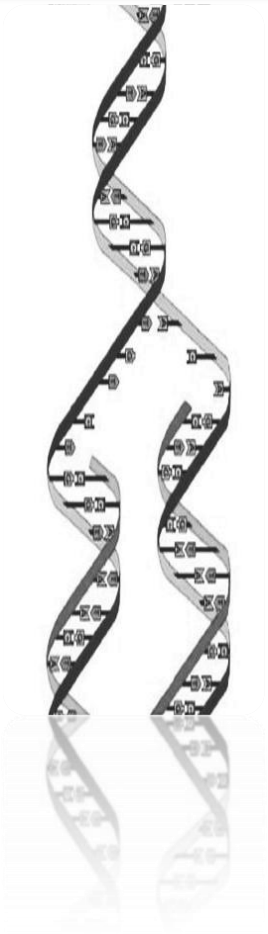
Trabajo de Diploma 2011

Agrupamiento en grafos bipartitos y optimización de parámetros basada en enjambre de partículas, aplicación en la detección de genes ortólogos.



AUTOR: Maray Montes de Oca Labrada

TUTORA: MsC.Deborah Galpert Cañizares

CONSULTANTE: MsC Mario Pupo

"Año 53 de la Revolución"

Santa Clara

2011

Ver después no vale, lo que vale es ver antes y estar preparado.

José Martí.

A mi familia por su guía y apoyo.

A mis amigos por su preocupación.

A mi facultad por la preparación que me brindo.

A mi tutora Deborah Galpert Cañizares por su dedicación.

Al colectivo de trabajo del laboratorio de Bioinformática por sus criterios.

El presente trabajo es una continuidad del trabajo de diploma “Herramientas Computacionales de Comparación de Genomas” donde se construía un grafo bipartito completo a partir de la medida de similaridad local-global entre genes y se implementaba un agrupamiento BUS sobre este grafo para detectar genes ortólogos. En el presente trabajo se realiza un estudio de diferentes técnicas de agrupamiento y co-agrupamiento sobre grafos bipartitos y se aplica la implementación del algoritmo de particionamiento multinivel METIS a la fase de agrupamiento del algoritmo de detección de ortólogos. Inicialmente se realiza una poda por umbral a la matriz del grafo bipartito completo formado para la comparación de los genomas S.Cerevisiae y S.pombe. Luego se aplica el METIS a la matriz podada y sus resultados de agrupamiento se comparan con la base de datos de ortólogos para estas dos especies curada manualmente en el Laboratorio Sanger. Esta comparación se realiza mediante el cálculo de la medida Indice Ajustado de Rand obteniéndose mejores valores que los obtenidos con el algoritmo BUS. La implementación de estos procedimientos se realizó en Matlab 9.0 con el uso del paquete LINKCLUE disponible en Internet.

Con vistas a mejorar la precisión de la detección de ortólogos, se realiza primeramente una estimación de los parámetros para el umbral de la poda minimizando la distancia de Hamming entre la matriz podada y la matriz formada a partir de los resultados de agrupamiento de Sanger. Este procedimiento se realizó en MATLAB implementando la metaheurística bioinspirada en enjambres de partículas PSO. Los resultados obtenidos muestran que elvalor más adecuado para la poda es 80. Seguidamente, se realiza otro proceso de estimación del parámetro de cantidad de grupos iniciales a formar por el METIS, optimizando el Indice Ajustado de Rand en comparación con la base de datos Sanger. Para esto se implementó también el PSO en Matlab y se utilizó la mejor matriz podada en el paso previo. Finalmente, los resultados de calidad obtenidos en el agrupamiento superan los del algoritmo BUS.

This Project is a continuation of the Diploma Paper “Computational Tools for genome comparison” where the authors built a complete bipartite graph from a local-global gene similarity measure. They implemented a BUS clustering algorithm to this graph to detect the orthologs. In our project, we studied some clustering and biclustering techniques for bipartite graphs. We applied the available implementation of the multilevel partitioning algorithm METIS to the grouping phase of the ortholog detection algorithm. First, we run a threshold pruning process to the complete graph of the comparison of the S.Cerevisiae and S.pombe genomes. Next, we applied METIS on the pruned matrix and its results are compared with the ones in the manually curated SANGER database. This comparison was made through the Adjusted Rand Index measure (ARI) with better results than the ones obtained with BUS. The implementation of the procedures was made in Matlab 9.0 with the use the LINKCLUE available package for the ARI calculation.

In order to improve the precision of the ortholog detection we first estimate the value of the prunning process minimizing the Hamming Distance between the pruned matrix and the binary matrix obtained from SANGER database. This process was based on the bio-inspired metaheuristic Particle Swarm Optimization (PSO). The best pruning parameter was 80. The pruned matrix built with this value was used in the next PSO step where we estimate the value of the number of clusters for the METIS algorithm maximizing the ARI measure in respect to SANGER. Both PSO implementations were made in Matlab. The maximum ARI obtained overdo the BUS ARI.

Introducción:

La mayoría de las características físicas, bioquímicas e inclusive el comportamiento en un ser humano, están influenciadas por múltiples variables genéticas transmitidas de generación en generación. Esta información se encuentra en todas las células del cuerpo, codificada en el ácido desoxirribonucleico (ADN) y recibe el nombre de genoma.

La genómica comparativa es la ciencia que se encarga de estudiar las semejanzas y desigualdades entre genomas de diferentes organismos. Estas investigaciones prometen adquirir nuevas percepciones, sobre muchos aspectos de la evolución de las especies modernas. Cuando comparamos genomas, podemos detectar homologías en la composición genética, como es el caso de los genes ortólogos.

Los genes ortólogos son aquellas secuencias homólogas que han evolucionado por un evento de especiación, en otras palabras son las secuencias que se encuentran en diferentes especies y que son altamente similares debido a que se han originado a partir de un ancestro común. Las secuencias ortólogas proveen información útil en taxonomía y en estudios filogenéticos. De hecho la detección de genes ortólogos entre especies que se comparan resulta imprescindible para el estudio de las funciones conservadas de los genes entre los genomas.

Como podemos apreciar la información contenida en los genomas es muy valiosa y a la vez muy amplia, por este motivo la genómica comparativa necesita de la automatización de sus métodos. En la actualidad contamos con herramientas de diseños muy variados para identificar y colectar secuencias ortólogas.

Los algoritmos de detección de ortólogos como PhyOP (Goodstadt L. 2006) se basan en el enfoque de árbol filogenético para la clasificación de los genes. Otra variante al enfoque de árbol, es el enfoque de grafos o de comparación par a par de genes. Estos algoritmos parten de construir un grafo bipartito con la similitud par a par de secuencias de los genes de dos genomas en comparación, y luego aplican heurísticas como “ReciprocalBest Hits” (RBH) (Tatusov RL, 1997) o “ReciprocallSmallestDistance” (RSD) (Wall, 2003) para podar el grafo antes de aplicar técnicas de agrupamiento. Otros

algoritmos como SOAR (Chen, 2005), MSOAR (Fu, 2007) y Multi-BUS (Rasmussen, 2005) toman en cuenta no sólo la similitud entre las secuencias sino también los reordenamientos globales de genes y los bloques de genes que conservan el orden para estimar la distancia evolutiva. Enfoques híbridos como (FadiTowfic, 2009) han explorado las propiedades de la red de interacción de proteínas en combinación con la similitud de las secuencias.

Diversas bases de datos han sido construidas a partir de la predicción basada en grafo como por ejemplo: NCBI euKaryoticOrthologousGroupdatabase (KOG) (Tatusov, 2003), INPARANOID (O'Brien et al., 2005), OrthoMCL (Li Li, 2003), OrthoMCL_DB (Feng, 2006), MultiParanoid (Alexeyenko, 2006), Eukaryotic Gene Orthologuesdatabase (EGO) (Lee Y., 2002) y más recientemente INPARANOID 7.0 (Gabriel O¨ stlund, 2010).

Siguiendo la tendencia a combinar rasgos de los genes, en el Laboratorio de Bioinformática de la Universidad Central de Las Villas (UCLV) se desarrolló el trabajo de Diploma "Herramientas Computacionales para la Comparación de Genomas" (Estopiñales, 2009), donde se define una nueva medida de disimilaridad local-global para la comparación de genes entre dos genomas de eucariotas cercanos en la evolución. Se trabajó el enfoque de grafo, construyendo el grafo bipartito a partir de la alineación par a par de secuencias utilizando el algoritmo de Needleman – Wunsh (Needleman 1970) implementado en Matlab 7.4. Sobre la base del esquema del algoritmo BUS (Kamvysselis, 2003) se combinó la similaridad de secuencias con la información de los bloques de orden conservado. Se aplicaron políticas de poda y agrupamiento similares a las de (Kamvysselis, 2003). Se realizó un experimento con el cromosoma cinco de Saccharomyces Cerevisiae y el genoma completo de Saccharomyces Bayanus con resultados prometedores.

Utilizando la posibilidad de combinar más rasgos con la nueva medida de disimilaridad local-global se construyó un nuevo algoritmo paralelo, implementado en Matlab 9.0, para la comparación par a par de genes que utiliza las estructuras algebraicas del código genético definidas en el propio Laboratorio publicadas en (Sanchez R. et. al, 2006). Se realizó la prueba de este algoritmo con el genoma Saccharomyces Cerevisiae y el

genoma Schizosaccharomyces Pombe. Se validaron los resultados con la lista de ortólogos curada manualmente (Wood, 2006). Los resultados obtenidos sugieren una mejora en cuanto a la fase de poda y agrupamiento del algoritmo general de detección de ortólogos. De aquí surge la motivación para realizar este trabajo de diploma.

0.1 Planteamiento del problema

Partiendo del enfoque basado en grafo, se necesita elevar la precisión de los algoritmos en el problema de la detección de genes ortólogos, tomando en cuenta distintas fases como el pre procesamiento y el agrupamiento.

0.2 Objetivos

0.2.1 Objetivos generales

Estudiar y aplicar algoritmos de agrupamiento sobre grafos bipartitos que se puedan utilizar en la fase de agrupamiento en la detección de genes ortólogos, para obtener resultados que sean comparables con una base de datos curada manualmente.

0.2.2 Objetivos específicos

1. Realizar un estudio de las técnicas de agrupamiento para grafos bipartitos y sus implementaciones.
2. Aplicar alguna de las implementaciones disponibles de los algoritmos de agrupamiento sobre grafo al problema de detección de ortólogos.
3. Estimar los parámetros de poda y número de grupos aplicando optimización basada en enjambres de partículas (Particle Swarm Optimization; PSO).
4. Validar los resultados del agrupamiento con los datos de Saccharomyces Cerevisiae y Schizosaccharomyce Spombe utilizando como referencia los ortólogos curados manualmente de la base de datos SANGER.

0.3 Justificación

El presente trabajo es parte de los proyectos de investigación del Laboratorio de Bioinformática de la UCLV.

0.4 Hipótesis de investigación

- La utilización de algoritmos reconocidos en la literatura para el agrupamiento sobre grafo bipartito podría mejorar el desempeño de un algoritmo de detección de ortólogos.
- El algoritmo de agrupamiento con enfoque de grafo que proponemos toma en cuenta valores adecuados para los parámetros de poda y de número de grupos en el agrupamiento.
- Los resultados de ortología del algoritmo propuesto serán comparables con una lista manual de ortólogos para Saccharomyces cerevisiae y Schizosaccharomyces pombe.

Teniendo en cuenta los objetivos propuestos con anterioridad, este proyecto se estructuró en dos capítulos. El primer capítulo ‘Marco Teórico’ se centra en el estudio de los algoritmos de agrupamiento sobre grafos, sus medidas de validación y el análisis de las metaheurísticas basada en inteligencia colectiva. En el capítulo dos ‘Implementación y Experimentación’ aplicamos un algoritmo de agrupamiento al conjunto de genes y validamos sus resultados utilizando la medida externa ‘índice de rand ajustado’, implementamos la metaheurística basada en enjambre de partículas (PSO) para la optimización de los parámetros del algoritmo de agrupamiento y del pre procesamiento.

Capítulo I: Marco Teórico

En la prehistoria, los seres humanos aplicaron sus intuiciones sobre los mecanismos de la herencia al cultivo de plantas y la cría de animales. En la actualidad, los enfoques computacionales en la comparación genómica, se han convertido en un tópico de investigación para la Bioinformática. El desarrollo de las matemáticas asistidas por ordenador con productos tales como ‘Mathematica’ o ‘Matlab’ha ayudado a ingenieros, matemáticos, biólogos e informáticos a comenzar a operar en este dominio, produciéndose una importante colección pública de casos de estudio y demostraciones.Sin embargo la genómica comparativa es un área que por su importancia se encuentra en constante desarrollo y ávida de nuevas propuestas. En el caso de la detección de genes ortólogos,aunque ha sido una rama dentro de la Bioinformática ampliamente trabajada, aún requiere de algoritmos más precisos (Gabaldón, 2009).

1.1 Genomas a comparar

La presente investigación parte de la formación de un grafo bipartito ponderado completo, que representa la correspondencia par a par de genes, de los organismos modelos: Saccharomyces cerevisiae (S. cerevisiae) y Schizosaccharomyce spombe (S. pombe).

Un grafo bipartito completo, es un grafo no dirigido, cuyos vértices se pueden separar en dos conjuntos disjuntos. Cada vértice de un grupo establece una relación con todos los vértices del otro grupo, sin llegar a existir aristas entre los vértices que pertenecen al mismo conjunto.

En el grafo, el peso de las aristas representa la distancia evolutiva entre los genes de las dos especies. Para determinarla se utilizó el principio local-global, aplicado sobre dosrasgos:la secuencia de aminoácidos y la longitud, que se define como la cantidad de nucleótidos que contiene la secuencia. Las disimilaridades locales se calcularon utilizando rutinas definidas en el ‘Matlab’, como es el caso del algoritmo Needleman–Wunsch (Needleman, 1970) para la alineación y por último se escogió la distancia euclidiana comomedida global.

Definición formal del grafo bipartito:

Sea$G = (V, E)$bipartito no dirigido, completo que describe las disimilaridades entre los genes del genoma *X* y los genes del genoma *Y* en las dos especies comparadas. El conjunto de vértices *V* tienen orden $n = n_1 + n_2$donde n_1 es el total de genes en *X* y n_2 es el total de genes en *Y*. Cada arista $x, z \in E$ que conecta los genes $x \in X$ y $y \in Y$ es pesada por la disimilaridad entre genes $d_g(x,y)$.

1.2 Pre-procesamiento

El problema del pre-procesamiento de los datos, es un área interesante para la investigación. Técnicas como la estandarización, discretización y filtrado de los datos, son métodos de pre-procesamiento que estructuran la información para su posterior análisis.

Teniendo en cuenta el enfoque de grafo de nuestro conjunto de datos, aplicamos la poda como técnica de pre-procesado, con el objetivo de descartar las relaciones entre genes evolutivamente lejanos y por lo tanto menos propensos a ser ortólogos. Con estaintención proponemos eliminar aquellas aristas, cuyo peso sobrepasen un umbral predeterminado.

1.3 Métodos de agrupamiento

El agrupamiento es una técnica de aprendizaje no supervisado, que comprende una serie de metodologías para la clasificación automática de datos en un determinado número de grupos o ‘cluster’, utilizando para ello una medida de asociación. Cada ‘cluster’ está formado por objetos que son similares entre ellos y distintos a los que forman el resto de los grupos. Mientras mayor sea la homogeneidad inter-grupo y mayor la diferencias entre-grupos, mejor es el agrupamiento.

Las técnicas de agrupamiento son aplicadas a disimiles áreas como: el procesamiento de imágenes, reconocimiento de patrones, en las ciencias económicas, en la clasificación de documentos, en la medicina, la química y estudios sociales.

Este proyecto se propone el estudio de estas metodologías aplicadas al problema de la deteccion de genes ortólogos, pero teniendo en cuenta la estructura de nuestro conjunto de datos, nos enfocaremos en las técnicas de agrupamiento sobre grafos bipartitos.

El procedimiento general de los métodos de agrupamiento sobre grafos, consiste en construir un grafo y luego aplicarle técnicas de particionamiento. El particionamiento de un grafo, es el proceso que divide el grafo en partes aproximadamente del mismo tamaño, de manera que prevalezcan pocas conexiones entre los grupos creados. En el caso de un grafo ponderado, la partición se encamina a minimizar la suma de los pesos de las aristas implicadas en el corte. Este proceso es un problema NP-completo, sin embargo se han desarrollado muchas metodologías capaces de encontrar buenas particiones.

En la bibliografía consultada para este proyecto, estudiamos diferentes algoritmos de agrupamiento con diseños muy variados. Algoritmos como ’CLuster Identification via Connectivity Kernels’(CLICK) (Shamir et al., 2000) que se encuentra disponible en http://www.tik.ee.ethz.ch/sop/bicat/, asumen tras la normalización de sus datos que los pesos de las aristas representan la probabilidad de que los vértices estén en el mismo grupo, en cambio ‘Cluster Affinity Search Technique’ (CAST) (Ben-Dor et al., 1999) va creando los grupos añadiendo o eliminando elementos en función de la afinidad, parámetro que es definido por el usuario; ambas metodologías constituyen una buena elección ante una base de datos con un alto nivel de ruido, en la que el número de posibles ‘cluster’ se desconoce.

Otros algoritmos como el Chamelon (Karypis et al., 1999) y RObust Clustering using linKs (ROCK) (Guha et al., 2000)después de particionar el grafo,fusionan los grupos en dependencia de las medidas: interconexión relativa y cercanía relativa. ‘Categorical Clustering Using Summaries’(CACTUS) (Ganti et al., 1999),‘Molecular COmplex DEtection’(MCODE) (Bader et al., 2003), ‘Restricted Neighborhood Search Clustering‘(RNSC) (King et al., 2004), ‘Super Paramagnetic Clustering’(SPC) (Brohue et al., 2006), (Tetko et al., 2005)y más reciente los métodos Spectral que emplean los valores y vectores propios de la matriz para determinar los ‘cluster’ (Alpert and Yao, 1995), (Fowlkes et al., 2004), (Kannan et al., 2004), (Ng et al., 2002), (Spielmat and

Teng, 1996), (Zien et al., 1997), (Zien et al., 1999), (Chang et al., 1994) son una pequeña muestra de la amplia gama de algoritmos existentes y en desarrollo.

Las implementaciones estudiadas de estos algoritmos son diseños orientados al análisis de datos de expresión genética, donde el objetivo es agrupar genes de un organismo en función de sus expresiones. Si recordamos las características de nuestro conjunto de datos, nos percatamos que el propósito es agrupar genes similares (ortólogos) en base a la distancia calculada a partir de sus rasgos. Por lo tanto estos diseños no son factibles para los resultados que deseamos obtener.

El 'biclustering' o co-agrupamiento es una rama del agrupamiento que permite aglomerar las filas y columnas de una matriz simultáneamente. Este término se introduce por primera vez en (Mirkin et al., 1996). El 'Statistical Algorithmic Method for Bicluster Analysis'(SAMBA) (Tanay et al., 2002) es uno de los algoritmos de 'biclustering' que utiliza el enfoque de grafo yse encuentra disponible como parte del paquete 'EXPression ANalyzer y Display ER' (Expander) que se puede acceder desdehttp://209.85.135.104/. El procedimiento del SAMBA se puede catalogar en tres pasos, primero se aplica un pre-procesamiento a los datos, segundo se buscan los K subgrafos más pesados y tercero se depuran los subgrafos mediante la eliminación o adición de vértices. Los algoritmos de 'biclustering' se caracterizan por admitir el solapamiento, característica que le imprime a los resultados una estructura que como explicaremos en la sección de 'Validación' resulta inadecuada.

Como podemos apreciar las técnicas tradicionales de particionamiento de grafos, operan directamente sobre el grafo original para producir los agrupamientos, este procedimiento resulta costoso cuando estamos trabajando con un amplio conjunto de datos. Para compensar esta limitación, los algoritmos de particionamiento de multinivel (Karypis et al., 1995),proponen primero reducir el tamaño del grafo y como resultado dividir un grafo más pequeño, luego reconstruir esta partición para el grafo original.

1.3.1 METIS

El METIS (Karypis et al,. 1998) es un paquete de software que implementa algoritmos de particionamiento multinivel, fue creado por George Karypis and Vipin Kumar. Su nombre hace referencia a la diosa griega dueña de la ‘sabiduría’ y el ‘conocimiento’.

-Procedimiento del METIS

De manera general el procedimiento de los algoritmos de particionamiento multinivel como el METIS se puede dividir en tres fases:

1- ‘Coarsening Phase’:

Durante esta etapa el tamaño del grafo sucesivamente decrece, hasta obtener un grafo con un máximo de cien vértices.

2- ‘Partición inicial’:

Realiza la división del grafo en dos partes.

3- ‘Uncoarsening phase’:

Se reconstruye el grafo original agregando vértices y aristas, a medida que se refinan las particiones utilizando el algoritmo de particionamiento KL (Kernighan–Lin) (Kernighan, 1970).

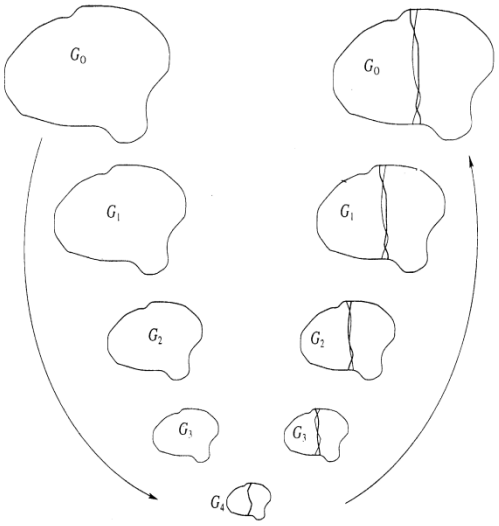


Figura 1: Representación gráfica del procedimiento general de los algoritmos multinivel. Donde G_i son los distintos grafos obtenidos durante las etapas previas.

El METIS implementa dos rutinas para trabajar:

1- ***kmetis fichero-grafo K***
2- ***pmetis fichero-grafo K***

Los parámetros fichero-grafo se refiere ala dirección del fichero que almacena la información del grafo y K es la cantidad de grupos que deseamos obtener como resultado del particionamiento.

Teniendo en cuenta las características afines de estos algoritmos con nuestros propósitos, proponemos aplicarlos al problema de la deteccion de genes ortologos y analizar sus resultados.

1.4 Validación del agrupamiento

Mediante la variación de los parámetros del algoritmo de agrupamiento, tenemos la posibilidad de obtener resultados diferentes, sin embargo algunos de ellos pueden ser no relevantes desde el punto de vista de la calidad del agrupamiento. Así, uno de los objetivos principales de las técnicas de agrupamiento es incluir alguna medida de calidad, para desestimar resultados poco interesantes. En la mayoría de los casos estas medidas son estadísticas o matemáticas y son útiles para observar la coherencia numérica de un grupo de valores. Pero hemos de tener en cuenta que tratamos con datos de origen biológico, que en principio no siguen una regla determinada o si la siguen todavía no ha sido descubierta, por lo tanto los resultados finales han de ser relevantes desde el punto de vista biológico. Es por ello que tomamos como referencia para la validación una base de datos de datos de ortólogos curada manualmente en el Laboratorio Sanger.

De acuerdo con (Steinbach et al., 2000), existen dos tipos de medidas de calidad: internas y externas. Las medidas internas comparan diferentes conjuntos de grupos sin referencias a conocimiento externo. Las medidas externas evalúan el desempeño del agrupamiento, al comparar los resultados obtenidos contra clasificaciones fehacientes del mismo conjunto de datos.

Entre los índices externos más usados se encuentran la medida F (F-measure) (Rijsbergen, 1979), el índice de Rand (Rand, 1971), (Hubert, 1985), el índice de Rand Ajustado (Hubert, 1985), el coeficiente de Jaccard (Halkidi, 2001) y el índice de Minkowski (Jardine et al., 1971). Sin embargo, un índice de validación externo puede ser cualquier medida de similaridad entre estructuraciones, siempre calculando la similitud existente entre la estructuración obtenida por cierto clasificador, respecto a una estructuración conocida, la cual es asumida como correcta o natural para el conjunto de objetos analizados (Shulcloper et al., 2010).

En (Milligan and Cooper, 1986) se hace un estudio de índices que miden la semejanza entre dos particionamiento con diferentes números de grupos. Tomando en cuenta el comportamiento de cada uno de ellos ante las pruebas realizadas, se arriba a la conclusión de que el índice de rand presenta un mejor desempeño.

En este proyecto para determinar la validez de los resultados obtenidos por el algoritmo de agrupamiento METIS, adoptaremos esta medida externa y nos apoyaremos en la base de datos ortólogos curada manualmente versión dos de Sanger, como referencia externa.

1.4.1 Índice Aleatorio Ajustado (ARI)

Esta medida es la versión corregida del Índice de Rand (AR), ambas determinan la similitud entre dos agrupamientos. Para poder utilizarlas los agrupamientos deben ser exclusivos, es decir cada objeto debe pertenecer a un solo grupo. Teniendo en cuenta esta característica, algoritmos que permiten el solapamiento como SAMBA; no se utilizan.

-Definición:
Sea U la clasificación correcta y V la solución generada por el algoritmo de agrupamiento. Tenemos que:

$$ARI\ U,V = \frac{a - \frac{(a+b)(a+c)}{\frac{N}{2}}}{\frac{a+b\ +(a+c)}{2} - \frac{(a+b)(a+c)}{\frac{N}{2}}}$$

donde:

$-a = \quad i,j\ C_u\ i\ = C_u\ j\ \wedge C_v\ i\ = C_v(j)\quad ,$

$-b = \quad i,j\ C_u\ i\ = C_u\ j\ \wedge C_v\ i\ \neq C_v(j)\quad ,$

$-c = \quad i,j\ C_u\ i\ \neq C_u\ j\ \wedge C_v\ i\ = C_v(j)\quad ,$

$-N = total\ de\ datos.$

Los valores de ARI oscilan entre cero y uno, mientras más cercano a uno estén los resultados, mejor es el agrupamiento con respecto a la clasificación externa.

La implementación de esta medida se encuentra en el paquete de ensamblaje de agrupamientos 'Link-Based Cluster Ensembles' (LinkCluE) disponible en http://users.aber.ac.uk/nii07/(Iam-on et al., 2010).

1.5 Estudio de los parámetros de la poda y del agrupamiento

En esta sección se realiza un estudio de los parámetros que influyen en el comportamiento del algoritmo METIS; con el objetivo de conocer cuáles son los valores adecuados para la asignación.

Si hacemos una recapitulación recordaremos que los parámetros del METIS consisten en:

1- Fichero Grafo: donde le indicamos la dirección del grafo que se desea particionar.
2- K: determina la cantidad de grupos que queremos obtener.

Podemos comprobar que para valores aleatorios de K el algoritmo arroja diferentes resultados, del mismo modo sucede si variamos el umbral de poda, obteniendo de esta forma distintos grafos. Los grafos pueden diferir en sus relaciones, porqué para umbrales diferentes excluimos o incluimos diferentes aristas para el análisis. Determinar un umbral de poda óptimo, permite que prevalezcan para el agrupamiento las relaciones relevantes entre genes ortólogos y eliminaría aquellas que no son imprescindibles. Conocer la cantidad de grupos ante la cual el METIS obtiene un mejor agrupamiento resulta apropiado y necesario ante la amplia gama de valores posibles que podemos asignarle.

Con el objetivo de conocer la combinación apropiada de valores para estos parámetros, utilizamos la metaheurística basada en enjambre de partículas (PSO por sus siglas en inglés Particle Swarm Optimization) (Kennedy, 1997), (Kennedy y Eberhart, 1995a), (Kennedy y Eberhart, 1995b), (Kennedy y Spears, 1998). Este método posee buena convergencia global y es fácil de implementar comparado con otros métodos de

optimización, además ha mostrado ser muy eficiente para resolver problemas de optimización de un sólo objetivo con rápidas tasas de convergencia (Kennedy et al. 2001).

1.6 Inteligencia Colectiva

La computación bioinformática crea analogías de sistemas naturales o sociales para la resolución de problemas. Los algoritmos bioinspirados, modelan de forma aproximada un fenómeno existente en la naturaleza y en la actualidad son uno de los campos más prometedores de investigación en el diseño de algoritmos. Ejemplos de estos modelos son las redes neuronales que imitan el comportamiento del sistema nervioso, los algoritmos evolutivos basados en los principios darwinianos de evolución natural, los algoritmos inmunológicos que simulan el desempeño del sistema inmunológico y las metaheurísticas basadas en Inteligencia Colectiva como la optimización basada en colonias de hormigas (Dorigo y Stützle 2002), (Dorigo y Stützle, 2004), (Dorigo et al. 2006), (Dorigo, 2007), donde se estudia la conducta de estos organismos cuando realizan una actividad en conjunto y la optimización basada en enjambres de partículas, que es una técnica de optimización inspirada en el comportamiento social de bandadas de aves o peces.

La "Inteligencia Colectiva" es una forma de inteligencia que surge de la colaboración y concurso de muchos individuos, esta expresión fue introducida por Gerardo Beni, Suzanne Hackwood y Jing Wang en 1989. Las metaheurísticas basadas en Inteligencia Colectiva son técnicas inspiradas en el estudio de comportamientos colectivos (Bonabeau, 1999), presentes en sistemas de la naturaleza, generalmente de carácter descentralizado y auto organizativo.

La principal característica de los algoritmos basados en la inteligencia colectiva viene determinada por la estrecha colaboración social que presentan, a través del sistema de comunicación que surge entre los individuos del colectivo (Engelbrecht, 2006). Esta comunicación, puede ser de forma directa o indirecta. La comunicación indirecta ocurre cuando un individuo altera el medio en que se desarrollan y los otros son capaces de captar estos cambios, por ejemplo las hormigas se guían por un rastro de feromona que dejan sus compañeras indicando la calidad de determinada ruta. En cambio la

comunicación directa es cuando se determina la ubicación de otros individuos mediante sonido, visibilidad u otra forma directa de interacción.

1.6.1 Optimización basada en Enjambre de Partículas

El método de optimización basado en enjambres de partículas fue desarrollada por James Kennedy y Russell C. Eberhart. Su procedimiento se inspira en el movimiento natural desarrollado por las comunidades de animales, tales como las emigraciones de las bandadas de pájaros, y en la actualidad se emplea en la optimización de distintos tipos de sistemas.

En PSO la población es un enjambre que está compuesto por una serie de partículas (pájaros, peces, abejas, etc.) que interactúan entre sí y representan soluciones a un problema determinado. La inteligencia de este enjambre no está en los individuos, sino en el colectivo 'Swarm Intelligence'.

Cada una de las partículas es tratada como un punto en un espacio N dimensional dentro del cual pueden desplazarse, es decir cambiar de posición en el espacio de búsqueda. Con este propósito cada partícula ajusta su propio "vuelo" de acuerdo a su propia experiencia y a la experiencia del resto de la banda la cual vuela por el espacio buscando regiones prometedoras(Kennedy et al., 1998).Tal comportamiento social se basa en la transmisión del éxito de cada individuo a los demás del grupo, lo que les permite satisfacer de la mejor manera posible sus necesidades más inmediatas, por ejemplo la localización de alimentos. Este procedimiento es el que define los movimientos de las variables de decisión en el espacio de búsqueda y las orienta hacia soluciones óptimas.

-Procedimiento general

El procedimiento de la metaheurística PSO se puede resumir en cinco pasos que se describen a continuación (Kennedy and Eberhart, 1995b):

1. Inicialización de las partículas.
2. Actualización de la velocidad de cada partícula.
3. Actualización de la posición de cada partícula.
4. Actualización de la memoria.
5. Chequeo de terminación.

1- Inicialización de las partículas

Dado un espacio de decisión N-dimensional, cada partícula i del enjambre se representa por su posición actual Pi ={Pi_1, Pi_2,…, Pi_n}, la velocidad Vi = {Vi_1, Vi_2,…,Vi_n} con la cual ha llegado a dicha posición, así como la mejor posición Pbest = {$Pbesti_1$, $Pbesti_2$,…, $Pbest_n$} en la que se ha encontrado, y por último la mejor posición encontrada dentro del enjambre Gbest denominada "mejor global".

Inicialmente los valores de la velocidad y la posición de las partículas se determinan aleatoriamente, dentro de los rangos predefinidos en función del problema. El total de partículas varía en dependencia de los objetivos de la optimización, aunque por lo generalse recomienda emplear poblaciones de 10 a 40 partículas. Es importante recordar que a mayor valor de los parámetros más oportunidad de encontrar el óptimo pero aumenta el costo computacional.

Una vez determinado los valores de posición y velocidad para cada partícula del enjambre, se evalúa la función objetivo y en dependencia del valor obtenido se determina la mejor posición alcanzada por cada partícula hasta el momento.A continuación teniendo en cuenta la mejor experiencia de cada partículase establece la mejor solución global del enjambre.

2- Actualización de la velocidad de cada partícula

En cada iteración se genera un nuevo desplazamiento para todas las partículas. El número de iteraciones es un parámetro que decide el usuarioen dependencia de sus objetivos, aunque por lo general se escoge entre 100 o 200 iteraciones. Para determinar la nueva velocidad se utiliza la siguiente ecuación:

$$V(\iota) = w * V(\iota) + C_1 * rand_1 \ \ Pbest(\iota) - \ P(\iota) \ \ + C_2 * rand_2 \ \ Gbest(\iota) - P(\iota)$$

Dónde:

-w:peso de la inercia

-C_1:razón de aprendizaje cognitivo

-C_2:razón de aprendizaje social

-$rand_1 y \ rand_2$: números aleatorios entre cero y uno.

-$Pbest \ \ i$: mejor posición alcanzada por la partícula i.

-$Gbest \ \ i$: mejor posición global alcanzada por el enjambre.

-$P \ \ i$: posición actual de la partícula i.

-$V \ \ i$: velocidad actual de la partícula i.

El peso de la inercia es un parámetro de usuario utilizado para controlar el impacto de la velocidad en el espacio de búsqueda. Determina el balance entre la exploración local y global, una adecuada selección de su valor permite en menos iteraciones encontrar el óptimo. Típicamente a w se le asigna un valor fijo por ejemplo 0.8 (Eberhart et al., 1998) y en otros casos se le asigna un valor inicial entre 1 y 1.5 que se hace decrecer durante la ejecución del algoritmo o se utilizan funciones como w = 0.5 + rand()/2. O sea, se proponen pesos de inercia altos al principio para iniciar una exploración global ya que al tomar w un valor grande las partículas se mueven lejos de la mejor posición alcanzada según su conocimiento y si reducimos el valor de w con el número de iteraciones nos enfocamos en una búsqueda local. Si w=0 la velocidad de la partícula se determina por las mejores posiciones ya sea su mejor posición o la mejor posición global alcanzada por todas las partículas.

La razón de aprendizaje cognitivo es un valor que indica la influencia de la mejor experiencia de la partícula en su nueva velocidad, mientras que la razón de aprendizaje social es el parámetro que determina el dominio de la información social en el valor de velocidad para la partícula.

Kennedy identifica cuatro tipos de algoritmos de PSO en función de los valores de C_1 y C_2(Kennedy and Eberhart., 1995b):

– Modelo completo: C_1, C_2 >0.
– Sólo cognitivo: C_1 >0 y C_2 =0.
– Sólo social: C_1 =0 y C_2 >0.
– Sólo social exclusivo: C_1 =0, C_2 >0

La selección de estos parámetros tiene impacto en la velocidad de convergencia del algoritmo para encontrar el óptimo. En (Grau et al. 2007), (Eberhart et al., 2001), (Eberhart et al., 1999) se tomaron por ejemplo los valores C1= C2=2, pero en realidad se recomienda en el trabajo que C1 y C2 no tomen necesariamente el mismo valor sino, que se generen aleatoriamente con distribución uniforme en el intervalo [0, 2]. En (Beielstein et al. 2002) se recomienda que la suma de estos valores sea menor o igual a 4. Para obtener una mayor información acerca de la influencia de estos parámetros en la efectividad del algoritmo PSO ver (Beielstein et al. 2002; Kennedy et al. 2001; Shi et al., 1998).

$Rand_1$ y $Rand_2$ son funciones que retornan un número aleatorio en el intervalo [0, 1], mediante el cual se determina la influencia real de las informaciones individual y social en la nueva velocidad para la partícula.

La velocidad suele llegar a ser muy grande por lo que es necesario ajustar su excesivo crecimiento. Si definimos muy pequeño el valor máximo que puede alcanzar la velocidad la búsqueda se enfoca en óptimos locales, mientras que si es grande se pueden sobrepasar buenas soluciones; generalmente para evitar pensar en un rango adecuado para la velocidad se le asigna el mismo que se usa para las posiciones, ofreciendo buenos resultados (Eberhart, et al., 2000).

Otra variante para controlar el excesivo crecimiento de la velocidad es utilizar el coeficiente de restricción *K*:

$$K = \frac{2}{2 - \varphi - \sqrt{\varphi^2 - 4\varphi}}$$

donde:

- $\varphi = C_1 + C_2$

Este coeficiente transforma la ecuación de la velocidad como se muestra a continuación:

$$V(\imath) = K * (w * V(\imath) + C_1 * rand_1 \ Pbest(\imath) - P(\imath) + C_2 * rand_2 \ Gbest(\imath) - P(\imath))$$

3- Actualización de la posición de cada partícula

A partir del valor de la velocidad se determina una nueva posición para todas las partículas:

$$P\ i = P\ i + V(i)$$

4- Actualización de la memoria

En cada iteración se determina una nueva posición para cada partícula, en esta paso se comprueba utilizando la función objetivo, si el nuevo valor supera la mejor posición alcanzada por la partícula hasta el momento. Una vez que han sido actualizadas o analizadas las mejores posiciones de cada partícula con respecto al nuevo desplazamiento, se determina la mejor posición global.

4.1-$Pbest\ \imath = P(\imath)$ si $F\ P\ \imath > F\ Pbest\ \imath$

4.2- $Gbest = Pbest\ \imath$ si $F\ Pbest\ \imath > F\ Gbest$

donde F(x) es la función objetivo que se optimiza en cada iteración; para este proyecto usamos la medida externa Índice de Rand Ajustado descrita en el epígrafe 1.4.1 para

determinar la cantidad óptima de grupos, mientras que para conocer el valor adecuado para la poda utilizamos la distancia de 'Hamming' (Deza, 2006).

-Medida Hamming:

La distancia de Hammingd_h es una medida en $\mathbb{R}^n$, definida por:

$i: 1 \leq i \leq n, X_i \neq Y_i$, donde X,Y $\in [0,1]$

5- Chequeo de terminación

Repetir el algoritmo del paso dos al paso cuatro, hasta que cumplir con el total de iteraciones definidas por el usuario.

PSO ha sido aplicado con éxito en diferentes campos de investigación para la resolución de problemas de optimización. Algunos ejemplos son: optimización de funciones numéricas (Xie et al., 2002), entrenamiento de redes neuronales (Gudise et al., 2003), aprendizaje de sistemas difusos (Parsopoulos et al., 2003) y registrado de imágenes (Omran et al., 2002). La mayoría de estos problemas requieren codificación continua y, aunque no existe un gran número de propuestas de PSO para trabajar con otro tipo de codificación como la binaria o para permutaciones de enteros; se utilizarán algunas variantes que permiten trabajar con estos parámetros.

Capitulo II: Implementación y experimentación

Apoyándonos en la investigación previa, el propósito de este capítulo es aplicar una metodología de agrupamiento sobre el grafo bipartito y orientar sus resultados a la detección de genes ortólogos. Para determinar la validez de los grupos de ortólogos detectados, utilizamos la medida externa 'Índice de Rand Ajustado' y con el propósito de mejorar el desempeño del algoritmo de agrupamiento, optimizamos los parámetros que influyen en sus soluciones utilizando la metaheurística PSO.

2.1 Organismos

En este proyecto trabajamos con los genomas de dos especies de levadura: S. cerevisiae y S. pombe. Analizamos un total de10923genes y29648630relaciones pesadas a partir de la medida de similaridad local-global en función de los rasgos: alineación y longitud. Los valores para esta medida oscilan dentro del rango [0,1].

2.2 Pre procesamiento

Antes de comenzar la fase de agrupamiento, realizamos un pre-procesamiento del conjunto de datos; con el objetivo de eliminar mediante la poda, aristas irrelevantes para el análisis y además reducir el tamaño del grafo.

El procedimiento general de la poda consiste en determinar la arista óptima de salida de un gen que sería la arista de salida con menor peso; teniendo en cuenta este valor determinamos un umbral que excluye las relaciones de ese gen cuyo peso sobrepase el umbral. Este procedimiento se realiza para todos los genes de cada genoma.

Como estamos trabajando con un grafo no dirigido, para poder visualizar mejor las aristas de salida de un gen, creamos una versión dirigida del grafo original por cada genoma. De manera que dado el grafo bipartito completo G=(V,E), construimos el grafo M=(V,E) como la versión dirigida del grafo G, sustituyendo cada arista no dirigida e=(x, y) por dos aristas dirigidas (x, y) y (y, x) conservándose el mismo peso.

Después de haber podado ambos grafos teniendo en cuenta el criterio expuesto, para acoplar los resultados en el grafo original, comprobaremos cuáles son las aristas eliminadas que tienen en común ambos grafos, estas serán las aristas podadas en el grafo inicial.

A continuación proponemos un seudocódigo del procedimiento explicado anteriormente:

2.2.1 Seudocódigo del pre-procesamiento

Definiciones:

1. matriz-similitud = $\begin{matrix} a_{1,1} & \cdots & a_{1,m} \\ \vdots & \ddots & \vdots \\ a_{n,1} & \cdots & a_{n,m} \end{matrix}$ donde $a_{i,j} \in [0,1]$
2. poda $\in [20,80]$

Parámetros de entrada*: matriz de similitud, poda*
1- *Crear versión dirigida del grafo bipartito*
- $matriz\_genoma\_1_{nxm} = matriz\_similitud_{nxm}$
- $matriz\_genoma\_2_{nxm} = (\ matriz\_similitud_{nxm}\ )'$

2- *Efectuar poda para cada matriz.*
- $\forall\ A_i$ tal que i = 1..n
- Dado A_i buscar el mínimo de los $a_{i,j}$
- Umbral = min $a_{i,j}$ \ poda $*$ 1000 + min $a_{i,j}$
- Aquellos $a_{i,j} > umbral;\ a_{i,j} = 0$

3- *Acoplar los grafos podados*
- $\forall a_{i,j}\ tal\ que\ i = 1..n; j = 1..m$
- $Si\ a_{i,j} = 0\ en\ matriz\_genoma\_1\ y\ \ a_{i,j} = 0\ en\ matriz\_genoma$

$Entonces a_{i,j} = 0\ en\ matriz\_adyacencia$

Salida:*matriz de similitud podada.*

Figura 1: Seudocódigo del método de pre procesamiento.

2.3 Agrupamiento

En esta sección aplicaremos el algoritmo de agrupamiento multinivel estudiado en el capítulo anterior,vinculado al problema de la detección de genes ortólogos. Entre las dos rutinas que implementa el paquete METIS vamos a utilizar *kmetis* por ser la mas rápida y la mejor que se desempeña ante grandes volúmenes de datos.

Este software es uno de los más utilizados para el particionamiento de grafos, carece de interfaz gráfica, por lo que las rutinas se ejecutan por línea de comando y muchos algoritmos de agrupamiento como el Chameleon lo incorporan como parte de su procedimiento. La versión que vamos a utilizar es la 4.0 programada para Windows y como recordaremos la rutina se invoca de la siguiente manera: kmetis Fichero_Matriz K Para que el software comprenda la estructura del grafo que se carga en el fichero de entrada, tenemos que transformar la información al formato que el programa exige. Este procedimiento se automatiza con la función *formato_ metis.m.* Una vez que hayamos decidido la cantidad de 'clusters' que deseemos obtener invocamos la rutina con los datos pertinentes.

Cuando termine de ejecutarse el algoritmo*kmetis*se produce como salida un listado, donde por cada vértice del grafo se indica el número del 'cluster' al que pertenece, comenzando por cero la enumeración. Para conocer la veracidad de los grupos de ortólogos detectados por el algoritmo, utilizamos la medida externa Indice de Rand ajustado (ARI).

La implementación de la medida ARI se encuentra en la función *adjusted_rand_index.m*y recibe como párametro dos agrupamientos, uno de ellos es el resultado producido por el algoritmo y el otro será la clasificación externa usada como referencia. Estas clasificacionesse representan como dos vectores columna, donde el índice del vector representa el número del gen y el valor del vector en ese índice indicael 'cluster' al que pertenece el gen. Para poder usar la medida, ambas clasificaciones deben trabajar sobre el mismo conjunto de genes sin solapamiento y pueden diferenciarse en el número de 'clusters' detectados. Básicamente, el próposito de ARI es detectar lassemajanzas entre los dos agrupamiento,contabilizando los genes

que fueron agrupados juntos en ambas clasificaciones y los que fueron separados, en‘clusters’ diferentes. El resultado de ARI es un valor entre cero y uno, mientras mas cercano a uno este el resultado mejor será el agrupamiento en base a la referencia externa.

Como referencia externa utilizamos la base de ortólogos curada manualmente Sanger, donde se clasifican como ortólogos o no ortólogos los genes con los que estamos trabajando. Inicialmente el Sanger incluía 4994 clasificaciones de sequencias S pombe y 5112 de Scerevisiae, algunas de las cuales no formaban parte de nuestro grafo, que contaba con 5885 secuencias de Scerevisiae y 5038 de S pombe, lo mismo sucedía a la inversa, es decir algunos de los genes del grafo no estaban clasificados en el Sanger. Para lograr trabajar sobre un conjunto de datos en común y hacer las inferencias pertinentes, eliminamos las incoherencias entre ambas clasificaciones obteniendocomo resultado el conjunto de datos actual.

El Sanger esta formado por grupos de genes, donde interpretamos cada grupo como un ‘cluster’ de genes ortólogos, aquellos genes que no pertenecen a ningún grupo al no ser ortólogos los ubicamos en el cluster 1 por defecto.Para lograr extraer un vector columna con estos datos del Sanger implementamos la función *vector_sanger.m.*

2.3.1 Resultados obtenidos del METIS

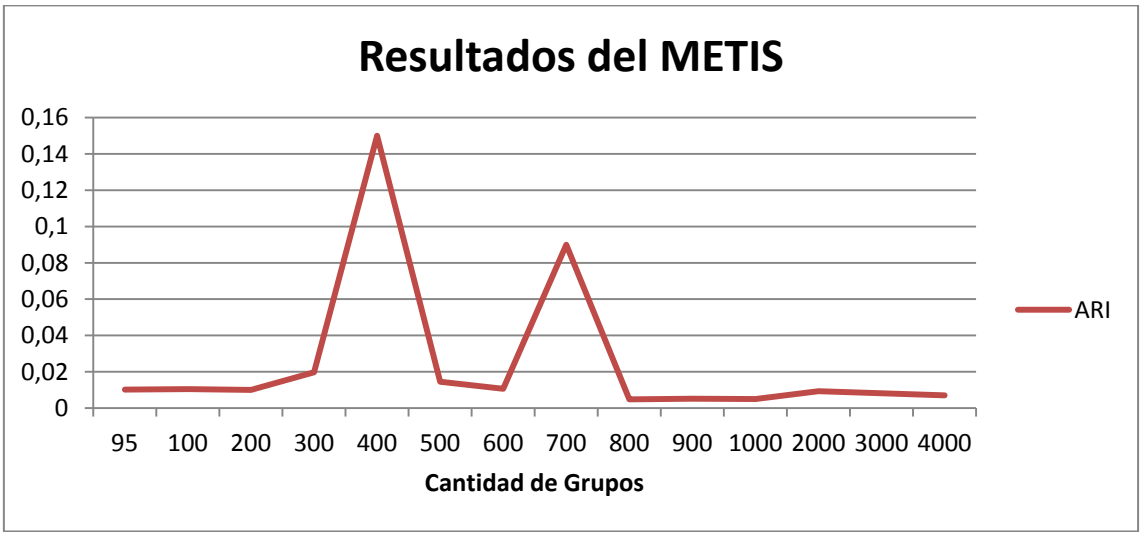


Figura 2 : Tabla donde se representan corridas del METIS y los valores de ARI.

Si analizamos la tabla anterior podemos apreciar que cuando variamos la cantidad de grupos a generar, obtenemos valores muy aleatorios de la medida ARI.Sin embargo, resulta díficil inferir el valor adecuado que debemos asignarle a la cantidad de 'cluster' para lograr la mejor clasificación posible con METIS.El párametro de poda de la etapa de pre procesamiento no deja de ser menos importante, sus resultados también influyen en las soluciones del algoritmo de agrupamiento.

Con el objetivo de conocer los valores óptimos para el párametro de poda de la etapa de pre procesamiento y la cantidad de 'clusters' para el algoritmo de multinivel, ante los cuales se obtienen los resultados más altos de la medida ARI, utilizamos la metaheurística basada en enjambre de particulas PSO de la inteligencia colectiva.

Es importante destacar que los valores de ambos párametros son números enteros mientras que el PSO trabaja preferentemente con codificación continua. En la literatura se presentan algunas variantes para tratar estas limitaciones:

1- Normalizar los parámetros discretos presentes en la partícula. El modo de normalización consiste en dividir cada parámetro discreto entre la suma de todos losparámetros discretos presentes en la misma; de manera que se obtienen valores continuosentre cero y uno. Todos los parámetros, continuos y discretos, calculan su velocidad de lamisma manera (Xie et al., 2002).

2- Tratar todos los parámetros, tanto continuos como discretos, de la misma manera y al efectuar operaciones cuyo resultado sea continuo en parámetros que necesitan ser enteros, se trunca dicho valor continuo (Quesada, 2010).

En este proyecto emplearemos la variante dos.
Otro aspecto importante para destacar es la comparación realizada entre el algoritmo BUS (Estopiñales, 2009) y el METIS en función de los valores de ARI. La figura 3 muestra los resultados de METIS generados para K= 1000 y K=2000 sobre grafos podados con parámetros de poda igual a 20 y 80 y los resultados de BUS ante los mismos grafos podados. Como se podrá apreciar el METIS presenta un mejor comportamiento que el BUS ante estos parámetros.

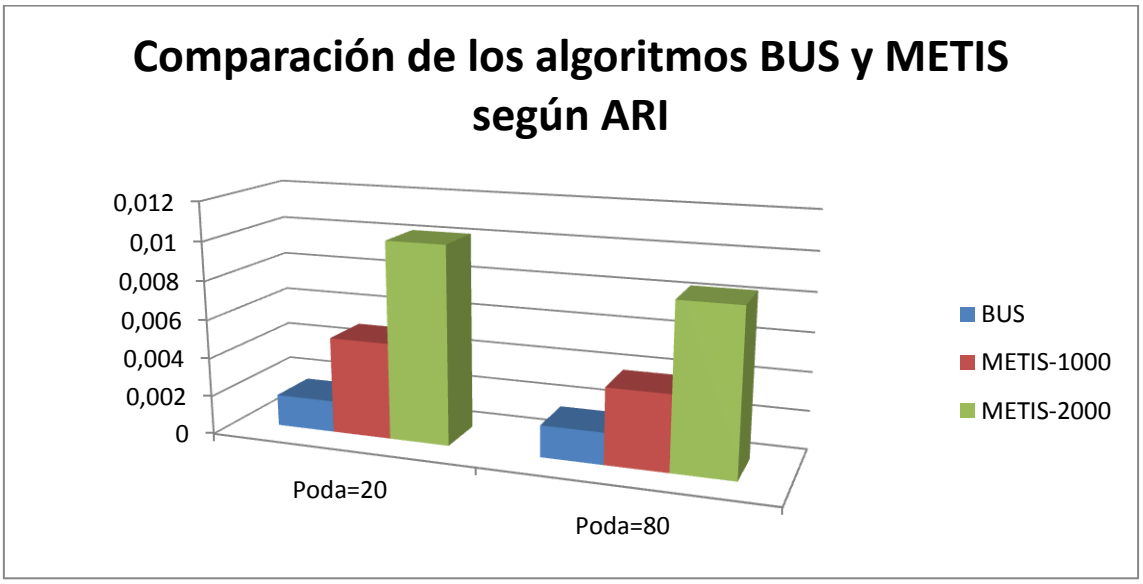


Figura 3: Comparación de los algoritmos BUS y METIS en función de los valores de ARI.

2.4 Optimización del parámetro poda

Cuando sobre el grafo aplicamos una poda óptima eliminamos las relaciones entre genes que no son ortólogos, evitando que el algoritmo de agrupamiento analice aristas irrelevantes que pudieran influir en sus resultados. Para conocer el valor de la poda que optimiza la etapa de pre procesamiento utilizamos la metaheurística PSO.

Como grafo podado de referencia óptima vamos a utilizar un grafo que extraemos del Sanger. La idea es tomar cada uno de los grupos de genes ortólogos del Sanger y establecer arista entre esos genes, aquellos genes que no pertenezcan al 'cluster' no se relacionan con los genes que si pertenecen, de esta manera eliminamos aristas entre genes que son ortólogos según el Sanger.

La función objetivo que evalúa el parámetro de poda es la distancia de Hamming, cuyo valor indica las aristas diferentes entre ambos grafos, mientras menor sea el valor de la distancia más se acerca el grafo podado al grafo de Sanger.La función que implementa este procedimiento es *pso_poda.m*

2.4.1 Seudocódigo del PSO para el parámetro poda

Entrada: *W, C_1, C_2, Cantidad de iteraciones, Total de partículas*

*1-**Inicializar las partículas***
-Para el total de partículas generar aleatoriamente los valores iniciales de posición y velocidad dentro de los rangos predefinidos.

*2-**Calcular la mejor posición inicial para cada partícula.***
-Tomar los valores de la posición y la velocidad de cada partícula, como los valores de poda.
-Calcular la distancia de Hamming de las matrices podadas contra la matriz Sanger.
-Tomar el valor para el cual fue menor la distancia de Hamming, como la mejor posición alcanzada por la partícula.
-Determinar la mejor posición global, como aquel valor para el cual se obtuvo la menor distancia de haming en todo el enjambre.

*3-**Actualizar la velocidad de cada partícula***
-Aplicar la ecuación de la velocidad para cada partícula, chequear que el valor generado no sobrepase el rango definido para la velocidad.

*4-**Actualizar la posición de cada partícula***
-Calcular la nueva posición de la partícula a partir de la nueva velocidad
-Chequear que la partícula no 'vuele' hacia regiones no factibles.

*5-**Actualizar memoria***
-Volver a determinar en base al nuevo desplazamiento y a la mejor posición alcanzada por la partícula la mejor posición global y la mejor posición de cada partícula, utilizando la función de Hamming.

*. 6-**Repetir del paso tres al cuatro hasta el total de iteraciones***

Salida: *Fichero donde se almacenan las mejores posiciones alcanzadas por las partículas durante su vuelo y por el enjambre.*

Figura 4: Esquema del seudocódigo del PSO para la optimización del parámetro de poda.

2.4.2 Diagrama de componentes

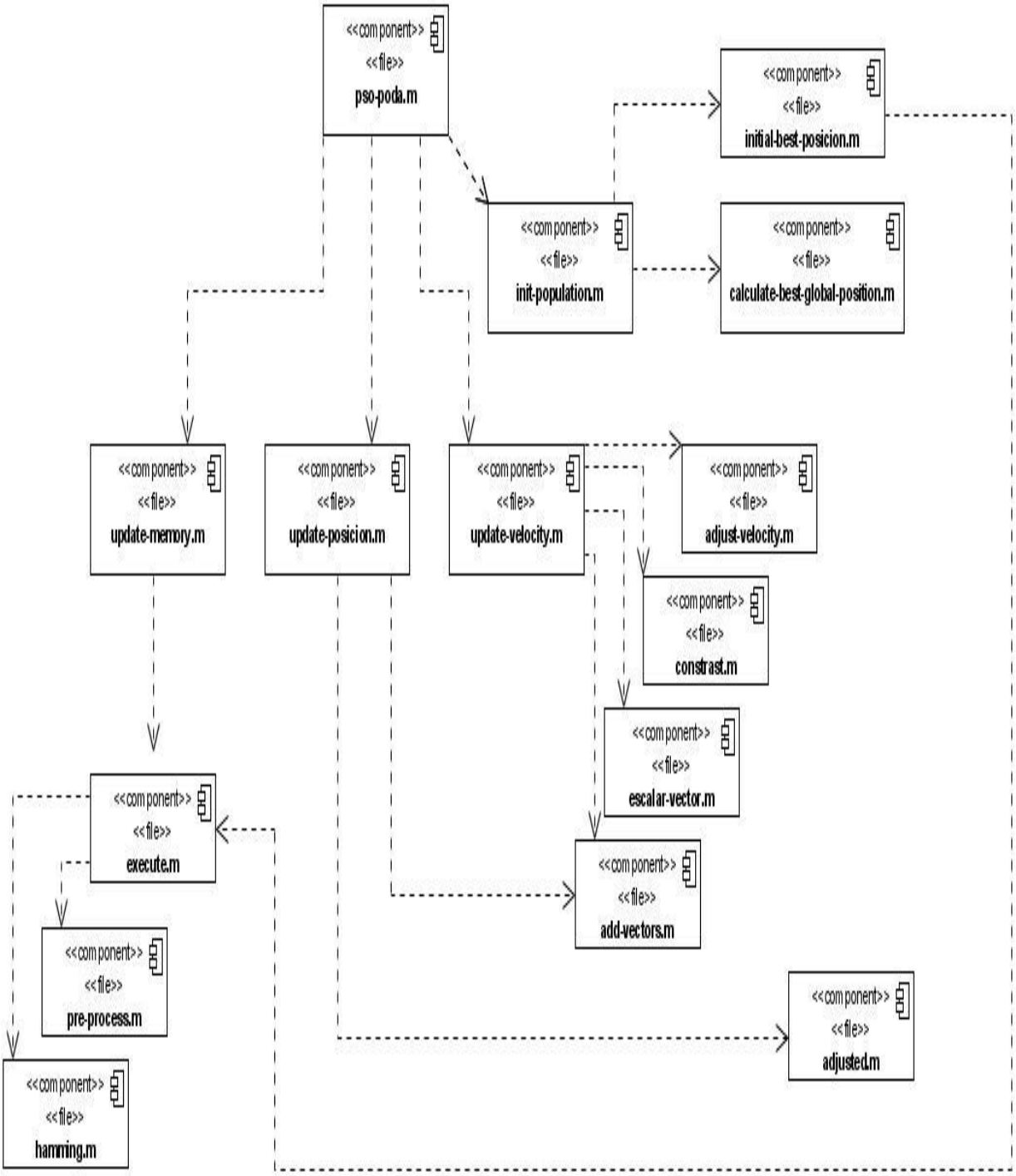


Figura 5: Diagrama de componentes del procedimiento PSO aplicado en la optimización del parámetro de poda.

-Definición de parámetros para el PSO

Teniendo en cuenta las características del problema y el análisis realizado en el capítulo anterior, para el PSO de la poda definimos una población de 20 partículas que representan el parámetro poda a optimizar de una sola dimensión, el rango de la posición y la velocidad estará determinado por los valores factibles para la poda [20,80]. El peso de la inercia (w) será 0.8 y los valores del aprendizaje cognitivo (C_1) y el aprendizaje social (C_2) es 2. El total de iteraciones fue 100.

2.4.3 Resultados del PSO para la poda

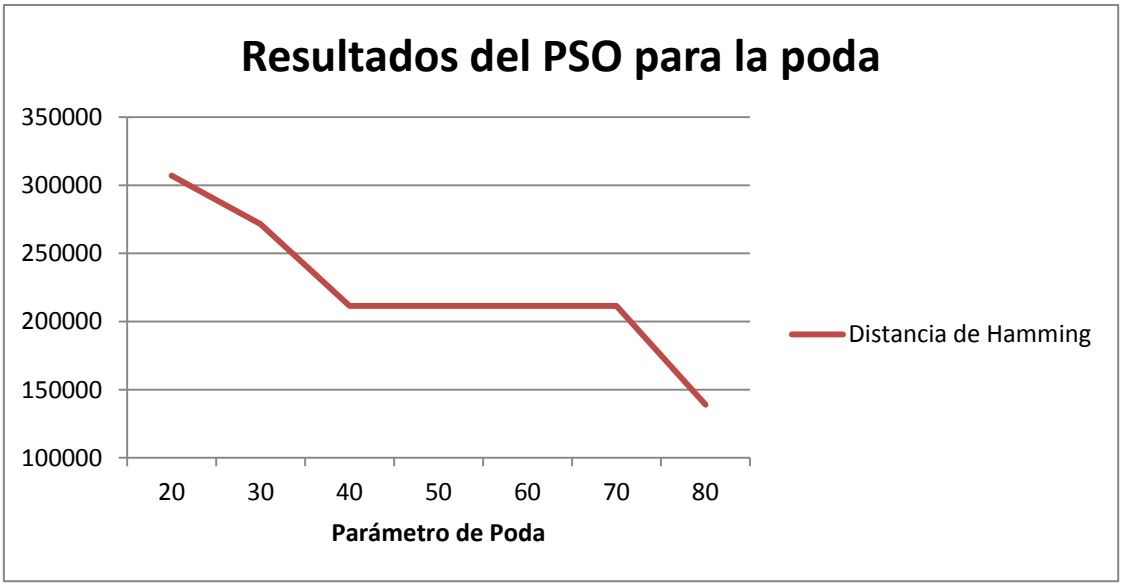


Figura 6: Distancia de Hamming entre la matriz podada y la matriz Sanger.

Como podemos observar en el gráfica la mayor la similitud que se obtiene entre las dos grafos es para la poda igual 80, donde la distancia de Hamming asume el valor más bajo de 138 965 aristas diferentes, de un total de 29 648 630. Por lo tanto el valor de poda óptimo que mejor estructura el grafo para la detección de genes ortólogos es: 80.

2.5 Optimización de parámetros del METIS

Teniendo en cuenta el comportamiento aleatorio del METIS ante diferentes valores del parámetro K, proponemos conocer utilizando el PSO el número apropiado de grupos ante el cual el algoritmo presente los mejores resultados en función de ARI.

Nos apoyaremos en la base de ortólogos curada manualmente Sanger para determinar por medio de la medida externa, la veracidad de los grupos de ortólogos generados por el algoritmo de agrupamiento. Además tomaremos en cuenta los resultados del epígrafe anterior por lo que en este procedimiento trabajaremos sobre el grafo podado óptimamente. El procedimiento que implementa este proceso es *pso_metis.m.*

2.5.1 Seudocódigo del PSO para el parámetro K

Entrada:*W,C_1,C_2, Cantidad de iteraciones, Total de partículas*

*1-**Inicializar las partículas***
-Para el total de partículas generar aleatoriamente los valores iniciales de posición y velocidad entre los rangos predefinidos.

*2-**Calcular la mejor posición inicial para cada partícula***
-Tomar los valores de la posición y la velocidad de cada partícula, como números de 'cluster' a generar; ejecutar el kmetis
-Calcular el valor de ARI para el agrupamiento producido por el METIS y el agrupamiento de referencia Sanger.
-Determinar la mejor posición global, como aquel valor para el cual se obtuvo el mayor resultado de ARI de todo el enjambre.

*3-**Actualizar la velocidad de cada partícula***
-Aplicar la ecuación de la velocidad para partícula, chequear que el valor generado no sobrepase el rango definido para la velocidad.

*4-**Actualizar la posición de cada partícula***
-Calcular la nueva posición de la partícula a partir de la nueva velocidad
-Chequear que la partícula no 'vuele' hacia regiones no factibles.

*5-**Actualizar memoria***
-Volver a determinar en base al nuevo desplazamiento y a la mejor posición alcanzada por la partícula la mejor posición global y la mejor posición de cada partícula, utilizando la función ARI.

*6-**Repetir del paso tercero al cuarto hasta el total de iteraciones.***
***Salida:** Fichero donde se almacenan las mejores posiciones alcanzadas por las partículas durante su vuelo y por el enjambre.*

Figura 7: Seudocódigo del proceso de PSO aplicado en la optimización del parámetro K del METIS

2.5.2 Diagrama de componentes

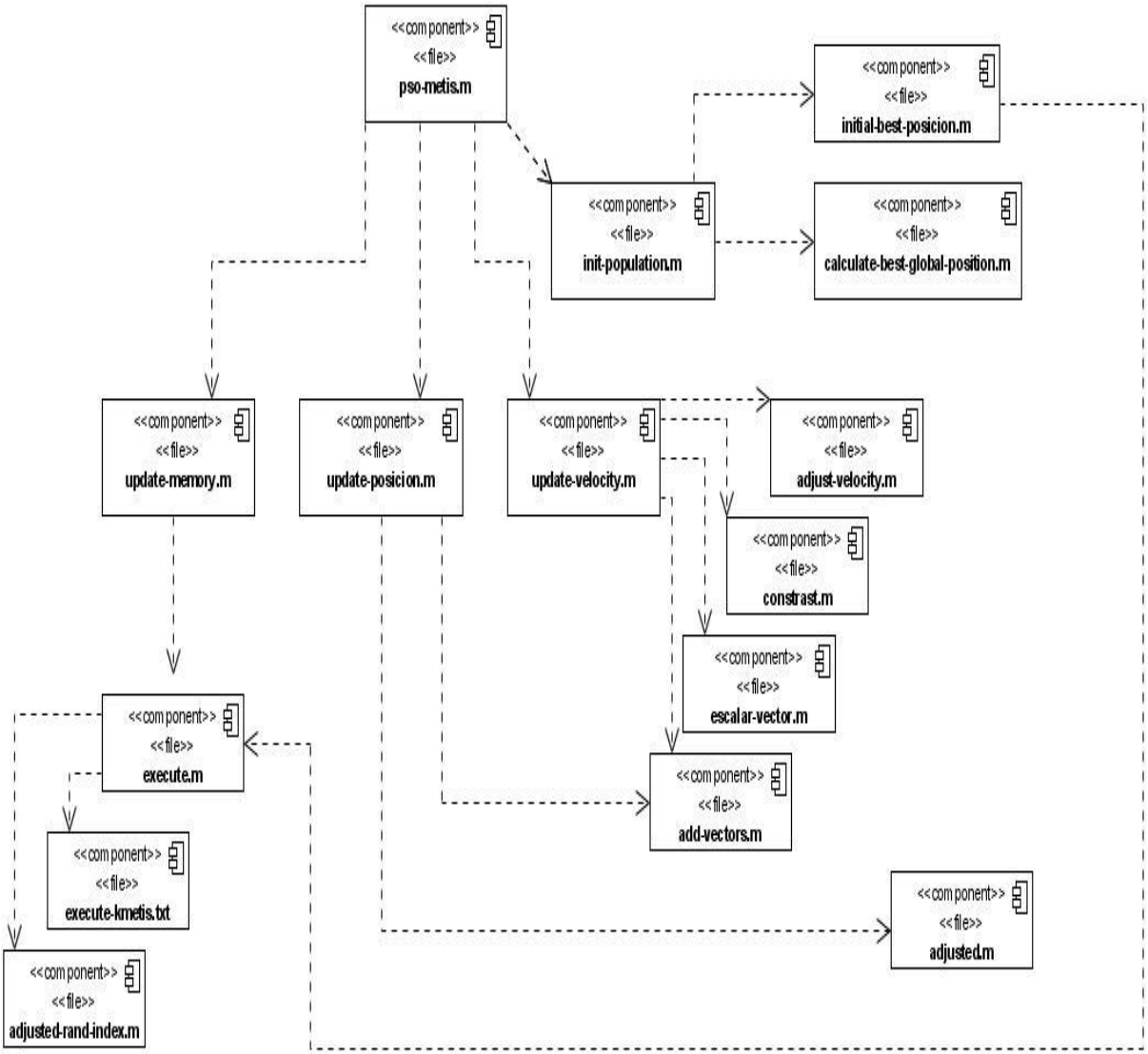


Figura 8: Diagrama de componentes del procedimiento de PSO para optimizar el parámetro K del algoritmo *kmetis.*

-Definición de parámetros para el PSO

Teniendo presente las características de este problema definimos una población de 100 partículas de una sola dimensión que representan los valores del parámetro K a optimizar y 200 iteraciones. Para determinar el valor mínimo del rango para la posición y la velocidad utilizamos la regla del pulgar (Kanti, 1979) que plantea: $K \approx \overline{N/2}$, el valor máximo lo determina el genoma con más genes en el conjunto de datos, en este caso es el S pombe con 4972 secuencias, por lo tanto el rango establecido es: [95,4972]. El resto de los parámetros asumen los valores W=0.8,C_1=C_2=2.

2.5.3 Resultados del PSO para K

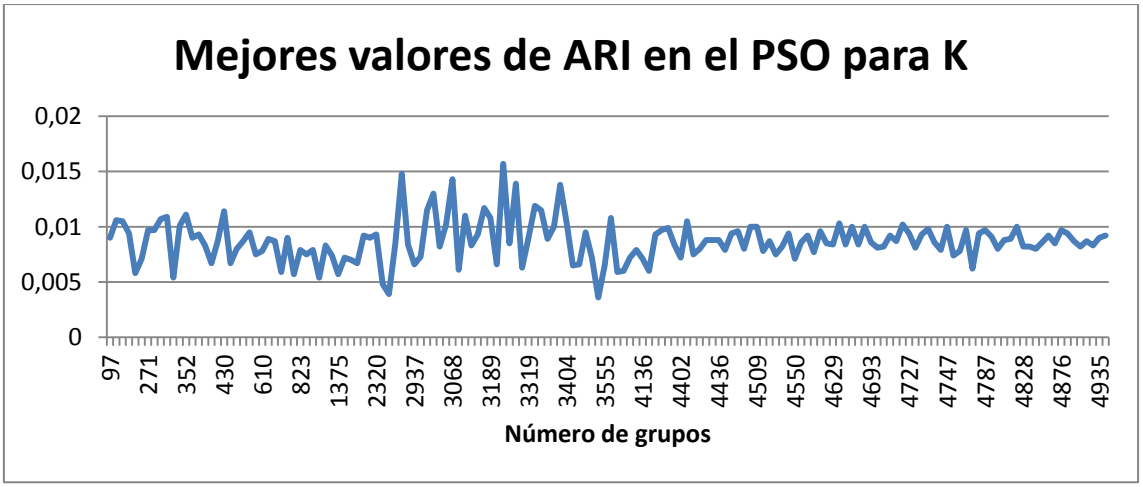


Figura 9: Gráfica que representa los mejores valores obtenidos en el PSO del parámetro K.

En la figura 9 se representan los resultados del proceso de PSO aplicado sobre el parámetro K del METIS, ilustramos los mejores valores de ARI obtenidos ante diferentes números de grupos, determinándose K=3263 como el valor óptimo ante el cual el METIS realiza la mejor clasificación de genes, tomando en cuenta el valor de ARI.

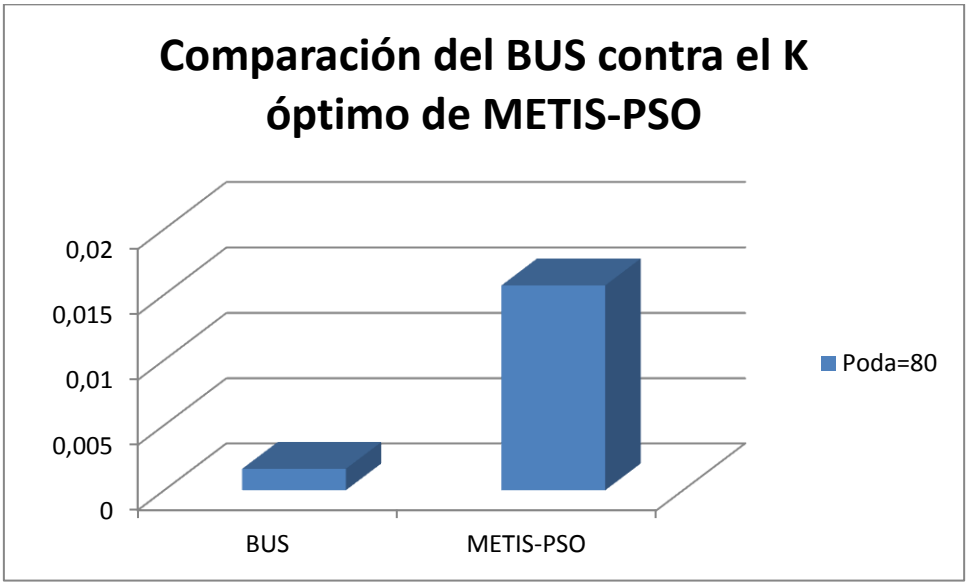


Figura 10: Figura que representa la comparación del BUS y el METIS en función de ARI tomando en cuenta el valor óptimo de K obtenido en el PSO.

Como podemos observar en la figura 10 ante el mejor valor obtenido de K mediante la optimización usando PSO, el METIS realiza una mejor clasificación que el BUS si tomamos en cuenta los valores de ARI resultantes.

Conclusiones Parciales

Se utilizó el METIS en la fase de agrupamiento de la detección de genes ortólogos y se compararon los resultados contra el BUS. Para mejorar la precisión se estima, usando PSO, el parámetro de poda que minimiza la Distancia de Hamming de la matriz podada con relación a la matriz binaria formada a partir de la base de SANGER. Luego se estima, usando PSO también, el parámetro de cantidad de grupos para el METIS que maximiza el ARI. La comparación de los resultados con el BUS muestra una mejora en la variante PSO por encima de la de BUS.

Conclusiones

-Se realizó un estudio de las técnicas de agrupamiento para grafos bipartitos

-Se aplicó la implementación del algoritmo METIS al problema de la detección de ortólogos.

-Se estimaron valores óptimos del parámetro de poda aplicando optimización basada en enjambres de partículas (Particle Swarm Optimization; PSO).

-Se implementó la optimización del parámetro de números de grupo del METIS usando PSO.

-Se validaron los resultados del agrupamiento con los datos de Saccharomyces Cerevisiae y Schizosaccharomyce Spombe utilizando como referencia los ortólogos curados manualmente de la base de datos SANGER.

Recomendaciones

- Probar otros algoritmos de agrupamiento sobre grafos bipartitos al problema de la detección de ortólogos.

- Utilizar una implementación paralela del METIS para reducir el tiempo computacional del agrupamiento y de la optimización de los parámetros.

- Estimar los valores de los parámetros de la poda y el número de grupos en solo algoritmo de PSO.

Referencias bibliográficas

ALEXEYENKO, A., ET AL. (2006) Automatic clustering of orthologs and inparalogs shared by multiple proteomes. Bioinformatics.

BADER, G., HOGUE, C.(2003) An automated method for finding molecular complexes in large protein interaction networks. BMC Bioinformatics.

BAENA, D.(2006) Análisis de datos de Expresión Genética mediante técnicas de Biclustering.

BEIELSTEIN, T., PARSOPOULOS, K. E. , VRAHATIS, M. N.(2002) Tuning PSO parameters through sensitivity analysis., Technical Report of the Collaborative Research Center, University of Dortmund
<http://citeseerx.ist.psu.edu/viewdoc/summary?doi=10.1.1.12.6598>.

BEN-DOR, A., SHAMIR, R., YAKHINI, Z.(1999) Clustering gene expression patterns.Journal of Computational Biology.

BONABEAU, E.(1999) Swarm Intelligence: From natural to artificial systems,Oxford University Press.

BROHUE, S., VAN HELDEN, J.(2006) Evaluation of clustering algorithms for protein-protein interaction networks.BMCBioinformatics.

CHÁVEZ, M.(2008) Modelos de redes bayesianas en el estudio de secuencias genómicas y otros problemas biomédicos. Universidad central “Marta Abreu” de las villas.

CHEN, X., ET AL.(2005) Assignment of Orthologous Genes via Genome Rearrangement.IEEE/ACM Trans. Comput.Biology Bioinform.

DEZA, E.(2006) Dictionary of Distances.

DORIGO, M., STÜTZLE, T.(2004) Ant Colony Optimization.

DORIGO, M., BIRATTARI, M.,STÜTZLE, T.(2006) Ant Colony OptimizationArtificial Ants as a Computational Intelligence Technique.IEEE Computational Intelligence Magazine 1.

DORIGO, M., STÜTZLE, T. (2007) An Introduction to Ant Colony Optimization. In T. F. Gonzalez, editor, Handbook of Approximation Algorithms and Metaheuristics, CRC Press.

DORIGO, M., ET AL. (2002) The Ant Colony Optimization Metaheuristic: Algorithms, Applications, and Advances.

ENGELBRECHT, A. P.(2006) Fundamentals of Computational Swarm Intelligence.John Wiley y Sons.

ESTOPIÑALES, M. (2009) Herramientas Computacionales para la Comparación de Genomas". Trabajo de Diploma, Ciencia de la Computación Cuba. Universidad Central de Las Villas.

FADI TOWFIC, ET AL.(2009) Detection of Gene Orthology Based On Protein-Protein Interaction Networks. IEEE International Conference on Bioinformatics and Biomedicine, BIBM. Washington, DC, USA.

FENG, C., ET AL. (2006) OrthoMCL-DB: querying a comprehensive multi-species collection of ortholog groups. Nucleic Acids Research.

FU, Z., ET AL. (2007) A High-Throughput Ortholog Assignment System Based on Genome Rearrangement. Journal of Computational Biology.

GABALDÓN, T. E. A. (2009) Joining forces in the quest for orthologs.

GABRIEL O¨ STLUND, T. S., ET AL.(2010) InParanoid 7: new algorithms and tools for eukaryotic orthology analysis. Nucleic Acids Research.

GOODSTADT, L., PONTING, C.P. (2006) Phylogenetic Reconstruction of Orthology, Paralogy, and Conserved Synteny for Dog and Human" PLoSComputBiol 2(9): e133. DOI: 10.1371/journal.pcbi.0020133.

GRAU, R., CHÁVEZ, M. C., SÁNCHEZ, R., MORGADO, E., CASAS, G. Y BONET, I.(2007) Boolean algebraic structures of the genetic code.Possibilities of applications. IN: TUYLS.

GUDISE, G., ET AL. (2003) Comparison Of Particle Swarm Optimization and Backpropagation as Training Algorithms for Neural Networks. In: Proceedings of the IEEE Swarm Intelligence Symposium. Indianapolis, USA.

HALKIDI, M., BATISTAKIS, Y., VAZIRGIANNIS, M.(2001) On clustering validation techniques. Intell.Inf.Syst. J.

HUBERT, A.(1985) Comparing partitions. Journal of Classification
KAMVYSSELIS, M. K., (2003) Computational comparative genomics: genes, regulation, evolution, in Department of Electrical Engineering and Computer Science. Massachusetts Institute of Technology: Massachusetts.

KANTI, M.(1979) Multivariate Analysis.Academic Press.

KARYPIS, G., KUMA,V.(1998) A fast and highly quality multilevel scheme for partitioning irregular graphs.SIAM Journal on Scientific Computing.

KARYPIS, G., KUMAR,V. (1998) MeTiS: A software package for partitioning unstructured graphs, partitioning meshes, and computing reducing orderings of sparse matrices, version 4.0. Technical report, University of Minnesota.

KARYPIS, G., KUMAR, V. (1998) Multilevel algorithms for multi-constraint graph partitioning. Department of Computer Science, University of Minnesota.

KARYPIS , G., ET AL. (1999) Hierarchical clustering using dynamic modeling.

KENNEDY, J., EBERHART, R. C.(1995a) A new optimizer using particle swarm theory. In: Sixth International Symposium on Micro Machine and Human Science.

KENNEDY, J., EBERHART, R. C.(1995b) Particle swarm optimization. In: Proceedings of IEEE International Conference on Neural Networks.

KENNEDY, J. (1997) The particle swarm: social adaptation of knowledge. International.Conference on Evolutionary Computation.

KENNEDY, J., SPEARS, W. M. (1998) Matching algorithms to problems: an experimental test of the particle swarm and some genetic algorithms on the multimodal problem generator. Proceedings of the IEEE International Conference on Evolutionary Computation.

KENNEDY, J., ET AL.(2001) Swarm Intelligence.Morgan Kaufmann Series in Artificial Intelligence.

KING, A., PRZULJ, N., JURISICA, I.(2000) Protein complex prediction via cost-based clustering.Bioinformatics.

KING, A., PRZULJ, N., JURISICA, I.(2004) Protein complex predictionvia cost-based clustering.Bioinformatics.

KUMAR., G., ET AL. (1995) Multilevel k-way partitioning scheme for irregular graphs. . Technical Report TR95-064, Department of Computer Science, University of Minnesota.

LEE Y, S., ET AL. (2002) Cross-referencing eukaryotic genomes: TIGR Orthologous Gene Alignments (TOGA). Genome Research.

LI LI, E. A. (2003) OrthoMCL: Identification of Ortholog Groups for Eukaryotic Genomes. Genome Research.

LIN, S., ET AL. (1970) An efficient heuristic procedure for partitioning graphs.

MIRKIN, B.(1996) Mathematical Classification and Clustering.Kluwer Academic Publishers.

O'BRIEN, K. P. (2005) Inparanoid: a comprehensive database of eukaryotic orthologs.Nucleic Acids Research

OMRAN, M., SALMAN, A., ENGELBRETCH.(2002) Image Classification Using Particle Swarm Optimizationin. In: Proceedings of the 4th Asia-Pacific Conference on Simulated Evolution and Learning. Singapore.

PARSOPOULOS, K., PAPAGEORGIOU, E., GROUMPOS, P., VRAHATIS. (2003) A first Study of Fuzzy cogntiveMapsLearning Using Particle Swarm Optimization. In: Proceedings of the IEEE Congress on Evolutionary Computation,.Canbella,Australia.

QUESADA, A., TIHANNY, L. (2010) GARLucene 2.0: Intermediación Diferencial y estimación de los parámetros aplicando una metaheurística bioinspirada. Universidad central "Marta Abreu" de las Villas

RAND, W.(1971) Objective criteria for the evaluation of clustring methods. . Journal of the Ameriacan Statistical Association.

RAND, W.(1976) Objective criteria for the evaluation of clustring methods. Journal of the AmeriacanStatistical Association 66

RASMUSSEN, M. (2005) Multi-BUS: An algorithm for resolving multi-species gene correspondence and gene family relationships.

RIJSBERGEN, C.(1979) Information Retrieval.second edition Butterworths

GUHA,S., ET AL. (1999) ROCK: A RobustClustering Algorithm for Categorical Attributes,". Proc.15th Int'l Conf. Data Eng., IEEE CS Press, Los Alamitos,Calif., pp. 512-521.

.

S.DHILLON, I. (2001) Co-clustering documents an words using bipartite spectral graph partition.

SANCHEZ R, ET AL. (2006) A Novel DNA Sequence Vector Space over an extended Genetic Code Galois Field MATCH Commun.

SHARAN, R. (2000) Click: A clustering algorithm for gene expression analysis. Proceedings of the 8th International Conference on Intelligent Systems for MolecularBiology

SHI, Y. ET AL.(1998) Parameter Selection in Particle Swarm Optimization. Proceedings of the Seventh Annual Conference on Evolutionary Programming: 591-601.

.

STEINBACH, M., KARYPIS,G., KUMAR,V.(2000) A Comparison of Document Clustering Techniques.

TANAY, A, ET AL.(2002) Discovering statistically significant biclusters in gene expression data.Bioinformatics.

TATUSOV, R., ET AL. (1997) A genomic perspective on protein families.Science.

TATUSOV, R. L. (2003) The COG database: an updated version includes eukaryotes. BMC Bioinformatics.

TETKO, I., FACIUS,A. RUEPP, A., ET AL., (2005) Super paramagnetic clustering of protein sequences. BMC Bioinformatics

WALL, D. P.(2003) Detecting putative ortholog.Bioinformatics.

WOOD, V.(2006) Schizosaccharomycespombe comparative genomics; from sequence to systems. In: Comparative Genomics using fungi as models

XIE, X., ET AL.(2002) Solving Numerical Optimization Problems by Simulating Particulates in Potencial Field with Cooperative Agents. In: Internation al Conference on Artifcial Intelligence, 2002 las Vegas, USA.