Revista de Matemática: Teoría y Aplicaciones **30**(2): 173-192
CIMPA - UCR ISSN: 1409-2433(Print), 2215-3373(Online)
DOI: https://doi.org/10.15517/rmta.v30i2.51162

OPTIMIZANDO CON BÚSQUEDA TABÚ EN PARTICIONAMIENTO SOBRE DATOS ESPACIALES CON MÚLTIPLES OBJETIVOS

OPTIMIZATION WITH TABU SEARCH IN SPATIAL DATA CLUSTERING WITH MULTIPLE OBJECTIVES

Maria Beatríz Bernábe Loranca[1] Marco Antonio Rodríguez Flores[2]
Carmen Cerón Garnica[3] Gerardo Martínez Guzmán[4]

Received: 15/Jul/2022; Accepted: 14/Apr/2023

[1]Benemérita Universidad Autónoma de Puebla, Facultad de Ciencias de la Computación, Puebla, México. E-Mail: beatriz.bernabe@gmail.com
[2]Cyprus University of Technology, Department of Electrical Engineering and Computer Engineering and Informatics, Limasol, Chipre.
E-Mail: marco89_rf@hotmail.com
[3]Benemérita Universidad Autónoma de Puebla, Facultad de Ciencias de la Computación, Puebla, México. E-Mail: carmen.ceron@correo.buap.mx
[4]Benemérita Universidad Autónoma de Puebla, Facultad de Ciencias de la Computación, Puebla, México. E-Mail: gmartinez54@hotmail.com

Resumen

El particionamiento sobre datos geográficos es de gran utilidad para resolver problemas relacionados con diseño territorial. Para instancias de tamaño pequeño, este problema incluso es resuelto por métodos exactos en un tiempo de respuesta aceptable. Sin embargo, para instancias de tamaño grande y debido a la naturaleza combinatoria de este problema, la complejidad computacional aumenta y el uso de métodos de aproximación se ha hecho necesario. Un caso en particular de este tipo de problemas que ha tenido nuestra atención en los últimos años es el agrupamiento por particiones para AGEBS (áreas geoestadísticas básicas). Algunos trabajos relacionados se han desarrollado para resolver la formación de grupos compactos de AGEBS, pero la incorporación de restricciones adicionales ha sido poco tratada. Un problema interesante de aplicación muy demandado, es la extensión del agrupamiento compacto para construir grupos bajo el criterio de homogeneidad y/o balanceo en el número de objetos que componen los grupos. Este problema se traduce en un problema multiobjetivo, el cual debe lidiar con dos objetivos para conseguir un compromiso entre ambos. Este trabajo presenta un modelo de programación matemática multiobjetivo y su asociada implementación para lograr el equilibrio entre compacidad y homogeneidad en la cardinalidad de objetos. La metaheurística incorporada a este problema de agrupamiento territorial multiobjetivo ha sido búsqueda tabú.

Palabras clave: particionamiento; multiobjetivo; búsqueda tabú; diseño territorial; metaheurísticas.

Abstract

Clustering spatial-geographic units, zones or areas is employed to solve problems related to territorial design. The clustering adapts to the definition of territorial design of a particular problem, which demands spatial data processing under clustering schemes with topological requirements in the zones. For small instances, when geographical compactness is attended as an objective function, this problem is solved by exact methods in an acceptable response time. However, for bigger instances and due to the combinatory nature of this problem, the computational complexity increases and the employment of approximated methods becomes a necessity, in such a way that when the geographical compactness was the only cost function, a couple of approximated methods were implemented, giving satisfactory results. A particular case of this kind of problems that has our attention in recent years is the classification of AGEBS (basic geographical units by its initials in Spanish) through partitions. Some works were made related to the formation of compact groups of AGEBS, but additional restrictions were not often considered. A very interesting and demanded application problem is extending the compact clustering to form groups under a homogeneity criterion to balance the number of objects in every group. This problem implies a multiobjective approach that has to tackle two objectives to attain a balance between the two. This work presents a mathematical model and the resulting implementation to achieve the equilibrium between compactness and homogeneity in the number of objects. The metaheuristic incorporated to this multiobjective clustering problem is tabu search.

Keywords: partitioning, multiobjective; tabu search; territorial design; metaheuristics.

Mathematics Subject Classification: Primario: 68R01; Secundario: 68R05, 90C27, 90C59.

1 Introducción

Los datos espaciales son tratados frecuentemente en problemas de diseño territorial o zonas para dar respuesta a diversos problemas geográficos. El diseño de zonas ocurre cuando pequeñas áreas básicas o unidades geográficas deben ser agrupadas en zonas que resulten aceptables según los requerimientos impuestos por el problema estudiado. En este punto, el agrupamiento es planteado como geográfico y es una tarea implícita en diseño de territorio (DT). Los problemas de este tipo tienen como principio fundamental, la creación de grupos de zonas espacialmente compactas, contiguas y/o conexas.

Las aplicaciones del diseño territorial incluyen distribución política, comercio, censos, clasificación para muestreo (como es el caso que tratamos), etc.

Para incorporar en diseño territorial un método de agrupamiento espacial, es importante duplicar esfuerzos en la revisión de métodos de clustering. En este trabajo, después de distintas observaciones, la clasificación por particiones, ha sido preferida tan solo por su satisfactorio y práctico resultado implícito: la creación de zonas compactas. El particionamiento compacto resuelve la creación de grupos "geométricamente poligonales" debido a que la medida utilizada facilita la compacidad de cada figura de los grupos (minimización de distancias). Aún resuelto este problema, la inclusión de otras medidas de calidad permite mayor aplicación, por ejemplo la homogeneidad en el número de objetos para problemas de localización y ruteo.

Es en este tipo de homogeneidad y compacidad geométrica donde se centra este trabajo. Se presenta un modelo de programación matemática para resolver el problema de particionamiento territorial con dos objetivos: 1) compacidad geométrica y 2) homogeneidad en la cardinalidad de los objetos. Se hace uso de búsqueda tabú para aproximar las dos funciones en conflicto.

De lo anterior, el artículo se organiza como sigue: en la sección 2 se presentan los preliminares y aspectos teóricos. La sección 3 se compone de la descripción del problema y su modelo matemático. En la sección 4 se muestra el diseño del algoritmo con búsqueda tabú que hemos llamado Particionamiento con Diseño Territorial Penalidad (PTSP). Por último se presenta la experiencia de cómputo y conclusiones.

2 Preliminares

El particionamiento geográfico es una herramienta muy útil en el diseño territorial, el cual se entiende como el problema de agrupar pequeñas áreas geográficas en grupos más grandes de tal manera que estos últimos son aceptables de acuerdo a cierto criterio de planificación relevante. Dependiendo del contexto, los criterios pueden estar motivados ya sea económicamente (promedio de ventas potenciales, carga de trabajo o número de clientes) o tener un fondo demográfico (número de habitantes, población votante). Además restricciones espaciales (contigüidad, compacidad) son demandadas con regularidad (ver [7] y [13]).

En los criterios geográficos destaca la compacidad, comprendida como aquella en donde la formación de zonas es de forma “circular”; y sin distorsiones. Hasta hoy, una definición concreta de compacidad en esta área no ha sido publicada, cada problema ajusta este concepto. En general, este criterio para formar grupos compactos se ha planteado como una función objetivo de minimización de distancias (las unidades geográficas estén más cerca al centro del grupo al que pertenecen que a otro centro de otro grupo). Es decir, un método de agrupamiento se responsabiliza de conseguir este objetivo minimizando una función de costo que considera la distancia total de los elementos hacia su respectivo centro de grupo. La solución del problema del agrupamiento geográfico es de carácter combinatorio (NP-difícil) y exige que su solución sea generada a partir de métodos no exactos de optimización como las metaheurísticas utilizadas en [3] y [1].

Sin embargo, una restricción interesante que hemos perseguido es el criterio demográfico entendido como la homogeneidad en cuanto al número de habitantes en México, ha sido Romero en el año 2002 quien inició esta línea de investigación considerando los datos AGEBS en el agrupamiento [12]. Otras propuestas manejan el mismo tipo de unidad geográfica, los AGEBS y abordan el problema con técnicas de optimización combinatoria como búsqueda por entorno variable (VNS) y recocido simulado (RS) [3].

Un problema evidente en el tratamiento de los mapas con AGEBS consiste en la irregularidad geométrica: la forma de AGEBS y su dispersión en el plano, como en la Figura 1. Esta situación limita su solución con métodos de adyacencia y triangulación de Dealunay como lo han hecho otros autores, por ejemplo en [12].

Figura 1: Mapa de AGEBS con alta concentración de unidades en el centro y disperso hacia los bordes.

Las soluciones compactas resueltas por los trabajos que incluyen AGEBS son útiles para pocas aplicaciones. Sin embargo, en problemas reales, surgen otras restricciones que deben ser consideradas, lo que implica que la optimización de un sólo objetivo resulte insuficiente. En este punto encontrar soluciones “óptimas” para diferentes criterios se hace necesario. El desafío es alcanzar una solución óptima para homogeneidad en el número de objetos y compacidad geométrica.

En [4], se presentan dos técnicas para lograr el tipo de homogeneidad en la cardinalidad de los objetos. Resultados importantes se consiguieron, pero el tiempo de cómputo parece incrementarse considerablemente a partir de 100 grupos.

El presente trabajo, expone una mejor propuesta con soluciones prometedoras. La principal contribución es un modelo de optimización multiobjetivo para encontrar soluciones óptimas aproximadas para dos criterios de interés: la compacidad y la homogeneidad intragrupo en cuanto al número de elementos (AGEBS). La metaheurística escogida como método de aproximación ha sido búsqueda tabú y se ha incorporado al modelo propuesto para lidiar con la complejidad del problema. Por otra parte, hemos aplicado nuestra propuesta a datos recientes que corresponden a distritos de Colima y Puebla.

3 Descripción del problema

Ejemplos representativos en diseño territorial con distintos objetivos pueden verse en distribución electoral, muestreo de censos y atención comercial donde destaca la generación de grupos homogéneos en cuanto al número de elementos, extensión territorial o concentración de la población. Algunas de las aplicaciones del problema de diseño territorial son el diseño de distritos electorales, territorios escolares, albergues, recolección de basura, ventas y servicios, etc. Sin embargo, los más importantes son el diseño de territorios electorales y las ventas, ver [7] y [13].

El algoritmo que exponemos en este trabajo busca el compromiso entre los objetivos de compacidad geométrica y homogeneidad en la cardinalidad de los objetos. Este algoritmo es de particionamiento sobre medoides incluyendo un método de aproximación con búsqueda tabú.

En términos amplios, este algoritmo genera una solución inicial de k grupos, la cual puede ser aleatoria. Intercambios entre los elementos de los grupos o con los representantes de los grupos se colocan en una nueva configuración y constituyen una mejor solución que la inicial. Estas nuevas soluciones, llamadas “vecinas”, son generadas hasta que se satisface alguna condición de parada. Los métodos más utilizados para generar soluciones vecinas son los siguientes (según [9]):

Swap Method: se selecciona un objeto al azar de un grupo y otro objeto de otro grupo y luego se hace un intercambio entre estos. Se describe como sigue: dada una solución A, selecciona 2 objetos al azar $A(i)$ y $A(j)$ tales que no están en el mismo grupo, luego se asigna $A(i)$ al grupo de $A(j)$ y viceversa.

Single Method: este es el modo más sencillo. Un objeto se mueve a la vez de un grupo a otro. Se describe como sigue: dada una solución actual A, selecciona al azar $A(i)$, asigna el valor X definido como un entero generado al azar en el rango $[1, k]$ (donde k es el número de grupos) y que es distinto del grupo al que pertenece $A(i)$. Entonces sea $A(i) = X$.

Threshold Method: el modo de umbral de probabilidad se utiliza comúnmente para establecer las soluciones vecinas en la búsqueda tabú basada en el método de agrupamiento. Modera la "sacudida" sobre la solución actual. Entre mayor sea el valor del umbral de probabilidad, menor es la sacudida permitida y por lo tanto, las soluciones de prueba son más similares a la actual y viceversa.

En [9] y [10] una clasificación de algoritmos de particionamiento es descrita, de la cual los más populares son: k-medias, k-medianas, PAM (particionamiento alrededor de los medoides) y CLARA (ver además [11], [2] y [8]). Los inconvenientes de estos algoritmos obedecen generalmente a la calidad de la solución inicial o bien cuando las perturbaciones de las soluciones vecinas son aleatorias y que usualmente conducen a óptimos locales de los criterios que optimiza.

Hemos considerado los métodos *Swap Method* y *Single Method* para desarrollar una función de vecindario que intercambia a un centroide de la solución actual con el mejor candidato elegido de entre las unidades geográficas asignadas a dicho centroide, sin embargo cuando el grupo tiene tamaño igual a uno (sólo tiene un centroide), su centroide es reemplazado por un objeto al azar de entre todos los objetos disponibles en todo el mapa.

Una vez creada la solución inicial con un centroide aleatorio, en la generación de la siguiente solución con la estrategia tabú, la selección de nuevos centroides no es totalmente aleatoria porque en el proceso de búsqueda, las soluciones malas son bloqueadas y marcadas como tabú con el fin de alejarse de vecindades previamente exploradas. Las unidades geográficas (UGS) de recién uso, etiquetadas como centroides, son bloqueadas durante un cierto número de iteraciones con el fin de alejarse de soluciones previamente exploradas. Del mismo modo, los centroides insertados recientemente son marcados como tabú para un cierto número de iteraciones, es decir no pueden ser cambiados para favorecer la exploración en la búsqueda local. Además, en comparación con otros métodos que usan centroides, en nuestro algoritmo, estos tienen una probabilidad mayor a cero de que sean cambiados muy pronto.

3.1 Modelación

Entendido que los objetivos a optimizar son la compacidad y la homogeneidad (balanceo de objetos), considerar el objetivo de homogeneidad como una restricción "dura", estima el balanceo ideal, sin embargo, la implicación probable de este enumerativo proceso, es la búsqueda intensificada con construcción de soluciones "totalmente balanceadas" con un gran costo computacional. En este escenario, la alternativa práctica es una restricción "blanda", y significa el balanceo como un objetivo adicional que beneficia el tiempo de cómputo penalizando el balanceo de la solución [5].

El modelo de optimización utiliza las siguientes definiciones dadas en [3]:

Definición 1 (Compacidad). *Si denotamos por $Z = 1, 2, ..., n$ al conjunto de n objetos a clasificar, se trata de dividir Z en k grupos $G_1, G_2, ..., G_k$ con $k < n$ de tal forma que:*

$$\begin{aligned}&\bigcup_{i=1}^{k} G_i = Z;\\ &G_i \cap G_j = \emptyset, \qquad i \neq j;\\ &|G_i| \geq 1, \qquad i = 1, 2, ..., k.\end{aligned}$$

Un grupo G_m con $|G_m| > 1$ es compacto si para cada objeto $t \in G_m$ cumple:

$$min_{i \in G_m} d(t, i) < min_{j \in Z - G_m} d(t, j), i \neq t. \tag{3.1}$$

Un grupo G_m con $|G_m| = 1$ es compacto si su objeto t cumple:

$$min_{i \in Z - t} d(t, i) > min_{j, l \in G_f} d(j, l), \qquad \forall f \neq m.$$

El criterio de vecindad entre objetos para lograr compacidad está dado por los pares de distancias descritos en la Ecuación 3.1.

Definición 2 (Homogeneidad (cardinalidad en los elementos)). *Sea $T_i = |G_i|$ para $i = 1, 2, \ldots, k$ y dado un porcentaje de tolerancia de homogeneidad $p \in [0, 1]$ se obtienen dos cotas: la cota inferior $I = \lfloor n/k \rfloor - \lceil n/k * p \rceil$ y la cota superior $S = \lfloor n/k \rfloor + \lceil n/k * p \rceil$ donde n es el número de unidades geográficas y k, el número de grupos a formar. De aquí que una solución no es homogénea cuando:*

$$\exists T_i \,(T_i \notin [I, S]), \qquad para\ i = 1, ...k. \tag{3.2}$$

3.1.1 Modelo para agrupamiento geográfico

Sea Ugt el número total de las unidades geográfica a agrupar (ugs). Ugt es de tamaño n tal que $Ugt = \{x_1, x_2, \ldots, x_n\}$, donde x_i es la i-ésima unidad geográfica y desde luego i, indica el índice en el conjunto Ugt. El número de zonas (grupos) es denotado por k. El objetivo es formar grupos y para referirnos a estos, definimos a Z_l como el conjunto de todas las ugs o bien Ugt, las cuales pertenecen a la zona l. El centroide es expresado como C_t, y $d(i, j)$ es la distancia euclidiana del nodo i al nodo j, es decir, de una unidad geográfica a otra. Entonces se tienen como restricciones: $Z_l \neq \emptyset$ para $l = 1, ..., k$ (los grupos no son vacíos), $Z_l \cap Z_m = \emptyset$ para $l \neq m$ (no existen ugs repetidas en distintos grupos), y $\bigcup_{l=1}^{k} Z_l = Ugt$ (la unión de todos los grupos es todos las ugs).

Una vez que se ha decidido el número k de centroides C_t, para $t = 1, ..., k$, a utilizar, hay que seleccionarlos en forma aleatoria y enseguida asignar las Ugt a los centroides de la manera tradicional como la función objetivo del clustering:

$$\min_{t=1,\ldots,k}\{d(i,c_t)\}.$$

Cada ug es asignada al centroide más cercano c_t.

El costo de homogeneidad de la solución está definido por la siguiente función:

$$h(t) = \begin{cases} T - Z & \text{si } T > Z, \\ I - T & \text{si } T < I, \\ 0 & \text{en otro caso.} \end{cases}$$

Para cada valor de k se calcula la suma de las distancias de las ugs asignadas a cada centroide y también se suman los elementos excedentes o faltantes de cada grupo con respecto a los valores dados por la cota inferior y superior. Estos valores son ponderados con pesos w_1 y w_2 tal que $w_1 + w_2 = 1$ y finalmente estos valores ponderados se suman. Se minimiza este valor durante nit iteraciones. Esto puede expresarse como:

$$\min_{l=1,\ldots,nit}\{w_1(\min\{\sum_{t=1}^{k}\sum_{i\in c_t}\{d(i,c_t)\}) + w_2(\sum_{j=1}^{k}h(T_j))\}\}. \tag{3.3}$$

4 Propuesta de búsqueda tabú

Situándonos en el modelo de la sección 3, la ecuación 3.2 expresa la función biobjetivo, aproximada con búsqueda tabú.

4.1 Aspectos básicos

Los orígenes de la búsqueda tabú (*"tabu search"*, TS) pueden hallarse en diversos trabajos publicados a finales de los 70. Oficialmente, el nombre y la metodología fueron introducidos posteriormente por Fred Glover y Manuel Laguna en 1989 en el libro homónimo *"Tabu Search"* [6].

La filosofía de TS es derivar y explotar una colección de principios inteligentes para la resolución de problemas. En este sentido, puede decirse que la TS está basada en selectos conceptos que unen los campos de la inteligencia artificial y la optimización.

En este apartado se expresan conceptos básicos del diseño territorial, pero en la subsección 4.3 se presenta el algoritmo.

TS siendo una metaheurística que guía a una heurística local de búsqueda para explorar el espacio de soluciones más allá de los óptimos locales. El procedimiento local es una búsqueda que utiliza una operación de movimiento sobre la solución actual para definir su vecindario. Uno de los componentes principales de TS es el uso de memoria adaptativa, la cual crea un comportamiento más flexible de búsqueda. Las estrategias basadas en memoria son el sello distintivo de los enfoques de búsqueda tabú.

TS se preocupa por imponer restricciones para guiar un proceso de búsqueda para negociar regiones que de otra manera serían difíciles de acceder. Las restricciones son impuestas o creadas al hacer referencia a estructuras de memoria, las cuales están diseñadas para este propósito en específico. Por otra parte, TS está basada en la premisa de que la resolución de problemas considera estrategias inteligentes manejando memoria adaptativa y exploración responsiva.

TS contrasta con las metaheurísticas sin memoria en que las últimas dependen fuertemente de procesos semi-aleatorios que implementan una especie de muestreo. Ejemplos de métodos sin memoria son aquellos inspirados por metáforas de la física y biología y heurísticas *"semi-greedy"* (ver [6] y [9]).

Diversas aplicaciones han hecho uso de TS para lograr buena calidad en sus soluciones, en particular, nuestro interés se ha centrado en mirar aquellos trabajos que revelan buenos resultados de TS en problemas de agrupamiento. Como caso particular, en [9] se propone un enfoque de búsqueda tabú modificado que consiste en dos fases: la fase constructiva en la cual se genera la solución inicial utilizando el algoritmo k-medias y una fase de mejora, donde se emplea una búsqueda tabú modificada con el objetivo de mejorar la solución de la fase constructiva. Además logran mejores soluciones que el algoritmo k-medias y que el de búsqueda tabú estándar, sin embargo sólo está enfocado a la minimización de distancias, esto quiere decir que logra la formación de grupos compactos pero no considera restricciones de homogeneidad ni de ningún otro tipo. Es en este escenario donde reside la importancia de nuestro trabajo: el algoritmo de agrupamiento extrae las principales propiedades de k medoides con un especial énfasis en PAM ([10] y [11]) y para el problema que tratamos logramos soluciones de buena calidad a un costo de cómputo razonable.

4.2 Estructuras de datos

La primera y principal estructura de datos es un arreglo de tamaño inicial k (número de grupos a formar) en el cual se almacenan los centroides de cada grupo. Para la búsqueda tabú es necesario definir una lista donde los centroides tabú residan. Uno de estos que recientemente reemplazó a otro de la solución es considerado como tabú-activo durante cierto número de iteraciones, de esta manera se prohíbe que este se reemplace en etapas tempranas. En la Figura 2 se puede ver la estructura de estos centroides.

El tamaño de esta lista es dinámico pero su tamaño máximo es $m = k - 1$ (ya que debe contarse con al menos un centroide que pueda ser reemplazado) y en todo momento la suma de los centroides reemplazables y los tabúes es k. Para ello,

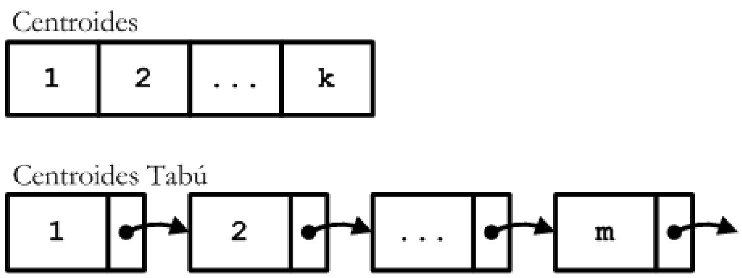

Figura 2: Estructura de Centroides

se definieron dos estructuras comprometidas: los que se pueden reemplazar y los que no.

Las unidades geográficas *ugs*, en nuestro caso AGEBS, se almacenan en un arreglo de tamaño inicial n (cantidad de unidades geográficas) y para la búsqueda tabú se define una lista en la que se almacenan las *ugs* tabú (ver Figura 3). Un centroide que recientemente fue reemplazado por uno nuevo (antes una *ug*) pasa a ser tabú-activo durante cierto número de iteraciones para prohibir que forme parte de la solución nuevamente y muy pronto. De esta manera la búsqueda se diversifica evitando volver a visitar soluciones antiguas.

En general, la lista tabú puede contener para este problema soluciones visitadas recientemente o últimos movimientos realizados que incluyen características de las soluciones ya visitadas. El tamaño de la lista tabú (*"tabu tenure"*), se determina por el tiempo o número de iteraciones que un elemento o movimiento permanece en la lista. Es posible que, en algún momento, todos los elementos posean la misma *"tenure"*, entonces se considera que esa es la longitud de la lista tabú, lo cual no siempre sucede, de otro modo, entonces la lista será variable. En particular, el factor tiempo hace que la lista sea dinámica porque van incorporándose los *ugs* y su estancia en la lista no tiene tiempo indefinido, es finito y al concluir su tiempo de aspiración, estos salen. Más aún, la lista de *ugs* tabú es de tamaño dinámico porque su tamaño máximo es $h = n - k$, ya que k *ugs* pasan a formar parte del arreglo de centroides y/o de la lista de centroides tabú. Por otro lado, cuando los elementos de la lista tabú no tienen la misma *"tenure"*, un elemento que se incorporó a la lista tabú antes que otro puede salir después y viceversa.

Con el objetivo de obtener homogeneidad, se requiere un control sobre las *ugs* asignadas a cada centroide. Por lo tanto, se ha incluido una matriz de tamaño $n{-}k{\times}k$ en la implementación, donde cada columna representa un grupo o centroide y cada grupo puede contener un máximo de $(n - k)$ elementos, excluyendo los centroides. En Figura 4 se puede ver la matriz correspondiente.

Además, se define otro arreglo de tamaño k para almacenar el de cada grupo. Inicialmente, cada grupo tiene un tamaño igual a 1, contando al centroide como

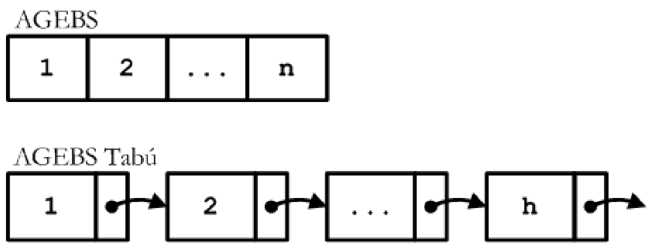

Figura 3: Estructura de Unidades Geográficas

elemento de este. Durante la búsqueda, se actualiza este arreglo cada vez que acepta una solución y se calcula su costo. Esta actualización permite mantener la homogeneidad y el control sobre el tamaño de cada grupo en cada solución generada, ya que es necesario asignar las *ugs* a cada centroide.

Matriz de Grupos

1,1	1,2	...	1,k
2,1	2,2	...	2,k
...	...	...	...
n−k,1	n−k,2	...	n−k,k

Figura 4: Matriz de agrupamiento

4.3 Algoritmo

El siguiente algoritmo ha sido adaptado con TS y con el modelo expuesto en la sección 3. La función que se optimiza está dada por la Ecuación 3.3. A este algoritmo se le conoce a lo largo del artículo como Particionamiento con TS Penalidad (PTSP).

Algoritmo 1. Agrupamiento Geográfico con Búsqueda Tabú (PTSP).

Entrada:
Número de grupos k
Número de iteraciones nit
Número de iteraciones para la segunda fase $nit2$
Número de iteraciones para perturbación ip

1: pc=0
2: ic=1
3: S=$SolucionInicial()$
4: $S*$=S (óptimo elite)

```
5: Mientras ic<nit hacer
6:     Si PenalidadSuperior(S)==0 entonces
7:         modoN=1
8:     si no
9:         modoN=2
10:    fin si
11:    costop=Costo(S) (óptimo actual)
12:    Movimiento(S, modoN)
13:    Si Costo(S)>costop entonces
14:        pc=pc+1
15:    fin si
16:    Si Costo(S)<Costo(S∗) entonces
17:        S∗=S
18:    fin si
19:    Si pc>ip entonces
20:        PerturbarSolucion(S)
21:        pc=0
22:    fin si
23:    ActualizaListasTabu()
24:    ic=ic+1
25: fin mientras
26: S=S∗
27: LimpiaEstadosTabu()
28: Para i=0 hasta nit2 hacer
29:    Movimiento(S, modoN)
30:    Si Costo(S)<Costo(S∗) entonces
31:        S∗=S
32:    fin si
33: ActualizaListasTabu()
34: ic=ic+1
35: fin para
36: Retorna S∗
```

En la línea 1, se inicializa el contador de perturbación en 0. Las líneas 13 y 14 indican que si el costo de la nueva solución es peor que el de la anterior, el contador de perturbación se incrementa en 1. Cuando este contador alcanza el valor máximo dado *ip* se perturba la solución actual en las líneas 19-21. La perturbación consiste en generar una nueva solución aleatoria y reiniciar las listas tabú, y a partir de esta se reiniciará la búsqueda.

En la línea 3 se genera una solución inicial, eligiendo k objetos al azar como centroides de grupo y esta se almacena en S. La línea 4 indica que esta solución pasará a ser la mejor de las encontradas hasta el momento y es representada por $S*$ (óptimo élite).

El primer ciclo de búsqueda comienza en la línea 5, y finaliza cuando la penalidad de la mejor solución encontrada llega a ser igual a 0 o bien cuando el contador de iteraciones ic alcanza el máximo número de iteraciones dado por el usuario. La condición "Si" dentro del ciclo, que se observa entre las líneas 6 a 10, modifica la manera en que se elige el centroide a reemplazar en la función de vecindario. Cuando la penalidad superior es igual a 0, es decir, cuando no hay elementos en los grupos que rebasan la cota superior de homogeneidad (ver el modelo en la sección 3), entonces se cambia el modo de elección de centroide a 1, de esta manera se empiezan a elegir los centroides no tabú cuyo grupo es el de menor tamaño. Si esta condición no se cumple, el modo de elección 2 elige un centroide no tabú cuyo grupo es el de mayor tamaño. Se ha comprobado por medio de experimentación que esta estrategia logra disminuir más rápidamente la penalidad de homogeneidad que la selección aleatoria de centroides.

En la línea 11 se guarda el costo de la solución S antes de realizar el movimiento en la línea 12 sobre la misma. Cuando se realiza el movimiento, en la condición "Si" de las líneas 13-15 se comprueba si el costo de la nueva solución S es mejor que el costo previo, si no lo es, se incrementa el contador de perturbación pc. La condición "Si" de las líneas 16-18 se encarga de actualizar la mejor solución encontrada $S*$ si la nueva solución encontrada S es aún mejor. En las siguientes líneas se actualizan las listas tabú y se incrementa el contador de iteraciones ic.

Por último, una segunda fase de búsqueda sobre la mejor solución encontrada en la primera fase es realizada. Esta búsqueda consiste en vaciar las listas tabú y así dar lugar a movimientos sobre la mejor solución durante un cierto número de iteraciones ($nit2$) con el fin de encontrar una mejor solución que pueda estar cerca de $S*$ (a unos cuantos movimientos de distancia). La solución retornada es la mejor encontrada después de las 2 fases de búsqueda.

4.4 Función de vecindad

En nuestra propuesta algorítmica, los vecinos de la solución actual se obtienen de dos maneras: se selecciona al grupo más pequeño o bien al más grande de la solución actual considerando el costo de penalidad de la solución actual (si existen elementos por debajo de la cota inferior o por encima de la superior). Si el grupo tiene tamaño igual a 1 entonces el centroide del grupo es reemplazado por una unidad geográfica no centroide al azar. Cuando el grupo tiene más de un elemento, el centroide del grupo es reemplazado por la unidad geográfica miembro del mismo grupo tal que este objeto que ahora es centroide, minimiza más el costo actual de la solución. En esta parte, la estrategia básica de la búsqueda tabú comienza; es etiquetado el nuevo centroide como tabú-activo para evitar su reemplazo y el antiguo centroide es también marcado para prevenir su pronto reingreso a la solución actual. El tabú *"tenure"* que hemos establecido es $k-1$ (número de grupos menos uno) que por medio de experimentación muestra el nivel de intensificación necesaria para lograr una aceleración a costos óptimos. Esta manera de obtener soluciones vecinas es una variante de las técnicas de selección más usuales para agrupamiento presentadas en la sección 3 de este artículo.

Es importante aclarar que, a pesar de que se están considerando dos costos, la compacidad se expresa claramente como la minimización de la función objetivo y el cálculo del balanceo se resuelve con la propuesta de penalidad. Aparentemente, se encuentran en contradicción, pero no es así porque una vez generada una solución de compacidad, a esa solución se le calcula el balanceo de los objetos. Finalmente, nuestro método no genera soluciones en competencia bajo el esquema de soluciones no dominadas, lo cual implica que no hemos construido el frente de Pareto y se traduce en un desafío posterior.

5 Experiencia computacional

Las pruebas realizadas utilizaron un hardware con las siguientes características:

- CPU: Dual Core AMD E-350 a 1.6 Ghz.
- RAM: 2GB DDR3.
- HDD: SATA-II 320GB 5400 RPM.
- OS: Windows 7 Ultimate 32bits.

Para efectos de comparación con pruebas anteriores como en [4], se ha considerado al mapa del valle de Toluca, México. Este mapa consta de 469 unidades geográficas y se han realizado las pruebas para formar desde 2 hasta 200 y 300 grupos.

En la Tabla 1 se concentran los resultados de un prototipo basado en PAM (PP) pero no excluye al modelo de optimización propuesto en este artículo. Este prototipo es previo a la más importante aportación de este trabajo. Es importante subrayar que PP no maneja algún método de aproximación, fue desarrollado para este trabajo con el fin de experimentar la dualidad de las funciones que se optimizan. Sin embargo, devolvió resultados importantes, los cuales permanecieron en la última implementación en al menos dos aspectos: 1) Soluciones de excelente calidad se observan para las pruebas realizadas, de tal modo que la mayor parte de las estrategias de programación en este punto son respetadas para la siguiente y última versión. 2) Para pruebas mayores a 200 grupos el costo computacional es elevado y no fue posible registrar los costos de las soluciones. La implicación de este costo computacional en PAM coloco a diseño territorial en nuestro trabajo como el método de aproximación adecuado. Esta nueva propuesta llamada particionamiento con TS penalidad (PTSP), es la contribución principal de nuestro artículo. Los resultados se han almacenado en la Tabla 2.

El primer algoritmo de la Tabla 2 (PDE) corresponde a una propuesta previa para resolver homogeneidad (ver [4]). Este algoritmo minimiza la desviación están-

Tabla 1: Pruebas con PAM Penalidad (PP) de acuerdo al modelo de la sección 3

	PAM Penalidad		
G	Compacidad	Penalidad	Tiempo(s)
2	36.5485	0	0.075
4	27.4569	0	0.222
6	31.4284	0	0.670
8	27.2486	0	1.051
10	19.9424	0	3.042
20	13.6542	0	24.041
40	8.6276	1	114.813
60	7.4026	9	261.200
80	5.3185	17	469.071
100	4.5248	6	860.164
120	3.9991	11	1391.139
140	2.8315	0	1894.436
160	2.8784	11	2249.452
180	2.0487	0	2325.100
200	1.6022	0	2482.393

dar del tamaño de los grupos con respecto al tamaño ideal. En la tabla también puede observarse las pruebas correspondientes a PTSP búsqueda tabú. Un análisis de las tablas deja ver que nuestro modelo de la sección 3 con una nueva medida de homogeneidad basada en minimizar los elementos excedentes, es muy superior con respecto a PAM Penalidad (Tabla 1) como a PDE. La incorporación de búsqueda tabú en dicho modelo, ha sido una interesante estrategia debido a los resultados alcanzados. Es fácil notar en la Tabla 2 que para PTSP, un mejor balance en los elementos de cada grupo se ha obtenido. Por otra parte, PTSP mantiene una mejor compacidad en la mayoría de los casos y rebasa por mucho, el tiempo de cómputo para las instancias más grandes de la tabla.

En la Figura 5 se muestra un resultado gráfico para 10 grupos que corresponde a la prueba de la Tabla 2. Este mapa responde a una interfaz con un sistema de información geográfico (SIG).

Este resultado justifica el ejercicio realizado para las instancias de la Tabla 3: una vez comprobada la eficiencia de PTSP, se ejecutaron 6 pruebas para de distritos electorales del estado de Puebla (2500 objetos).

Para las Figuras 5 y 6 hemos desarrollado una aplicación que genera mapas coloreados considerando los grupos proporcionados. Debido a que esta implementación ha sido realizada sólo para este artículo, no ha sido publicada.

Tabla 2: Pruebas con PAM (Desviación Estándar [4]) y PTSP

	PAM Desviación Estándar (PDE)			Tabu Search Penalidad (PTSP)		
G	Compacidad	Penalidad	Tiempo(s)	Compacidad	Penalidad	Tiempo(s)
2	37.2245	0	0.234	36.3865	0	990
4	30.9555	0	0.190	27.3666	0	2.010
6	29.4603	0	0.610	23.3953	0	3.360
8	24.4460	0	2.645	21.0318	0	4.293
10	17.1270	2	4.101	17.4750	0	5.531
20	13.0818	1	19.277	13.6844	0	24.261
40	7.5209	16	136.340	8.9531	0	89.232
60	5.1136	51	334.054	6.3707	26	93.405
80	3.8886	97	549.179	5.0451	28	202.395
100	3.0497	106	1042.438	3.9835	32	105.953
120	2.5883	103	1464.341	3.4795	39	93.612
140	2.1947	103	2080.419	2.9383	19	241.207
160	1.8786	92	2474.121	2.6919	49	89.398
180	1.6211	69	2566.385	2.5472	29	241.301
200	1.4206	49	2829.424	2.0897	19	79.366
220	1.2410	38	3249.674	1.9316	11	73.661
240	1.0895	100	3380.657	1.7650	58	70.450
260	0.9491	89	2508.559	1.5874	40	140.298
280	0.8077	77	2394.728	1.3897	31	140151
300	0.6807	60	2610.345	1.2878	21	127.297

Tabla 3: Pruebas con Búsqueda Tabú

	Tabu Search Penalidad (PTSP)		
G	Compacidad	Penalidad	Tiempo(s)
4	705.5231	0	732.700
60	215.4017	14	1043.120
100	142.9411	19	985.768
400	62.8906	170	809.757
800	40.4055	179	525.847
1200	28.4216	129	464.946

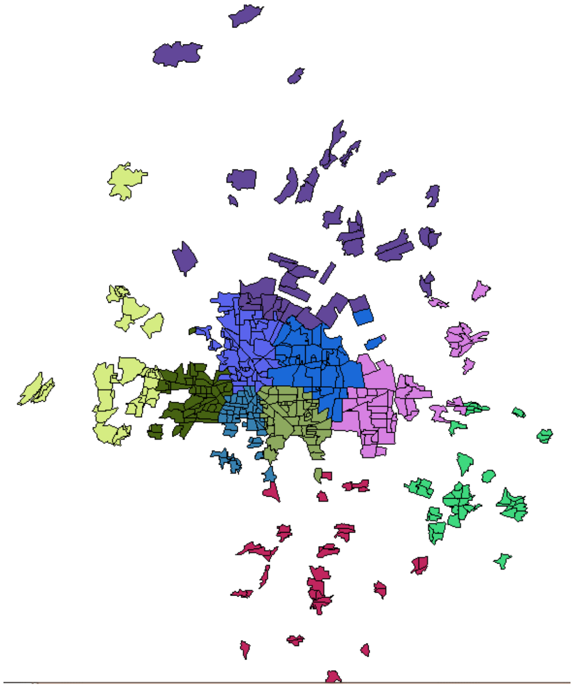

Figura 5: Mapa de Toluca de la prueba obtenida por PTSP en la Tabla 2 para $G = 10$.

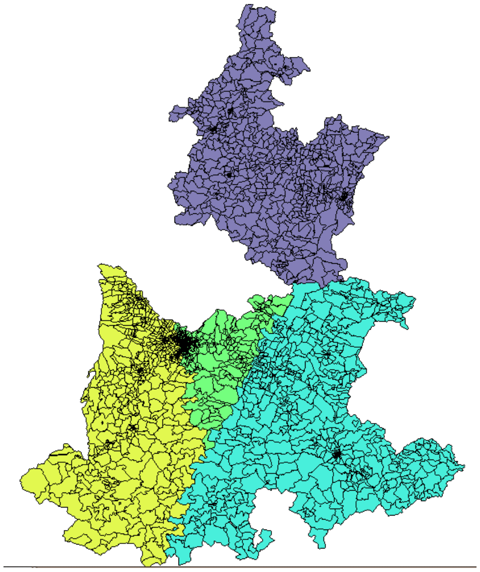

Figura 6: Mapa de Puebla para la prueba obtenida por PTSP en la tabla 3 con G = 4.

6 Conclusiones

Los resultados de la experiencia computacional indican que podemos concluir con seguridad que de los algoritmos presentados, PTSP consigue resultados de mejor calidad en cuanto a costo y tiempo. Destaca que PTSP acepta instancias de gran tamaño que no pueden ser probadas con los algoritmos tradicionales de particionamiento debido al alto costo de cómputo.

Se distingue que nuestra propuesta de homogeneidad en PTSP es de mejor calidad que el algoritmo PAM Penalidad (PP de la Tabla 1) donde el tiempo de cómputo se incrementa muy rápido y exponencialmente mientras que búsqueda tabú en PTSP logra ejecutar alrededor de 20,000 iteraciones sobre el mismo mapa (469 objetos) en un tiempo mucho menor justo por una combinación estratégica de movimientos aleatorios e inteligentes.

Es posible reducir el costo de la penalidad llevando a cabo más iteraciones pero depende del tomador de decisiones si desea obtener una solución rápidamente sacrificando la optimalidad o viceversa.

Como trabajo futuro esperamos mejorar la búsqueda tabú, con nuevas estrategias de selección y búsqueda de soluciones o inclusive un procedimiento constructivo para generar buenas soluciones iniciales. En general con el número de iteraciones utilizado para las pruebas, búsqueda tabú ha logrado mantener una penalidad por abajo del 10 % de las unidades geográficas totales de los mapas, sólo en algunos casos se ha rebasado (Tabla 2, G = 160 y 240).

Nuestra contribución promete lograr mejores resultados para otras instancias y problemas biobjetivo similares relacionados a restricciones de homogeneidad, lo cual se traduce como una de nuestras metas: adaptar y probar el algoritmo para diversos problemas similares.

Agradecimientos

Queremos agradecer totalmente a los evaluadores de la Revista de Matemáticas, Teoría y Aplicaciones por el tiempo, comentarios y valoración que tuvieron en la revisión de nuestro artículo. Su arbitraje dio lugar a que nuestro trabajo mejorara considerablemente en calidad y contenido.

Financiamiento

Este trabajo ha sido financiado y apoyado por la Benemérita Universidad Autónoma de Puebla

Referencias

[1] M. Altman, *The computational complexity of automated redistricting: Is automation the answer*. Rutgers Computer & Tech. LJ **23**(1997), 81.

[2] M. R. Anderberg, *Cluster analysis for applications: probability and mathematical statistics: a series of monographs and textbooks*. Vol. 19. Academic press, 2014.

[3] B. Bernabé Loranca, J. E. Espinosa Rosales, J. Ramírez Rodríguez y M. A. Osorio Lama, *A Statistical comparative analysis of Simulated Annealing and Variable Neighborhood Search for the Geographic Clustering Problem*. Español. Computación y Sistemas (2011).

[4] B. Bernábe-Loranca et al., *Extensions to K-Medoids with Balance Restrictions over the Cardinality of the Partitions*. Journal of Applied Research and Technology **12**(2014), no. 3, 396-408. DOI: 10.1016/S1665-6423(14)71621-9

[5] J. García y J. Maheut, *Modelos de programación lineal: Definición de objetivos*. Modelos y Métodos de Investigación de Operaciones. Procedimientos para Pensar (2011), 42-44.

[6] F. Glover y M. Laguna, *Tabu Search Kluwer*. Handbook of Combinatorial Optimization. Academic Publishers, Boston, MA, 1997. DOI: 10.1007/978-1-4615-6089-0

[7] J. Kalcsics, S. Nickel y M. Schröder, *Towards a unified territorial design approach—applications algorithms and GIS integration*. Top **13**(2005), no. 1. With discussion and a rejoinder by the authors, 1-74. DOI: 10.1007/BF02578982

[8] L. Kaufman y P. J. Rousseeuw, *Finding groups in data: an introduction to cluster analysis*. John Wiley & Sons, 2009. DOI: 10.1002/9780470316801

[9] A. Kharroushe, S. Abdullah y M. Z. Ahmad Nazr, *A modified tabu search approach for the clustering problem*. Journal of Applied Sciences **11**(2011), no. 19, 3447-3453. DOI: 10.3923/jas.2011.3447.3453

[10] S. A. Leiva-Valdebenito y F. J. Torres-Avilés, *Una revisión de los algoritmos de partición más comunes en el análisis de conglomerados: un estudio comparativo*. Revista Colombiana de Estadística **33**(2010), no. 2, 321-339. DOI: 10.15446/rce

[11] J. MacQueen, *Classification and analysis of multivariate observations*. 5th Berkeley Symp. Math. Statist. Probability. University of California Los Angeles LA USA. 1967, 281-297.

[12] D. Romero, J. Burguette Constantino, L. E. Martínez Stiker y J. R. Velasco Ocampo, *Formación de unidades primarias de muestreo*. Boletín de los Sistemas Nacionales Estadístico y de Información Geográfica **2.1**(2006), 42-51.

[13] M. A. Salazar-Aguilar, J. L. González-Velarde y R. Z. Ríos-Mercado, *A divide-and-conquer approach to commercial territory design*. Computación y sistemas **16**(2012), no. 3, 309-320.