin{aligned} \min \quad & \vec{c}\vec{x}^T \\ \text{sujeto a:} \quad & A\vec{x} \geq \vec{b}; \\ & \vec{x} \geq 0, \end{aligned} \tag{2.5}$$

donde A es una matriz que contiene los coeficientes de las restricciones, \vec{b} es un vector que contiene los valores del lado derecho de las restricciones y \vec{c} son los coeficientes de la función objetivo. Al problema primal se le puede asociar otro problema de programación lineal llamado problema *dual*, que en ocasiones es más fácil de resolver que el primal. Su representación en forma matricial es

$$\begin{aligned} \max \quad & \vec{w}\vec{b}^T \\ \text{sujeto a:} \quad & \vec{w}^T A \leq \vec{c}; \\ & \vec{w} \geq 0. \end{aligned} \tag{2.6}$$

La representación anterior es nombrada forma *canónica* de dualidad, la cual se da cuando se tienen restricciones de desigualdad. Otra representación que es equivalente es la forma *estándar* de dualidad, la cual se aplica cuando se tienen restricciones de igualdad en el primal como a continuación se muestra:

$$\begin{aligned} \min \quad & \vec{c}\vec{x}^T \\ \text{sujeto a:} \quad & A\vec{x} = \vec{b}; \\ & \vec{x} \geq 0, \end{aligned} \tag{2.7}$$

entonces el dual se define como:

$$\begin{aligned} \max \quad & \vec{w}\vec{b}^T \\ \text{sujeto a:} \quad & \vec{w}^T A \leq \vec{c}; \\ & \vec{w} \text{ no restringida.} \end{aligned} \tag{2.8}$$

2.3 RELAJACIÓN LAGRANGIANA

En ocasiones un problema de programación entera es tan difícil de resolver directamente que es necesario quitar del conjunto de restricciones aquellas que hacen más difícil el problema. Sin embargo, este nuevo problema de optimización más sencillo ofrece un valor óptimo que es al menos tan grande (pequeño para el caso de minimizar), caso de maximizar, como el valor óptimo del problema original. El problema de optimización sencillo se le denomina *relajación* del problema original y ofrece una cota superior para el caso de maximización (cota inferior para el caso de minimizar). Cada solución factible es una cota inferior en el caso de maximizar (cota superior en el caso de minimizar). Una extensión de este enfoque de reemplazar un problema entero difícil por un problema de optimización más sencillo es la relajación Lagrangiana. La idea es no eliminar las restricciones difíciles sino agregarlas a la función objetivo con los multiplicadores de Lagrange. Estos últimos en realidad son las variables duales asociadas a un problema de optimización.

A continuación se brinda un panorama general sobre la relajación Lagrangiana y las condiciones necesarias para garantizar haber encontrado el valor óptimo de un problema de optimización.

Dada una función $f(\vec{x})$ continua y diferenciable sujeta a restricciones de igualdad $g_r(\vec{x}) = 0$ con $r \in R$, se puede lograr la solución óptima resolviendo el problema no restringido (problema Lagrangiano):

$$\min f(\vec{x}) - \sum_{r \in R} \mu_r g_r(\vec{x}),$$

donde μ_r son valores no negativos, además son llamados multiplicadores de Lagrange. Si se tiene un problema de optimización con restricciones de desigualdad de la forma

$$\begin{aligned} \min \quad & f(\vec{x}) \\ \text{sujeto a:} \quad & g_r(\vec{x}) \geq 0, \end{aligned} \tag{2.9}$$

las restricciones pueden convertirse en restricciones de igualdad agregando nuevas

variables e_r ,

$$e_r^2 = -g_r(\vec{x}),$$

por lo que el problema Lagrangiano para este caso se define como:

$$\min \mathcal{L} = f(\vec{x}) - \sum_{r \in R} \mu_r [g_r(\vec{x}) + e_r^2].$$

Para encontrar el mínimo de $f(\vec{x})$ deben cumplirse las siguientes tres condiciones, para valores no negativos de μ_r :

$$\frac{\partial \mathcal{L}}{\partial x_j} = \frac{\partial f(\vec{x})}{\partial x_j} - \sum_{r \in R} \mu_r \frac{\partial g_r(\vec{x})}{\partial x_j} = 0, \quad j \in N; \quad (2.10)$$

$$\frac{\partial \mathcal{L}}{\partial \mu_r} = g_r(\vec{x} + e_r^2) = 0 \leftrightarrow g_r \geq 0, \quad r \in R; \quad (2.11)$$

$$\frac{\partial \mathcal{L}}{\partial e_r} = 2\mu_r e_r = 0 \leftrightarrow \mu_r g_r = 0, \quad r \in R. \quad (2.12)$$

Finalmente, en la siguiente sección se describe una idea general sobre el funcionamiento de los algoritmos de búsqueda local.

2.4 ALGORITMOS DE BÚSQUEDA LOCAL

Los algoritmos de búsqueda local son estrategias usadas para mejorar la calidad de una solución inicial de forma iterativa, trabajan con la hipótesis de que existe un vecindario $N(\vec{x}^*)$ definido sobre el espacio de todas las posibles soluciones que son similares a la mejor solución actual \vec{x}^* [1, 29]. La elección de la apropiada solución inicial y del movimiento definido para generar el vecindario influirá enormemente para la creación de soluciones cuyo valor de la función objetivo sea mejor que lo que se tiene hasta el momento [8].

La idea principal de la búsqueda local es generar un nuevo vecindario en base a la mejor solución actual, repitiendo el procedimiento hasta que ya no pueda encontrarse una mejor solución actual. Hay dos formas de elegir la mejor solución actual: la

primer solución que es mejor que lo que se tiene hasta el momento y la mejor solución del vecindario, esta última suele ser muy tardada, por lo que se usa más la primera.

A continuación se muestra el algoritmo de una búsqueda local para el caso de maximización.

Algoritmo 1 (Búsqueda local)

Entrada: Solución inicial \vec{x} .**mientras** *Exista* $\vec{x}_l \in N(\vec{x})$ tal que $f(\vec{x}_l) > f(\vec{x})$ y \vec{x} factible **hacer**| $\vec{x} \leftarrow \vec{x}_l$;**fin****devolver** $(\vec{x}, \text{EvaluaBeneficio}(\vec{x}))$.

CAPÍTULO 3

TEORÍA DE CAMPO MEDIO

En el presente capítulo se proporcionan ideas generales sobre el tema tomadas de los textos [14, 24, 36, 37]. Mismas que permitirán una mejor comprensión sobre la metodología propuesta. Se presenta la distribución de Maxwell–Boltzmann como una función de distribución de probabilidad general de un sistema, la cual en términos físicos indica las probabilidades de los diferentes estados de energía de ser ocupados, por lo que para cada estado en un sistema cerrado se tiene una máxima entropía y mínima energía. La entropía es una medida de desorden o aleatoriedad de partículas, de manera que las posiciones de las partículas son menos predecibles y aumenta la entropía en un sistema desordenado [12]. Un sistema cerrado es un sistema que está conectado a otro sistema, no puede intercambiar materia pero sí energía.

3.1 CONCEPTOS BÁSICOS

La teoría de campo medio (TCM) estudia el comportamiento de modelos complejos de gran tamaño a través de un modelo más sencillo, la idea principal es centrarse en una determinada entidad y sustituir su interacción con el resto de las entidades, por una interacción promedio reduciendo un problema de múltiples cuerpos por uno solo. Teniendo como objetivo principal el deducir a partir de la noción de sus componentes microscópicas, propiedades o características de un sistema macroscópico.

TCM ha sido usado exitosamente en problemas con estructuras de tipo grafo en los que se ha podido estimar la distribución de probabilidad de cualquier variable, es decir, la probabilidad marginal de un nodo añadiendo la probabilidad conjunta con respecto al resto de las variables. Por lo que, en términos de estadística, se quiere encontrar una función de distribución de probabilidad de las variables del sistema; en términos de física, se desea hallar una función de mínima energía que se expresa como la suma de la energía promedio y la entropía. En un sistema cerrado surgen dos principios que a continuación se mencionan.

- Principio de entropía máxima: la entropía es máxima cuando el sistema alcanza el equilibrio con energía interna fija.
- Principio de mínima energía: la energía total es mínima cuando el sistema alcanza el equilibrio con entropía fija.

Una forma de abordar el enfoque de campo medio es usando relajaciones Lagrangianas para el problema de mínima energía, la cual se usa en el presente trabajo. La idea es resolver el problema primal via dual, es decir, dar posibles valores a los multiplicadores de manera que mejore el valor de la función objetivo primal.

A continuación, consideremos un sistema aislado en el que la energía no puede escapar por lo que la energía total del sistema U se mantiene constante. Además, se tiene un gran número de partículas N que interactúan de tal forma que exista transferencia de energía entre ellas y que no impidan el tratamiento de las partículas de forma independiente. En un determinado instante de tiempo, la energía del sistema se distribuye entre las partículas formando estados de energía. Esta distribución indica cuántas partículas hay en cada estado de energía, por lo que representa el macroestado. El número de partículas en un estado E_i se le denomina *partición* n_i . El número de partículas y la energía total del sistema se definen como

$$N = \sum_i n_i, \quad U = \sum_i n_i E_i, \quad (3.1)$$

respectivamente. Una expresión que indica el número total de microestados en un macroestado dado, o bien, el número total de formas diferentes en las que se puede obtener todas las particiones es

$$\Omega_n = \frac{N!}{\prod_i n_i!}, \quad (3.2)$$

la cual proporciona información respecto a la partición que tiene mayor probabilidad.

En un contexto estadístico es importante señalar que una vez encontrada la partición con mayor probabilidad de aparecer que el resto de las particiones, se dice que el sistema ha logrado un equilibrio estadístico, lo cual indica que el sistema no cambiará la partición más probable a menos que un factor externo lo perturbe. El reto ahora es encontrar una ley de distribución del sistema aislado que permita identificar la probabilidad de ocurrencia de cada una de las particiones y así obtener propiedades macroscópicas o colectivas observables a partir de un nivel microscópico. Una ley de distribución que se adecúa a nuestro sistema es la distribución de Maxwell-Boltzmann que se define en la siguiente sección.

3.2 DISTRIBUCIÓN DE MAXWELL-BOLTZMANN

Para determinar cuándo la ecuación (3.2) alcanza el valor máximo es necesario primero convertir las variables discretas n_i en variables continuas x_i . Además, tiene términos factoriales por lo que necesitamos otra expresión que se pueda derivar, una posible es la aproximación conocida como la *fórmula de Stirling*

$$\ln(n!) \approx n \ln(n) - n, \quad (3.3)$$

cuyo error absoluto puede ser despreciable cuando n es muy grande. Definiendo una función en términos de \vec{x} de la siguiente forma $\Omega_x = \frac{N!}{\prod_i x_i!}$ y usando la aproximación de la ecuación (3.3), se tiene que

$$\ln(\Omega_x) = N \ln(N) - N - \sum_i (x_i \ln(x_i) - x_i), \quad (3.4)$$

con $x_i \geq 0$ y las condiciones $N = \sum_i n_i$, $U = \sum_i n_i E_i$. Se puede conocer el máximo del modelo anterior usando la relajación lagrangiana y la ecuación (2.10). Se asignan dos multiplicadores no negativos, α y β , a las condiciones del problema; por lo que se tiene la relajación

$$\mathcal{L}_\Omega = \ln(\Omega_x) - \alpha \sum_i x_i - \beta \sum_i x_i E_i.$$

Usando la ecuación (2.10) se tiene

$$-\ln(x_i) - \alpha - \beta E_i = 0,$$

entonces el valor máximo de Ω_x se obtiene al calcular los valores de x_i con la ecuación

$$x_i = e^{-\alpha - \beta E_i} = A e^{-\beta E_i}.$$

Dado que $n_i \approx x_i$, entonces $N = A \sum_i e^{-\beta E_i}$. Se define la probabilidad de que la partición n_i sea seleccionada del total de particiones es

$$\frac{n_i}{N} = \frac{e^{-\beta E_i}}{\sum_i e^{-\beta E_i}},$$

la cual denotaremos por p_i y $\sum_i e^{-\beta E_i}$ es conocida como la *función de partición*, la cual se representa por \mathcal{Z} . De ella se pueden conocer la energía libre, temperatura, entropía, entre otras cosas. En física estadística, p_i indica la probabilidad de que el sistema pueda ocupar un microestado i ,

$$p_i = \frac{1}{\mathcal{Z}} e^{-\beta E_i}.$$

Ahora bien, dado que la suma de todas probabilidades debe ser uno, se llega a

$$\sum_i p_i = \frac{1}{\mathcal{Z}} \sum_i e^{-\beta E_i}, \quad (3.5)$$

donde \mathcal{Z} es una constante de normalización, $\beta = \frac{1}{\varphi T}$ con φ la constante de Boltzmann que es igual a $1.3807 \times 10^{-29} J/K$ y T es la temperatura absoluta del sistema (la temperatura más baja es cero, se le denomina cero absoluto).

Para fines prácticos y reemplazando las sumatorias por integrales, la ecuación (3.5) se denota por

$$P(\vec{x}) = \frac{1}{\mathcal{Z}} e^{\left(\frac{-V(\vec{x})}{\tau}\right)}, \quad (3.6)$$

en la cual $V(\vec{x})$ representa una función que contiene la función objetivo y restricciones de un problema de optimización.

CAPÍTULO 4

METODOLOGÍA

Un problema de optimización (véase la sección 2.1) puede representarse con una función $V(\vec{x})$ en términos de la función objetivo y sus restricciones, empleando una relajación lagrangiana. Además, se le puede asociar una función de distribución de probabilidad como la ecuación (3.6), en la que se supone que el sistema se encuentra en equilibrio térmico y por simplicidad de notación, τ y \mathcal{Z} toman el valor de uno.

En el presente trabajo de investigación se estima una aproximación de la función de distribución de probabilidad de la ecuación (3.6) para problemas de optimización binarios usando una versión de campo medio llamada campo medio variacional, la cual permite definir una ecuación que calcula la probabilidad de ocurrencia de las variables de decisión y así poder construir una solución por medio del redondeo. Por otro lado, se describen los problemas binarios utilizados para validar la eficiencia del algoritmo; nombrados problema de la mochila, problema de la mochila cuadrática y problema de la mochila multidimensional

4.1 CAMPO MEDIO VARIACIONAL

Campo medio variacional aproxima una función de distribución intratable P , ver ecuación (3.6), por una función de distribución tratable Q [37]. La distribución P

generalmente es una función intratable debido a la complejidad de su estructura y la distribución Q pertenece a una familia de funciones mucho más sencillas. Q es elegida de tal forma que una cierta medida de distancia entre P y Q es minimizada, denominada en la literatura como la divergencia de Kullback-Leibler o entropía relativa [25].

Para el desarrollo de campo medio variacional se asume que todas las variables x_i son independientes, por lo que una distribución Q puede representarse como el producto de sus distribuciones marginales; esto es

$$Q(\vec{x}) = \prod_i q_i(x_i).$$

Una forma en las que se pueden representar las funciones marginales para un problema de optimización binario es

$$q_i(x_i) = 1 + (2m_i - 1)x_i - m_i,$$

en las que cada $m_i \in [0, 1]$ denota una variable continua de campo medio que denota la probabilidad de ocurrencia de x_i bajo la distribución Q . Para encontrar la diferencia mínima entre P y Q se usa la divergencia de *Kullback-Leibler* expresada como

$$D_{KL}(Q||P) = \langle \ln Q(\vec{x}) \rangle - \langle \ln P(\vec{x}) \rangle,$$

en la que $\langle \cdot \rangle$ representa la media aritmética de cada una de las dos distribuciones. Por otra parte, es necesario introducir la entropía S_Q bajo la distribución Q ,

$$S_Q = -\tau \langle \ln Q(\vec{x}) \rangle,$$

la cual está relacionada con el número de microestados del sistema. Además, se define el problema variacional de mínima energía o energía libre F_Q ,

$$F_Q = \frac{1}{\tau} [\langle V(\vec{x}) \rangle - S_Q]. \quad (4.1)$$

Una expresión para $\langle V(\vec{x}) \rangle$ puede construirse en términos de la relajación Lagrangiana del problema de la ecuación (2.4),

$$\langle V(\vec{x}) \rangle = \mathcal{L}(\vec{x}) = f(\vec{x}) + \sum_{r \in R} \mu_r g_r(\vec{x}),$$

donde μ contiene valores no negativos, los cuales representan los multiplicadores de Lagrange asociados a cada restricción.

La relajación continua del problema en la ecuación (2.4) es equivalente a sus promedios bajo la distribución Q , esto es,

$$\begin{aligned} \min \quad & f(\vec{m}) \\ \text{sujeto a: } \quad & g_r(\vec{m}) \geq 0, \quad r \in R; \\ & \vec{m} \in [0, 1]^N. \end{aligned} \tag{4.2}$$

Por otro lado, la entropía también puede representarse para el problema variacional como $S_Q = -\sum_{i=1}^N p_i \ln p_i$, de modo que

$$S_Q = -\sum_{i=1}^N [(1 - m_i) \ln(1 - m_i) + m_i \ln m_i]. \tag{4.3}$$

Es evidente que el problema variacional de la ecuación (4.1) en términos del vector \vec{m} tiene la forma

$$\begin{aligned} \min \quad & F_Q = \left\{ f(\vec{m}) + \sum_{r \in R} \mu_r g_r(\vec{m}) \right. \\ & \left. + \sum_{i \in N} [(1 - m_i) \ln(1 - m_i) + m_i \ln m_i] \right\}. \end{aligned} \tag{4.4}$$

Por último, como la ecuación (4.4) es un problema de optimización a minimizar debe satisfacer la condición de la ecuación (2.10) de KKT, en consecuencia el problema variacional se reduce a un conjunto de ecuaciones independientes de la forma

$$m_i = (1 + \exp[\partial_i \mathcal{L}(\vec{m})])^{-1}. \tag{4.5}$$

4.2 CASOS DE PRUEBA

En la presente sección se da una breve explicación de los problemas que fueron seleccionados, en base a su fácil interpretación y múltiples aplicaciones que pueden

surgir en la vida real, en diferentes áreas del conocimiento. Por otro lado fueron seleccionados por su gran complejidad lo cual permite evaluar el desempeño computacional de la metodología propuesta. Principalmente se muestra la descripción y modelación matemática del problema, e inclusive algunas de sus aplicaciones reales.

4.2.1 PROBLEMA DE LA MOCHILA

El problema de la mochila (KP, siglas en inglés de *knapsack problem*) es un problema muy conocido por la comunidad de investigación de operaciones, cuya función objetivo y restricción son lineales. Se sabe que es un problema que entra en la clasificación de los problemas NP-completos [39]. Se define de la siguiente manera: dados N artículos con un peso w_i y un beneficio b_i asociados a cada artículo, ambos con valores no negativos. Además, se tiene un límite de capacidad c , es decir, la mochila puede resistir cierta cantidad de peso. La finalidad del problema es encontrar aquellos artículos que ofrezcan el mayor beneficio sin exceder la capacidad de la mochila [32, 45]. Existen muchos problemas de decisión con una sola restricción en los que puede aplicarse este modelo, en áreas como economía, ingeniería y negocios [6]. El modelo matemático se representa como sigue:

$$\begin{aligned} \max \quad & \sum_{i \in N} b_i x_i & (4.6) \\ \text{sujeto a:} \quad & \sum_{i \in N} w_i x_i \leq c; \\ & \vec{x} \in \{0, 1\}^N. \end{aligned}$$

4.2.2 PROBLEMA DE LA MOCHILA CUADRÁTICA

El problema de la mochila cuadrática (QKP, siglas en inglés de *quadratic knapsack problem*) es una extensión del KP: se tiene un doble beneficio, esto es, el beneficio de incorporar un artículo a la mochila y el beneficio de incorporar dos artículos

al mismo tiempo. Lo anterior hace que la función objetivo sea cuadrática, lo cual incrementa su complejidad. Este problema se clasifica como un problema NP-difícil [44, 50]. Algunas de sus aplicaciones se presentan en telecomunicaciones, problemas de ubicación, problemas de clique máximo, entre otros [18, 40, 17, 49]. El QKP se define de la siguiente manera: dado N artículos con un peso w_i asociado a cada artículo y una matriz cuadrada B de beneficios, ambos con valores no negativos. El objetivo es seleccionar aquellos artículos que brinden el mayor beneficio sin exceder una capacidad máxima c . Cabe mencionar que B es una matriz simétrica, es decir, $b_{ji} = b_{ij}$; b_{ii} es el beneficio de incorporar el artículo i a la mochila; $b_{ij} + b_{ji}$ es el beneficio conjunto de incorporar el artículo k y l a la mochila. El modelo matemático es

$$\begin{aligned} \max \quad & \sum_{i \in N} \sum_{j \in N} b_{ij} x_i x_j & (4.7) \\ \text{sujeto a:} \quad & \sum_{i \in N} w_i x_i \leq c; \\ & \vec{x} \in \{0, 1\}^N. \end{aligned}$$

4.2.3 PROBLEMA DE LA MOCHILA MULTIDIMENSIONAL

El problema de la mochila multidimensional (MKP, siglas en inglés de *multidimensional knapsack problem*) es un problema de optimización muy estudiado por la comunidad de investigación de operaciones, debido a que es un problema NP-difícil [20] y es un gran desafío resolverlo conforme incrementa el número de sus restricciones. Cabe mencionar que este problema es una generalización del KP cuando sólo se tiene una restricción. Algunas de sus aplicaciones son en modelos de evaluación de proyectos, presupuestación de capital, problemas de carga, políticas de inversión para el sector turístico, entre muchas otras [33, 42, 46]. El objetivo, al igual que en los tres problemas anteriores, es encontrar la mejor configuración de selección de artículos, dado un conjunto de artículos N , de tal forma que el beneficio sea máximo sin exceder la capacidad de las mochilas. Dado un conjunto de restricciones R , se definen un

beneficio b_i , un peso w_{ri} correspondiente a la restricción r y al artículo i . Además de un peso máximo c_r para cada restricción [2, 20, 53]. El modelo matemático es

$$\begin{aligned} \max \quad & \sum_{i \in N} b_i x_i \\ \text{sujeto a: } & \sum_{i \in N} w_{ri} x_i \leq c_r, \quad r \in R; \\ & \vec{x} \in \{0, 1\}^N. \end{aligned} \tag{4.8}$$

4.3 ECUACIONES DE CAMPO MEDIO PARA LOS CASOS DE PRUEBA

Para cada uno de los problemas anteriores se puede encontrar una expresión que permita encontrar, de forma iterativa, el mejor valor esperado de cada una de las variables bajo la distribución Q de campo medio.

A continuación se muestran las expresiones que permiten calcular los valores esperados de cada una de las variables x_i , para los tres casos de estudio en los que $i \in N$.

- Para el KP:

$$m_i = [1 + \exp[-b_i + \mu w_i]]^{-1}. \tag{4.9}$$

- Para el QKP:

$$m_i = \left[1 + \exp[-2b_{ii}m_i^* - \sum_{j \in N, j \neq i} b_{ij}m_j^* + \mu w_i] \right]^{-1}. \tag{4.10}$$

- Para el MKP:

$$m_i = \left[1 + \exp[-b_i + \sum_{j \in N} \mu_j w_{ji}] \right]^{-1}. \tag{4.11}$$

Las ecuaciones anteriores (4.9)–(4.11) tienen dos valores constantes (beneficio y peso) que se extraen del caso prueba en cuestión. Todas las expresiones dependen de

un parámetro μ (conjunto de parámetros para el MKP) que es necesario estimar. Observe que si $\mu = 0$ (en el caso del MKP todos cero), se genera un vector \vec{m} con todos sus elementos cercanos a uno; ahora bien, si μ es grande (en el caso del MKP todos uno) tendrá elementos parecidos al cero. Además, la ecuación (4.10) tiene elementos de un vector solución previo \vec{m}^* al que se está calculando, es decir, es necesario conocer una solución anterior para poder calcular una actual. La solución precedente puede ser un vector con valores aleatorios en el intervalo $[0, 1]$, o bien, un vector constante.

CAPÍTULO 5

EXPERIMENTOS COMPUTACIONALES

En este capítulo se presentan de forma detallada las características del equipo de cómputo y software utilizado para realizar las pruebas computacionales. Se describen las propiedades de cada conjunto de casos de prueba para cada uno de los problemas. Asimismo, se exhiben los algoritmos implementados para lograr los resultados que se muestran en la sección 5.3.

5.1 ALGORITMOS DE CAMPO MEDIO PARA LOS CASOS DE PRUEBA

Los algoritmos que se muestran en esta sección logran contruir una solución factible en cada uno de los casos de prueba. Recordando que la ecuación (4.5) de campo medio, entrega valores continuos entre 0 y 1. Dicha ecuación genera un vector \vec{m} con la probabilidad de ocurrencia de cada una de las variables del problema binario, así que una solución puede obtenerse redondeando los valores \vec{m} . Sin embargo, las ecuaciones que se presentan en la sección 4.3 dependen de los valores de los multiplicadores denotados por $\vec{\mu}$. Como también se menciona, valores de $\vec{\mu}$ muy cercanos a cero crearán un vector \vec{m} con valores cercanos a uno; conforme crece $\vec{\mu}$, \vec{m} tenderá a convertirse en un vector de ceros.

Una buena estrategia para construir la mejor solución \vec{m} de campo medio en los algoritmos (2) y (3) es usar una técnica de bisección, la cual encuentra el mejor valor del multiplicador que, en términos de física estadística, maximiza la entropía y minimiza la energía libre. Dados dos multiplicadores de los cuales uno genera una solución infactible y el otro una solución factible, la idea de la bisección es dividir un intervalo en dos partes iguales descartando la subregión que produzca malas soluciones. Un multiplicador comienza en cero y el otro es un valor no muy grande, puesto que si es muy grande se realizarían más cortes del intervalo lo cual provocaría un incremento en el tiempo para finalizar el algoritmo. Ahora bien, si los dos multiplicadores iniciales producen soluciones infactibles se implementa una estrategia para incrementar el valor del segundo multiplicador, hasta lograr encontrar uno que ofrezca una solución factible, evidentemente también cambiará el valor del primer multiplicador conforme incrementa el segundo, éste se quedará con su valor anterior. Ambos algoritmos finalizan hasta que la diferencia entre ambos multiplicadores sea pequeña. Una gran ventaja de estos dos algoritmos es que hasta finalizar se evalúa el beneficio obtenido dado que únicamente se trabaja con el dual de campo medio, por lo que el mejor valor encontrado de μ construirá la mejor solución de campo medio para el problema binario usando. Cabe mencionar que el dual encontrado es el mejor para el problema de mínima energía y no para el de la relajación Lagrangiana. La única diferencia entre estos dos algoritmos es que el de la ecuación (3) requiere una solución continua inicial que es fijada con valores de uno.

El algoritmo (4) está diseñado para el problema de la mochila multidimensional en el que se tienen r restricciones, por lo tanto hay que calcular r multiplicadores de Lagrange. Para este caso no es tan sencillo encontrar los mejores valores de $\vec{\mu}$ que minimicen la entropía libre del sistema. Una simple bisección ya no es del todo útil, al igual genera soluciones factibles en poco tiempo pero la calidad disminuye. Por esta razón se implementa la siguiente estrategia de incremento en los multiplicadores. Comenzando con todos sus valores en cero, se actualizan los multiplicadores cuya restricción ha sido violada con un incremento de θ veces el valor de evaluar la solución

en la restricción. El algoritmo finaliza hasta que todas las restricciones son satisfechas.

Algoritmo 2 (MF para KP)

Entrada: $\mu_s \leftarrow 0; \mu_f \leftarrow 1; \epsilon \leftarrow \infty; \epsilon_m \leftarrow 0.01$.

Calcular \vec{m} usando ecuación (4.9) con μ_f ;

$\vec{x} \leftarrow \text{redondear}(\vec{m})$;

Calcular factibilidad de \vec{x} ;

mientras \vec{x} sea *infactible* **hacer**

$\mu_s \leftarrow \mu_f$;

$\mu_f \leftarrow 3 * \mu_f$;

 Calcular \vec{m} usando ecuación (4.9) con μ_f ;

$\vec{x} \leftarrow \text{redondear}(\vec{m})$;

fin

mientras $(\epsilon > \epsilon_m)$ **hacer**

$\mu \leftarrow (\mu_s + \mu_f) / 2$;

 Calcular \vec{m} usando ecuación (4.9) con μ ;

$\vec{x} \leftarrow \text{redondear}(\vec{m})$;

 Calcular factibilidad de \vec{x} ;

si \vec{x} es *infactible* **entonces**

$\mu_s \leftarrow \mu$

en otro caso

$\mu_f \leftarrow \mu$;

$\vec{x}^* \leftarrow \vec{x}$;

fin

$\epsilon \leftarrow \mu_f - \mu_s$;

fin

devolver $(\vec{x}^*, \text{EvaluaBeneficio}(\vec{x}^*))$.

Algoritmo 3 (MF para QKP)

Entrada: $\mu_s \leftarrow 0; \mu_f \leftarrow 1; \epsilon \leftarrow \infty; \epsilon_m \leftarrow 0.01; \vec{m}^* \leftarrow \vec{1}$.Calcular \vec{m} usando ecuación (4.10) con \vec{m}^* y μ_f ; $\vec{x} \leftarrow \text{redondear}(\vec{m});$ Calcular factibilidad de \vec{x} ;**mientras** \vec{x} sea *infactible* **hacer** $\mu_s \leftarrow \mu_f;$ $\mu_f \leftarrow 3 * \mu_f;$ Calcular \vec{m} usando ecuación (4.10) con \vec{m}^* y μ_f ; $\vec{x} \leftarrow \text{redondear}(\vec{m});$ **fin****mientras** $(\epsilon > \epsilon_m)$ **hacer** $\mu \leftarrow (\mu_s + \mu_f)/2;$ Calcular \vec{m} usando ecuación (4.10) con \vec{m}^* y μ ; $\vec{x} \leftarrow \text{redondear}(\vec{m});$ Calcular factibilidad de \vec{x} ; **si** \vec{x} es *infactible* **entonces** $\mu_s \leftarrow \mu$ **en otro caso** $\mu_f \leftarrow \mu;$ $\vec{x}^* \leftarrow \vec{x};$ **fin** $\epsilon \leftarrow \mu_f - \mu_s;$ **fin****devolver** $(\vec{x}^*, \text{EvaluaBeneficio}(\vec{x}^*)).$

Algoritmo 4 (MF para MKP)

Entrada: $\vec{\mu} \leftarrow \vec{0}$, $\theta \leftarrow 1/10^7$.**repetir** Generar \vec{m} con ecuación (4.11) usando $\vec{\mu}$ $\vec{x} \leftarrow \text{redondear}(\vec{m})$; $Rvi \leftarrow 0$; **mientras** $r \in R$ **hacer** **si** $\sum_{j \in N} w_{rj} m_j - c_r > 0$ **entonces** $\mu_r \leftarrow \mu_r + \theta * (\sum_{j \in N} w_{rj} m_j - c_r)$; $Rvi \leftarrow Rvi + 1$; **fin** **fin****hasta que** $Rvi = 0$;**devolver** $(\vec{x}, \text{EvaluaBeneficio}(\vec{x}))$.

5.2 IMPLEMENTACIÓN

Los experimentos computacionales se realizaron en una Workstation HP con un procesador Intel Xeon(R) E3-1240v3@3.40GHz de ocho núcleos, memoria RAM de 15.5 GB y un sistema operativo Ubuntu 14.04 LTS de 64 bits. Los algoritmos se implementaron en el lenguaje de cómputo estadístico R¹. A continuación se describen cada uno de los conjuntos de prueba utilizados en los cuatro problemas seleccionados. a continuación:

- Para el KP se generaron instancias con el generador de Pisinger [43] bajo las condiciones de una fuerte correlación lineal entre b_i y w_i , siendo w_i valores aleatorios uniformes en el intervalo $[1, R]$ y $b_i = w_i + R/10$. Se crean diez casos de tamaños $10^2, 5 \times 10^2, 10^3, 5 \times 10^3, 10^4, 5 \times 10^4, 10^5, 5 \times 10^5, 10^6, 5 \times 10^6, 10^7, 5 \times 10^7$, con valores de R de $10^2, 10^3, 10^4, 10^5, 10^6, 10^7$.

¹<http://cran.r-project.org/>

- Para el QKP se dividen tres clases de casos sintéticos reportados en la literatura con cuatro diferentes densidades (25%, 50%, 75% y 100%). La densidad indica el porcentaje de conectividad de un objeto con el resto de los objetos. La primera clase [7] con valores de N igual a 100, 200 y 300. La segunda clase contiene casos creados con generador [11], los valores de N son 1000, 2000. La tercer clase [13] contiene casos con valores para N de 5000 y 6000.
- Para el MKP se usan 270 casos diferentes [15] que se clasifican de acuerdo a tres tamaños diferentes (100, 250 y 500), tres cantidades para el número de restricciones (5, 10, 30) y tres tipos de correlación (25, 50 y 75); en cada combinación se crean 10 casos diferentes.

5.3 RESULTADOS

Antes de comenzar a describir los resultados obtenidos, se introduce la diferencia porcentual DP como una medida de calidad de solución, se define como el valor porcentual del cociente de la diferencia del mejor valor conocido y el valor obtenido, dividido entre el mejor valor conocido.

Casos de gran tamaño se pueden resolver de manera exacta y eficiente para el KP, inclusive en problemas con valores de R moderados. En valores altos de R , los algoritmos exactos tienen mayor dificultad para resolver casos de grandes dimensiones [43].

Para validar la eficiencia de nuestro algoritmo en KP se utiliza el método de ramificación y acotamiento de CPLEX (al cual denotamos BB, finaliza cuando una solución factible es encontrada). En la figura 5.1 se puede apreciar el crecimiento exponencial del tiempo de ejecución de BB. Además, dicho método no puede encontrar una solución factible en problemas con valores de N superiores a 10^6 con R igual a 10^3 , valores superiores a 5×10^5 en N con R igual a 10^4 o 10^5 , y N mayores a 5×10^4 con R iguales a 10^6 o 10^7 . A diferencia, CM encuentra una solución factible a un

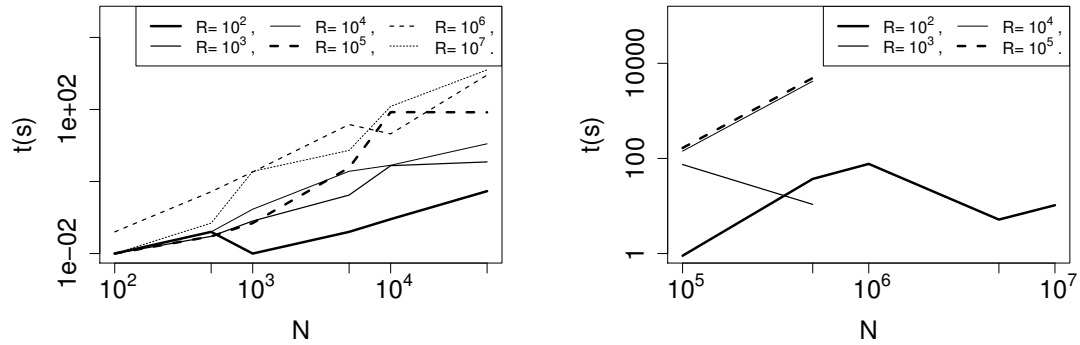


Figura 5.1: Tiempo promedio obtenido por BB para el KP.

costo computacional muy pequeño para cualquier caso, por lo que es una estrategia robusta con respecto a R , ver figura 5.2. Heurísticas del estado del arte para este problema utilizan tamaños de hasta 10^5 y valores de R no mayores a 2000 [5, 10, 27].

En la figura 5.3 se puede observar el comportamiento en cuanto a calidad de solución se refiere ya sea por grupos de R o grupos de N . En ambos casos se puede observar un patrón, la DP para cada grupo de instancias es poco variable y no tan alejado del cero, excepto en R y N con valores de 10^2 , los cuales son los valores atípicos que aparecen en la gráfica; por ende CM funciona muy bien en valores grandes de N y R . Dichos resultados pueden observarse también en los cuadros A.1 y A.2, de las que podemos también saber el costo-beneficio (CB) de usar las dos metodologías. Para calcular el CB se divide el mejor valor encontrado entre el tiempo, es evidente que mientras el CB de una estrategia sea mucho mayor que el de otra indica ser mejor estrategia. En el caso del KP, se puede apreciar en la figura 5.4 que CM es muy superior a BB ya que ofrece soluciones de alta calidad en mucho menor tiempo.

En el caso del QKP el lograr la solución óptima es muy difícil hasta para casos de cientos de variables [7, 19]. Los resultados obtenidos por CM para los casos de tamaño 100, 200 y 300 se pueden observar en el cuadro A.3 y se resume su comportamiento en cuestión de calidad de la solución y tiempo de ejecución en la figura 5.5. El algoritmo propuesto entrega soluciones muy buenas en un tiempo muy

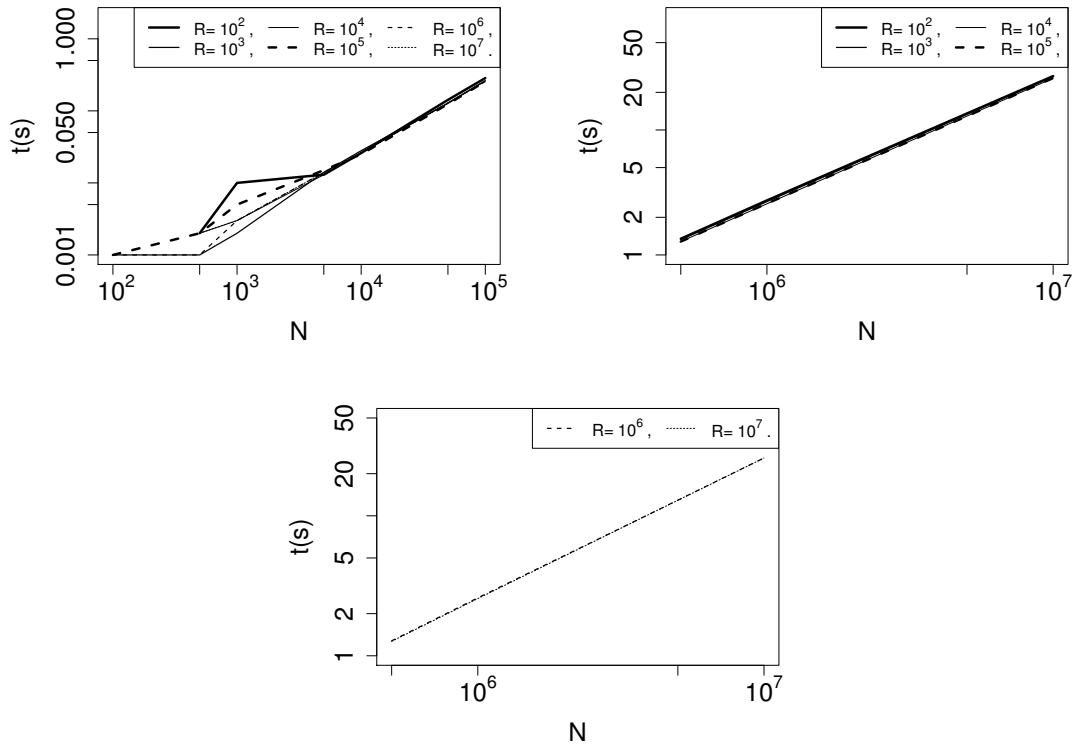


Figura 5.2: Tiempo promedio obtenido por CM para el KP.

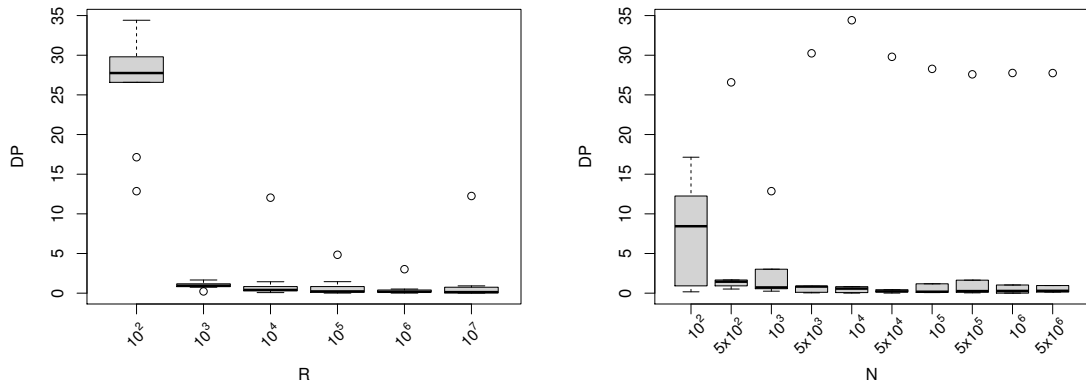


Figura 5.3: Diagramas de cajas para el KP.

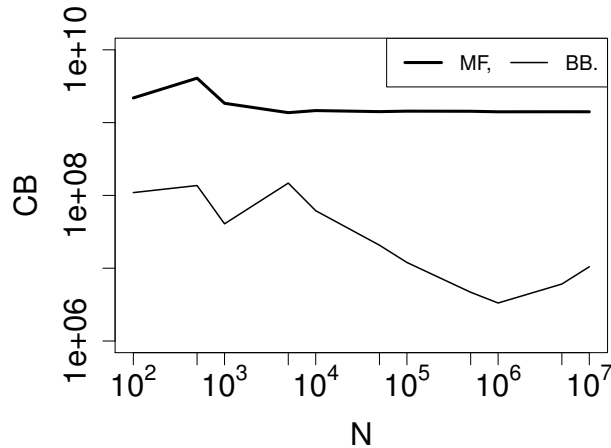


Figura 5.4: Comparativo del CB entre CM y BB para el KP con $R=10^6$.

pequeño, inclusive en casos en los que existe poca conectividad y que es sabido son más difíciles de resolver. En promedio se tiene una DP inferior al 3.5% en todos los tamaños del QKP.

Para evaluar el rendimiento de CM se usan casos más grandes de los cuales se desconoce el valor óptimo: 1000 y 2000 propuestos por [52]; 5000 y 6000 generados por [13]. Los primeros casos mencionados han sido utilizados para validar la eficiencia de algoritmos heurísticos como programación dinámica (DP+FE) [19], GRASP con búsqueda Tabú (GRASP+Tabu) [52], exploración de hiperplano iterado (IHEA) [13]. Los resultados de estos algoritmos y el propuesto se muestran en los cuadros A.4 y A.5. Los casos de 5000 y 6000 fueron creados para validar la eficiencia del IHEA, en los cuadros A.6 y A.7 se muestran los tiempos promedio de ejecución y el mejor valor encontrado por este algoritmo. También en estos cuadros se presentan los resultados obtenidos por el algoritmo propuesto. Como se desconoce el valor óptimo para los casos grandes, se toma como referencia el mejor valor reportado por IHEA para saber qué tan buena es la solución de CM y obtener una DP.

A manera de resumen se puede observar en las figura 5.6 que se tiene un comportamiento similar a los casos pequeños, CM brinda soluciones competitivas a un costo computacional muy pequeño, las DPs entre casos con las mismas características

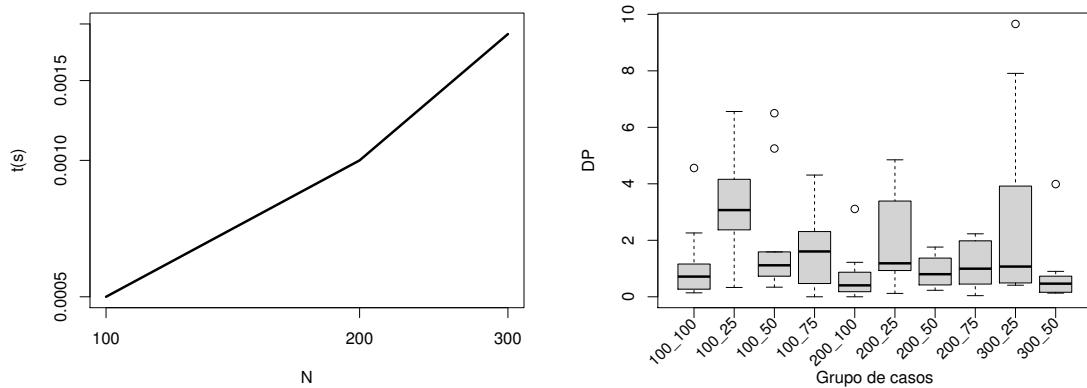


Figura 5.5: Tiempos promedio y diagramas de cajas de las DPs para el QKP con tamaños de 100, 200 y 300.

son más cerrados cuando existe una alta conectividad entre las variables y en el caso contrario los grupos son más amplios. Cabe mencionar que en promedio las DPs suelen disminuir conforme crece el tamaño de los casos: en el caso de 1000 se tiene una DP de 0.91%, en 2000 una DP de 0.26%, en 5000 una DP de 0.13% y en 6000 una DP de 0.09%.

Finalmente, a continuación se muestra el comportamiento de CM en un problema en el que ya no es tan sencillo construir una solución de alta calidad porque en el MKP se tiene un conjunto de restricciones a satisfacer, por lo que trabajar en el espacio dual (como en los casos anteriores en los que se encontraba el mejor multiplicador) es más difícil puesto que hay que encontrar más de un multiplicador que minimice la función de energía libre. Los resultados que se obtienen del algoritmo propuesto de campo medio se muestran en los cuadros A.8, A.9, A.10, A.11 y A.12. Para este problema en ciertos casos se conoce el valor óptimo y en otros se tiene el mejor valor conocido hasta el momento, por lo que con ellos se calcula una DP para validar el rendimiento de CM. En la figura 5.9 se puede apreciar que al parecer, al menos para este grupo de casos prueba, es más complicado el estimar los multiplicadores de CM cuando el número de variables es pequeño puesto que los tiempos son menores que las más grandes. En cuánto a calidad de solución se puede observar que como era de

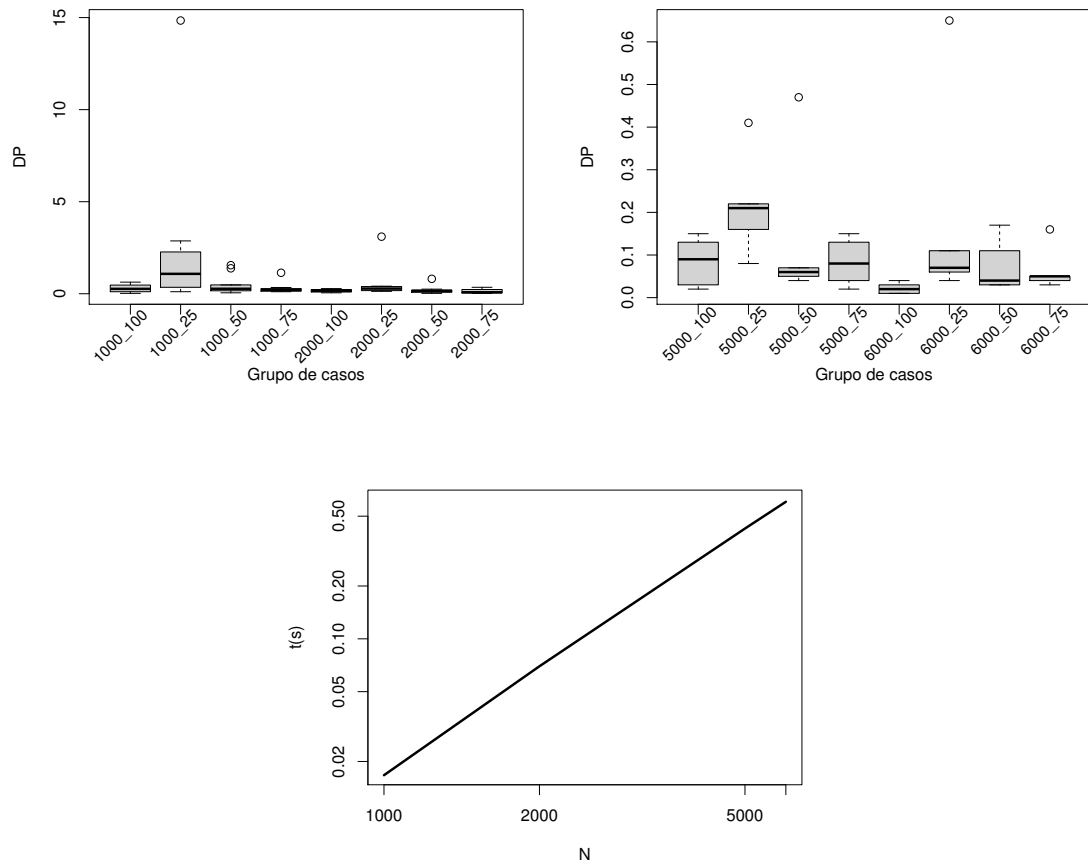


Figura 5.6: Tiempos promedio y diagramas de cajas de las DPs para el QKP con tamaños de 1000, 2000, 5000 y 6000.

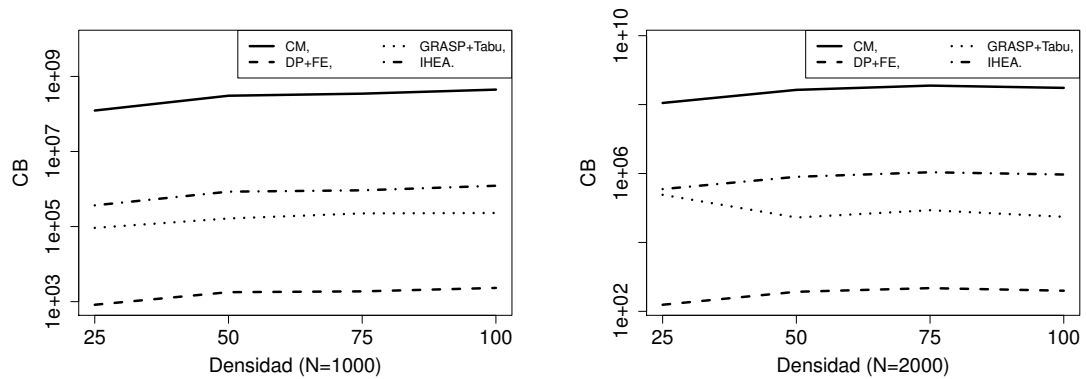


Figura 5.7: Comparativo del CB de cuatro heurísticos para el QKP en tamaños de 1000 y 2000.

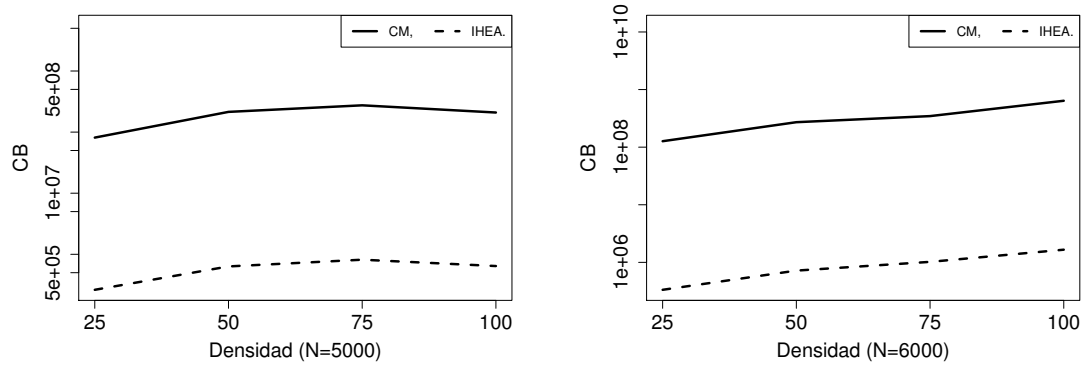


Figura 5.8: Comparativo del CB de dos heurísticos para el QKP en tamaños de 5000 y 6000.

esperarse, en los grupos de instancias con más restricciones se obtiene soluciones de menor calidad que el resto, aunque la diferencia no es muy grande. Sin embargo, en la figura 5.10 se puede apreciar que conforme incrementa el número de variables la DP disminuye. Se sabe que los casos fuertemente correlacionados son más difíciles de resolver [22], sin embargo, en la figura anterior se puede observar que CM tiene un comportamiento contrario, es decir, CM es eficiente en problemas con alta correlación.

Recientemente este conjunto de casos para el MKP han sido usados en el trabajo [28], cuyo algoritmo es nombrado TPTEA y usado para hacer notar el costo beneficio de usar dicha estrategia y CM. En la figura 5.11 se resume que conforme crece el número de variables el CB es mucho menor para el TPTEA, por lo que CM entrega soluciones de buena calidad en un tiempo mucho menor.

Al igual que en el problema anterior se observa que la tendencia es disminuir la DP promedio conforme crece el tamaño de los casos. En tamaños de 100 se tiene una DP promedio de 6.09%, para los de 250 una DP promedio de 4.15% y para los de 500 una DP promedio de 2.85%.

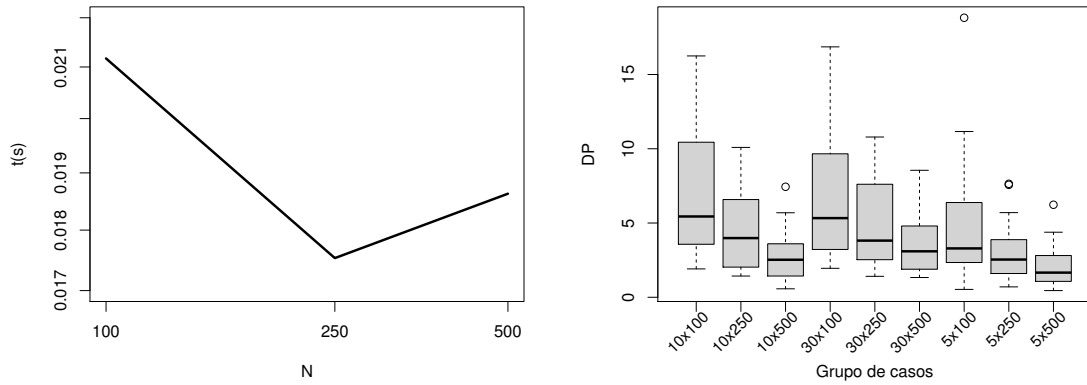


Figura 5.9: Tiempos promedio y diagramas de cajas de las DPs para el MKP con tamaños de 100, 250 y 500.

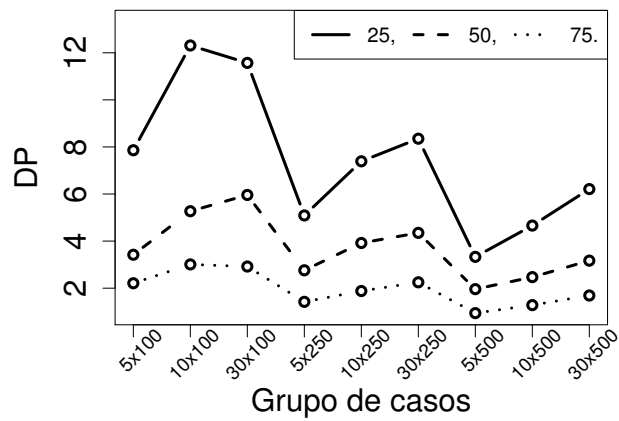


Figura 5.10: DP promedio de grupo de casos separados por tipos de correlación.

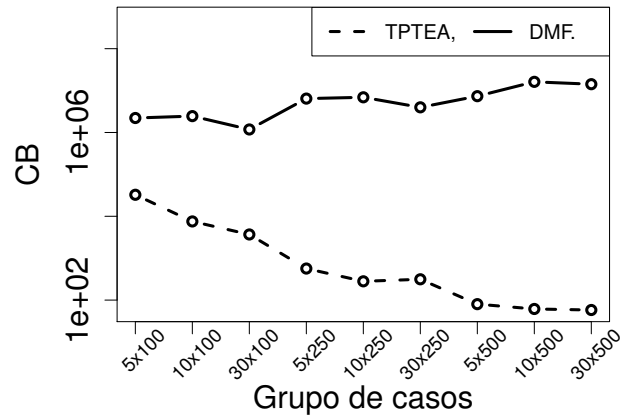


Figura 5.11: Comparativo del CB de dos heurísticos para el MKP.

5.4 DISCUSIÓN

A pesar de que la metodología propuesta de campo medio depende de una aproximación a la función de distribución de Boltzmann y que es combinada con una búsqueda en el espacio dual que toma en cuenta el efecto de las restricciones, se obtienen soluciones prometedoras para el problema binario a partir de un vector solución continuo que representa la probabilidad de ocurrencia de cada una de las variables de decisión.

CM es un método de búsqueda completamente determinista que a diferencia de los llamados algoritmos de estimación de distribución (EDAs, siglas en inglés de *estimation of distribution algorithms*), no depende de una población probabilística para construir un modelo probabilístico, característica que hace muy lentos a los EDAs.

El método propuesto, en lugar de iniciar el proceso de búsqueda desde una temperatura lo suficientemente alta, como lo hacen en otras versiones de campo medio, usa una temperatura lo suficientemente pequeña y fija.

Cabe mencionar que la exploración del espacio de solución se realiza a través

de las variables duales de problema de mínima energía, por lo que los mejores valores de los multiplicadores encontrados no son los mismos para la relajación Lagrangiana del problema de optimización. Tampoco, el modelo probabilístico de campo medio indica ser la mejor distribución de probabilidad de las variables. Recordemos que se estima una distribución promedio del sistema que en casos grandes la aproximación de campo medio se aproximará cada vez más a la real.

Una línea de investigación evidente abierta del presente trabajo es el diseño de nuevas heurísticas basadas en campo medio o la hibridación con algoritmos ya existentes.

Banda, Berrones y Velasco [4] presentan un algoritmo que incorpora a CM una búsqueda local simple para el problema de la mochila cuadrática. La búsqueda local comienza de la solución de CM, iterativamente genera un vecindario a partir de una mejor solución que las que se tiene y finaliza cuando no hay mejora. Para reducir el espacio de búsqueda se fijan aquellas variables que tienen una alta probabilidad de ser seleccionadas, al hacer esto se corre el riesgo de no tomar en cuenta elementos que puedan pertenecer a la solución óptima. Los resultados computacionales muestran que se mejoran las soluciones de CM, más no se logra alcanzar el valor óptimo (como era de esperarse dado que la búsqueda local implementada es muy básica) pero si se reduce significativamente el costo computacional.

CAPÍTULO 6

CONCLUSIONES

La metodología propuesta es altamente competitiva para los tres problemas de prueba con respecto a enfoques presentados en el estado del arte. A pesar de que es utilizada una aproximación de la distribución de Boltzmann, se obtienen resultados de alta calidad en muy poco tiempo. La tendencia es que, al menos hay evidencia para el QKP y el MKP, conforme crece el tamaño del caso las DPs se reducen de forma significativa. El algoritmo de CM tiene un excelente desempeño en la mayoría de los casos, inclusive en casos en los que se sabe tienen características en los que cualquier algoritmo tardaría mucho tiempo en resolverlos.

6.1 CONTRIBUCIONES

Se logran algoritmos deterministas que a diferencia de otros heurísticos, requieren de pocos parámetros a ajustar para problemas tipo mochila.

En los casos de prueba, la metodología propuesta construye buenas soluciones en poco tiempo y como era de esperarse en enfoques basados en promedios, la calidad de las soluciones es más precisas a medida que crece el número de variables.

Parte de la investigación ha sido presentada en un congreso nacional llamado *Congreso de la Sociedad Mexicana de Investigación de Operaciones* en el año 2015

y dos congresos internacionales: *Congreso Latino-Iberoamericano de Investigación Operativa* en el 2016 y *Congreso de Computación Evolutiva* en el 2017.

De este trabajo surgen dos artículos ya publicados [3] y [4], en revistas científicas de prestigio.

6.2 TRABAJO FUTURO

CM es un algoritmo que genera una distribución de probabilidad inicial que puede ser mejorada mediante alguna estrategia evolutiva, por lo que una interesante investigación sería comparar los resultados de CM con algún EDA conocido.

Otra forma de aprovechar las bondades del algoritmo de CM es usar sus soluciones como una solución inicial de un algoritmo exacto o heurístico, reduciendo el espacio de búsqueda y por ende el tiempo de cómputo.

Aplicar la teoría de campo medio en problemas enteros, no binarios, de los cuales se sabe son muy difíciles de resolver.

APÉNDICE A

RESULTADOS COMPUTACIONALES
DETALLADOS

Cuadro A.1: Resultados comparativos de CM y BB para el KP con valores de R desde 10^2 hasta 10^5 para los diferentes valores de N .

R	N	BB		CM		
		Óptimo	$t(s)$	V.sol	DP	$t(s)$
10^2	10^2	210	0.020	174	17.14	0.000
	5×10^2	380	0.020	279	26.58	0.002
	10^3	490	0.020	427	12.86	0.010
	5×10^3	1554	0.010	1084	30.24	0.013
	10^4	3101	0.010	2034	34.41	0.026
	5×10^4	15753	0.040	11058	29.80	0.142
	10^5	31681	0.080	22723	28.28	0.285
	5×10^5	159001	0.490	115139	27.59	1.348
	10^6	317912	1.080	229668	27.76	2.731
	5×10^6	1594216	6.020	1151839	27.75	13.558
10^7	3186502	12.280	2300004	27.82	27.003	
5×10^7	15937215	76.860	11516487	27.74	134.000	
10^3	10^2	2397	0.020	2375	0.92	0.001
	5×10^2	3798	0.090	3735	1.66	0.001
	10^3	5198	0.140	5159	0.75	0.002
	5×10^3	17402	0.100	17251	0.87	0.014
	10^4	35996	0.090	35702	0.82	0.028
	5×10^4	182139	0.050	181743	0.22	0.127
	10^5	358973	0.110	354740	1.18	0.262
	5×10^5	1801732	0.590	1772136	1.64	1.326
	10^6	3618023	1.280	3580799	1.03	2.671
	5×10^6	18050073	6.980	17876489	0.96	13.342
10^7	36120220	14.320	35858324	0.73	26.751	
5×10^7	180702786	101.890	179774006	0.51	133.252	
10^4	10^2	23982	0.020	21095	12.04	0.000
	5×10^2	37984	0.120	37437	1.44	0.002
	10^3	52990	0.070	52695	0.56	0.003
	5×10^3	176832	0.110	175370	0.83	0.013
	10^4	361885	0.200	361622	0.07	0.026
	5×10^4	1833844	0.100	1825889	0.43	0.129
	10^5	3661996	0.350	3655225	0.18	0.263
	5×10^5	18266392	0.660	18201824	0.35	1.280
	10^6	36623230	1.300	36512624	0.30	2.565
	5×10^6	182619555	7.370	181921074	0.38	12.928
10^7	365430250	15.880	364163642	0.35	25.981	
5×10^7	1827322539	121.990	1823805885	0.19	129.082	
10^5	10^2	199748	0.020	190084	4.84	0.001
	5×10^2	379971	0.090	374454	1.45	0.002
	10^3	529972	0.070	526250	0.70	0.005
	5×10^3	1853174	0.140	1837868	0.83	0.015
	10^4	3749745	0.220	3749292	0.01	0.025
	5×10^4	18261137	0.690	18218877	0.23	0.124
	10^5	36719765	1.000	36678173	0.11	0.256
	5×10^5	182645787	26.370	182433725	0.12	1.279
	10^6	364987982	28.910	364946856	0.01	2.577
	5×10^6	1827055310	37.500	1823275777	0.21	12.887
10^7	3655087059	47.410	3652255571	0.08	26.027	
5×10^7	-	-	18275722164	-	129.296	

Cuadro A.2: Resultados comparativos de CM y BB para el KP con valores de R desde 10^6 hasta 10^7 para los diferentes valores de N .

R	N	BB		CM			
		Óptimo	$t(s)$	V.sol	DP	$t(s)$	
10^6	10^2	2186550	0.020	2182815	0.17	0.001	
	5×10^2	4099065	0.030	4077720	0.52	0.001	
	10^3	5699204	0.140	5527137	3.02	0.003	
	5×10^3	19188239	0.130	19180096	0.04	0.014	
	10^4	38207899	0.620	38053456	0.40	0.026	
	5×10^4	182643485	8.820	182619790	0.01	0.129	
	10^5	367005239	30.610	366345913	0.18	0.255	
	5×10^5	1834431237	392.550	1831236761	0.17	1.279	
	10^6	3661246312	1100.610	3651622928	0.26	2.593	
	5×10^6	18266367437	3013.340	18240789763	0.14	12.931	
10^7	10^7	36559794204	3492.500	36530177104	0.08	25.916	
	5×10^7	-	-	182825574825	-	128.785	
	10^7	10^2	23907447	0.020	20979371	12.25	0.000
		5×10^2	40995443	0.040	40616721	0.92	0.002
		10^3	55992816	0.060	55852924	0.25	0.003
		5×10^3	191766632	0.230	191560961	0.11	0.014
		10^4	373993929	0.460	371207275	0.75	0.026
		5×10^4	1877021294	6.500	1874124082	0.15	0.128
		10^5	3697607061	23.330	3694880890	0.07	0.258
		5×10^5	18386289561	580.710	18379692940	0.04	1.268
10^6		36621578668	2591.540	36618938686	0.01	2.563	
5×10^6		-	-	182783379500	-	12.912	
10^7	-	-	364675427837	-	25.875		

Cuadro A.3: Resultados de CM para valores de $N \in \{100, 200, 300\}$.

d,l	Óptimo	$N=100$			d,l	Óptimo	$N=200$			d,l	Óptimo	$N=300$		
		V.sol	DP	$t(s)$			V.sol	DP	$t(s)$			V.sol	DP	$t(s)$
25_1	18558	17341	6.56	0.0003	25_1	204441	202122	1.13	0.001	25_1	29140	26835	7.91	0.003
25_2	56525	56339	0.33	0.0004	25_2	239573	239287	0.12	0.001	25_2	281990	278475	1.25	0.002
25_3	3752	3596	4.16	0.0005	25_3	245463	-	-	-	25_3	231075	-	-	-
25_4	50382	48785	3.17	0.0005	25_4	222361	220296	0.93	0.001	25_4	444759	442569	0.49	0.002
25_5	61494	60983	0.83	0.0004	25_5	187324	185000	1.24	0.001	25_5	14988	14401	3.92	0.002
25_6	36360	35162	3.29	0.0006	25_6	80351	77406	3.67	0.001	25_6	269782	268686	0.41	0.002
25_7	14657	14221	2.97	0.0004	25_7	59036	57884	1.95	0.001	25_7	485263	482847	0.5	0.002
25_8	20452	19317	5.55	0.0003	25_8	149433	147990	0.97	0.001	25_8	9343	8440	9.66	0.002
25_9	35438	34535	2.55	0.0005	25_9	49366	47693	3.39	0.001	25_9	250761	246879	1.55	0.002
25_10	24930	24340	2.37	0.0002	25_10	48459	46109	4.85	0.001	25_10	383377	379966	0.89	0.002
50_1	83742	82959	0.94	0.0003	50_1	372097	371231	0.23	0.001	50_1	513379	509643	0.73	0.002
50_2	104856	104389	0.45	0.0007	50_2	211130	209332	0.85	0.001	50_2	105543	104980	0.53	0.002
50_3	34006	32220	5.25	0.0004	50_3	227185	223330	1.7	0.001	50_3	875788	874538	0.14	0.002
50_4	105996	105640	0.34	0.0004	50_4	228572	224544	1.76	0.001	50_4	307124	304354	0.9	0.002
50_5	56464	56010	0.8	0.0008	50_5	479651	477627	0.42	0.001	50_5	727820	726850	0.13	0.002
50_6	16083	15038	6.5	0.0005	50_6	426777	424724	0.48	0.001	50_6	734053	732855	0.16	0.002
50_7	52819	51979	1.59	0.0008	50_7	220890	218224	1.21	0.001	50_7	43595	41854	3.99	0.002
50_8	54246	53850	0.73	0.0005	50_8	317952	316732	0.38	0.001	50_8	767977	764955	0.39	0.002
50_9	68974	68082	1.29	0.0003	50_9	104936	103494	1.37	0.001	50_9	761351	758314	0.4	0.002
50_10	88634	87347	1.45	0.0003	50_10	284751	282609	0.75	0.001	50_10	996070	988765	0.73	0.002
75_1	189137	189137	0	0.0004	75_1	442894	437929	1.12	0.001					
75_2	95074	93532	1.62	0.0006	75_2	286643	283839	0.98	0.001					
75_3	62098	61806	0.47	0.0007	75_3	61924	60698	1.98	0.001					
75_4	72245	70576	2.31	0.0007	75_4	128351	127050	1.01	0.001					
75_5	27616	26950	2.41	0.0005	75_5	137885	134814	2.23	0.001					
75_6	145273	144731	0.37	0.0003	75_6	229631	228765	0.38	0.001					
75_7	110979	109214	1.59	0.0003	75_7	269887	268312	0.58	0.001					
75_8	19570	18726	4.31	0.0004	75_8	600858	600631	0.04	0.001					
75_9	104341	102313	1.94	0.0002	75_9	516771	514428	0.45	0.001					
75_10	143740	142676	0.74	0.0002	75_10	142694	139787	2.04	0.001					
100_1	81978	81760	0.27	0.0003	100_1	937149	935422	0.18	0.001					
100_2	190424	188993	0.75	0.0007	100_2	303058	300427	0.87	0.001					
100_3	225434	225124	0.14	0.0004	100_3	29367	28455	3.11	0.001					
100_4	63028	-	-	-	100_4	100838	99610	1.22	0.001					
100_5	230076	224885	2.26	0.0006	100_5	786635	785980	0.08	0.001					
100_6	74358	73855	0.68	0.0005	100_6	41171	41171	0	0.001					
100_7	10330	9859	4.56	0.0007	100_7	701094	698219	0.41	0.001					
100_8	62582	61858	1.16	0.0004	100_8	782443	779304	0.4	0.001					
100_9	232754	231760	0.43	0.0002	100_9	628992	626879	0.34	0.001					
100_10	193262	191578	0.87	0.0004	100_10	378442	376219	0.59	0.001					

Cuadro A.4: Resultados comparativos de CM con tres algoritmos del estado del arte para $N=1000$.

d,l	CM		DP+FE [19]		GRASP+Tabu [52]		IHEA [13]	
	V.sol	$t(s)$	V.sol	$t(s)$	V.sol	$t(s)$	V.sol	$t(s)$
25_1	6150753	0.017	6172407	1682.280	6172407	18.234	6172407	2.765
25_2	224731	0.016	229833	2103.290	229941	20.448	229941	5.390
25_3	167473	0.017	172418	1919.350	172418	13.429	172418	5.892
25_4	359192	0.016	367365	2537.720	367426	16.188	367426	7.293
25_5	4872931	0.016	4885569	2626.970	4885611	23.368	4885611	5.543
25_6	13361	0.017	15528	608.550	15689	5.072	15689	1.635
25_7	4940406	0.016	4945741	2725.220	4945810	22.636	4945810	4.804
25_8	1699843	0.016	1709954	3762.890	1710198	44.150	1710198	7.104
25_9	489517	0.016	496315	2839.990	496315	18.619	496315	6.891
25_10	1164491	0.016	1173686	3607.270	1173792	36.537	1173792	7.573
50_1	5652232	0.016	5663517	3722.470	5663590	31.459	5663590	6.870
50_2	178332	0.016	180831	1450.870	180831	0.893	180831	3.692
50_3	11363521	0.017	11384139	2071.250	11384283	19.753	11384283	3.338
50_4	317240	0.017	322184	1868.860	322226	13.677	322226	5.433
50_5	9967800	0.016	9983477	2570.760	9984247	25.315	9984247	3.662
50_6	4092241	0.018	4106186	3801.720	4106261	36.010	4106261	7.691
50_7	10493537	0.016	10498135	2322.160	10498370	20.727	10498370	3.584
50_8	4975984	0.018	4981017	3826.980	4981146	72.100	4981146	9.155
50_9	1720453	0.016	1727727	3382.020	1727861	32.717	1727861	9.381
50_10	2329531	0.017	2340590	3605.070	2340724	59.074	2340724	7.416
75_1	11554653	0.017	11569498	3334.210	11570056	39.680	11570056	4.892
75_2	1894859	0.017	1901119	3094.560	1901389	20.131	1901389	6.492
75_3	2092412	0.017	2096415	3208.980	2096485	24.713	2096485	8.742
75_4	7293839	0.017	7305195	3821.020	7305321	34.156	7305321	6.846
75_5	13950705	0.017	13969705	2887.190	13970240	23.182	13970842	6.022
75_6	12274739	0.017	12288299	3178.950	12288738	20.733	12288738	4.463
75_7	1092797	0.018	1095837	2580.270	1095837	14.359	1095837	7.119
75_8	5564147	0.016	5575592	3804.420	5575813	42.451	5575813	7.833
75_9	687813	0.017	695595	2171.330	695774	14.062	695774	4.624
75_10	2501816	0.017	2507627	3349.440	2507677	29.338	2507677	6.863
100_1	6231812	0.016	6243330	3849.500	6243494	44.646	6243494	7.018
100_2	4837713	0.017	4853927	3627.050	4854086	52.601	4854086	7.092
100_3	3157656	0.017	3171955	3320.520	3172022	29.177	3172022	6.391
100_4	749978	0.017	754542	1990.800	754727	14.651	754727	5.207
100_5	18626930	0.017	18646607	2829.350	18646620	24.273	18646620	4.070
100_6	16004630	0.017	16019697	3247.810	16018298	25.780	16020232	5.204
100_7	12933578	0.017	12936205	3587.160	12936205	27.590	12936205	5.533
100_8	6887903	0.017	6927342	3850.890	6927738	59.551	6927738	7.298
100_9	3870342	0.017	3874959	3463.920	3874959	32.414	3874959	7.085
100_10	1328223	0.016	1334389	2474.890	1334494	14.651	1334494	6.270

Cuadro A.5: Resultados comparativos de CM con tres algoritmos del estado del arte para $N=2000$.

$d.l$	CM		DP+FE [19]		GRASP+Tabu [52]		IHEA [13]	
	V.sol	$t(s)$	V.sol	$t(s)$	V.sol	$t(s)$	V.sol	$t(s)$
25_1	5251589	0.071	5268004	57726.920	5268188	320.273	5268188	22.264
25_2	13270768	0.070	13293940	51050.130	13294030	205.053	13294030	24.917
25_3	5477886	0.071	5500323	57419.270	5500433	496.081	5500433	28.933
25_4	14606794	0.070	14624769	46620.160	14625118	215.072	14625118	17.050
25_5	5957249	0.070	5975645	57416.960	5975751	457.765	5975751	28.102
25_6	4474141	0.071	4491533	56155.800	4491691	294.252	4491691	23.442
25_7	6372821	0.070	6388475	57116.940	6388756	346.090	6388756	25.178
25_8	11746351	0.069	11769395	52832.060	11769873	277.109	11769873	22.584
25_9	10941184	0.070	10959388	54258.650	10960328	278.882	10960328	22.420
25_10	134917	0.070	139233	14686.960	139236	68.070	139236	7.551
50_1	7053129	0.069	7070736	52860.690	7070736	294.078	7070736	28.016
50_2	12562364	0.070	12586693	57518.440	12587545	331.619	12587545	23.943
50_3	27246961	0.070	27266846	48397.300	27268336	191.506	27268336	22.691
50_4	17720044	0.070	17754391	57376.090	17754434	485.249	17754434	24.506
50_5	16773712	0.070	16804699	57563.580	16805490	923.936	16806059	32.057
50_6	23047769	0.070	23075693	52613.210	23076155	285.256	23076155	21.579
50_7	28753633	0.069	28757657	46437.960	28759759	442.792	28759759	25.365
50_8	1567412	0.070	1580242	32416.870	1580242	102.412	1580242	13.937
50_9	26492874	0.070	26523637	48529.930	26523791	212.114	26523791	19.695
50_10	24727531	0.069	24746249	50565.420	24747047	253.202	24747047	20.613
75_1	25098111	0.070	25121327	57579.990	25121998	500.371	25121998	22.721
75_2	12635538	0.070	12663927	54629.120	12664670	316.231	12664670	21.584
75_3	43928427	0.070	43943294	45151.420	43943994	171.362	43943994	18.723
75_4	37476687	0.070	37496414	50255.520	37496613	219.561	37496613	19.901
75_5	24793604	0.070	24835254	56840.030	24834948	424.285	24835349	27.439
75_6	45127942	0.069	45137702	44437.730	45137758	190.011	45137758	20.862
75_7	25486552	0.068	25502503	57480.680	25502608	303.887	25502608	21.848
75_8	10035317	0.070	10067752	52566.820	10067892	213.795	10067892	21.560
75_9	14173568	0.069	14177079	55684.210	14171994	329.877	14177079	32.008
75_10	7788713	0.069	7815419	48717.480	7815755	201.636	7815755	20.537
100_1	37911125	0.069	37929562	57195.970	37929909	270.140	37929909	21.622
100_2	33606556	0.069	33665281	57844.250	33647322	490.736	33665281	34.322
100_3	29876545	0.069	29951509	57198.420	29952019	923.360	29952019	23.249
100_4	26925001	0.068	26948234	57484.560	26949268	440.690	26949268	23.800
100_5	22009071	0.070	22040523	58316.780	22041715	466.252	22041715	23.346
100_6	18850028	0.070	18868630	56282.860	18868887	339.878	18868887	22.315
100_7	15829907	0.070	15850198	54333.570	15850597	358.472	15850597	22.555
100_8	13607904	0.070	13628210	52206.350	13628967	231.923	13628967	22.250
100_9	8374905	0.071	8394440	45817.310	8394562	188.672	8394562	18.686
100_10	4909671	0.070	4923413	38243.750	4923559	124.031	4923559	15.041

Cuadro A.6: Resultados comparativos de CM con algoritmo del estado del arte para $N=5000$.

$d.l$	CM		IHEA [13]	
	V.sol	$t(s)$	V.sol	$t(s)$
25_1	23614840	0.432	23667450	130.664
25_2	37833091	0.421	37914560	143.679
25_3	68241357	0.424	68295820	126.904
25_4	33811271	0.431	33866053	139.453
25_5	9494449	0.421	9533115	111.366
50_1	45173276	0.420	45194685	144.125
50_2	88291369	0.443	88355678	143.188
50_3	152355756	0.426	152447303	143.813
50_4	170934350	0.423	171000228	148.015
50_5	1181814	0.424	1187339	61.106
75_1	28128149	0.439	28170819	105.745
75_2	195365500	0.426	195434758	149.977
75_3	64272514	0.430	64324704	141.571
75_4	247307461	0.420	247348595	144.213
75_5	46403603	0.418	46462750	136.119
100_1	214357297	0.413	214425886	150.076
100_2	18758392	0.427	18783132	76.661
100_3	10768973	0.425	10784650	61.450
100_4	160509127	0.426	160539947	153.082
100_5	33135895	0.412	33166524	105.708

Cuadro A.7: Resultados comparativos de CM con algoritmo del estado del arte para $N=6000$.

$d.l$	CM		IHEA [13]	
	V.sol	$t(s)$	V.sol	$t(s)$
25_1	69789341	0.609	69832542	204.230
25_2	3673316	0.609	3697236	123.770
25_3	79246984	0.605	79300092	246.285
25_4	191462861	0.591	191531304	238.917
25_5	36080566	0.614	36121510	208.762
50_1	194286675	0.591	194344567	214.187
50_2	323635258	0.609	323753804	272.235
50_3	31859452	0.615	31913824	220.343
50_4	225486858	0.598	225556641	198.893
50_5	40885584	0.603	40931924	186.351
75_1	204401513	0.599	204512250	267.433
75_2	42353252	0.604	42422207	182.990
75_3	524314637	0.604	524508156	177.873
75_4	196912744	0.598	197004931	220.513
75_5	74332278	0.610	74350712	282.668
100_1	292228015	0.610	292257056	219.599
100_2	219704463	0.610	219791358	257.679
100_3	376854314	0.587	376967122	266.202
100_4	355591433	0.600	355609720	245.857
100_5	686256995	0.612	686364195	211.295

Cuadro A.8: Resultados comparativos de CM con un algoritmo del estado del arte para problemas de 100 variables con 5 y 10 restricciones.

R_N	C	i	Óptimo	CM			TPTEA [28]			
				V.sol	DP	t(s)	V.solprom	D.est	t.prom(s)	DP
5×100	25	1	24381	21659	11.16	0.012	24381.0	0	0.6	0
		2	24274	23319	3.93	0.008	24274.0	0	0.4	0
		3	23551	22007	6.56	0.016	23551.0	0	0.4	0
		4	23534	19104	18.82	0.016	23534.0	0	1.5	0
		5	23991	22203	7.45	0.010	23991.0	0	0.8	0
		6	24613	23552	4.31	0.019	24613.0	0	0.4	0
		7	25591	24541	4.10	0.014	25591.0	0	0.4	0
		8	23410	21777	6.98	0.029	23410.0	0	0.3	0
		9	24216	22621	6.59	0.012	24216.0	0	1.7	0
		10	24411	22164	9.20	0.009	24411.0	0	0.5	0
	50	1	42757	41341	3.31	0.022	42757.0	0	1.1	0
		2	42545	41168	3.24	0.009	42545.0	0	9.3	0
		3	41968	40367	3.81	0.010	41968.0	0	0.9	0
		4	45090	43807	2.85	0.023	45090.0	0	10.8	0
		5	42218	41229	2.34	0.023	42218.0	0	0.5	0
		6	42927	41779	2.67	0.024	42927.0	0	0.5	0
		7	42009	40634	3.27	0.019	42009.0	0	0.4	0
		8	45020	42149	6.38	0.026	45020.0	0	0.5	0
		9	43441	41362	4.79	0.008	43441.0	0	1.7	0
		10	44554	43912	1.44	0.009	44554.0	0	2.7	0
	75	1	59822	58394	2.39	0.102	59822.0	0	0.3	0
		2	62081	60952	1.82	0.017	62081.0	0	0.6	0
		3	59802	58614	1.99	0.018	59802.0	0	0.3	0
		4	60479	59099	2.28	0.011	60479.0	0	0.3	0
		5	61091	60069	1.67	0.029	61091.0	0	0.4	0
		6	58959	57197	2.99	0.015	58959.0	0	0.4	0
		7	61538	60110	2.32	0.009	61538.0	0	0.3	0
		8	61520	59404	3.44	0.013	61520.0	0	0.3	0
		9	59453	57830	2.73	0.014	59453.0	0	0.3	0
		10	59965	59650	0.53	0.010	59965.0	0	0.6	0
10×100	25	1	23064	20110	12.81	0.011	23064.0	0	1.5	0
		2	22801	20420	10.44	0.030	22801.0	0	1.4	0
		3	22131	18830	14.92	0.011	22131.0	0	1.1	0
		4	22772	19476	14.47	0.014	22772.0	0	19.7	0
		5	22751	20063	11.81	0.022	22751.0	0	0.4	0
		6	22777	19860	12.81	0.012	22777.0	0	5.9	0
		7	21875	19230	12.09	0.015	21875.0	0	0.5	0
		8	22635	20681	8.63	0.008	22635.0	0	5.0	0
		9	22511	18852	16.25	0.027	22511.0	0	0.5	0
		10	22702	20668	8.96	0.009	22702.0	0	0.5	0
	50	1	41395	39479	4.63	0.011	41395.0	0	16.6	0
		2	42344	39205	7.41	0.007	42344.0	0	0.9	0
		3	42401	40545	4.38	0.018	42401.0	0	13.1	0
		4	45624	44258	2.99	0.009	45624.0	0	21.5	0
		5	41884	40148	4.14	0.006	41884.0	0	6.5	0
		6	42995	40442	5.94	0.007	42995.0	0	0.6	0
		7	43574	40961	6.00	0.014	43574.0	0	16.2	0
		8	42970	40232	6.37	0.023	42970.0	0	15.3	0
		9	42212	39729	5.88	0.004	42212.0	0	0.5	0
		10	41207	39145	5.00	0.006	41207.0	0	18.8	0
	75	1	57375	56170	2.10	0.009	57375.0	0	0.4	0
		2	58978	57540	2.44	0.006	58978.0	0	1.0	0
		3	58391	56142	3.85	0.021	58391.0	0	0.5	0
		4	61966	59883	3.36	0.012	61966.0	0	1.7	0
		5	60803	59630	1.93	0.011	60803.0	0	0.5	0
		6	61437	59192	3.65	0.010	61437.0	0	5.7	0
		7	56377	55302	1.91	0.005	56377.0	0	7.9	0
		8	59391	57332	3.47	0.023	59391.0	0	0.4	0
		9	60205	58057	3.57	0.020	60205.0	0	0.6	0
		10	60633	58377	3.72	0.099	60633.0	0	0.5	0

Cuadro A.9: Resultados comparativos de CM con un algoritmo del estado del arte para problemas de 100 y 250 variables con 30 y 5 restricciones, respectivamente.

R_N	C	i	Óptimo	CM			TPTEA [28]			
				V.sol	DP	t(s)	V.solprom	D.est	t.prom(s)	DP
30×100	25	1	21946	19093	13.00	0.171	21946.0	0	7.5	0
		2	21716	18302	15.72	0.025	21716.0	0	17.7	0
		3	20754	17892	13.79	0.023	20754.0	0	10.4	0
		4	21464	19946	7.07	0.008	21464.0	0	12.9	0
		5	21844	19666	9.97	0.009	21844.0	0	15.6	0
		6	22176	18438	16.86	0.018	22176.0	0	0.7	0
		7	21799	19864	8.88	0.029	21799.0	0	19.5	0
		8	21397	18474	13.66	0.033	21397.0	0	15.8	0
		9	22525	20907	7.18	0.007	22525.0	0	15.0	0
		10	20983	18955	9.66	0.044	20983.0	0	0.8	0
	50	1	40767	38610	5.29	0.007	40767.0	0	22.5	0
		2	41308	38347	7.17	0.029	41308.0	0	19.2	0
		3	41630	37386	10.19	0.014	41630.0	0	28.5	0
		4	41041	39621	3.46	0.007	41041.0	0	22.1	0
		5	40889	38413	6.06	0.029	40889.0	0	21.6	0
		6	41058	37296	9.16	0.021	41058.0	0	0.9	0
		7	41062	38897	5.27	0.051	41062.0	0	11.3	0
		8	42719	41344	3.22	0.015	42719.0	0	19.4	0
		9	42230	40322	4.52	0.010	42230.0	0	1.2	0
		10	41700	39461	5.37	0.052	41700.0	0	14.3	0
	75	1	57494	55929	2.72	0.008	57494.0	0	0.6	0
		2	60027	58261	2.94	0.008	60027.0	0	1.4	0
		3	58052	56912	1.96	0.016	58052.0	0	18.2	0
		4	60776	59590	1.95	0.020	60776.0	0	4.5	0
		5	58884	57142	2.96	0.009	58884.0	0	4.9	0
		6	60011	57977	3.39	0.047	60011.0	0	2.3	0
		7	58132	56374	3.02	0.086	58132.0	0	0.7	0
		8	59064	57154	3.23	0.040	59064.0	0	0.7	0
		9	58975	57199	3.01	0.012	58975.0	0	19.6	0
		10	60603	58196	3.97	0.131	60603.0	0	2.3	0
5×250	25	1	59312	55931	5.70	0.010	59312.0	0	67.0	0
		2	61472	58244	5.25	0.011	61472.0	0	252.0	0
		3	62130	59885	3.61	0.009	62130.0	0	83.0	0
		4	59463	54923	7.64	0.036	59462.3	2.49	1017.0	0
		5	58951	57058	3.21	0.008	58951.0	0	99.0	0
		6	60077	57200	4.79	0.008	60069.5	7.5	742.0	0.01
		7	60414	57438	4.93	0.010	60414.0	0	97.0	0
		8	61472	56820	7.57	0.026	61472.0	0	153.0	0
		9	61885	59730	3.48	0.011	61885.0	0	85.0	0
		10	58959	56131	4.80	0.009	58959.0	0	79.0	0
	50	1	109109	105080	3.69	0.006	109109.0	0	289.0	0
		2	109841	106886	2.69	0.031	109841.0	0	118.0	0
		3	108508	106477	1.87	0.034	108508.0	0	107.0	0
		4	109383	107185	2.01	0.006	109383.0	0	145.0	0
		5	110720	108076	2.39	0.004	110720.0	0	611.0	0
		6	110256	105977	3.88	0.010	110256.0	0	258.0	0
		7	109040	106434	2.39	0.013	109040.0	0	115.0	0
		8	109042	105331	3.40	0.010	109042.0	0	103.0	0
		9	109971	108210	1.60	0.009	109971.0	0	212.0	0
		10	107058	103138	3.66	0.011	107058.0	0	211.0	0
	75	1	149665	146500	2.11	0.008	149665.0	0	250.0	0
		2	155944	154198	1.12	0.009	155943.9	0.72	61.0	0
		3	149334	146199	2.10	0.010	149334.0	0	119.0	0
		4	152130	150537	1.05	0.007	152130.0	0	56.0	0
		5	150353	148592	1.17	0.011	150353.0	0	58.0	0
		6	150045	147410	1.76	0.008	150045.0	0	42.0	0
		7	148607	146552	1.38	0.014	148607.0	0	36.0	0
		8	149782	147699	1.39	0.010	149782.0	0	52.0	0
		9	155075	152820	1.45	0.018	155075.0	0	42.0	0
		10	154668	153588	0.70	0.010	154668.0	0	118.0	0

Cuadro A.10: Resultados comparativos de CM con un algoritmo del estado del arte para problemas de 250 variables con 10 y 30 restricciones.

R_N	C	i	Óptimo	CM			TPTEA [28]			
				V.sol	DP	t(s)	V.solprom	D.est	t.prom(s)	DP
10×250	25	1	59187	54789	7.43	0.053	59187.0	0	195.0	0
		2	58781	54481	7.32	0.008	58743.1	35.42	715.0	0.06
		3	58097	54017	7.02	0.014	58097.0	0	190.0	0
		4	61000	57230	6.18	0.008	60998.6	4.55	839.0	0
		5	58092	52228	10.09	0.007	58090.6	5.36	822.0	0
		6	58824	53580	8.91	0.013	58822.6	5.24	462.0	0
		7	58704	54844	6.58	0.004	58704.0	0	385.0	0
		8	58936	54218	8.01	0.007	58932.1	2.47	733.0	0.01
		9	59387	56607	4.68	0.013	59387.0	0	102.0	0
		10	59208	54603	7.78	0.021	59208.0	0	327.0	0
50	1	1	110913	106175	4.27	0.009	110913.0	0	371.0	0
		2	108717	104515	3.87	0.006	108717.0	0	529.0	0
		3	108932	104966	3.64	0.007	108932.0	0	77.0	0
		4	110086	104158	5.38	0.007	110086.0	0	1071.0	0
		5	108485	105206	3.02	0.009	108485.0	0	129.0	0
		6	110845	105780	4.57	0.007	110843.7	1.89	1064.0	0
		7	106077	101628	4.19	0.022	106075.7	0.96	239.0	0
		8	106686	102317	4.10	0.013	106686.0	0	563.0	0
		9	109829	105760	3.70	0.006	109827.4	1.96	845.0	0
		10	106723	104146	2.41	0.008	106723.0	0	81.0	0
75	1	1	151809	148983	1.86	0.023	151809.0	0	177.0	0
		2	148772	144981	2.55	0.011	148772.0	0	25.0	0
		3	151909	148272	2.39	0.010	151909.0	0	86.0	0
		4	151324	148981	1.55	0.008	151324.0	0	629.0	0
		5	151966	148880	2.03	0.014	151961.8	7.61	414.0	0
		6	152109	149908	1.45	0.009	152109.0	0	51.0	0
		7	153131	150233	1.89	0.036	153131.0	0	36.0	0
		8	153578	150849	1.78	0.012	153578.0	0	96.0	0
		9	149160	147026	1.43	0.008	149160.0	0	59.0	0
		10	149704	146922	1.86	0.029	149704.0	0	56.0	0
30×250	25	1	56842	50962	10.34	0.013	56824.0	0	131.0	0.03
		2	58520	53124	9.22	0.010	58520.0	0	216.0	0
		3	56614	52308	7.61	0.007	56614.0	0	216.0	0
		4	56930	53998	5.15	0.004	56930.0	0	91.0	0
		5	56629	53625	5.30	0.014	56629.0	0	74.0	0
		6	57205	53822	5.91	0.006	57205.0	0	374.0	0
		7	56357	51236	9.09	0.009	56333.4	26.79	1155.0	0.04
		8	56457	50367	10.79	0.015	56457.0	0	103.0	0
		9	57474	51829	9.82	0.010	57458.9	15.21	971.0	0.03
		10	56447	50599	10.36	0.031	56447.0	0	99.0	0
50	1	1	107770	103476	3.98	0.006	107763.1	8.6	1034.0	0.01
		2	108392	103862	4.18	0.010	108387.2	6.26	438.0	0
		3	106442	99856	6.19	0.011	106439.6	3.67	587.0	0
		4	106876	104169	2.53	0.008	106876.0	0	205.0	0
		5	107414	101716	5.30	0.009	107414.0	0	230.0	0
		6	107271	103761	3.27	0.003	107271.0	0	294.0	0
		7	106372	102491	3.65	0.010	106371.8	1.26	682.0	0
		8	104032	100482	3.41	0.006	104019.0	8.03	497.0	0.01
		9	106856	103768	2.89	0.007	106852.5	7.83	322.0	0
		10	105780	97197	8.11	0.358	105779.2	4.49	441.0	0
75	1	1	150163	147277	1.92	0.008	150163.0	0	457.0	0
		2	149958	145799	2.77	0.022	149958.0	0	101.0	0
		3	153007	150283	1.78	0.101	153007.0	0	131.0	0
		4	153234	149110	2.69	0.023	153234.0	0	84.0	0
		5	150287	147137	2.10	0.009	150287.0	0	51.0	0
		6	148574	146481	1.41	0.005	148574.0	0	77.0	0
		7	147477	143766	2.52	0.007	147477.0	0	79.0	0
		8	152912	149459	2.26	0.058	152912.0	0	71.0	0
		9	149570	146179	2.27	0.010	149570.0	0	61.0	0
		10	149668	145560	2.74	0.009	149668.0	0	742.0	0

Cuadro A.11: Resultados comparativos de CM con un algoritmo del estado del arte para problemas de 500 variables con 5 y 10 restricciones.

R_N	C	i	M.conocido	CM			TPTEA [28]			
				V.sol	DP	t(s)	V.solprom	D.est	t.prom(s)	DP
5×500	25	1	120148	117446	2.25	0.005	120126.9	9.53	3753.0	0.02
		2	117879	115311	2.18	0.006	117850.8	11.96	3876.0	0.02
		3	121131	116611	3.73	0.007	121112.2	6.74	3148.0	0.02
		4	120804	117410	2.81	0.006	120786.4	10.32	2918.0	0.01
		5	122319	116960	4.38	0.010	122319.0	0	1936.0	0
		6	122024	119365	2.18	0.008	122008.8	10.14	4421.0	0.01
		7	119127	115500	3.04	0.008	119120.5	4.79	3419.0	0.01
		8	120568	117572	2.48	0.009	120548.1	10.4	2739.0	0.02
		9	121586	114012	6.23	0.008	121559.2	9	2719.0	0.02
		10	120717	115969	3.93	0.023	120695.0	8.52	3353.0	0.02
50	1	1	218428	212667	2.64	0.020	218411.3	11.61	4100.0	0.01
		2	221202	213380	3.54	0.012	221184.9	4.24	3560.0	0.01
		3	217542	214854	1.24	0.005	217525.9	6.62	3836.0	0.01
		4	223560	220767	1.25	0.025	223558.9	0.99	2139.0	0
		5	218966	214801	1.90	0.119	218966.0	0	171.0	0
		6	220530	217163	1.53	0.006	220528.1	2.1	3183.0	0
		7	219989	216498	1.59	0.435	219985.9	4.58	2799.0	0
		8	218215	211726	2.97	0.009	218200.4	4.78	3700.0	0.01
		9	216976	214359	1.21	0.010	216976.0	0	608.0	0
		10	219719	215931	1.72	0.013	219715.6	3.61	2633.0	0
75	1	1	295828	293420	0.81	0.015	295828.0	0	552.0	0
		2	308086	304089	1.30	0.009	308081.9	2.23	3225.0	0
		3	299796	297004	0.93	0.010	299796.0	0	638.0	0
		4	306480	304426	0.67	0.006	306478.5	1.69	2626.0	0
		5	300342	295529	1.60	0.013	300340.7	2.89	3454.0	0
		6	302571	301198	0.45	0.028	302565.4	4.29	2157.0	0
		7	301339	298384	0.98	0.012	301330.7	3.73	590.0	0
		8	306454	303424	0.99	0.008	306454.0	0	1707.0	0
		9	302828	301177	0.55	0.007	302820.7	8.05	2852.0	0
		10	299910	296689	1.07	0.006	299901.8	3.66	3492.0	0
10×500	25	1	117821	111643	5.24	0.007	117736.2	20.87	3665.0	0.07
		2	119249	114349	4.11	0.003	119137.5	26.48	3355.0	0.09
		3	119215	112433	5.69	0.036	119108.3	20.13	4470.0	0.09
		4	118829	115069	3.16	0.009	118793.9	16.73	3150.0	0.03
		5	116530	109932	5.66	0.012	116405.2	19.27	3582.0	0.11
		6	119504	115196	3.60	0.010	119441.8	20.5	3647.0	0.05
		7	119827	114244	4.66	0.007	119739.7	16.4	3790.0	0.07
		8	118344	114988	2.84	0.005	118258.3	23.63	4229.0	0.07
		9	117815	109049	7.44	0.010	117706.0	28.53	3169.0	0.09
		10	119251	114171	4.26	0.006	119161.9	18.67	3339.0	0.07
50	1	1	217377	213456	1.80	0.003	217313.7	15.31	3984.0	0.03
		2	219077	211410	3.50	0.007	219022.7	15.56	3598.0	0.02
		3	217847	211560	2.89	0.009	217786.7	16.25	4010.0	0.03
		4	216868	212020	2.24	0.018	216836.3	18.66	3404.0	0.01
		5	213873	207212	3.11	0.007	213780.3	13.61	4095.0	0.04
		6	215086	209835	2.44	0.019	215049.6	14.45	3622.0	0.02
		7	217940	214515	1.57	0.006	217884.8	17.07	4281.0	0.03
		8	219990	213103	3.13	0.014	219947.4	18.07	3908.0	0.02
		9	214382	208795	2.61	0.010	214327.4	15.24	3999.0	0.03
		10	220899	217736	1.43	0.006	220864.4	16.89	3212.0	0.02
75	1	1	304387	300425	1.30	0.008	304364.5	10.35	3025.0	0.01
		2	302379	298625	1.24	0.007	302364.5	10.05	2884.0	0
		3	302417	298611	1.26	0.032	302398.1	10.22	3498.0	0.01
		4	300784	297530	1.08	0.008	300758.8	6.73	967.0	0.01
		5	304374	301773	0.85	0.075	304361.1	6.21	3509.0	0
		6	301836	296093	1.90	0.010	301740.4	14.11	3809.0	0.03
		7	304952	303214	0.57	0.005	304952.0	0	3133.0	0
		8	296478	291693	1.61	0.012	296455.5	6.61	3139.0	0.01
		9	301359	298373	0.99	0.009	301349.3	11.82	3187.0	0
		10	307089	301104	1.95	0.016	307088.2	3.2	2352.0	0

Cuadro A.12: Resultados comparativos de CM con un algoritmo del estado del arte para problemas de 500 variables con 30 restricciones.

R_N	C	i	M.conocido	CM			TPTEA [28]			
				V.sol	DP	t(s)	V.solprom	D.est	t.prom(s)	DP
30×500	25	1	116056	108936	6.13	0.004	115897.2	23.8	3969.0	0.14
		2	114810	109480	4.64	0.007	114733.0	20.01	3549.0	0.07
		3	116741	111236	4.72	0.009	116619.1	29.95	4413.0	0.1
		4	115354	105494	8.55	0.010	115251.6	25.19	3499.0	0.09
		5	116525	107621	7.64	0.014	116364.8	32.83	3436.0	0.14
		6	115741	109083	5.75	0.007	115674.0	22.98	3848.0	0.06
		7	114181	106274	6.92	0.010	114037.1	38.04	4785.0	0.13
		8	114403	106959	6.51	0.007	114164.4	33.17	4110.0	0.21
		9	115419	109879	4.80	0.004	115221.4	26.06	4043.0	0.17
		10	117116	109540	6.47	0.013	116984.4	35.94	3779.0	0.11
50	1	1	218104	211117	3.20	0.007	218069.6	9.28	3163.0	0.02
		2	214648	207098	3.52	0.009	214544.9	35.69	3796.0	0.05
		3	215978	210105	2.72	0.019	215898.8	17.46	4008.0	0.04
		4	217910	209647	3.79	0.024	217831.3	30.8	3259.0	0.04
		5	215689	207539	3.78	0.009	215602.1	24.6	3945.0	0.04
		6	215919	207814	3.75	0.016	215766.2	26.09	3558.0	0.07
		7	215907	209459	2.99	0.023	215857.2	19.46	3177.0	0.02
		8	216542	210955	2.58	0.006	216459.7	25.8	3643.0	0.04
		9	217340	210840	2.99	0.007	217304.3	11.59	3461.0	0.02
		10	214739	209560	2.41	0.011	214671.3	30.71	3418.0	0.03
75	1	1	301675	297040	1.54	0.020	301641.6	11.43	2849.0	0.01
		2	300055	293142	2.30	0.067	300035.7	19.46	3863.0	0.01
		3	305087	299868	1.71	0.008	305080.5	7.8	3785.0	0
		4	302032	297320	1.56	0.018	301983.6	19.13	3200.0	0.02
		5	304462	300150	1.42	0.023	304427.5	11.95	3102.0	0.01
		6	297012	291258	1.94	0.008	296965.0	17.11	3743.0	0.02
		7	303364	297621	1.89	0.013	303335.6	13.11	2828.0	0.01
		8	307007	302932	1.33	0.007	306972.5	15.25	3922.0	0.01
		9	303199	298559	1.53	0.015	303168.5	13.57	3283.0	0.01
		10	300572	295549	1.67	0.038	300530.2	16.91	3937.0	0.01

BIBLIOGRAFÍA

- [1] ANDERSON, E., «Mechanisms for local search», *European Journal of Operational Research*, **88**(1), págs. 139–151, 1996.
- [2] ATILGAN, C. y U. NURIYEV, «Hybrid heuristic algorithm for the multidimensional knapsack problem», en *Problems of Cybernetics and Informatics, 2012 IV International Conference*, IEEE, págs. 1–4, 2012.
- [3] BANDA, J., J. VELASCO y A. BERRONES, «Dual mean field search for large scale linear and quadratic knapsack problems», *Physica A: Statistical Mechanics and its Applications*, **478**, págs. 158–167, 2017.
- [4] BANDA, J., J. VELASCO y A. BERRONES, «A hybrid heuristic algorithm based on Mean-Field Theory with a Simple Local Search for the Quadratic Knapsack Problem», en *IEEE Congress on Evolutionary Computation*, págs. 2559–2565, June 2017.
- [5] BANITALEBI, A., M. AZIZ y Z. AZIZ, «A self-adaptive binary differential evolution algorithm for large scale binary optimization problems», *Information Sciences*, **367-368**, págs. 487–511, 2016.
- [6] BARTHOLDI, J., *The Knapsack Problem*, Springer US, Boston, MA, págs. 19–31, 2008.
- [7] BILLIONNET, A. y E. SOUTIF, «An exact method based on Lagrangian decomposition for the 0–1 quadratic knapsack problem», *European Journal of Operational Research*, **157**(3), págs. 565–575, 2004.

- [8] BLUM, C., J. PUCHINGER, G. RAIDL y A. ROLI, «Hybrid metaheuristics in combinatorial optimization: A survey», *Applied Soft Computing*, **11**(6), págs. 4135–4151, 2011.
- [9] BLUM, C. y A. ROLI, «Metaheuristics in combinatorial optimization: overview and conceptual comparison», *ACM Computing Surveys (CSUR)*, **35**(3), págs. 268–308, 2003.
- [10] BOYER, V., D. E. BAZ y M. ELKIHHEL, «Solving knapsack problems on GPU», *Computers and Operations Research*, **39**(1), págs. 42–47, special Issue on Knapsack Problems and Applications, 2012.
- [11] CAPRARA, A., D. PISINGER y P. TOTH, «Exact Solution of the Quadratic Knapsack Problem», *INFORMS Journal on Computing*, **11**(2), págs. 125–137, 1999.
- [12] CENGEL, Y. y M. BOLES, *Thermodynamics: an engineering approach*, McGraw-Hill, NY, USA.
- [13] CHEN, Y. y J. HAO, «An iterated hyperplane exploration approach for the quadratic knapsack problem», *Computers and Operations Research*, **77**, págs. 226–239, 2017.
- [14] CHISHOLM, J. y D. A. BORDE, *An introduction to statistical mechanics*, Internat. Ser. Mono. Phys., Pergamon, London, 1958.
- [15] CHU, P. y J. BEASLEY, «A Genetic Algorithm for the Multidimensional Knapsack Problem», *Journal of Heuristics*, **4**(1), págs. 63–86, Jun 1998.
- [16] DANTAS, B. y E. CÁCERES, «A Parallelization of a Simulated Annealing Approach for 0-1 Multidimensional Knapsack Problem Using GPGPU», en *Computer Architecture and High Performance Computing (SBAC-PAD), 2016 28th International Symposium on*, IEEE, págs. 134–140, 2016.
- [17] DE SIMONE, C., «The cut polytope and the Boolean quadric polytope», *Discrete Mathematics*, **79**(1), págs. 71–75, 1990.

- [18] DIJKHUIZEN, G. y U. FAIGLE, «A cutting-plane approach to the edge-weighted maximal clique problem», *European Journal of Operational Research*, **69**(1), págs. 121–130, 1993.
- [19] FOMENI, F. y A. LETCHFORD, «A Dynamic Programming Heuristic for the Quadratic Knapsack Problem», *INFORMS Journal on Computing*, **26**(1), págs. 173–182, 2014.
- [20] FRÉVILLE, A., «The multidimensional 0–1 knapsack problem: an overview», *European Journal of Operational Research*, **155**(1), págs. 1–21, 2004.
- [21] GAREY, M. y D. JOHNSON, *Computers and Intractability: A Guide to the Theory of NP-Completeness*, W.H. Freeman and Company, 2011.
- [22] GAVISH, B. y H. PIRKUL, «Efficient algorithms for solving multiconstraint zero-one knapsack problems to optimality», *Mathematical Programming*, **31**(1), págs. 78–105, Jan 1985.
- [23] GILL, P., W. MURRAY y M. WRIGHT, *Practical optimization*, Academic press, 1981.
- [24] GLAZER, M. y J. WARK, *Statistical mechanics: a survival guide*, Oxford University Press, London, 2002.
- [25] HERSHEY, J. y P. OLSEN, «Approximating the Kullback Leibler divergence between Gaussian mixture models», en *Acoustics, Speech and Signal Processing, 2007. IEEE International Conference on*, tomo 4, IEEE, págs. IV–317, 2007.
- [26] KELLERER, H., U. PFERSCHY y D. PISINGER, «Knapsack problems», capítulo Introduction to NP-Completeness of knapsack problems, Springer Berlin Heidelberg, Berlin, Heidelberg, págs. 483–493, 2004.
- [27] KONG, X., L. GAO, H. OUYANG y S. LI, «A simplified binary harmony search algorithm for large scale 0–1 knapsack problems», *Expert Systems with Applications*, **42**(12), págs. 5337–5355, 2015.

- [28] LAI, X., J. HAO, F. GLOVER y Z. LÜ, «A two-phase tabu-evolutionary algorithm for the 0–1 multidimensional knapsack problem», *Information Sciences*, **436-437**, págs. 282–301, 2018.
- [29] LENSTRA, J., *Local search in combinatorial optimization*, Princeton University Press, 2003.
- [30] LOZANO, M., M. LAGUNA, R. MARTÍ, F. RODRÍGUEZ y C. GARCÍA-MARTÍNEZ, «A genetic algorithm for the minimum generating set problem», *Applied Soft Computing*, **48**, págs. 254–264, 2016.
- [31] MAHI, M., Ö. BAYKAN y H. KODAZ, «A new hybrid method based on particle swarm optimization, ant colony optimization and 3-opt algorithms for traveling salesman problem», *Applied Soft Computing*, **30**, págs. 484–490, 2015.
- [32] MARTELLO, S., D. PISINGER y P. TOTH, «New trends in exact algorithms for the 0–1 knapsack problem», *European Journal of Operational Research*, **123**(2), págs. 325–332, 2000.
- [33] MEIER, H., N. CHRISTOFIDES y G. SALKIN, «Capital Budgeting Under Uncertainty—An Integrated Approach Using Contingent Claims Analysis and Integer Programming», *Operations Research*, **49**(2), págs. 196–206, 2001.
- [34] MITCHELL, J., «Branch-and-cut algorithms for combinatorial optimization problems», *Handbook of Applied Optimization*, págs. 65–77, 2002.
- [35] MOHSEN, A., «Annealing Ant Colony Optimization with Mutation Operator for Solving TSP», *Computational Intelligence and Neuroscience*, 2016.
- [36] MUSSARDO, G., *Statistical field theory: an introduction to exactly solved models in statistical physics*, Oxford University Press, NY, USA, 2010.
- [37] OPPER, M. y D. SAAD, *Advanced mean field methods: theory and practice*, MIT press, Massachusetts, USA, 2001.
- [38] PAPADIMITRIOU, C., *Computational complexity*, Addison Wesley, USA, 1994.

- [39] PAPANIMITRIOU, C. y K. STEIGLITZ, *Combinatorial optimization: algorithms and complexity*, Prentice Hall Inc., NJ, USA, 1998.
- [40] PARK, K., K. LEE y S. PARK, «An extended formulation approach to the edge-weighted maximal clique problem», *European Journal of Operational Research*, **95**(3), págs. 671–682, 1996.
- [41] PELIKAN, M., D. GOLDBERG y F. LOBO, «A survey of optimization by building and using probabilistic models», *Computational Optimization and Applications*, **21**(1), págs. 5–20, 2002.
- [42] PETERSEN, C., «Computational Experience with Variants of the Balas Algorithm Applied to the Selection of R&D Projects», *Management Science*, **13**(9), págs. 736–750, 1967.
- [43] PISINGER, D., «Where are the hard knapsack problems?», *Computers and Operations Research*, **32**(9), págs. 2271–2284, 2005.
- [44] PISINGER, D., «The quadratic knapsack problem: a survey», *Discrete Applied Mathematics*, **155**(5), págs. 623–648, 2007.
- [45] SALKIN, H. y C. DE KLUYVER, «The knapsack problem: a survey», *Naval Research Logistics Quarterly*, **22**(1), págs. 127–144, 1975.
- [46] SHIH, W., «A Branch and Bound Method for the Multiconstraint Zero-One Knapsack Problem», *Journal of the Operational Research Society*, **30**(4), págs. 369–378, 1979.
- [47] TALBI, E., *Metaheuristics: from Design to Implementation*, Wiley Publishing, NJ, USA, 2009.
- [48] TOUMI, S., B. JARBOUI, M. EDDALY y A. REBAÏ, «Branch-and-bound algorithm for solving blocking flowshop scheduling problems with makespan criterion», *International Journal of Mathematics in Operational Research*, **10**(1), págs. 34–48, 2017.

- [49] WITZGALL, C., «Mathematical methods of site selection for Electronic Message Systems», *NASA STI/Recon Technical Report N*, **76**, 1975.
- [50] XIE, X. y J. LIU, «A mini-swarm for the quadratic knapsack problem», en *Swarm Intelligence Symposium*, IEEE, págs. 190–197, 2007.
- [51] YANG, X., *Nature-inspired metaheuristic algorithms*, Luniver press, 2010.
- [52] YANG, Z., G. WANG y F. CHU, «An effective GRASP and tabu search for the 0–1 quadratic knapsack problem», *Computers and Operations Research*, **40**(5), págs. 1176–1185, 2013.
- [53] ZHOU, Y., Z. KUANG y J. WANG, «A chaotic neural network combined heuristic strategy for multidimensional knapsack problem», en *Advances in Computation and Intelligence*, Springer, Berlin, Heidelberg, págs. 715–722, 2008.

RESUMEN AUTOBIOGRÁFICO

Juan Antonio Banda Moreno

Candidato para obtener el grado de
Doctor en Ingeniería
con especialidad en Ingeniería de Sistemas

Universidad Autónoma de Nuevo León
Facultad de Ingeniería Mecánica y Eléctrica

Tesis:

ANÁLISIS DEL DESEMPEÑO DE ALGORITMOS BASADOS EN LA
TEORÍA DE CAMPO MEDIO PARA PROBLEMAS TIPO MOCHILA

Nací el 30 de diciembre de 1986 en Cd. Valles, SLP, lugar dónde realicé mis estudios de educación básica y media superior. En el año 2006 comencé a estudiar la Licenciatura en Matemáticas en la Facultad de Ciencias Físico–Matemáticas de la Universidad Autónoma de Nuevo León. Posteriormente en la misma institución pero ahora en la Facultad de Ingeniería Mecánica y Eléctrica, obtuve el grado de Maestro en Ciencias en Ingeniería de Sistemas en 2014. En 2015 inicié con el presente tema de investigación en mencionada institución.