

# El Problema de la Dispersión Máxima en un entorno Multi-Objetivo

Mauricio Machuca Cabral, Benjamín Barán, and Fernando Sandoya

Facultad Politécnica  
Universidad Nacional de Asunción,  
Asunción, Paraguay  
School of Natural Sciences and Mathematics  
Escuela Superior Politécnica del Litoral  
Guayaquil, Ecuador  
mmachuca78@gmail.com, bbaran@pol.una.py, fsandoya@espol.edu.ec  
<http://www.pol.una.py>, <http://www.espol.edu.ec>

**Resumen** El problema de la diversidad máxima o dispersión máxima (*MDP - Maximum Diversity Problem*) presenta un gran número de aplicaciones prácticas que surgen de la búsqueda de los elementos más disímiles de un conjunto de datos. Este tipo de problema utiliza modelos de diversidad y definiciones de distancia o disimilitud (observar que en el contexto de este trabajo distancia y disimilitud son considerados sinónimos) como forma de medir que tan diferentes son los elementos de un conjunto dado. Como diversos expertos pueden preferir diferentes definiciones de distancia para problemas específicos, se propone resolver el *MDP* con un enfoque multi-objetivo, considerando por primera vez, la utilización simultánea de múltiples definiciones de distancia. En este nuevo contexto multi-objetivo, este trabajo propone la utilización de un algoritmo evolutivo multi-objetivo (el reconocido NSGA-II), presentando varios casos de prueba que demuestran la eficiencia del algoritmo propuesto en comparación con la búsqueda exhaustiva.

**Keywords:** Maximum Dispersion Problem, Maximum Diversity Problem, MultiObjective Optimization, MultiObjective Evolutionary Algorithm.

## 1. Introducción

El problema de la diversidad máxima o dispersión máxima (observar que en el contexto de este trabajo diversidad y dispersión son sinónimos) consiste en seleccionar un subconjunto  $M = \{i\}, i = 1, \dots, |M|$  de un conjunto  $N = \{j\}, j = 1, \dots, |N|$  ( $|M| < |N|$ ), de forma a maximizar la diversidad entre los elementos del subconjunto  $M$  seleccionado. En los últimos años se han propuesto diferentes modelos para la resolución de este tipo de problema, como por ejemplo:

- *Maximum Diversity Problem (MDP)* [6];
- *Max-Min Diversity Problem (MMDP)* [18];

- *Maximum Mean Dispersion Problem (Max-Mean DP)* [17];
- *Minimum Differential Dispersion Problem (Min-Diff DP)* [17].

Cada uno de estos modelos utiliza una definición diferente para interpretar el concepto de diversidad/dispersión, pero todas estas definiciones tienen la característica común de necesitar para su implementación de una *medida de diversidad* ( $div(M)$ ) a ser aplicada sobre el conjunto de elementos estudiados. En la literatura podemos encontrar varios métodos que se han planteado para la resolución del *MDP* (mono-objetivo), como *GRASP and Path Relinking* [13], *Iterated Local Search* [4], *Variable Neighborhood Search (VNS)* [1], *Iterated Tabu Search (ITS)* [16], *Memetic self-adaptive evolution strategies* [5], *a tabu search based memetic algorithm* [23], *Opposition-based Memetic Search* [24], *Maximum-score Diversity Selection* [15], entre otros.

Las áreas de aplicación son las más diversas, y entre estas se pueden citar: selección eficiente de equipos de trabajo [20], preservación de la diversidad biológica [7], descubrimiento temprano de medicamentos [15], diseño de la estructura molecular, existencias de cría agrícola, composición de jurados [8], ubicación de instalaciones, diseño de productos, genética, sistemas ecológicos, tratamientos médicos y promoción de la diversidad étnica entre inmigrantes [12] por citar solo las aplicaciones más conocidas.

La medida de diversidad o medida de dispersión está relacionada a una definición de distancia, por lo tanto, para resolver el *MDP* necesitamos generar la matriz de distancias inter-elemento, para luego maximizar la diversidad (la definición de distancia utilizada depende en general del campo de aplicación del problema sobre el que se esté trabajando y del experto que lo escoje, es decir, diferentes expertos, utilizan diferentes definiciones de distancia). En este contexto, este trabajo propone resolver el *MDP* multi-objetivo, utilizando varias matrices de distancia (obtenidas cada una de ellas con una definición diferente de distancia) sobre la misma base de datos, convirtiéndose así cada una, en la matriz de entrada que permite calcular cada función objetivo del problema. De entre las meta-heurísticas disponibles para la resolución de problemas multi-objetivo, en este trabajo se utiliza el *Nondominated Sorting Genetic Algorithm II (NSGA-II)* por ser el algoritmo evolutivo más utilizado para resolver problemas multi-objetivo [11]. Este algoritmo es implementado y comparado con un algoritmo de búsqueda exhaustiva (BE) para resolver el *Maximum Diversity Problem (MDP)* en un contexto puramente multi-objetivo.

El presente trabajo está organizado de la siguiente manera: en la Sección 2 se discuten los trabajos relacionados al *Maximum Dispersion Problem*, en la Sección 3 se presentan definiciones de distancia, diversidad y el problema de la diversidad máxima, la Sección 4 presenta el planteamiento y la formulación matemática del problema; la Sección 5 presenta la propuesta de aplicación del *NSGA-II* para resolver el *MDP* multi-objetivo; en la Sección 6 se discuten los resultados experimentales obtenidos, y finalmente, la Sección 7 presenta las conclusiones y líneas de trabajo futuro.

## 2. Trabajos Relacionados

Diferentes heurísticas y metaheurísticas se han presentado en el pasado para el problema de dispersión máxima.

En el trabajo de Martí et al. [13] se resuelve el *Max-Mean Dispersion Problem*, aplicando un *GRASP (Greedy Randomized Adaptive Search Procedure)* con un *Path Relinking* en el cual la búsqueda local fue basada en la metodología *Variable Neighborhood*. Los valores de distancia que consideraron podían tomar tanto valores positivos como negativos, y no necesariamente satisfacer las propiedades usuales de una distancia, como la desigualdad triangular. Los resultados fueron comparados con trabajos previos también basados en *GRASP* demostrando las ventajas del método propuesto.

Sandoya et al. [21] describen modelos matemáticos para los modelos de diversidad y equidad; así como un compendio de heurísticas y meta-heurísticas para los mismos. La mayoría de estas heurísticas y meta-heurísticas, están basadas en métodos de construcción y búsqueda local, como lo son *GRASP* y *Tabu Search*. Dentro de las conclusiones finales se explica que existen implementaciones muy específicas para estos modelos, quedando todavía pendiente, la implementación de soluciones más genéricas.

Zhou et al. [24] presentan un *Opposition-based memetic algorithm (OBMA)* para la resolución del *MDP*. En él, se integraron el concepto de *opposition-based learning (OBL)* con el *memetic search framework*, explorando las soluciones candidatas y sus soluciones opuestas durante los procesos de inicialización y evolución. La diferenciación del *OBMA* con otros algoritmos meméticos radica en las siguientes tres características: un procedimiento de búsqueda de trayectoria doble que simultáneamente busca una solución candidata y su opuesta, la aplicación de *tabu search* para una optimización local efectiva y para mantener la diversidad de la población propusieron un *rank-based quality-and-distance pool updating strategy* el cual utiliza una fórmula de distancia para determinar si la nueva solución generada, debe o no ser agregada a la población.

## 3. Distancias, Diversidad y el Problema de la Diversidad Máxima

El problema de la Diversidad Máxima consiste en seleccionar un subconjunto de elementos de forma que la medida de diversidad del conjunto sea maximizada [10]. Este problema es ya conocido como un problema *NP – Hard* [6].

En Smyth y McClave [22] se define a la diversidad de un conjunto  $M$  como la disimilitud promedio entre todos los pares de elementos. A los efectos de medir dicha diversidad  $div(M)$ , se debe establecer una relación  $d_{ij}$  que represente la distancia o disimilitud entre cada pareja de elementos del conjunto  $M$ . La definición específica de distancia a ser utilizada depende del problema con el que se trabaje y de las consideraciones de cada experto.

Siendo  $d_{ij}$  la distancia entre los elementos  $i$  y  $j$ , y teniendo cada elemento  $K$  atributos, se definen las variables  $x_{ik}$  y  $x_{jk}$  como el  $k$ -ésimo atributo de los

elementos  $i$  y  $j$  respectivamente. Como ejemplo de medida de disimilitud, se presenta la definición de la distancia Euclidiana utilizando la notación arriba presentada.

$$d_{ij} = \sqrt{\sum_{k=1}^K (x_{ik} - x_{jk})^2} \quad (1)$$

Establecida la distancia  $d_{ij}$  entre elementos de un conjunto  $M$ , se puede pasar a definir la medida de diversidad en dicho conjunto. La medida de diversidad utilizada en este trabajo, es la dispersión de la suma, que se define como la suma de las distancias inter-elementos de todos los elementos del conjunto  $M$ . Esta medida es utilizada en el problema de optimización *Max-Sum* [21]:

$$div(M) = \sum_{i < j, i, j \in M} d_{ij} \quad (2)$$

El objetivo del *MDP* es obtener el subconjunto  $M$  más diverso, a través de la selección de los elementos más alejados unos de otros. Sin embargo, esta medida de diversidad tiene la característica de que elementos cercanos pueden quedar en el subconjunto óptimo, si estos se encuentran muy distantes de los demás elementos. En consecuencia, varios autores prefieren evaluar también otros modelos de diversidad como los citados en la Sección 1.

## 4. Planteamiento del Problema

### 4.1. Métricas de distancias consideradas

Cada área que trabaja con un problema específico considerando la diversidad máxima, puede utilizar una definición de distancia diferente. En consecuencia, surge la necesidad de resolver este problema como un problema de optimización multi-objetivo, dado que diferentes investigadores pueden preferir definiciones distintas de distancia (o disimilitud). En <http://www.ug.edu.ec/mdp/bases/>, se presentan 65 definiciones de distancia encontradas en la literatura [2], [14] y [9]. Inicialmente se ha calculado la correlación ( $\rho$ ) entre todos los pares de definiciones relevadas de distancia, y luego, las que presentaron una correlación unitaria ( $\rho = 1$ ) fueron re-clasificadas para este trabajo, quedando 50 definiciones diferentes de disimilitud con correlación de a par diferentes a la unidad, i.e.  $|\rho| < 1$ . En consecuencia, la novedad en este trabajo consiste en permitir la realización de una optimización del problema *MDP* utilizando todas las definiciones de distancia que necesiten los diversos expertos que analizan un mismo problema, por lo que el *MDP* es tratado en este trabajo como un problema con múltiples objetivos.

### 4.2. Formulación del Problema

El problema *MDP* multi-objetivo puede ser planteado como:

$$\text{Maximizar } \vec{div}(M) = [div^1(M) \dots div^p(M)]^T \in \mathbb{R}^p \quad (3)$$

donde la  $l$ -ésima función objetivo  $div^l(M)$  se define como:

$$div^l(M) = \sum_{i < j} d_{ij}^l; \text{ donde } l \in \{1 \dots p\} \quad (4)$$

$d_{ij}^l$  representa la  $l$ -ésima definición de distancia considerada.

## 5. Algoritmo Multiobjetivo para el Problema de la Máxima Diversidad

### 5.1. Búsqueda Exhaustiva

El problema de la diversidad máxima considerado en este trabajo es el *Multi-Objective Max-Sum Dispersion Problem*, definido en (3) y (4).

Por ser la primera vez que se plantea este problema en un contexto puramente multi-objetivo, no se cuenta todavía con un conjunto Pareto conocido. En consecuencia, se ha implementado un algoritmo de *Búsqueda Exhaustiva (BE)* para resolver el *MDP* multi-objetivo.

Si bien la BE puede encontrar la totalidad de los elementos más disímiles de una base de datos dada, queda claro que este algoritmo no representa una solución escalable para problemas *NP Hard*, pues si aumenta el tamaño de la base de datos o la cantidad de objetivos utilizados, el mismo ya no es capaz de realizar toda la búsqueda en tiempo útil, haciendo intratable el problema para la mayoría de los casos prácticos.

### 5.2. Algoritmo Evolutivo

El presente trabajo utiliza el *Nondominated Sorting Genetic Algorithm* versión 2, conocido como NSGA-II [3], un MOEA (*MultiObjective Evolutionary Algorithm*) reconocido como el más representativo y por ende utilizado en el estado del arte [19]. El Algoritmo 1 presenta el pseudo-código básico del NSGA-II para resolver el problema de la Máxima Diversidad, el cual recibe como datos de entrada: una base de datos de  $|N|$  elementos, que conforman el conjunto  $N$ , un vector de definiciones de distancia  $dd$  (para este trabajo se tienen disponibles 50 definiciones distintas de distancia<sup>1</sup>), la cantidad de elementos disímiles a seleccionar  $|M|$ , el tamaño de la población *PopSize*, probabilidad de cruzamiento  $p_c$ , probabilidad de mutación  $p_m$  y la cantidad de iteraciones *iter* a realizar por el algoritmo. Finalmente, este algoritmo retorna el mejor conjunto de soluciones no dominadas  $P$  y el frente Pareto correspondiente  $PF$  al final del proceso evolutivo.

<sup>1</sup> <http://www.ug.edu.ec/mdp/bases/>

---

**Algoritmo 1** Algoritmo Genético NSGA-II para MDP

---

**Entrada:**  $N$  base de datos de elementos,  $dd$  vector de definiciones de distancia (2 a 50 métricas disponibles),  $M$  cantidad de elementos disímiles a seleccionar,  $PopSize$  Tamaño de la población,  $p_c$  probabilidad de cruzamiento,  $p_m$  probabilidad de mutación,  $iter$ : número de iteraciones a realizar.

**Salida:** Solución P conjunto Pareto, PF frente Pareto  $\max(div(M))$ .

```

1:  $i \leftarrow 0$ 
2:  $P_i \leftarrow$  InicializarPoblación( $PopSize$ )
3: EvaluarPoblación( $P_i$ )
4:  $Q_i \leftarrow \emptyset$  [Población Auxiliar]
5: mientras CondiciónDeParada = falso hacer
6:    $Q_i \leftarrow$  CrearNuevaPoblación( $P_i, PopSize$ ) [usar selección, cruzamiento y mutación sobre  $J_i$  para crear una nueva población  $Q_i$  de  $PopSize$  individuos]
7:   EvaluarPoblación( $Q_i$ )
8:    $F_i \leftarrow Q_i \cup P_i$ 
9:    $\mathcal{F}_i \leftarrow$  OrdenamientoNoDominado( $F_i$ )
10:   $j \leftarrow 0$ 
11:  mientras  $|P_{i+1}| < PopSize$  hacer
12:    AsignarDistancia( $\mathcal{F}_{ij}$ )
13:     $P_{i+1} \leftarrow P_{i+1} \cup \mathcal{F}_{ij}$ 
14:     $j \leftarrow j + 1$ 
15:  fin mientras
16:  ORDENAR( $P_{i+1}, \geq_n$ )
17:   $P_{i+1} \leftarrow P_{i+1}[0 : PopSize]$ 
18:   $i \leftarrow i + 1$ 
19: fin mientras
20:  $PF \leftarrow \mathcal{F}_{i1}$ 
21: devolver P, PF

```

---

## 6. Resultados Experimentales

En esta sección se exponen los experimentos realizados. Como no se cuenta con un frente Pareto óptimo conocido, se toman los datos obtenidos por la BE como frente Pareto de referencia, cuando este conjunto puede ser computacionalmente calculado. Se ha trabajado con dos bases de datos a los efectos de buscar los  $|M|$  elementos mas disímiles. Cada una de estas bases de datos consisten en un conjunto de datos numéricos generados aleatoriamente. Las bases de datos utilizadas en los experimentos están disponibles en <http://www.ug.edu.ec/mdp/bases/>. En los experimentos se han utilizado diferentes definiciones de distancia. Los datos que se presentan a continuación corresponden a  $P = 2$  y  $P = 5$  definiciones distintas de distancia, representando cada definición una matriz de entrada diferente que permite calcular una función objetivo del problema. En la Tabla 1, se detallan 4 de las 5 definiciones de distancia utilizadas en los experimentos, dado que la distancia Euclidiana fue definida en (1).

| # | Nombre, Definición y Referencia   | # | Nombre, Definición y Referencia   |
|---|---|---|---|
| 1 | <i>Canberra Distance</i> $\sum_{k=1}^K \frac{ x_{ik} - x_{jk} }{ x_{ik} + x_{jk} }$ [2]     | 3 | <i>Jaccard</i> $\frac{\sum_{j=1}^d (x_{ik} - x_{jk})^2}{\sum_{k=1}^K x_{ik}^2 + \sum_{k=1}^K x_{jk}^2 - \sum_{k=1}^K x_{ik}x_{jk}}$ [2] |
| 2 | <i>Sorensen</i> $\frac{\sum_{k=1}^K  x_{ik} - x_{jk} }{\sum_{k=1}^K  x_{ik} + x_{jk} }$ [2] | 4 | <i>Clark</i> $\sqrt{\sum_{k=1}^K \left( \frac{ x_{ik} - x_{jk} }{x_{ik} + x_{jk}} \right)^2}$ [2]                                       |

Tabla 1: Algunas definiciones de distancia utilizadas

A continuación se presentan tres casos diferentes de pruebas en los que se fue variando el tamaño de la base de datos de entrada, la cantidad de elementos disímiles a seleccionar y la cantidad de objetivos considerados.

Los algoritmos fueron implementados con el lenguaje de programación Java y ejecutados sobre un computador Intel i3 de 1,40 GHz, 12 GB de memoria RAM y sistema operativo Windows 10. El procedimiento básico consistió en tomar la base de datos de entrada y calcular todas las matrices de distancia inter-elemento (tantas como objetivos se necesiten considerar). Una vez obtenidas las matrices de distancia, se ha resuelto el problema de la Máxima Diversidad por Búsqueda Exhaustiva; luego el mismo procedimiento fue resuelto mediante la utilización del NSGA-II. A continuación se explican con más detalles los tres experimentos realizados. Para el experimento 1 se tomó una base de datos de 40 elementos y las definiciones de distancia *Euclidean Distance* y *Canberra Distance* como objetivos. Cuatro corridas fueron realizadas, buscando los 3, 6, 8 y 10 elementos más disímiles. Claramente se puede notar como aumenta significativamente el tiempo de procesamiento de la Búsqueda Exhaustiva en la medida que aumenta la cantidad de elementos disímiles a seleccionar. Para el caso de 10 elementos, se necesitaron casi 3 horas de cómputo para encontrar todas las soluciones. En la Tabla 2 se muestran los resultados obtenidos con la BE.

| BE utilizando 2 objetivos |    |          |         | BE utilizando 5 objetivos |    |          |         | BE utilizando 2 objetivos |    |          |         |
|---------------------------|----|----------|---------|---------------------------|----|----------|---------|---------------------------|----|----------|---------|
| Elem.                     | M  | Cnt.Sol. | t (seg) | Elem.                     | M  | Cnt.Sol. | t (seg) | Elem.                     | M  | Cnt.Sol. | t (seg) |
| 40                        | 3  | 6        | 1,1     | 40                        | 3  | 15       | 1,7     | 100                       | 3  | 18       | 2,4     |
| 40                        | 6  | 23       | 65,5    | 40                        | 6  | 890      | 329     | 100                       | 6  | 35       | 10.826  |
| 40                        | 8  | 31       | 772     | 40                        | 8  | 852      | 7.163   | 100                       | 8  | **       | **      |
| 40                        | 10 | 43       | 10.142  | 40                        | 10 | *        | *       | 100                       | 10 | **       | **      |

Tabla 2: Resultados de la Búsqueda Exhaustiva

En el experimento 2 se tomó la misma base de datos de 40 elementos, pero en vez de dos, se consideraron 5 definiciones de distancia como objetivos (*Euclidean Distance*, *Canberra Distance*, *Sorensen Distance*, *Jaccard Distance* y *Clark Distance*). En este caso, para la selección de los 10 elementos más disímiles ya no se pudieron encontrar todas las soluciones en un tiempo razonable por Búsqueda Exhaustiva (el tiempo de procesamiento fue de más de 7 horas).

Para el experimento 3, se tomó una base de datos de 100 elementos, las definiciones *Euclidean Distance* y *Canberra Distance* como objetivos y se repitieron las corridas para la selección de 3, 6, 8 y 10 elementos. Bajo éstas condiciones, la BE solo pudo encontrar soluciones para  $|M| = 3$  y 6 elementos. Para la búsqueda de 8 y 10 elementos se tuvieron más de 8 horas de procesamiento sin poder encontrar todas las soluciones.

En la segunda parte de los experimentos, se utilizó el Algoritmo NSGA-II para resolver los mismos problemas arriba citados, utilizando  $|M| = 3, 6, 8$  y  $10$ , y considerando 20, 50, 100 y 250 iteraciones. El NSGA-II fue parametrizado de la siguiente manera: popSize (Tamaño de la Población) = 100,  $p_c$  (probabilidad de cruzamiento) = 0,9 y  $p_m$  (probabilidad de mutación) =  $1/|M|$

En la Tabla 3 se muestran los resultados obtenidos por el NSGA-II. En la columna  $t$  (seg) se muestra el tiempo de corrida del NSGA-II para las 250 iteraciones. Se puede notar que para estos casos, el NSGA-II es capaz de conseguir soluciones que se aproximan razonablemente al frente de referencia y en la medida que se aumentan la cantidad de iteraciones, lógicamente se encuentran soluciones más próximas al frente Pareto. En la Figura 1 se muestra la evolución para el experimento 3.

| Instancias del Problema |           |       | BE       |         | NSGA-II             |    |     |     | t (seg) |
|-------------------------|-----------|-------|----------|---------|---------------------|----|-----|-----|---------|
| Cnt.Obj.                | Elementos | $ M $ | Cnt.Sol. | t (seg) | Cnt. de Iteraciones |    |     |     |         |
|                         |           |       |          |         | 20                  | 50 | 100 | 250 |         |
| 2                       | 40        | 3     | 6        | 1,1     | 4                   | 3  | 4   | 5   | 4,29    |
| 2                       | 40        | 6     | 23       | 65,5    | 9                   | 14 | 17  | 21  | 4,81    |
| 2                       | 40        | 8     | 31       | 772     | 16                  | 20 | 21  | 26  | 4,64    |
| 2                       | 40        | 10    | 43       | 10.142  | 10                  | 14 | 30  | 33  | 4,61    |
| 5                       | 40        | 3     | 15       | 1,7     | 3                   | 11 | 14  | 14  | 21,51   |
| 5                       | 40        | 6     | 890      | 329     | 7                   | 33 | 52  | 186 | 19,27   |
| 5                       | 40        | 8     | 852      | 7.163   | 22                  | 48 | 67  | 93  | 17,69   |
| 5                       | 40        | 10    | *        | *       | 37                  | 86 | 83  | 83  | 18,24   |
| 2                       | 100       | 3     | 18       | 2,4     | 8                   | 10 | 13  | 15  | 8,47    |
| 2                       | 100       | 6     | 35       | 10.826  | 11                  | 27 | 22  | 32  | 7,25    |
| 2                       | 100       | 8     | *        | *       | 8                   | 13 | 29  | 37  | 7,58    |
| 2                       | 100       | 10    | *        | *       | 10                  | 16 | 18  | 41  | 8,43    |

Tabla 3: Resultados experimentales del NSGA-II comparados con la Búsqueda Exhaustiva (BE). La columna “Cnt.Sol.” muestra el número de soluciones encontradas por búsqueda exhaustiva mientras que las columnas “Cnt. de Iteraciones” muestran la cantidad de soluciones utilizando el NSGA-II con un número diferente de iteraciones

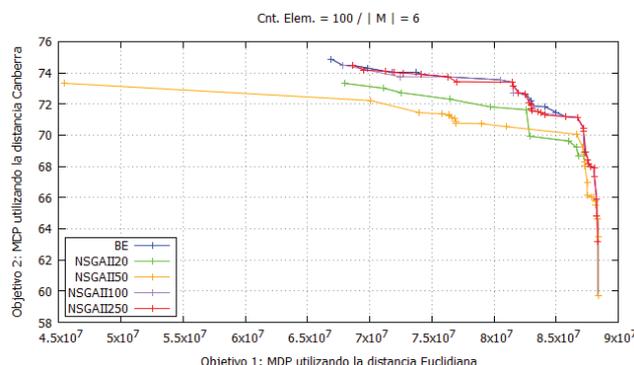


Figura 1: Experimento 3. BE vs NSGA-II con 20, 50, 100 y 250 Iteraciones

## 7. Conclusiones y trabajos futuros

Como se ha mencionado, el problema de la Dispersión Máxima *MDP* resulta relevante para áreas como biología, genética, medicina, para la justicia en la conformación de jurados, en las empresas para la conformación de equipos de trabajo entre otras aplicaciones. Existen varios modelos de diversidad que pueden ser aplicados, y a su vez en cada modelo se pueden utilizar diferentes definiciones de distancia. Dadas las características mencionadas, y como no existe un consenso entre los expertos sobre que definiciones de distancia utilizar, este trabajo propone por primera vez maximizar la diversidad de una base de datos determinada, con tantas definiciones de distancia como lo requieran los diferentes especialistas interesados en un problema. Se han presentado experimentos con bases de datos de 40 y 100 elementos respectivamente, y seleccionado los 3, 6, 8 y 10 elementos más disímiles de cada base de datos, para lo cual se utilizó un algoritmo evolutivo capaz de resolver el problema con tantas definiciones diferentes de distancias como sean necesarias. Como lo demuestran los resultados experimentales presentados, el algoritmo propuesto basado en el NSGA-II resuelve el problema de una manera razonable en tiempos admisibles. Como trabajo futuro se podrán considerar no solamente utilizar diferentes definiciones de distancia, sino también definiciones de diversidad combinadas, así como otros algoritmos evolutivos multi-objetivo, según sea el problema específico y la necesidad de los especialistas, ampliando también así, la visión que puedan tener los mismos para problemas específicos en los que estén trabajando.

## Referencias

1. Brimberg, J., Mladenović, N., Urošević, D., Ngai, E.: Variable neighborhood search for the heaviest  $k$ -subgraph. *Computers & Op. Research* 36(11), 2885–2891 (2009)
2. Cha, S.H.: Comprehensive survey on distance/similarity measures between probability density functions. *City* 1(2), 1 (2007)

3. Deb, K., Agrawal, S., Pratap, A., Meyarivan, T.: A fast elitist non-dominated sorting genetic algorithm for multi-objective optimization: Nsga-ii. In: International Conference on Parallel Problem Solving From Nature. pp. 849–858. Springer (2000)
4. Della Croce, F., Grosso, A., Locatelli, M.: A heuristic approach for the max–min diversity problem based on max-clique. *Comp. & Op. Research* 36(8), 2429–2433 (2009)
5. de Freitas, A.R.R., Guimarães, F.G., Silva, R.C.P., Souza, M.J.F.: Memetic self-adaptive evolution strategies applied to the maximum diversity problem. *Optimization Letters* 8(2), 705–714 (2014)
6. Ghosh, J.B.: Computational aspects of the maximum diversity problem. *Operations research letters* 19(4), 175–181 (1996)
7. Glover, F., Ching-Chung, K., Dhir, K.S.: A discrete optimization model for preserving biological diversity. *Applied mathematical modelling* 19(11), 696–701 (1995)
8. Glover, F., Kuo, C.C., Dhir, K.S.: Heuristic algorithms for the maximum diversity problem. *Journal of information and Optimization Sciences* 19(1), 109–132 (1998)
9. Huang, A.: Similarity measures for text document clustering. In: Proceedings of the sixth new zealand computer science research student conference (NZCSRSC2008), Christchurch, New Zealand. pp. 49–56 (2008)
10. Kuo, C.C., Glover, F., Dhir, K.S.: Analyzing and modeling the maximum diversity problem by zero-one programming. *Decision Sciences* 24(6), 1171–1185 (1993)
11. von Lüken, C., Barán, B., Brizuela, C.: A survey on multi-objective evolutionary algorithms for many-objective problems. *Computational Optimization and Applications* 58(3), 707–756 (2014)
12. Martí, R., Gallego, M., Duarte, A., Pardo, E.G.: Heuristics and metaheuristics for the maximum diversity problem. *Journal of Heuristics* 19(4), 591–615 (2013)
13. Martí, R., Sandoya, F.: Grasp and path relinking for the equitable dispersion problem. *Computers & Operations Research* 40(12), 3091–3099 (2013)
14. McCune, B., Grace, J.B., Urban, D.L.: Analysis of ecological communities, vol. 28. MjM software design Glenden Beach (2002)
15. Meinel, T., Ostermann, C., Berthold, M.: Maximum-score diversity selection for early drug discovery. *J. of chemical information and modeling* 51(2), 237–247 (2011)
16. Palubeckis, G.: Iterated tabu search for the maximum diversity problem. *Applied Mathematics and Computation* 189(1), 371–383 (2007)
17. Prokopyev, O.A., Kong, N., Martinez-Torres, D.L.: The equitable dispersion problem. *European Journal of Operational Research* 197(1), 59–67 (2009)
18. Resende, M.G., Martí, R., Gallego, M., Duarte, A.: Grasp and path relinking for the max–min diversity problem. *Computers & Op. Research* 37(3), 498–508 (2010)
19. Riquelme, N., Von Lüken, C., Baran, B.: Performance metrics in multi-objective optimization. In: Computing Conference (CLEI), 2015. pp. 1–11. IEEE (2015)
20. Sandoya, F., Aceves, R.: Grasp and path relinking to solve the problem of selecting efficient work teams. INTECH Open Access Publisher (2013)
21. Sandoya, F., Martínez-Gavara, A., Aceves, R., Duarte, A., Martí, R.: Diversity and equity models. In: Handbook of Heuristics, pp. 1–20. Springer (2015)
22. Smyth, B., McClave, P.: Similarity vs. diversity. In: International Conference on Case-Based Reasoning. pp. 347–361. Springer (2001)
23. Wang, Y., Hao, J.K., Glover, F., Lü, Z.: A tabu search based memetic algorithm for the maximum diversity problem. *Eng. App. of Artificial Intell.* 27, 103–114 (2014)
24. Zhou, Y., Hao, J.K., Duval, B.: Opposition-based memetic search for the maximum diversity problem. *IEEE Transactions on Evolutionary Computation* (2017)