Integración de un sistema de información geográfica para algoritmos de particionamiento

María Beatriz Bernábe Loranca, Rogelio González Velázquez
Benemérita Universidad Autónoma de Puebla, Facultad de Ciencias de la Computación, Puebla, Pue. México
beatriz.bernabe@gmail.com

Resumen. Para problemas de Particionamiento Geográfico (PG), se buscan agrupaciones de objetos de acuerdo a las condiciones geográficas. Generalmente, las respuestas del particionamiento geográfico en otros trabajos han sido mostradas textualmente o en un grafo, sin embargo, para problemas donde datos geográficos son los que se agrupan, representar gráficamente las particiones resultantes es un proceso complicado pero necesario. Esto implica que la agrupación use recursos adecuados para este propósito como geometría computacional o herramientas de interfaz con un Sistema de Información Geográfica (SIG).

En este trabajo nos ocupamos de presentar una breve revisión del particionamiento geográfico y el proceso e implementación que hace posible observar resultados de opciones de particionamiento en mapas. Para este propósito se diseñó un conjunto de módulos que se comunican con un SIG. Este proceso suele iniciarse con la selección de datos que continua con la escogencia de un algoritmo de particionamiento geográfico en distintas categorías (compacto, homogéneo para variables poblacionales, P-mediana, multiobjetivo, Relajación Lagrangeana, homogéneo en el número de objetos, etc.). El resultado del particionamiento genera archivos de salida, sin embargo, el sistema acepta un documento de texto compuesto de una lista con los objetos gráficos y la relación al grupo que pertenecen. El procedimiento final está constituido de una interfaz con un SIG con el fin de distinguirlos resultados de las diferentes agrupaciones en mapas. A este sistema le hemos llamado Sistema de Interfaz Gráfico para Particionamiento (SIGP).

Palabras clave: Particionamiento, Sistema de Información Geográfica (SIG).

1. Introducción

Una breve discusión de Particionamiento es necesaria en este trabajo considerando que SIGP resuelve el problema de generar el mapa asociado a distintas opciones de agrupamiento. La importancia del sistema gráfico que se ha desarrollado se sitúa en la explotación de las herramientas para desarrolladores de SIG con el propósito de construir un sistema capaz de mostrar resultados de agrupaciones en modo gráfico.

Los datos que el sistema admite son de naturaleza geográfica y como caso de estudio, se han considerado las Agebs (ÁreasGeoestadísticas Básicas).

Se describen aspectos importantes de las estructuras de datos implícitas en la implementación de cada uno de los módulos que permiten la comunicación entre ellos. El objetivo es lograr un deseable desempeño de todos los componentes del sistema SIGP para conseguir que los resultados de particiones se reflejen en un mapa. Por último, el sistema se integra de 3 pasos que funciona a nivel usuario de la siguiente manera:

1: Se elige la entidad federativa de interés, cada entidad está dividida en un número de zonas geográficas Agebs. En este paso, el usuario puede separar de la entidad un subconjunto de Agebs que satisfagan características deseables para un problema específico. Un subsistema de consulta para selección de variables ha sido implementado para este propósito [1].

2: Particionar las Agebs con: a) Compacidad con Recocido Simulado(RS), b) Compacidad con Búsqueda por Entorno Variable(VNS, por sus siglas en ingles Variable NeighborhoodSearch) [2], c) Homogeneidad en el número de grupos[3], d) Homogeneidad en variables[4], e) Particionamiento multiobjetivo [4], P-mediana[5], f) Particionamiento Alrededor de los Medoides (PAM), g) Relajación Lagrangeana (RL) [5].

3: Mostrar las particiones gráficamente en mapas.

De acuerdo a lo anterior el presente trabajo se encuentra organizado como sigue: Introducción como sección 1. En la sección 2 se presentan algunos puntos de los métodos de agrupamiento. En la sección 3 se expone el desarrollo del sistema gráfico y en la sección 4 se presentan algunos resultados experimentales. Finalmente en la sección 6 se exponen conclusiones y trabajo futuro.

2. Parte experimental

Motivados por su amplia gama de aplicaciones, distintos trabajos han desarrollado técnicas para agrupar datos de diferentes tipos. Especial atención ha merecido el Particionamiento.

2.1. Agregación

Un término significativo para agrupar datos espaciales, es la agregación, la cual es citada cuando a agrupación con restricciones de compacidad geométrica se refiere. La agregación, no es más que un caso particular de cluster donde debe asegurarse la continuidad geográfica entre los elementos agrupados. Este asunto especial de análisis cluster es llamado generalmente análisis cluster con restricción de continuidad espacial.

Los métodos de clasificación usan generalmente una noción de proximidad entre grupos de elementos, para medir la separación entre las clases que se buscan. Se introduce el concepto de agregación, entendida como una disimilitud entre grupos de individuos:

Sean $A, B \subset \Omega$, entonces la agregación entre A y B es $\delta(A,B)$, tal que δ es una disimilitud en el conjunto de partes $\mathrm{P}(\Omega)$:i) $\delta(A,A) = 0$ para todo $A \in \mathrm{P}(\Omega)$

y ii) $\delta(A,B)=\delta(B,A)$ para todo $A,B\in \mathrm{P}(\Omega)$ .Usualmente, la medida de agregación está basada en la disimilitud d medida sobre Ω [6].

2.2. Métodos clásicos de Particionamiento

En los métodos de Particionamiento, se busca una única partición de los objetos en estudio en k clases disjuntas. En la clasificación por Particionamiento se tiene que siendo $\{x_1,x_2,..,x_n\}$ el conjunto finito de n objetos a clasificar y $k<n$ el número de clases en los cuales que se desea clasificar a los objetos. Una partición $P=\{C_1,..,C_k\}$ de Ω en k clases $C_1,..,C_k$ está caracterizada por las siguientes 2 condiciones: 1) $C_i\cap C_j=\varnothing$ 2) $\Omega=\bigcup_{i=1}^{k} C_i$ $i\neq j$. Es posible permitir eventualmente que algunas de las clases C_i sea vacía, de manera que en realidad las particiones $P=\{C_1,..,C_k\}$ que se consideran son particiones Ω en k o menos clases. Sin embargo, las particiones óptimas de acuerdo al criterio de inercia contienen exactamente k clases no vacías [6].Generalmente, la clasificación por particiones es planteada como un problema de optimización. Esto es, dado un conjunto de n objetos denotado por $X=\{x_1,x_2,..,x_n\}$ en que $x_i\in R^D$, sea K un número entero positivo conocido apriori, el problema del clustering consiste en encontrar una partición $P=\{C_1,..,C_k\}$ de X, siendo C_j un conglomerado conformado por objetos similares, satisfaciendo una función objetivo $f:R^D\to R$ y las condiciones: $C_i\cap C_j=\varnothing$ C para i ≠ j, y $C_i\cup C_j=X$.

Para medir la similitud entre dos objetos x_a y x_b se usa una función de distancia denotada por $d(x_a,x_b)$, siendo la distancia euclidiana la más usada para medir la similaridad. Así la distancia entre dos diferentes elementos $x_i=(x_{i1},.,x_{iD})$ y $x_j=(x_{j1},.,x_{jD})$ es $d(x_i,x_j)=\sqrt{\sum_{l-1}^{D}(X_{il}-X_{jl})^2}$.

Los objetos de un conglomerado son similares cuando las distancias entre ellos es mínima; esto permite formular la función objetivo f , como

$$\sum_{j=1}^{k}\sum_{x_i\in C_j} d(x_i,\overline{x_j})^2 \qquad (1),$$

es decir, se desea minimizar (1), donde x_j conocido como elemento representativo del conglomerado, es la media de los elementos del conglomerado C_j ,

$$x_j = \frac{1}{|C_j|} \sum X_i \in C_j \quad (2)$$ y corresponde al centro del conglomerado.

Bajo esas características, el clustering es un problema de optimización combinatoria, y ha sido demostrado que es un NP-difícil [7]. Dada la naturaleza combinatoria de este problema, su resolución requiere del uso de métodos aproximados, lo cual justifica el uso e incorporación de heurísticas [2, 3, 4, 5].

3. Resultados

Algunos problemas de optimización combinatoria, requieren resolver clasificación por particiones. Otras aplicaciones demandan particiones que respeten la compacidad geométrica y/o homogeneidad para variables o balanceo en el número de objetos que conforman los grupos. En estas propuestas, las condiciones espaciales de las variables geográficas en la clasificación favorecen la creación de regiones espacialmente compactas, lo cual se traduce en la satisfacción de la restricción de continuidad geográfica. Por otra parte, la utilización de este tipo de aproximaciones implica, en algunos casos, otorgar gran importancia a las variables geográficas para garantizar la satisfacción de la restricción de continuidad geográfica. Sin embargo, esto supone que el papel de las variables no geográficas (por ejemplo variables socioeconómicas) dentro del proceso de agregación pasaría a ser secundario, aun así, este tipo de variables es muy importantes cuando la agregación resuelve problemas como equilibrio entre variables, homogeneidad o balanceo en el número de grupos.

Es posible representar un esquema funcional del sistema SIGP en un diagrama. Supóngase que solo 2 opciones de clasificación están disponibles: grupos compactos y grupos homogéneos. En la figura 1, se puede observar que cada uno de los módulos funciona independiente o en combinación con otros módulos. Destaca una propiedad interesante de la estructura del sistema: la posibilidad de que estos módulos sean trasladados a otras aplicaciones o agregar otros módulos. En la figura 2, el módulo de selección de datos dentro del módulo de consultas entrega un subconjunto de Agebs de interés, de tal modo que genera una matriz de distancias Euclídeas ajustada para ser procesada por las disponibles opciones de agrupamiento.

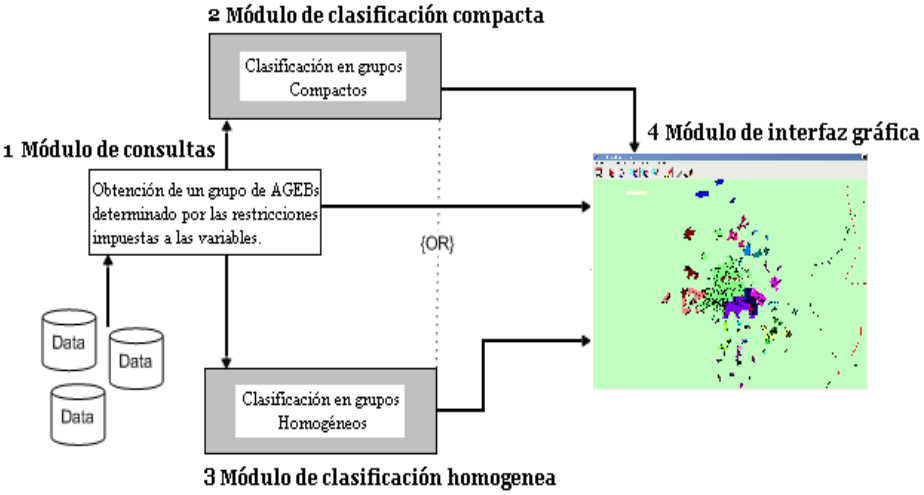
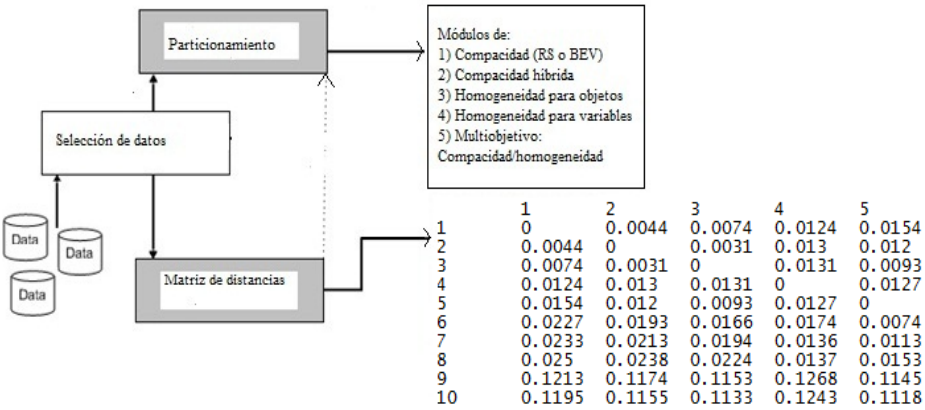


Fig. 1. SIGP básico



	1	2	3	4	5
1	0	0.0044	0.0074	0.0124	0.0154
2	0.0044	0	0.0031	0.013	0.012
3	0.0074	0.0031	0	0.0131	0.0093
4	0.0124	0.013	0.0131	0	0.0127
5	0.0154	0.012	0.0093	0.0127	0
6	0.0227	0.0193	0.0166	0.0174	0.0074
7	0.0233	0.0213	0.0194	0.0136	0.0113
8	0.025	0.0238	0.0224	0.0137	0.0153
9	0.1213	0.1174	0.1153	0.1268	0.1145
10	0.1195	0.1155	0.1133	0.1243	0.1118

Fig. 2. Matriz de distancias*

3.1. Módulo de clasificación compacta

El módulo de clasificación se describe en pseudocódigo para que pueda distinguirse las estructuras de datos de cada componente y el enlace que permite la comunicación entre estos. El módulo de clase *CGroup* es independiente y contiene dos archivos: *modCompactness* y *modVarDt*. El componente dependiente es una instancia de esta clase llamada *compactenessGroup*. El archivo *modVarDt* define dos estructuras de datos utilizadas para la entrada y salida de este módulo:

1) *PublicTypeTItemClusters* *y* *2) PublicTypeTClusters*

```
agebKey As Integer                    n As Integer
cluster As Integer                    nClusters As Integer
End Type                              ítem () As TItemClusters
                                      End Type
```

Como parámetro de entrada, el modulo Clasificación requiere del método *obtain-Compactness*, el cual, almacena los datos que se agruparán de dos maneras: 1)Con una variable de tipo *TCluster* que indica el número de Agebs que se particionan (*n*). El número de grupos a obtener (*nClusters*) se realiza con un arreglo de tipo *TItem-Cluster* para guardar las claves de los Agebs (*agebKey*) además del cluster al que pertenece (*cluster*) y 2) Mediante un apuntador a la base de datos que almacena los Agebs a clasificar y las variables asociadas que denotan el nombre del campo que señalan las Agebs (además de los nombres de los campos con las distancias entre los Agebs y los nombres de las tablas utilizadas). El resultado de la clasificación es recogida en una variable de tipo *TClusters* (ver figura3).

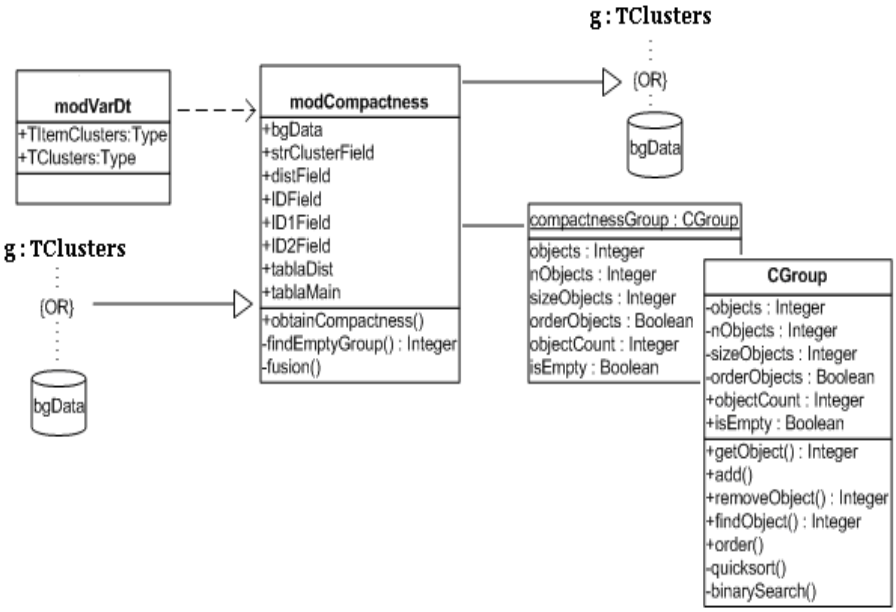


Fig. 3. Módulo de clasificación en grupos compactos

3.2. Módulo de clasificación homogénea

Este módulo reutiliza los componentes *modVarDt* y *CGroup* que maneja el módulo de clasificación compacta y definen una plantilla que puede ser usada en otras rutinas. Un archivo módulo *modHomogeneity* produce una instancia de la clase *CGroup* y utiliza la definición de las estructuras de datos *TItemClusters* y *TClusters* para el control de datos de entrada-salida (ver figura 4).

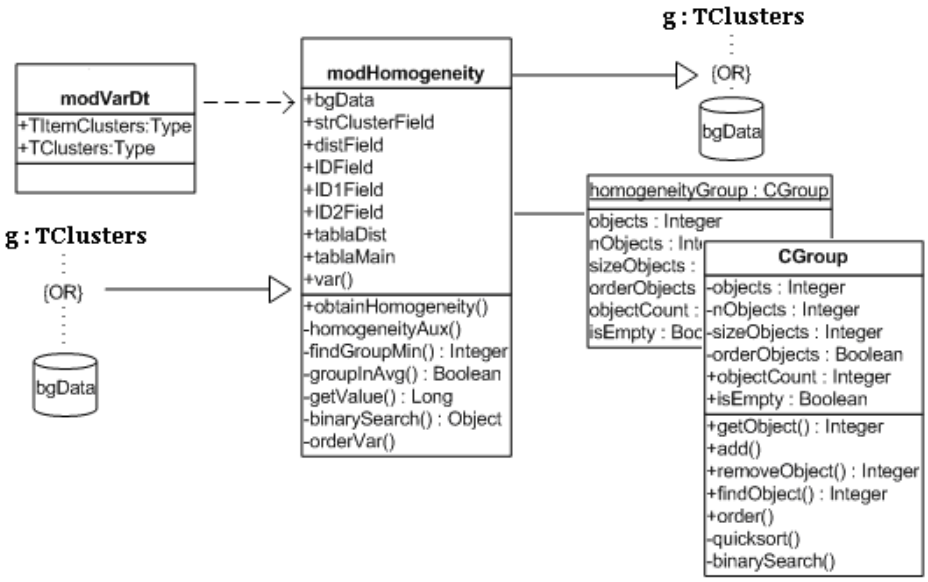


Fig. 4. Módulo de clasificación en grupos homogéneos

3.3. Módulo de interfaz gráfica

En general, los algoritmos de Particionamiento que se han desarrollado, generan tres archivos de resultados [2, 3, 4, 5]. El archivo que nos interesa es el código de Ageb con el número de grupo que le corresponde (formato de archivo para la interacción con MapX) [8]. Las bondades de este formato aseguran que un mapa es generado aun cuando el agrupamiento haya sido implementado con otro lenguaje y el archivo de entrada respete propiedades del formato (ver figura 5).

Los mapas pueden tener diferentes formatos y la extensión *tab* para capas de mapas es aceptada por el SIG MapInfo. Dependiendo de las capas (de población, mares ríos, carreteras etc.), es posible implementar procedimientos que habiliten funciones de componentes del SIG en combinación con el lenguaje visual. La capa que hemos utilizado consiste en la división geográfica de Agebs.

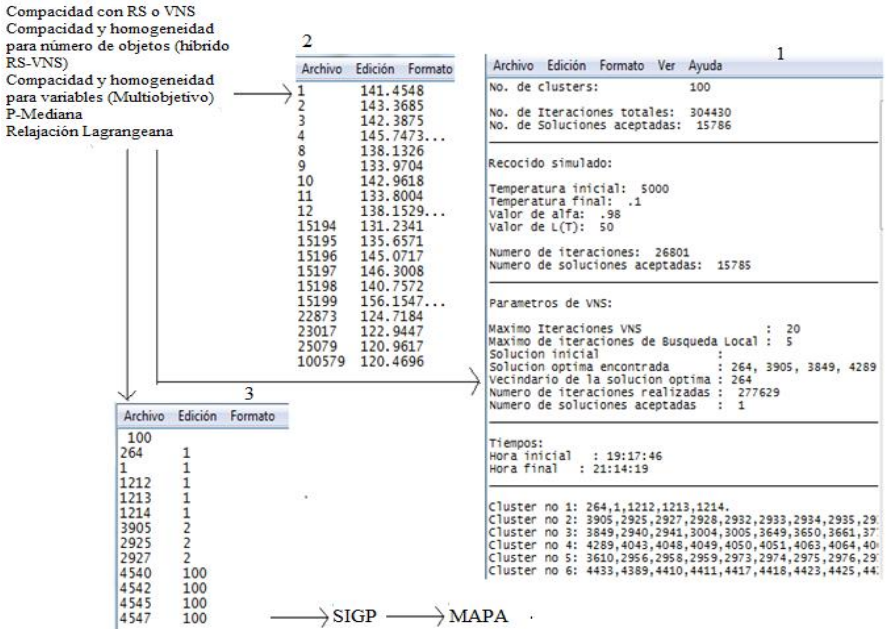


Fig. 5. Diagrama de flujo de datos

En este punto, MapInfo cuenta con una herramienta para el control y administración de las capas (layers). Este control se conoce como MapX. Un análisis exhaustivo de MapX ha sido necesario para explotar sus funciones y propiedades con el fin de construir la interfaz particionamiento-mapa para el SIGP. MapX es un componente ActiveX que permite integrar la funcionalidad de MapInfo en distintas aplicaciones. Se integra usando lenguajes de programación estándar: Visual Basic, Visual C++, Delphi, PowerBuilder y Oracle Express Objects. Desde luego estas características determinaron la elección de MapX.

Los componentes fundamentales de este módulo central son dos "formas" y 3 módulos: 1) *frmMapXInterfaz*, 2) *frmClusterProperties*, 3) un módulo de clase *Color* y 4) dos archivos de modulo llamados *modFuncMapymodMap*. El módulo *Color* y *modFuncMap* son independientes del módulo central, pero la instancia de la clase *Color* si es dependiente del módulo. El módulo de interfaz gráfica, establece un llamado a la forma *frmMapXInterfaz*. Obedeciendo a los valores de las variables que son enviadas como argumentos, se colorean determinadas zonas de un estado para mostrar el mapa correspondiente en una ventana. La figura 6 muestra la interacción entre los componentes de este módulo.

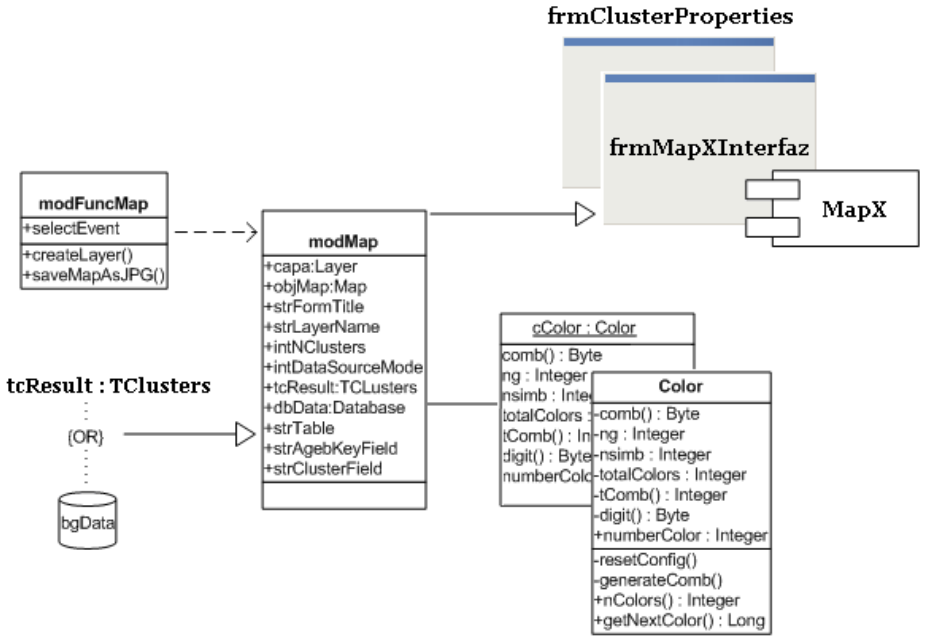


Fig. 6. Módulo de interfaz gráfica.

La forma *frmclusterProperties* ofrece opciones para que el usuario final elija el color de los grupos u ocultar algunos de estos.

El propósito de *modFuncMap* es articular y reunir todas las funciones generales a un control MapX, mientras que *modMap* tiene propiedades y métodos que interactúan sobre la información gráfica mostrada en *frmMapXInterfaz*.

La declaración de los parámetros de entrada, a nivel código, se reduce como sigue:

Public Const DATABASE_MODE As Integer = 1
Public Const TCLUSTER_MODE As Integer = 2
Public formTitle As String
Public layerName As String
Public nClusters As Integer
Public dataSourceMode As Integer
Public tcResult As TClusters
Public dbData As DAO.Database
Public strTable As String
Public strAgebKeyField As String
PublicstrClusterField As String

formTitle. Texto que va a contener la ventana.
layerName. Nombre de la capa (layer) de MapX a ser mostrada.
nClusters. Número de grupos de zonas, por lo que se generaran para los *nCluster*, tonos diferentes para cada grupo utilizando la clase *Color*.

dataSourceMode. Especifica el modo en que se van a introducir los argumentos. Esta estructura cuenta con dos modos y dependiendo del módulo escogido, se especifican los siguientes parámetros:

1. si *datasourceMode* = TCLUSTER_MODE

tcResult. Objeto que contiene los Agebs y el grupo al que pertenece cada uno.

2. si *datasourceMode* = DATABASE_MODE

dbData. Referencia a la base de datos que contiene la información

strTable. Nombre de la tabla en donde se encuentran los datos

strAgebKeyField. Nombre del campo que contiene a las claves de los Agebs

strClusterField. Nombre del campo que contiene la pertenencia de grupo de cada AGEB.

El tipo de dato *TClusters* también es una opción que comunica información de entrada al módulo como se puede apreciar en la figura 6.

Por cada grupo (*nCluster*), un llamado es hecho al método *createLayer* definido en *modFuncMap*. El método *createLayer* elabora una capa (*Layer*) con el color que es dado como argumento (se forma con la clase *Color* para no repetir tonos entre los grupos) y este método la sobrepone en la capa actual (*layerName*). Las operaciones zoom, colocar texto en el mapa, figuras, cambiar el estilo del texto y copiar el mapa al portapapeles son habilitadas por *frmMapXInterfaz,*y se facilitan a través del modelo de objetos de MapX.

3.4. Integración de los módulos

Integrados los módulos principales, la inserción de ventanas (*forms*) es importante para la comunicación con el usuario. Los archivos *modProject* y *modDatos*, han sido implementados para permitir la incorporación de otros módulos de agrupamiento, alcanzando así, una propiedad indispensable de calidad en software: Portabilidad. Por otra parte, *ModProject* y *modDatos* definen variables que registran información en tiempo real de las acciones del usuario y se acompañan de métodos que acceden a las bases de datos. En la figura 7 se observan los llamados entre módulos (línea punteada). La línea sólida especifica entrada y salida, mientras que la línea punteada define las llamadas entre componentes. La palabra "cat." se refiere a la tabla catálogos de la base de datos principal. En esta figura, también se distinguen "módulos separados", los cuales pueden ser reemplazados por otros módulos y generar el mapa correspondiente. Por otra parte, es necesaria la disponibilidad de la capa geográfica de Agebs (censo, ríos, montañas, etc.). En la figura 7 se aprecian distintos módulos de agrupamiento, sin embargo, la primera versión de este sistema disponía de dos opciones: compacidad y homogeneidad. En recientes actualizaciones, se incorporaron al SIGP otros procedimientos de particionamiento con el fin de medir la eficiencia de compatibilidad, flexibilidad, reusabilidad y portabilidad de los procedimientos vistos como módulos. Un reto es reunir todas las opciones de agrupamiento que hemos venido desarrollando en un sistema integrado, además de actualizar la base de datos con todas las entidades federativas de México y ofrecer un sistema robusto capaz de resolver problemas de agrupamiento de muestreo o de zonas para extractos poblacionales e incluso reagrupamiento para fines de distritación básica.

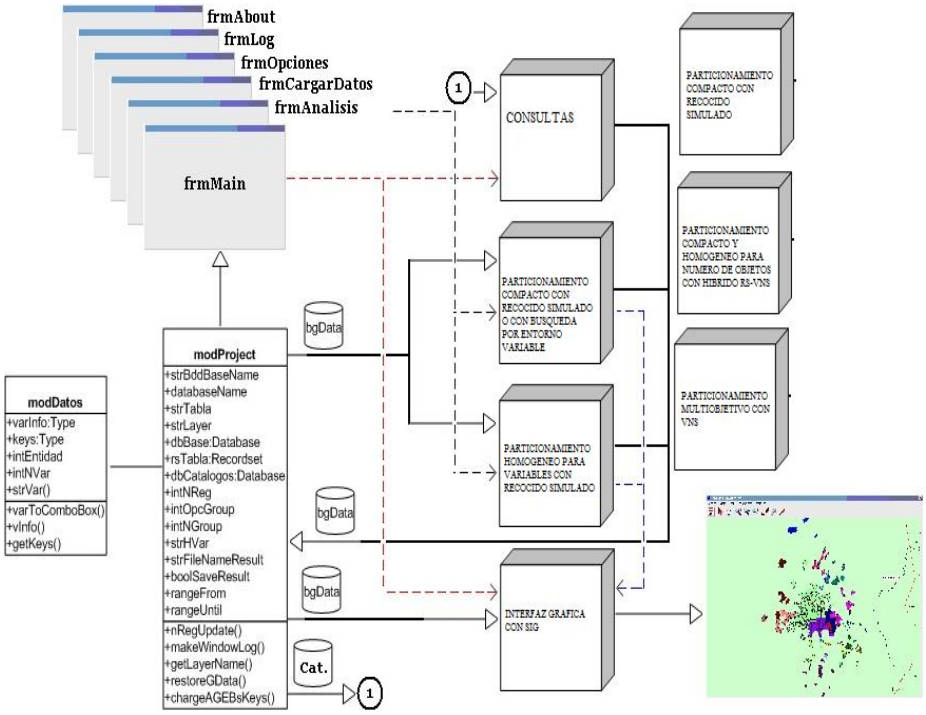


Fig. 7. Integración de todos los módulos en la aplicación final.

4. Discusión de resultados

En esta sección exponemos brevemente 2ejemplos de particionamiento: homogéneo para variables poblacionales y Relajación Lagrangeana RL. Como caso de estudio se ha considerado la Zona Metropolitana del Valle de Toluca (ZMVT), compuesta de 469 Agebs.

Ejemplo 1: Supóngase que serán agrupados los Agebs de la ZMVT, a la consulta *población femenina por encima del promedio* donde la variable *"población total" (Z001)*mantiene homogeneidad. Se requieren 16 grupos. La tabla 1 que muestra el valor que tiene cada grupo para la variable Z001 y el número de elementos que tiene cada grupo y concluimos que la homogeneidad resultante es satisfactoria. La figura 8muestra el mapa para la división de 16 grupos a la consulta población femenina por encima del promedio, manteniendo homogeneidad en la variable *"población total"*.

Grupo	Valor	Elementos
1	62579	14
2	64280	14
3	62842	12
4	62849	15
5	63470	13

6	64880	14
7	65012	12
8	64149	15
9	62547	12
10	63575	13
11	64656	10
12	65166	15
13	64286	15
14	65204	13
15	64250	15
16	64658	14

Tabla 1. Resultados de homogeneidad para el ejemplo 1

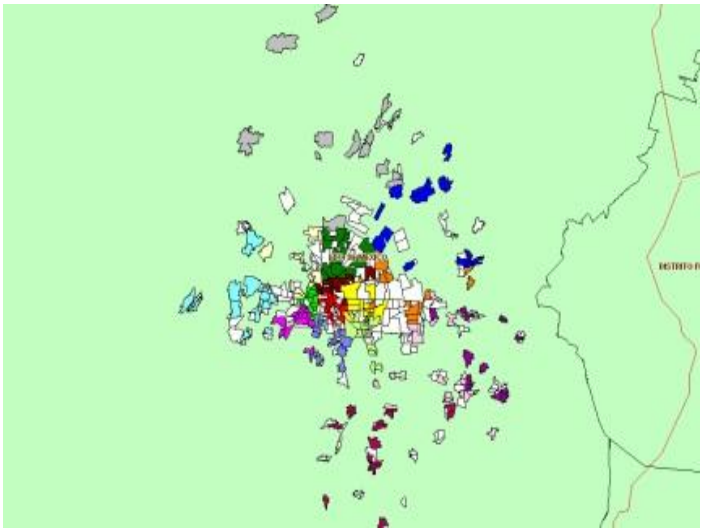


Fig. 8. 16 grupos para *población femenina por encima del promedio*

Ejemplo 2: Se desarrolló un esquema de relajación Lagrangena para el problema de la P-mediana. Para obtener cotas inferiores, se resuelve el dual Lagrangeano utilizando un algoritmo de optimización subgradiente. Este problema se implementó en FicoXpress [5] y las soluciones se han comparado con los resultados de un algoritmo exhaustivo PAM [9]. En la tabla 2 se concentraron resultados para 24, 47 y 94 grupos y en la figura 9 se presenta el mapa para 24 grupos y los resultados del agrupamiento se describieron en el formato que SIGP requiere (ver figura 9).

			FICO XPRESS		PAM
Instancia	Grupos	Solución Óptima	Cota Inferior	Tiempo	Tiempo
			Relajación lineal		
1	24	9.1986	9.1986	42.5	79

2	47	5.7338	5.7338	35.05	431
3	94	3.2089	3.2086	33.26	2188

Tabla 2. Resultados RL para 3 instancias

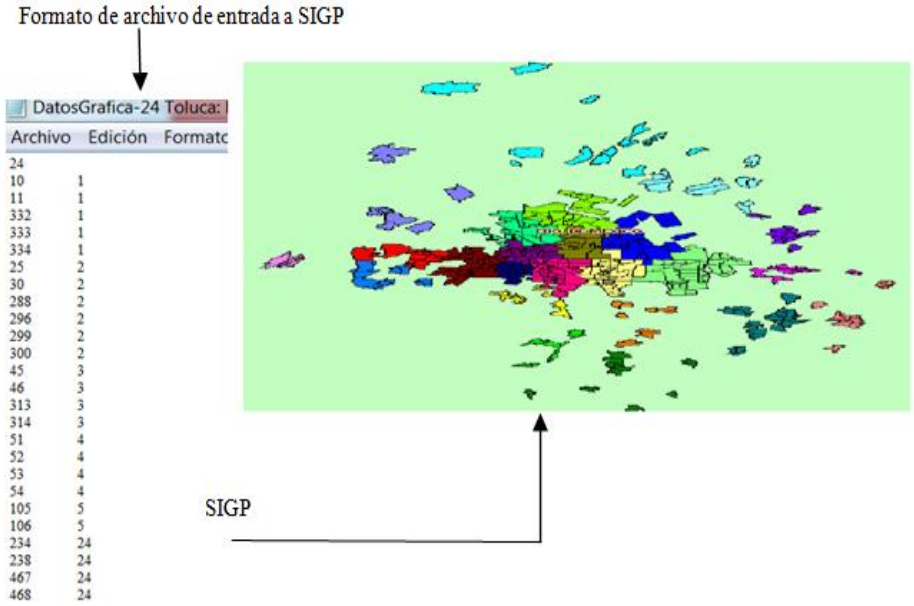


Fig. 9. Resultado de 24 grupos con RL

5. Conclusiones

El trabajo que hemos expuesto significa una importante contribución en el desarrollo de sistemas para problemas de agrupamiento geográfico que requieran resultados en mapas. La mayoría de los trabajos presentan sus resultados concentrados en una tabla con el costo de la función objetivo, el tiempo de cómputo y valor de los parámetros del algoritmo. Sin embargo, las agrupaciones deber ser visibles para asegurarse de que el agrupamiento responde correctamente en cuando a compacidad, conexidad y homogeneidad. El sistema que hemos expuesto en este trabajo responde con claridad los resultados de particiones compactas en un mapa. Subrayamos que el resultado en mapas es posible siempre que se encuentre disponible en Agebs, la capa de una zona a agrupar con formato MapX (extensión tab). La desventaja reside justamente para el caso contrario, cuando la capa de la zona no es extensión tab, el SIGP no puede construir el mapa debido tanto a las especificaciones del sistema como a las propiedades de MapX. Por otra parte, si de Particionamiento compacto se trata, es deseable que las agrupaciones logradas también sean conexas, y se han tenido iniciativas para demostrar analíticamente que los algoritmos que hemos implementado son compactos y

conexos. Atendiendo este aspecto, los resultados vistos en mapas revelan que se cumple conexidad y compacidad gráficamente.

Referencias

1. E. Zamora. Implementación de un Algoritmo Compacto y Homogéneo para la Clasificación de AGEBs bajo una Interfaz Gráfica. Tesis de Ingeniería en Ciencias de la Computación, Benemérita Universidad Autónoma de Puebla, Puebla, México, 18-27, 2006.
2. B. Bernábe, J. Espinosa, J. Ramírez, and M. A. Osorio. Statistical comparative analysis of Simulated Annealing and Variable Neighborhood Search for the Geographical Clustering Problem. *Computación y Sistemas*, vol. 42-3, pp. 295-308, 2009.
3. B. Bernábe, D. Pinto, E. Olivares, J. Vanoye, R. González, J. Martínez. El problema de homogeneidad y compacidad en diseño territorial. XVI CLAIO Congreso Latino-Iberoamericano de Investigación Operativa, 2012.
4. B.Bernábe, C. Coello, M. A. Osorio. A Multiobjective Approach for the Heuristic Optimization of Compactness and Homogeneity in the Optimal Zoning. *JART Journal of Applied Research and Technology*, vol.10-3, pp. 447-457, 2012.
5. J. Díaz., B. Bernábe B., Luna E., Olivares, J. L. Martínez. Relajación Lagrangeana para el problema de particionamiento en datos geográficos. *Revista de Matemática Teoría y Aplicaciones* vol. 19-2, pp. 43-55, 2012.
6. E. Pizza, A. Murillo., &J. Trejos. Nuevas técnicas de particionamiento en clasificación automática. *Revista de Matemática: Teoría y Aplicaciones*, vol. 6-1, pp.1–66, 1999.
7. E. Vicente, L. Rivera, D. Mauricio. Grasp en la resolución del problema de cluster. ISSN: 1815-0268, vol. 2- 2, pp. 16-25, 2005.
8. MapX Developer´s guide. MapInfo Corporation. Troy, NY.
9. L. Kaufman, P. Rousseeuw. Clustering by means of medoids. Statistical Data Analysis based on the L1 Norm, North-Holland, Amsterdam , pp. 405-416, 1987.