CENTRO DE CIENCIAS BÁSICAS

DEPARTAMENTO DE SISTEMAS DE INFORMACIÓN

TESIS

CLUSTERIZACIÓN DE HONGOS MEDIANTE METAHEURÍSTICAS HÍBRIDAS A PARTIR DE SU INFORMACIÓN PROTEÓMICA.

Presenta

LI. JESÚS IZAC RINCÓN MIRANDA.

QUE PARA OBTENER EL GRADO DE MAESTRÍA EN INFORMÁTICA Y TECNOLOGÍAS COMPUTACIONALES.

Tutora

Dra. MARÍA DOLORES TORRES SOTO.

Co-Tutora

Dra. AURORA TORRES SOTO.

Comité tutoral

Dr. CARLOS ARGELIO ARÉVALO MERCADO.

Aguascalientes, Aguascalientes a sábado, 28 de mayo de 2016

AUTORIZACIONES.

UNIVERSIDAD AUTONOMA DE AGUASCALIENTES

JESÚS IZAC RINCÓN MIRANDA
MAESTRÍA EN INFORMÁTICA Y TECNOLOGÍAS COMPUTACIONALES
P R E S E N T E.

Estimado alumno:

Por medio de este conducto me permito comunicar a Usted que habiendo recibido los votos aprobatorios de los revisores de su trabajo de tesis y/o caso práctico titulado: **"Clusterización de hongos mediante metaheurísticas híbridas a partir de su información proteómica",** hago de su conocimiento que puede imprimir dicho documento y continuar con los trámites para la presentación de su examen de grado.

Sin otro particular me permito saludarle muy afectuosamente.

A T E N T A M E N T E
Aguascalientes, Ags., a 26 de mayo de 2016
"Se lumen proferre"
EL DECANO

M. en C. JOSE DE JESUS RUIZ GALLEGOS

c.c.p.- Archivo.

UNIVERSIDAD AUTONOMA DE AGUASCALIENTES

FORMATO DE CARTA DE VOTO APROBATORIO

M.C. JOSÉ DE JESÚS RUÍZ GALLEGOS
DECANO (A) DEL CENTRO DE CIENCIAS BÁSICAS

PRESENTE

Por medio del presente como Tutor designado del estudiante ***JESÚS IZAC RINCÓN MIRANDA*** con ID 69327 quien realizó *el trabajo de tesis* titulado: **CLUSTERIZACIÓN *DE HONGOS MEDIANTE METAHEURÍSTICAS HÍBRIDAS A PARTIR DE SU INFORMACIÓN PROTEÓMICA***, y con fundamento en el Artículo 175, Apartado II del Reglamento General de Docencia, me permito emitir el **VOTO APROBATORIO**, para que *él* pueda proceder a imprimirla, y así como continuar con el procedimiento administrativo para la obtención del grado.

Pongo lo anterior a su digna consideración y sin otro particular por el momento, me *permito enviarle un cordial saludo.*

A T E N T A M E N T E
"Se Lumen Proferre"
Aguascalientes, Ags., a 20 de *mayo* de *2016.*

Dra. María Dolores Torres Soto
Tutora de *tesis*

c.c.p.- Interesado
c.c.p.- Secretaría de Investigación y Posgrado
c.c.p.- Jefatura del Depto. de Sistemas de Información
c.c.p.- Consejero Académico
c.c.p.- Minuta Secretario Técnico

UNIVERSIDAD AUTÓNOMA DE AGUASCALIENTES

FORMATO DE CARTA DE VOTO APROBATORIO

M.C. JOSÉ DE JESÚS RUÍZ GALLEGOS
DECANO (A) DEL CENTRO DE CIENCIAS BÁSICAS

PRESENTE

Por medio del presente como Co-Tutor designado del estudiante ***JESÚS IZAC RINCÓN MIRANDA*** con ID 69327 quien realizó *el trabajo de tesis* titulado: **CLUSTERIZACIÓN *DE HONGOS MEDIANTE METAHEURÍSTICAS HÍBRIDAS A PARTIR DE SU INFORMACIÓN PROTEÓMICA***, y con fundamento en el Artículo 175, Apartado II del Reglamento General de Docencia, me permito emitir el **VOTO APROBATORIO**, para que *él* pueda proceder a imprimirla, y así como continuar con el procedimiento administrativo para la obtención del grado.

Pongo lo anterior a su digna consideración y sin otro particular por el momento, me *permito enviarle un cordial saludo.*

A T E N T A M E N T E
"Se Lumen Proferre"
Aguascalientes, Ags., a 20 de *mayo* de *2016*.

Dra. Aurora Torres Soto
Co-Tutora de *tesis*

c.c.p.- Interesado
c.c.p.- Secretaría de Investigación y Posgrado
c.c.p.- Jefatura del Depto. de Sistemas de Información
c.c.p.- Consejero Académico
c.c.p.- Minuta Secretario Técnico

UNIVERSIDAD AUTÓNOMA DE AGUASCALIENTES

FORMATO DE CARTA DE VOTO APROBATORIO

M.C. JOSÉ DE JESÚS RUÍZ GALLEGOS
DECANO (A) DEL CENTRO DE CIENCIAS BÁSICAS

PRESENTE

Por medio del presente como Asesor designado del estudiante ***JESÚS IZAC RINCÓN MIRANDA*** con ID 69327 quien realizó *el trabajo de tesis* titulado: ***CLUSTERIZACIÓN DE HONGOS MEDIANTE METAHEURÍSTICAS HÍBRIDAS A PARTIR DE SU INFORMACIÓN PROTEÓMICA***, y con fundamento en el Artículo 175, Apartado II del Reglamento General de Docencia, me permito emitir el **VOTO APROBATORIO**, para que *él* pueda proceder a imprimirla, y así como continuar con el procedimiento administrativo para la obtención del grado.

Pongo lo anterior a su digna consideración y sin otro particular por el momento, me *permito enviarle un cordial saludo.*

A T E N T A M E N T E
"Se Lumen Proferre"
Aguascalientes, Ags., a 20 de *mayo* de *2016*.

Dr. Carlos Argelio Arévalo Mercado
Asesor de *tesis*

c.c.p.- Interesado
c.c.p.- Secretaría de Investigación y Posgrado
c.c.p.- Jefatura del Depto. de Sistemas de Información
c.c.p.- Consejero Académico
c.c.p.- Minuta Secretario Técnico

AGRADECIMIENTOS.

Aprovecho este espacio para agradecer el invaluable apoyo que he tenido por todos y cada uno de quienes de alguna manera han contribuido para culminar esta importante etapa de mi vida. A pesar de las barreras y obstáculos, sé que he dado todo mi esfuerzo por alcanzar este sueño y logro que comparto con todos ustedes.

A mis padres, Esperanza Miranda y José Rincón por haberme dado la vida misma, porque me han brindado su apoyo y amor de forma incondicional; sin dudarlo han estado para apoyarme con un consejo, con una palabra que me fortalece para seguir adelante en este sueño llamado vida. Porque son un ejemplo de tenacidad y perseverancia. Gracias por confiar en mí.

A mis hermanos Eric, Lucero, Máyela y Fátima que se preocupan siempre por mí, me demuestran su gran apoyo para mantener unida a la familia, también me hacen sentir orgulloso de saber que puedo ser su consejero y un ejemplo a seguir, claro además de que me permiten disfrutar de los nuevos miembros de la familia (Christian, Oscar, Yelitza, Abigail).

A Memo, un gran ser humano a quien tengo la dicha de conocer y convivir, alguien que se preocupa por mí, me entiende, me escucha, comparte mis ideas y mi sentir por muy rebuscado que sea.

A la maestra Dolores Torres Soto (Lolita), una persona a quien admiro y respeto mucho, alguien que ha estado para escucharme, para apoyarme en muchos de los momentos difíciles que he tenido. Además de ello mi directora de tesis de quien he aprendido mucho y es un ejemplo a seguir en lo humano, académico y profesional. Gracias por permitirme colaborar y seguir despertando la curiosidad que siempre he tenido en el mundo de la investigación, gracias impulsarme a seguir y confiar en mí.

A la Dra. Aurora Torres, una persona firme y decisiva, ejemplo de tenacidad y conocimiento, sin duda parte de mi formación y de este logro.

Al Dr. Carlos Arévalo, por su apoyo y la confianza que ha depositado en mí.

A mis amigos, compañeros de trabajo, de carrera, ya que cada uno de ustedes ha contribuido a su manera en mi crecimiento mediante una palabra, una frase, una felicitación o incluso hasta un regaño pero siempre creyendo en mí. Don Héctor, CPC. Graciela, Armando, Dra. Loe, Gaby, Denise, Mtra. Liz, Mtro. Paco, Mtra. Paty, Nachita (mi abuelita), Paz, Dr. Mora, Héctor, Mtra. Lorena, Mtra. Lolita, Vianney, Cesar, David. No quiero omitir a nadie de ser así pido disculpas pero saben que tengo un espacio en mi corazón para todos ustedes.

A mis maestros, que gracias a ellos y a su invaluable conocimiento me han permitido llegar hasta este punto.

A la universidad misma (UAA), mi alma mater que ha contribuido a mi formación desde la licenciatura, mi lugar de trabajo bajo un ambiente de sana convivencia y muy agradable.

A CONACYT, agradezco por apoyarme hasta donde le fue posible hacerlo.

A todos ustedes les doy las gracias

DEDICATORIAS.

A mis padres y hermanos, Memo, amigos y maestros. ¡GRACIAS POR CREER EN MÍ!

ÍNDICE GENERAL.

Resumen. 8
Abstract. 9
Capítulo I: Introducción. 10
1. Antecedentes. 10
1. Trabajos relacionados con bioinformática. 11
2. Trabajos donde se aplica clusterización en bioinformática. 12
2. Problema de Investigación. 14
1. Problema. 14
2. Problemática. 14
3. Objetivos de la investigación. 16
1. Objetivo General. 16
2. Objetivos Específicos. 16
4. Preguntas de Investigación. 17
5. Justificación. 18
6. Estructura del Trabajo. 20
Capitulo II: Marco Teórico. 22
1. Inteligencia Artificial. 22
2. Clusterización. 25
Técnicas de Clusterización. 28
3. Metaheurísticas. 33
1. Introducción a la Optimización. 33
2. Heurística. 35
3. Metaheurística. 38
4. Hibridación. 59
1. Antecedentes. 59
2. Clasificación. 61
5. Fundamentos básicos de Bioinformática y Proteómica. 63
1. Bioinformática. 63
2. Proteómica. 66
Capítulo III: Metodología. 71
1. Antecedentes y Caso de Estudio. 72
2. Diseño de la Investigación. 75
Aportaciones, y Generalidades de los Algoritmos. 75
Esquema de Trabajo. 77
Capítulo IV: Metaheurística Evolutiva AGH-CHIP. 79

1. Mecanismos Diseñados. 79
Función de Adaptabilidad. 79
Mutación Inteligente. 82
2. Generalidades. 85
1. Establecer parámetros iniciales. 86
2. Mecanismo de Representación. 88
3. Generar Población Inicial. 88
4. Mecanismo para Calificar cada Individuo. 90
5. Método de Ordenamiento. 93
6. Mecanismos para generar las siguientes poblaciones. 95
7. Establecer Criterio de Paro. 98
3. Experimentación. 99
1. Condiciones Generales. 99
2. Diseño de Experimentos. 99
3. Experimentos. 102
4. Pruebas estadísticas. 106
5. Interpretación de Resultados. 109
6. Aportes. 112
7. Conclusiones del AGH-CHIP. 124
Capítulo V: Metaheurística Evolutiva UMDA - CHIP. 125
1. Mecanismos Diseñados. 125
Función de Adaptabilidad. 125
2. Generalidades. 129
1. Establecer parámetros iniciales. 132
2. Mecanismo de Representación. 133
3. Generar Población Inicial. 134
4. Mecanismo para calificar cada individuo (función de adaptabilidad). 135
5. Método de Ordenamiento. 140
6. Mecanismo para Estimar la Distribución de Probabilidad de la Población. 141
7. Mecanismos para generar las siguientes poblaciones. 143
8. Establecer Criterio de Paro. 146
3. Experimentación. 147
1. Condiciones Generales. 147
2. Diseño de Experimentos. 147
3. Experimentos. 150
4. Pruebas Estadísticas. 153
5. Interpretación de Resultados. 156
6. Conclusiones del UMDA-CHIP. 160

Capítulo VI: Conclusiones. 161
1. Conclusiones Generales. 161
2. Conclusiones de los Algoritmos. 163
3. Principales Contribuciones de la Tesis. 165
4. Conclusiones de los Objetivos de la Investigación. 166
5. Limitaciones. 168
6. Trabajo Futuro. 169
Glosario. 170
Bibliografía. 172
ANEXOS. 193
ANEXO A. 193
ANEXO B. 196
ANEXO C. 209
ANEXO D. 210
ANEXO E. 221
ANEXO F. 222
ANEXO G. 224

ÍNDICE DE TABLAS.

TABLA 1- ALGUNAS DEFINICIONES DE LOS ENFOQUES DENTRO DE LA IA. 23
TABLA 2 - FORMAS DE CLASIFICACIÓN DE LA METAHEURÍSTICAS (GÓMEZ, 2014). 40
TABLA 3 - TABLA DE EJEMPLO PARA ILUSTRAR LA SELECCIÓN POR RULETA. 50
TABLA 4 - EJEMPLO DE UNA MATRIZ DE SEMEJANZAS. 72
TABLA 5 - LISTA DE HONGOS CONSIDERADOS EN LA INVESTIGACIÓN. 73
TABLA 6 - EJEMPLO DE MATRIZ DE SEMEJANZAS. 80
TABLA 7 - EJEMPLO DE MATRIZ DE SEMEJANZAS (2). 83
TABLA 8 - DESCRIPCIÓN DE PARÁMETROS DEL AGH-CHIP. 86
TABLA 9 - EJEMPLO DE POBLACIÓN. 90
TABLA 10 – EJEMPLO DE CÁLCULO DEL VALOR DE ADAPTABILIDAD DEL AGH-CHIP. 91
TABLA 11 - FACTORES CONSIDERADOS PARA EL EXPERIMENTO. 100
TABLA 12 - DISEÑO FACTORIAL SOBRE EL AGH-CHIP. 100
TABLA 13 - ELEMENTOS QUE CONFORMAN EL ARCHIVO DE SALIDA DE CADA REPLICA. 101
TABLA 14 - DESCRIPCIÓN DE LOS PARÁMETROS DE SALIDA. 102
TABLA 15 – RESUMEN DE RESULTADOS DEL EXPERIMENTO AGH-CHIP. 103
TABLA 16 - PRUEBA DE HOMOGENEIDAD DE VARIANZAS. 107
TABLA 17 - RESULTADOS DE LA PRUEBA DE KRUSKAL-WALLIS. 108
TABLA 18 - CLASIFICACIÓN PROPUESTA POR EL AGH-CHIP. 110
TABLA 19 - CONJUNTO DE TABLAS CON INFORMACIÓN BIOLÓGICA BÁSICA DE LOS HONGOS DE ESTUDIO. 112
TABLA 20 - DESCRIPCIÓN DE PARÁMETROS DEL UMDA-CHIP. 132
TABLA 21 - EJEMPLO DE POBLACIÓN DE CLÚSTERES. 135
TABLA 22 - PASOS DEL EDA-UMDA (CÁLCULO DE SEMEJANZA). 136
TABLA 23 - PASOS DEL EDA-UMDA (CALCULO DIFERENCIAS). 138
TABLA 24 - EJEMPLO DE ESTIMACIÓN DE LA MATRIZ DE PROBABILIDADES. 141
TABLA 25 - EJEMPLO DE ESTIMACIÓN DE INDIVIDUOS. 144
TABLA 26 - FACTORES CONSIDERADOS PARA EL EXPERIMENTO. 148
TABLA 27 - DISEÑO FACTORIAL SOBRE EL UMDA-CHIP. 148
TABLA 28 - ELEMENTOS QUE CONFORMAN EL ARCHIVO DE SALIDA DE CADA REPLICA. 149
TABLA 29 - DESCRIPCIÓN DE LOS PARÁMETROS DE SALIDA. 149
TABLA 30 – RESUMEN DE RESULTADOS DEL EXPERIMENTO. 150
TABLA 31 - PRUEBA DE KRUSKAL-WALLIS UMDA-CHIP. 154
TABLA 32 - CLASIFICACIÓN DE LOS HONGOS BASADA EN CUATRO FAMILIAS. 157
TABLA 33 - CLASIFICACIÓN PROPUESTA POR EL UMDA-CHIP. 158

ÍNDICE DE FIGURAS.

FIGURA 1 - MAPA CONCEPTUAL DEL PROBLEMA. 15
FIGURA 2 - OBJETIVOS DE LA CLUSTERIZACIÓN A PARTIR DE (PANDRE, 2011). 26
FIGURA 3 – DIFERENTES CLÚSTERES PARA LOS MISMOS DATOS (DOS DIMENSIONES) (JUSTEL, 2012). 27
FIGURA 4 - DENDOGRAMA DE UN CONJUNTO DE DATOS (EDNA, 2006). 29
FIGURA 5 - EJEMPLO DE CLUSTERIZACIÓN PARTICIONAL. 30
FIGURA 6 - AGRUPAMIENTO BASADO EN DENSIDAD CON DBSCAN (CHIRE, 2011). 31
FIGURA 7 – EJEMPLO DE ESPACIO DE SOLUCIONES. 37
FIGURA 8 - FUNCIONAMIENTO DE LAS METAHEURÍSTICAS (HERRERA, 2006). 39
FIGURA 9- ALGORITMO GENÉTICO SIMPLE (GOLBERG, 1989). 45
FIGURA 10 - REPRESENTACIÓN SIMBÓLICA BINARIA. 46
FIGURA 11 - REPRESENTACIÓN SIMBÓLICA REAL. 46
FIGURA 12 - REPRESENTACIÓN SIMBÓLICA ENTERA. 46
FIGURA 13 – EJEMPLO DE GENERACIONES 47
FIGURA 14 - RULETA QUE REPRESENTA LOS PORCENTAJES DE LA TABLA 3. 50
FIGURA 15 - CRUZAMIENTO SIMPLE O DE UN SOLO PUNTO. 51
FIGURA 16 - CRUZAMIENTO CON DOS PUNTOS DE CORTE. 52
FIGURA 17 - OPERADOR DE CRUCE CON MASCARA. 52
FIGURA 18 - METODOLOGÍA GENERAL DE UN EDA (ABDELMALIK MOUJAHID ET AL., 2015) A PARTIR DE (LARRANAGA & LOZANO, 2001). 56
FIGURA 19 - PSEUDOCÓDIGO DE LA APROXIMACIÓN DE UN EDA (LARRANAGA & LOZANO, 2001). 57
FIGURA 20 - CLASIFICACIÓN DE LAS METAHEURÍSTICAS HÍBRIDAS. 62
FIGURA 21 - MARCO DE TRABAJO DE LA INVESTIGACIÓN. 77
FIGURA 22 - EJEMPLO DE SOLUCIÓN (1). 80
FIGURA 23 - EJEMPLO DE SOLUCIÓN (3) 83
FIGURA 24 - EJEMPLO DE SOLUCIÓN CON MMI. 84
FIGURA 25- ALGORITMO GENÉTICO SIMPLE (GOLBERG, 1989) 85
FIGURA 26 - PSEUDOCÓDIGO SIMPLIFICADO DEL AGH-CHIP. 86
FIGURA 27 - EJEMPLO DE REPRESENTACIÓN DE CLÚSTERES EN UN CROMOSOMA. 88
FIGURA 28 – DFD DE LA FUNCIÓN DE ADAPTABILIDAD DEL AGH (SEMEJANZA) 92
FIGURA 29 - EJEMPLO DE POBLACIÓN COMPLETA. 93
FIGURA 30 - PSEUDOCÓDIGO DEL ALGORITMO DE LA BURBUJA. 94
FIGURA 31 - ELEMENTOS A ORDENAR. 95
FIGURA 32 - EJEMPLO DE SELECCIÓN ELITISTA. 96
FIGURA 33 - CAPTURA DE PANTALLA DEL AGH-CHIP EN EJECUCIÓN. 103
FIGURA 34 - PRUEBA DE NORMALIDAD DEL VALOR DE ADAPTABILIDAD. 106
FIGURA 35 - PRUEBA DE NORMALIDAD DEL TIEMPO. 107
FIGURA 36 - EJEMPLO DE ARCHIVOS DONDE SE ENCONTRÓ EL MEJOR VALOR DE ADAPTABILIDAD. 109
FIGURA 37 - EJEMPLO DE SOLUCIÓN (2) 127
FIGURA 38 - EJEMPLO DE CÁLCULO DE DEG. 128
FIGURA 39 - PSEUDOCÓDIGO DE LA APROXIMACIÓN DE UN EDA (LARRANAGA & LOZANO, 2001) 129

FIGURA 40 - PSEUDOCÓDIGO DEL UMDA (MÜHLENBEIN, 1997; MÜHLENBEIN & PAASS, 1996). 130
FIGURA 41 - SEUDOCÓDIGO SIMPLIFICADO DEL UMDA-CHIP. 131
FIGURA 42 - EJEMPLO DE CLÚSTERES CREADOS EN UN INDIVIDUO. 133
FIGURA 43 - EJEMPLO DE POBLACIÓN COMPLETA 140
FIGURA 44 - EJEMPLO DE SELECCIÓN POR TRUNCAMIENTO. 144
FIGURA 45 - PRUEBA DE NORMALIDAD DEL VALOR DE ADAPTABILIDAD. 153
FIGURA 46 - PRUEBA DE NORMALIDAD DEL TIEMPO. 154
FIGURA 47 - CAPTURA DE PANTALLA DE DONDE SE ENCONTRÓ EL VALOR MÁXIMO. 156

ÍNDICE DE ECUACIONES.

ECUACIÓN 1 57
ECUACIÓN 2 80
ECUACIÓN 3 126
ECUACIÓN 4 126
ECUACIÓN 5 127
ECUACIÓN 6 130

Resumen.

La ciencia y la tecnología se han convertido en medios importantes de la vida de los seres humanos desde hace décadas y constituyen hoy en día, un poderoso pilar del desarrollo cultural, social, económico y, en general, de la vida en la sociedad moderna y cambiante.

Por otra parte las técnicas de inteligencia artificial, representan un poderoso instrumento para resolver muchos problemas de cualquier área del conocimiento humano y en particular los métodos metaheurísticos han demostrado obtener muy buenos resultados en un tiempo y costo computacional aceptable. Dichas técnicas, apoyando a ciencias como la bioinformática en áreas como la proteómica, tienen un futuro prometedor.

En la introducción de este documento se presentó el problema que dio origen a la investigación, el cual consiste en crear clústeres de hongos mediante su información proteómica, posteriormente en los objetivos se planteó una propuesta para atacar el problema.

A partir de lo anterior, durante el desarrollo de esta investigación se logró consolidar un marco de trabajo robusto y flexible que permite la aplicación de técnicas metaheurísticas en problemas normalmente intratables como la clusterización (NP-completos). Además, se logró desarrollar y poner a punto dos técnicas metaheurísticas que a partir de una matriz de semejanzas proteómicas llevan a cabo la creación de clústeres.

Cabe destacar que una de las técnicas se combinó con un operador de mejora MMI (Mecanismo de Mutación Inteligente), que es una aportación valiosa al área de conocimientos para la tarea de clusterización y que reportó muy buenos resultados.

Los resultados empíricos de los experimentos fueron confirmados estadísticamente y se lograron identificar los mejores parámetros para cada algoritmo. Los resultados dentro del área de la aplicación fueron confirmados por expertos en el área de bioinformática y de biología de nuestra universidad.

Otro hecho a resaltar son los resultados mismos, ya que al llevar una minuciosa investigación en la literatura existente se logró comprobar que cada uno de los algoritmos empató sus resultados de una forma muy aceptable con una clasificación utilizada por investigadores.

Abstract.

Science and technology have been important areas in the life of human beings for decades and represent a powerful pillar of cultural, social, economic development and, in general, life in modern society and changing.

Artificial intelligence techniques represent a powerful tool for solving problems in many areas of human knowledge and in particular metaheuristic methods have shown to produce very good results, with acceptable computational time and cost. Such techniques, in sciences like bioinformatics and areas related to proteomics, have a promising future.

The problem that originated the research is described, which consists in creating clusters of fungi through their proteomic information. In the objectives section a potential solution is proposed in order to address the problem.

During the development of this research a robust and flexible framework was consolidated, which allows the enforcement of metaheuristics techniques in problems usually untreatable like clustering (NP-hard). In addition, it was possible to develop and tune two metaheuristics techniques that from a similarity matrix perform the creation of clusters.

It is important to notice that one of the techniques is combined with an operator of improvement named MMI (Mecanismo de Mutación Inteligente, for its acronym in Spanish); it is a valuable contribution into the knowledge in clustering tasks and has reported good results.

Empirical results were statistically confirmed and it was possible to identify the best parameters for each algorithm. The results within the knowledge area were confirmed by experts in the field of bioinformatics and biology of our university.

Thorough a literature review it was possible to verify that each algorithm matched their results with an acceptable way to classify used by the researchers.

Capítulo I: Introducción.

1. Antecedentes.

La ciencia y la tecnología han llegado a convertirse en medios importantes de la vida de los seres humanos desde hace décadas y constituyen hoy en día, un poderoso pilar del desarrollo cultural, social, económico y, en general, de la vida en la sociedad moderna y cambiante.

Tanto el auge en las técnicas para secuenciar el ADN (Ácido Desoxirribonucleico) y las proteínas, así como el volumen cada vez mayor de secuencias almacenadas en los bancos de datos, favorecieron la necesidad de creación de algoritmos a fin de catalogar y comparar secuencias, en los que se reconoce como pionera a Margaret Oakley Dayhoff (1925-1983), connotada investigadora del Centro Médico de la Universidad de Georgetown.

La doctora Dayhoff desarrolló métodos computacionales que le permitieron comparar secuencias de proteínas y a partir de los alineamientos entre ellas investigar las relaciones y por ende el proceso evolutivo entre los diferentes reinos, phyla y taxa biológicos. Su monumental trabajo, que recopilaba todas las secuencias proteicas entonces conocidas, se publicó en 1965 en un pequeño libro titulado “Atlas de secuencia y estructura de proteínas” (Franco, Cediel, & Payán, 2008).

De aquí, surge la bioinformática como un área relativamente nueva en el procesamiento y descubrimiento de relaciones en la enorme cantidad de datos que se generan en esta y otras ciencias y representan un gran reto para los sistemas computacionales actuales.

Por ello, “*las herramientas computacionales y la inteligencia artificial son cruciales para almacenar e interpretar estos datos de un modo eficiente y robusto*” (Altamiranda, Aguilar, & Hernández, 2008).

Entrando en contexto general sobre la problemática a la que se pretende dar solución en el presente trabajo, se tiene en primera instancia una matriz de semejanzas proteómicas de un grupo de hongos, el objetivo es aplicar técnicas metaheurísticas a fin de crear clústeres únicamente con su información proteómica.

Al llevar a cabo una revisión de la literatura se tiene que es un tema relativamente nuevo, es decir pocos autores se han enfocado en buscar clasificaciones diferentes de las que ya existen. Puede consultarse los trabajos de (Cooke, 1958; Guarro, 2012; Guarro, Gené, & Stchigel, 1999; Montes, Restrepo, & McEwen, 2003).

Sin embargo, en seguida se hace mención de algunas investigaciones relacionados con los primeros avances prácticos de la bioinformática, así como algunos trabajos recientes relacionados a la aplicación de clustering (agrupamiento) en problemas de bioinformática.

1. Trabajos relacionados con bioinformática.
2. Trabajos donde se aplica clusterización en bioinformática.

1. Trabajos relacionados con bioinformática.

- **"Proyecto del Genoma Humano".** Ha sido la primera gran incursión de las comunidades de investigación biológica y médica.
  Es un esfuerzo conjunto y coordinado de la comunidad internacional para clonar y secuenciar el genoma humano completo, este audaz proyecto, estimo un costo de 200 millones de dólares por año y un tiempo de 15 años en ser completado, prometió ser uno de los más revolucionarios y cautivadores esfuerzos científicos jamás concebido por la humanidad. Al conocer la secuencia de los cerca de 3 mil millones de pares de bases del genoma humano haploide y sus más de 30.000 genes (Sawicki, Samara, Hurwitz, & Passaro Jr., 1993).
- **"Comparación de secuencias".** El principal objetivo de la comparación de secuencias es analizar la semejanza entre estas para tratar de encontrar homología entre ellas. El interés en dicho análisis se debe a que las bases de datos de proteínas se organizan por familias con estructura similar, funciones similares, historial evolutivo similar, por tanto. Según Nieto[1] "*Una nueva proteína será admitida en la familia si tiene simultáneamente similitud con los miembros actuales de la familia y no sólo con una de ellas*" (Nieto, 2005).

[1] Consultado en http://mathgene.usc.es/cursoverano/cv2005/materiales/Nieto/ComparacionSecuencias.pdf (14/04/2016).

2. Trabajos donde se aplica clusterización en bioinformática.

- **"Use of proteomic patterns in serum to identify ovarian cancer".** En este trabajo, se desarrolló una herramienta bioinformática por medio de un algoritmo genético, dicha herramienta fue utilizada para identificar patrones proteómicos en el suero con la finalidad de distinguir (clasificar) la neoplásica de la no-neoplásica en el ovario (Petricoin III et al., 2002) y (Zapico Muñiz, Mora Brugés, & Blanco Vaca, 2005).

- **"Inferencia filogenética molecular en ambiente de alto Procesamiento computacional".** Dicha investigación se enfocó en proveer a la comunidad científica de un módulo de Análisis Filogenético empleando los programas PhyML y MrBayes, el cual aprovecha el clúster de computadoras Átropos del departamento de Bioinformática de la Universidad de las Ciencias Informáticas (UCI), con el fin de disminuir el tiempo de respuesta de los análisis filogenéticos, con lo que se contribuyendo al desarrollo de la Bioinformática en Cuba (Ledesma Tamayo, Tamayo, & Pérez, 2014).

- **"Árbol filogenético de hongos construido por medio de un agrupamiento jerárquico".** En este trabajo se llevó a cabo una investigación la cual concluyó que a partir de la metodología que se usó, basada en proteomas y mejores aciertos bidireccionales se logró obtener un árbol filogenético que agrupa los hongos de manera similar a las agrupaciones obtenidas con métodos basadas en sus características físicas (Ponce de León Sentí, Díaz, Martínez Guerra, Torres Soto, & Torres Soto, 2014).

- "**Aplicación de un algoritmo evolutivo híbrido para la clasificación hongos**". En este trabajo práctico se encontró que a partir de la información proteómica de los hongos estudiados fue posible hacer una clasificación a dos grupos (Levaduras y No Levaduras) (Rincón Miranda, Torres Soto, & Torres Soto, 2014).

- **“Classifying Parasitic Fungi by their Proteome using a Univariate Estimation of Distribution Algorithm with a Simplified Design”.** En ésta investigación el autor desarrollo y aplicó un EDA con un diseño simplificado en un problema de clasificación de hongos en base a su proteoma y se obtuvo un resultado que contrasta con una clasificación filogenética existente (Mendoza, 2011).

- **“KClust: fast and sensitive clustering of large protein sequence databases”.** En este trabajo se desarrolló un método para agrupar secuencias de proteínas en bases de datos grandes, con una reducción considerable en tiempo (Hauser, Mayer, & Söding, 2013).

2. Problema de Investigación.

1. Problema.

Encontrar un mecanismo para clusterizar hongos, por medio de técnicas metaheurísticas utilizando únicamente la información de semejanza proteómica de hongos.

2. Problemática.

El resultado del trabajo de investigación que se ha llevado a cabo por diferentes autores, hace que se cuente con algunos mecanismos para clasificar organismos biológicos a partir de sus características físicas como son: su color, textura, tamaño, forma, etc., es decir su fenotipo; incluso existe una clasificación milenaria denominada "*árbol filogenético*" que es una clasificación científica de las especies, basada únicamente en las relaciones de proximidad evolutiva entre las distintas especies, reconstruyendo la historia de su diversificación (filogénesis) desde el origen de la vida en la tierra hasta la actualidad (Babitsch Soler, 2010).

Con base a lo anterior se cree que a partir de la información de semejanza proteómica entre entes biológicos, es posible distinguir especies, de ahí que surge la inquietud de encontrar un instrumento para hacer este trabajo y para la investigación que se llevó a cabo, un grupo de hongos en particular fueron el objeto de interés.

Teniendo como insumo principal el resultado (matriz de semejanzas) de la investigación: "Aprendizaje heurístico de modelos probabilísticos para identificación de secuencias genómicas" realizada en 2014-2015 por las investigadoras: Eunice Esther Ponce de León Sentí, María Dolores Torres Soto, Aurora Torres Soto, Elva Díaz; se busca encontrar un mecanismo para clasificar un grupo de hongos, de hecho el mismo grupo de hongos, pero para este trabajo se pretendió establecer si con la información de la semejanza proteómica entre dichos hongos es posible establecer una relación que permita su agrupamiento (clusterización).

La Figura 1 representa la interacción que existe entre las disciplinas que apoyan la investigación llevaba a cabo para dar solución de la problemática.

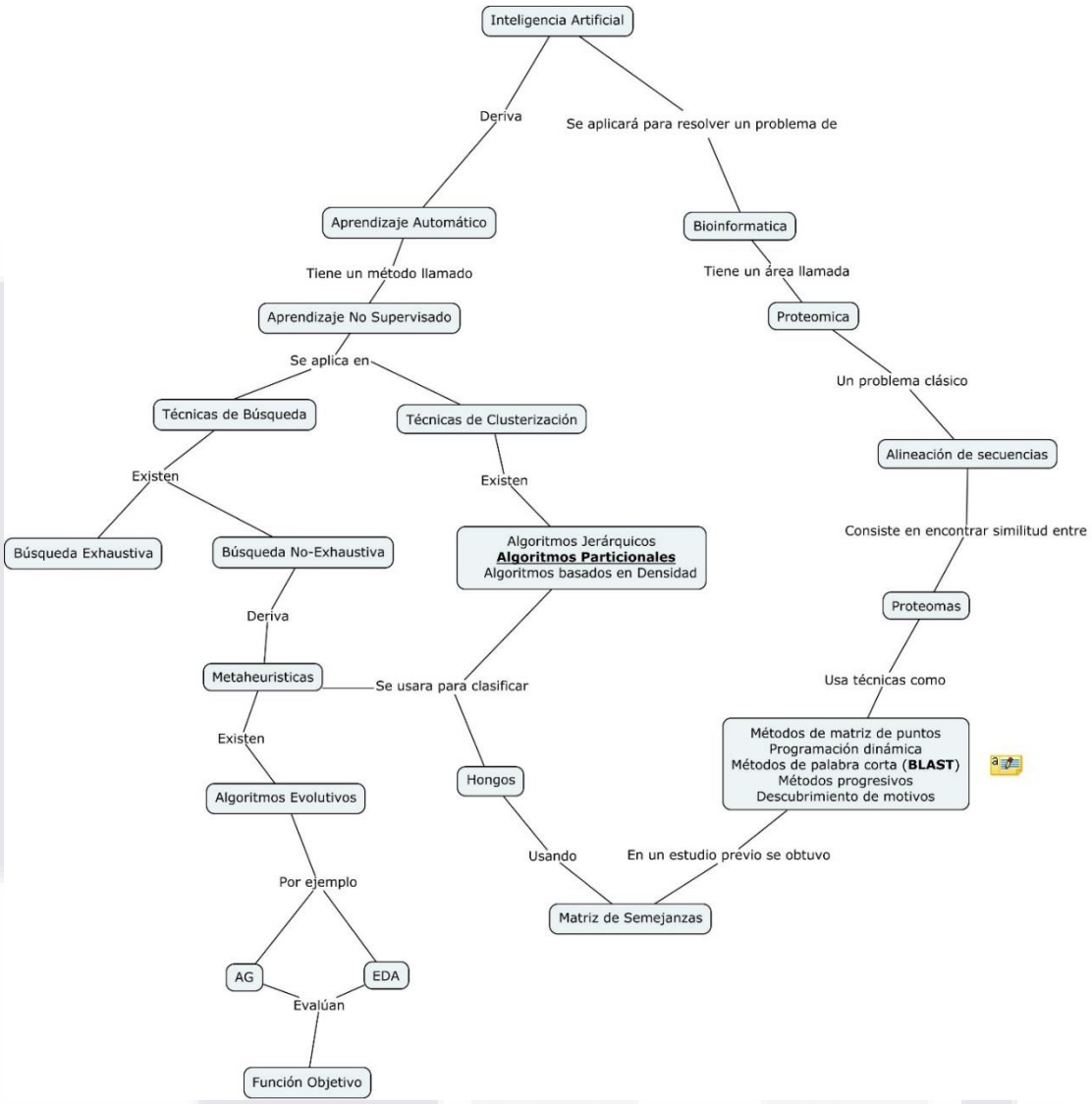


Figura 1 - Mapa conceptual del problema.

3. Objetivos de la investigación.

1. Objetivo General.

El objetivo general de esta investigación es buscar una alternativa rápida, eficiente y confiable para clusterizar hongos a partir de su información proteómica mediante el diseño, desarrollo y ajuste de los parámetros de dos algoritmos metaheurísticos (una de ellas híbrida).

2. Objetivos Específicos.

- Diseño e implementación de un Algoritmo Genético Híbrido.
- Diseño e implementación de un Algoritmo de Estimación de Distribución.
- Realización del diseño de experimentos factoriales para el ajuste de los parámetros de ambos algoritmos.
- Análisis de resultados de los algoritmos metaheurísticos.

4. Preguntas de Investigación.

¿Se puede establecer una alternativa para para clasificar hongos además de las existentes (basada en sus características físicas o en su ancestro común, por ejemplo)?

¿Es posible clasificar hongos u otros entes biológicos conociendo únicamente su información proteómica?

¿Con la aplicación de metaheurísticas híbridas se puede obtener un mecanismo de clusterización que sea rápida, confiable y eficiente en términos computacionales?

5. Justificación.

Se justifica llevar a cabo la investigación de aplicación de técnicas metaheurísticas hibridas para la clusterización de hongos mediante su información proteómica porque es de gran trascendencia e impacto en diversas áreas como: biología, medicina, genética, biología molecular y muchas otras relacionadas con ciencias de la vida y salud, además de que es una técnica novedosa, ágil, económica y confiable.

Encontrar una solución óptima a partir de un algoritmo de clusterización es un problema NP-completo (Gil-Garcia & Badía, 2002).

Al ser un problema de explosión combinatoria (NP-completo), tratar de resolverlo con métodos deterministas, la naturaleza de su complejidad impacta directamente en el tiempo y costo computacional requerido, sin embargo las técnicas metaheurísticas son capaces de proporcionar soluciones aceptables muy cercanas al óptimo global en tiempo y costos computacionales razonables.

Por otra parte, la clusterización encuentra aplicación en un sinfín de áreas, como por ejemplo en biología para clasificar animales y plantas, en medicina para identificar enfermedades, en marketing para identificar personas con hábitos de compras similares, en teoría de la señal pueden servir para eliminar ruidos, en biometría para identificación del locutor o de caras y un largo etcétera.

En el contexto del problema, al aplicar la técnica, se busca, por ejemplo: distinguir los hongos patógenos de los no patógenos, los micro de los macro hongos, incluso poder empatar los resultados con la clasificación del árbol filogenético, ello resultaría de gran utilidad y apoyo en estudios de micología.

La técnica también es novedosa y actual, ya que hay pocos trabajos relacionados con la clusterización de entes biológicos conociendo únicamente su información proteómica.

Si bien la técnica estará enfocada a clusterizar un grupo de hongos en particular basándose en su información proteómica eso no limita la posibilidad que se pueda extrapolar dicha técnica para aplicar con cualquier otro organismo biológico; este conocimiento tiene un enorme impacto social pues puede aplicarse a áreas de detección temprana de cáncer u otras patologías que aquejan a la sociedad.

Al fin y al cabo sin que se conozca excepción alguna todos los seres vivos están formados por proteínas y el avance en las ciencias computacionales ha permitido su fácil y rápida obtención, procesamiento y almacenamiento.

La proteómica, es el área que estudia las proteínas, reacciones e interacción entre sí, será objeto de investigación durante mucho tiempo, debido a que se perfila como una potente herramienta para el estudio y descubrimiento de aspectos fisiopatológicos en los seres humanos y ayudará a dilucidar las bases de la salud y enfermedad.

TIPO DE INVESTIGACIÓN

El desarrollo de ésta investigación corresponde a una combinación en el campo experimental y exploratorio ya que como se ha comprobado en la sección de antecedentes, es un área a la que pocos investigadores se han acercado pero por otro lado los algoritmos a desarrollar se deberán someter a distintas pruebas y comprobaciones.

6. Estructura del Trabajo.

Este documento de tesis está estructurado en 6 capítulos principales y se distribuye de la siguiente forma:

- ***Capítulo I – Introducción.*** (Actual) En este capítulo de inicio se describieron los antecedentes generales de los tópicos sobre los que trata la investigación, algunas investigaciones relacionadas a dichos tópicos; luego se describe el problema y la problemática, los objetivos y preguntas de la investigación y finalmente la justificación de llevar a cabo la investigación que se describe.

- ***Capítulo II – Marco Teórico.*** Este capítulo establece la teoría base y los conceptos fundamentales que se deben conocer para entender sobre que trata la investigación. Este capítulo está subdividido en 5 apartados principales en los cuales se detallan los conceptos, características, clasificación, entre otros puntos de interés sobre: Inteligencia Artificial, Clusterización, Metaheurísticas, Hibridación y Fundamentos básicos de Bioinformática y proteómica.

- ***Capítulo III – Metodología:*** El tercer capítulo del documento de tesis, describe la forma en cómo se lleva a cabo la aplicación de las técnicas metaheurísticas para la solución del problema, en primer instancia se describen los antecedentes y caso del estudio, para luego describir a detalle el diseño de la investigación en donde se incluyen las generalidades de los algoritmos propuestos, las aportaciones más sobresalientes del trabajo y el esquema de trabajo de la investigación.

- ***Capitulo IV – Metaheurística Evolutiva AGH-CHIP (Algoritmo Genético Híbrido para la Clusterización de Hongos mediante su Información Proteómica).*** Aquí se describe a detalle la construcción y aplicación de la primer técnica metaheurística. En el apartado se describen las generalidades del algoritmo y la estructura principal de funcionamiento; se definen las condiciones del experimento y su aplicación, en seguida se describen las pruebas estadísticas aplicadas, para luego dar interpretación a los resultados obtenidos por el algoritmo, también se

incluye una serie de tablas que son un aporte adicional y finalmente se exponen algunas conclusiones del capítulo y de la técnica.

- ***Capítulo V – Metaheurística Evolutiva UMDA-CHIP (Univariate Marginal Distribution Algorithm para la Clusterización de Hongos mediante su Información Proteómica).*** Este capítulo corresponde a la segunda técnica desarrollada y aplicada. También se describen las generalidades del algoritmo, condiciones para la experimentación y el experimento mismo, las pruebas estadísticas, interpretación de resultados para finalizar con unas conclusiones del capítulo.

- ***Capítulo VI – Conclusiones y Trabajo Futuro.*** El último capítulo de este documento presenta las conclusiones generales de la investigación y sus objetivos, a nivel de cada uno de los algoritmos y sus aportaciones y el trabajo futuro de la investigación.

En la parte final de documento se incluyen:

Glosario. Se incluye un glosario con los principales términos y siglas utilizados.

Bibliografía. Aquí se desglosa las fuentes bibliográficas que fueron consultadas para la escritura y soporte de la investigación.

Anexos. Corresponde a un espacio para incluir publicaciones relacionadas con la investigación, la matriz de semejanzas utilizada, las pruebas estadísticas y alguno que otro detallado en su correspondiente sección.

Capitulo II: Marco Teórico.

Durante el desarrollo de este capítulo que enmarca el documento de tesis, se va a abordar la parte fundamental que sostiene y respalda la investigación que se está llevando a cabo.

Algunos de los conceptos relevantes que se definen son: inteligencia artificial, posteriormente se hace mención de las técnicas de clusterización, metaheurísticas y dentro de este tema se hace énfasis a dos subtemas para explicar los Algoritmos Genéticos (AG's) y Algoritmos de la Distribución de la Estimación (EDA's) debido a que son precisamente las técnicas aplicadas a la problemática que se está tratando, enseguida se explica acerca de hibridación para luego describir brevemente algunos conceptos relacionados con bioinformática y proteómica.

1. Inteligencia Artificial.

No existe una definición universal sobre el concepto de Inteligencia Artificial (AI – Artificial Intelligence en Inglés) ya que incluso el propio término "inteligencia" causa controversia, de hecho cada autor propone su propia definición e interpretación dependiendo de su enfoque y especialidad.

A través de la historia se han seguido cuatro enfoques principales que fueron planteados por Russell y Norvig (Russell & Norvig, 1996) con el fin de diferenciar los tipos de inteligencia, dichos enfoques son: Sistemas que piensan como humanos, Sistemas que actúan como humanos, Sistemas que piensan racionalmente y Sistemas que actúan racionalmente (ideal).

En la Tabla 1 se muestra un resumen sobre lo mencionado en el párrafo anterior, los que aparecen en la parte superior se refieren a procesos mentales y al razonamiento, las de la parte inferior se refieren a la conducta. A la izquierda están las definiciones que miden el éxito en términos de la fidelidad en la forma de actuar de los humanos, mientras que las de la derecha toman como referencia un concepto ideal de inteligencia, llamado racionalidad.

Tabla 1- Algunas definiciones de los enfoques dentro de la IA.

Sistemas que piensan como humanos	***Sistemas que piensan racionalmente***
"El nuevo y excitante esfuerzo de hacer que los computadores piensen… máquinas con mentes, en el más amplio sentido literal" (Haugeland, 1989). Corresponde a un enfoque de modelo cognitivo.	*"El estudio de las facultades mentales mediante el uso de modelos computacionales"* (Charniak, Riesbeck, McDermott, & Meehan, 2014). Se hace énfasis a las leyes del pensamiento (lógica, silogismos).
Sistemas que actúan como humanos	***Sistemas que actúan racionalmente***
"El arte de desarrollar máquinas con capacidad para realizar funciones que cuando son realizadas por personas requieren de inteligencia (Kurzweil, 1990)". *"El estudio de cómo lograr que los computadores realicen tareas que, por el momento, los humanos hacen mejor"* (Rich & Knight, 1991). Dentro de este enfoque se aplica la prueba de Turing (1950), sin embargo pasar la prueba no es el objetivo principal de la IA.	*"IA...está relacionada con conductas inteligentes en artefactos"* (Nilsson, 1998). *"La rama de la ciencia de la computación que se ocupa de la automatización de la conducta inteligente"* (Luger & Stubblefield, 1993). Enfoque del agente racional (percepción y acciones).

Turing (Turing, 1950) enuncia que para que una computadora se considere inteligente se deben tomar en cuenta las siguientes capacidades:

- **Procesamiento de lenguaje natural** que le permita comunicarse de manera satisfactoria.
- **Representación del conocimiento** una forma para representar y almacenar lo que se conoce.
- **Razonamiento automático** para utilizar la información almacenada para resolver problemas y extraer nuevas conclusiones.

- **Aprendizaje automático** para adaptarse a nuevas circunstancias y para detectar y extrapolar patrones.

Se puede extender la prueba añadiendo video y la capacidad de transferir objetos.

- **Visión computacional** para percibir objetos.
- **Robótica** para manipular y mover objetos.

Existe un concepto que se desprende de la prueba de Turing y es el aprendizaje automático o aprendizaje de maquina (machine learning en inglés), que en ciencias de la computación se hace mención de que su objetivo es desarrollar técnicas que permitan a las computadoras aprender de forma automática y utilizar ese conocimiento para intentar dar solución a distintos problemas y toma de decisiones.

Dichos sistemas pueden trabajar con una gran cantidad de datos que incluso están incompletos o con mucho ruido.

Así mismo, con base al paradigma que siguen y de forma general es posible decir que están definidos en dos categorías principales (existen más formas de clasificación y enfoques):

- **Supervisados.** Con los datos de entrada, el algoritmo predice los resultados (ejemplo: regresión y clasificación).
- **No supervisados.** El algoritmo no conoce la salida, sino que se busca desarrollar nuevo conocimiento, un ejemplo de ello es la clusterización, a la que se dedicará un apartado por ser de interés y parte de la problemática a resolver en este documento.

2. Clusterización.

En este apartado se presentan los antecedentes, una introducción a los conceptos básicos de clusterización (clustering en inglés), además de una la descripción breve de algunas técnicas.

Si bien la clusterización no es un área purista, ya que tiene sus orígenes en la estadística multivariante aquí se explica desde el contexto de la minería de datos.

La minería de datos es una de las técnicas incluida en el proceso KDD (Knowledge Discovery in Databases, por sus siglas en inglés), en la que la clusterización juega un rol importante en el descubrimiento de nuevo conocimiento útil; generalmente, los elementos de un mismo clúster comparten propiedades comunes, dicho conocimiento puede permitir una extrapolación de ahí su aplicación.

Según Weiss e Indurkhya (Weiss & Indurkhya, 1998) las técnicas de minería de datos pueden dividirse en dos principales categorías: aprendizaje supervisado (predictivas) y aprendizaje no supervisado (descriptivas).

Las técnicas predictivas se utilizan para prever el comportamiento futuro de algún tipo de entidad mientras que las descriptivas pueden ayudar a su comprensión. Si bien dentro de la categoría descriptiva están las técnicas de clusterización, los modelos predictivos pueden ser descriptivos (hasta donde sean comprensibles) y los modelos descriptivos pueden emplearse para realizar predicciones. De esta forma, hay algoritmos o técnicas que pueden servir para distintos propósitos.

Como ya se dijo, las técnicas de clusterización son parte de los métodos de aprendizaje no supervisado (mencionados en el apartado de Inteligencia Artificial) ya que no requieren de clases predefinidas para hacer nuevos descubrimientos por lo que son muy útiles en situaciones donde existe poco conocimiento sobre los datos con los que se está trabajando.

Entrando en materia, primeramente hay que definir el concepto de clusterización; pueden formularse varias definiciones ya que cada autor llega a dar su propia interpretación, incluso el concepto puede tomar sinónimos como: segmentación, agrupamiento, conglomerados, etcétera, según el enfoque que se esté estudiando. Sin embargo todos comparten el mismo

fin y es el de poder crear grupos altamente homogéneos entre sí pero a su vez lo más heterogéneo con respecto a otros grupos, lo anterior con base a una medida de semejanza.

En seguida se presenta una definición más formal del concepto, explicada por Han y Kamber (Han & Kamber, 2000) en la que enuncian que la clusterización "*Es una de las tareas de aprendizaje no supervisado en la que no se requiere una clasificación predefinida. El objetivo es partir los datos obteniendo el conocimiento de acuerdo a las características de los mismos. En general, las clases de los datos no se presentan en el conjunto de datos y los objetos son agrupados basándose en el principio de maximización de similitud dentro de los clústeres y minimización de similitud entre clústeres diferentes*".

Lo anterior queda expresando gráficamente con la Figura 2.

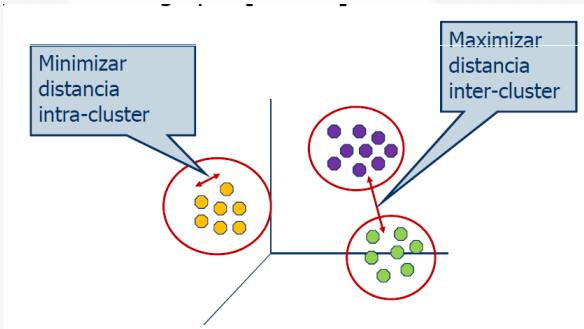


Figura 2 - Objetivos de la clusterización a partir de (Pandre, 2011).

"*La clusterización representa la división de datos en grupo de objetos similares llamados clústeres*" (Mitra & Acharya, 2003). Los grupos o clústeres, son un conjunto de elementos con características similares y desempeñan un papel importante en la manera de percibir y describir el mundo que nos rodea.

Un conjunto de datos puede contener varias dimensiones o atributos, visualmente se pueden percibir hasta tres dimensiones y considerando que existe una baja densidad, el ojo humano puede llevar a cabo la división en grupos (clusterización) para asignar objetos particulares a dichos grupos (clasificación), sin embargo si los datos están altamente concentrados y distorsionados es difícil distinguirlos como en la Figura 3 que se puede percibir que a partir de un conjunto de datos que están muy concentrados se pueden representar diferentes soluciones, además de esto y cuando la dimensión crece hay que recurrir a otro tipo de técnicas.

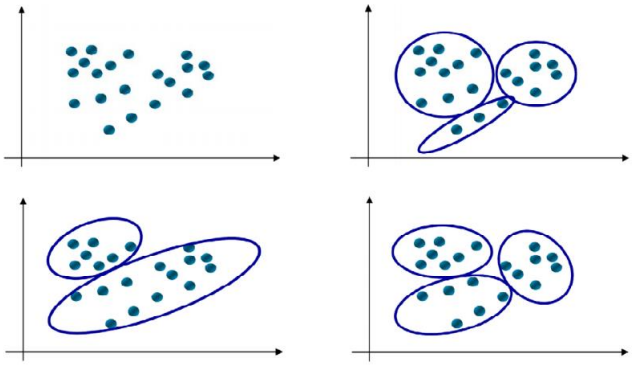


Figura 3 – Diferentes clústeres para los mismos datos (dos dimensiones) (Justel, 2012).

Como ya se ha hecho mención, la clusterización es una técnica que ayuda al descubrimiento de nuevo conocimiento y es aplicada en una amplia variedad de áreas como: biología, reconocimiento de patrones, segmentación de mercados, procesamiento de imágenes, cómputo y multimedia, análisis médico, economía, bioinformática y biometría.

Jain y Dubes (Anil K. Jain & Dubes, 1988) en su libro "Algorithms for clustering data" enuncian que las actividades del análisis de clusterización típicamente implican los siguientes 5 pasos.

1. *Representación de patrones.* Establecer las características iniciales de los datos, su tipo, tamaño así como el número de clases, número de patrones.
2. *Definición de proximidad.* Se refiere al establecimiento de una medida (o patrón) de similitud.
3. *Clusterización.* Aplicar la técnica correspondiente para crear los clústeres o grupos.
4. *Abstracción de datos.* La forma de representar dichos clústeres en forma simple y compacta.
5. *Verificación de resultados.* Llevar a cabo un análisis y evaluación de los resultados obtenidos.

Así mismo Edna (Edna, 2006) menciona en su documento de tesis que las características deseables de la mayoría de los algoritmos de clusterización son: "*escalabilidad, habilidad para trabajar con distintos tipos de atributos, descubrimiento de clústeres con formas arbitrarias, requerimientos mínimos en el conocimiento del dominio para determinar los parámetros de entrada, habilidad para tratar con datos ruidosos, insensibilidad al orden de*

las observaciones de entrada, alta dimensionalidad, clusterización basado en restricciones, interpretación y uso ".

Técnicas de Clusterización.

Los algoritmos de clusterización varían en función a las reglas heurísticas que usan, así como en el enfoque para el que fueron diseñadas. Básicamente los algoritmos de clusterización pueden ser clasificados en función de:

1. El tipo de dato que se maneja (numérico, categórico o mixto).
2. Criterio utilizado para medir la semejanza.
3. Conceptos y técnicas de clusterización utilizados (ej. Lógica difusa, estadística).

Dentro de la literatura existen una gran cantidad de técnicas de clusterización que varían en función de la arquitectura utilizada. Jain (Anil K. Jain & Dubes, 1988) propone una clasificación general que divide los algoritmos en: clusterización particional, clusterización jerárquica, clusterización basada en densidad y clusterización basada en grid.

Para cada categoría existen varias sub-categorías y para encontrar clústeres en los datos existen diferentes técnicas, Edna (Edna, 2006) en su trabajo de tesis menciona algunas de ellas:

- *Técnicas estadísticas, basadas en la utilización de medidas de similitud y análisis estadístico para agrupar los datos.*
- *Técnicas conceptuales, basadas en la clasificación de características cualitativas de los datos.*
- *Técnicas excluyentes, basas en el agrupamiento de datos sin traslape, es decir un dato único y exclusivamente puede pertenecer a una sola clase. La mayoría de los algoritmos de clusterización se basan en esta técnica.*
- *Técnicas con traslape, basadas en técnicas de lógica difusa que consideran grados de pertenencia en los datos. Los objetos pueden pertenecer a más de una clase.*

En seguida se explican brevemente las técnicas más representativas en minería de datos para crear clústeres.

Algoritmos Jerárquicos.

Un método jerárquico crea un dendograma (árbol) a partir de un conjunto de datos, de forma recursiva divide dicho conjunto de datos en conjuntos cada vez más pequeños hasta organizar todos los datos en forma jerárquica. La Figura 4 muestra un ejemplo de una representación gráfica de un dendograma en el que se obtuvieron dos clústeres.

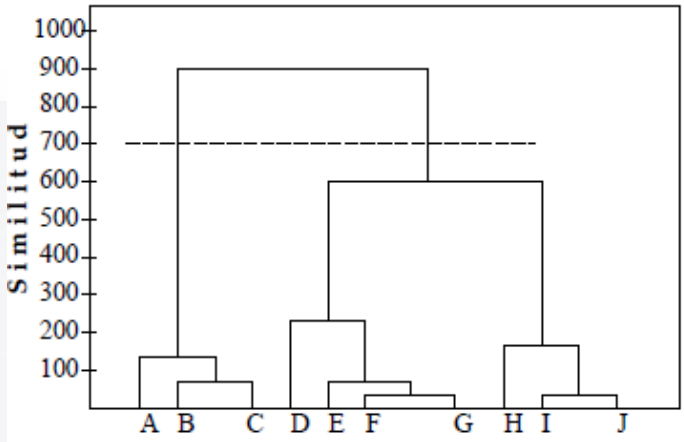


Figura 4 - Dendograma de un conjunto de datos (Edna, 2006).

Algunos algoritmos pertenecientes a esta categoría son: CURE (Clustering Using Representatives) (Sudipto Guha, Rastogi, & Shim, 1998), CHAMALEON (Karypis, Han, & Kumar, 1999), BIRCH (Balanced Iterative Reducing and Clustering using Hierarchical) (Zhang, Ramakrishnan, & Livny, 1996) y ROCK (RObust Clustering algorithm using linKs) (Saikat Guha, Rastogi, & Shim, 1999).

Algoritmos Particionales.

"*Un algoritmo de clusterización particional obtiene una partición simple de los datos en vez de la obtención de la estructura del clúster tal como se produce con los dendogramas de la técnica jerárquica*" (A. K. Jain, Murty, & Flynn, 1999). En la Figura 5 se puede apreciar un ejemplo de clusterización particional con valores arbitrarios, en este caso se crearon 4 clústeres.

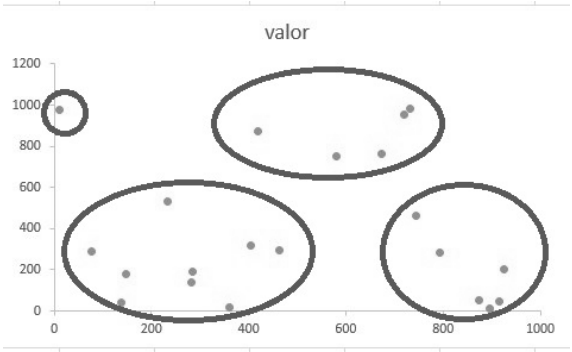


Figura 5 - Ejemplo de clusterización particional.

En la clusterización particional los datos son agrupados en k clústeres, de tal forma que sea minimizada la distancia total de cada dato desde el centro de su clúster o desde una distribución de clústeres. La distancia de un punto puede ser evaluada en forma distinta según el algoritmo, y es llamada generalmente función de similitud.

La principal ventaja que tienen los algoritmos particionales con respecto a los jerárquicos es al momento de procesar grandes cantidades de datos, debido a que sería complicada la construcción de los dendogramas. Sin embargo uno de los problemas es a la hora de decidir el número de clústeres que son creados.

Las técnicas particionales generalmente producen clústeres con base al mecanismo de evaluación definido. "*En la práctica, el algoritmo se ejecuta múltiples veces con diferentes estados de inicio y la mejor configuración que se obtenga es la que se utiliza como la clusterización de salida*." (Edna, 2006).

Algunos algoritmos de clusterización que pertenecen a esta clasificación son: CLARA (Clustering Large Applications) (Rousseeuw & Kaufman, 1990), CLARANS (Clustering Large Applications based on Randomized Search) (Ng & Han, 2002), K-prototypes (Milenova & Campos, 1997), K-mode (Huang, 1997), K-Means (Huang, 1998).

Algoritmos Basados en Densidad.

Estos algoritmos enfocan el problema de la división de un conjunto de datos en clústeres para lo que consideran la distribución de densidad de los puntos, de modo tal que los clústeres que se forman tienen una alta densidad de puntos en su interior mientras que alrededor de ellos aparecen zonas que son de baja densidad (dichos elementos aislados

representan ruido). La Figura 6 muestra un ejemplo ilustrativo de clusterización con la técnica mencionada.

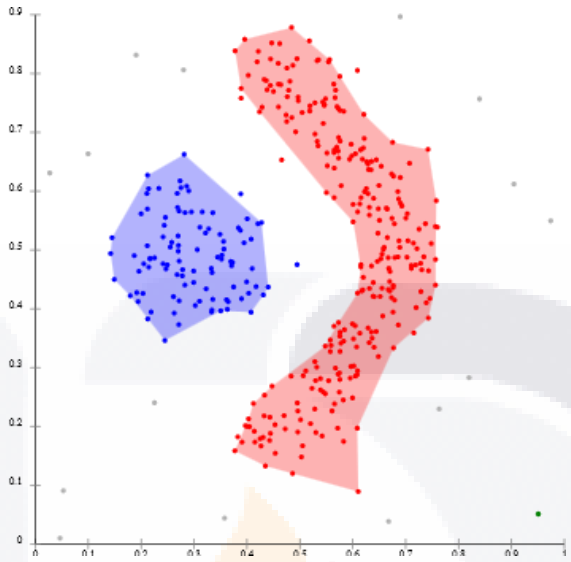


Figura 6 - Agrupamiento basado en densidad con DBSCAN (Chire, 2011).

Este tipo de algoritmo tiene la ventaja que puede encontrar clústeres de diversas formas en donde existe mucha densidad en los datos, además de reducir el ruido a diferencia de los clústeres particionales que pueden encontrar solo clústeres esféricos.

Algunos algoritmos de clusterización que pertenecen a esta clasificación son: DBSCAN (Density-Based Spatial Clustering of Aplications with Noise) (Ester, Kriegel, Sander, & Xu, 1996), OPTICS (Ordering Points To Identify the Clustering Structure) (Ankerst, Breunig, Kriegel, & Sander, 1999) y DENCLUE (DENsity-based CLUstEring) (Hinneburg & Keim, 1998).

Algoritmos Basados en Grid.

Mollá (Mollá Santiago, 2014) comenta que este tipo de técnicas dividen el espacio en un número finito de celdas que forman una estructura de rejilla donde se lleva a cabo la clusterización. Los objetos que contiene cada una de las celdas son representados mediante atributos estadísticos de esa misma celda. La clusterización se lleva a cabo usando dicha información estadística y no el de todos los datos. Por el hecho que el tamaño de las rejillas es inferior al número de objetos, la velocidad del proceso aumenta

considerablemente. Para distribuciones de datos aglomeradas o irregulares, se necesita aumentar la granularidad de la rejilla para mejores resultados.

Algunos algoritmos de clusterización que pertenecen a esta clasificación son: STING (Statistical Information Grid-based method) (Wang, Yang, Muntz, & others, 1997), CLIQUE (Agrawal, Gehrke, Gunopulos, & Raghavan, 1998) y Waveclúster (Sheikholeslami, Chatterjee, & Zhang, 2000).

Otras Técnicas de Clusterización.

- Clusterización difusa (Fuzzy clustering en inglés): En los algoritmos de clusterización tradicionales, cada patrón (elemento) pertenece única y exclusivamente a un clúster. En esta técnica de clusterización se asocia cada dato con cada clúster utilizando funciones de pertenencia (Zadeh, 1965).
- Clusterización con redes neuronales: Las redes neuronales artificiales (ANN)(Hertz, Krogh, & Palmer, 1991), están inspiradas en las redes neuronales biológicas y tienen una extensa utilización tanto en técnicas de clasificación como de clusterización en minería de datos.
- Clusterización por medio de algoritmos evolutivos: "*Los algoritmos evolutivos, motivados por la evolución natural, utilizan operadores genéticos y una población de soluciones para obtener el óptimo global en la partición de los datos*"(A. K. Jain et al., 1999). Las soluciones que son candidatas en los problemas de clusterización se muestran codificadas por medio de cromosomas.

3. Metaheurísticas.

En este apartado se describe el concepto de heurística, metaheurística, características y una clasificación general ya que de aquí se desprenden dos de las técnicas utilizadas en el desarrollo del trabajo.

1. Introducción a la Optimización.

Según Santiago (Santiago, 2006) cualquier sistema tiene tres elementos básicos (entradas, modelo o proceso, y salidas). Desde dicha perspectiva se puede distinguir entre 3 tipos de problemas: "*Problemas de optimización*", "*Problemas de sistemas de identificación*" y "*Problemas de simulación*".

Pardalos (Pardalos & Resende, 2001) enuncia que cotidianamente surgen situaciones que obligan a seleccionar la mejor decisión entre un conjunto de soluciones, lo cual comúnmente se conoce como "optimización".

"Matemáticamente, un problema de optimización consiste en el ajuste de un conjunto de variables de entrada, tales que maximizan o minimizan una determinada función de optimización, denominada *función objetivo*, que está compuesta por un conjunto determinado de variables definidas sobre un conjunto discreto" (López, 2009).

Para llevar a cabo el proceso de optimizar se deben aplican técnicas de búsqueda específicas, las cuales pueden ser exactas (obtienen el óptimo) o aproximadas (cercanas al óptimo). Para el caso en que los problemas requieren un tiempo polinomial, debido a que el espacio de búsqueda es muy grande se pueden aplicar técnicas "metaheurísticas" que como ya se hizo mención en los apartados anteriores buscan aproximarse al óptimo global.

Referente a la función objetivo se puede decir que en caso que exista una sola función por optimizar se le denomina "optimización mono-objetivo", por el contrario, si se requiere optimizar un conjunto de dos o más funciones objetivo se le denomina "optimización multi-objetivo", para cualquiera de los casos las funciones pueden estar sujetas a restricciones (o no) ya que son las que definen la región factible del problema.

Optimización Multi-objetivo.

Según Santiago (Santiago, 2006) debido a su complejidad, dichas funciones muchas veces entran en conflicto unas con otras, por lo que mejorar una función implica empeorar el desempeño de otras, sin embargo se debe encontrar el mejor compromiso (balance) entre sus objetivos. "*Al mejor compromiso generalmente se le denomina óptimo de Pareto*" (Cagnina, 2010).

Existen técnicas para la solución de este tipo de problema (es decir donde se requiere encontrar el balance) que muchos autores han utilizado y combinado con técnicas heurísticas, en particular con Algoritmos Evolutivos, estos métodos pueden englobarse como "MOEA" (Multi-Objective Evolutionary Algorithms, en español algoritmos evolutivos multiobjetivo) (Coello, Van Veldhuizen, & Lamont, 2002; Deb, 2001).

De acuerdo al trabajo de Abraham y sus colegas (Abraham, Jain, & Goldberg, 2005) se propone la siguiente clasificación para MOEA:

- *Métodos basados en funciones agregativas*. Consiste en combinar (o agregar) todas las funciones objetivo con operaciones aritméticas simples (sumas, restas, etc.) para obtener una sola función objetivo.
- *Métodos basados en población*. Se caracterizan por utilizar la población de un algoritmo evolutivo para diversificar la búsqueda.
- *Métodos basados en el concepto de Pareto*. Estos métodos cuentan con un esquema de selección que se basa en el concepto de optimización de Pareto.

2. Heurística.

En primera instancia puede ser útil la definición nominal etimológica de la heurística. La palabra *heurística* procede del término griego ***εὑρίσκειν***[2] *(*heurisken*)*, que significa *«hallar, inventar»* (etimología que comparte con eureka[3]).

La heurística usualmente propone métodos que guían el descubrimiento, en consecuencia se han hecho adaptaciones al término en diferentes áreas (incluida la IA), según la enciclopedia en línea Wikipedia establece que "*la 'heurística' puede ser definida como un arte, técnica o procedimiento práctico o informal, para la solución de problemas*[4]" ("Heurística", 2016).

En seguida se presentan algunas definiciones:

- "*A un proceso que puede resolver un cierto problema, pero que no ofrece ninguna garantía de lograrlo, se le denomina una 'heurística'*" (Newell, Shaw, & Simon, 1959).
- Un heurístico es un "*procedimiento simple, a menudo basado en el sentido común, que se supone que ofrecerá una buena solución (aunque no necesariamente la óptima) a problemas difíciles, de un modo fácil y rápido*" (Zanakis & Evans, 1981).
- "*Una heurística es una técnica que busca soluciones buenas (óptimas o casi óptimas) a un coste computacional razonable, aunque sin garantizar la factibilidad de las mismas. En algunos casos ni siquiera puede determinar la cercanía al óptimo de una solución factible*" (Ruiz-Rodríguez, 2012).

Siendo esta última la más acertada para el propósito del documento que se está presentando.

Los heurísticos se utilizan, por ejemplo, cuando no existe un método exacto de resolución, existe un método exacto pero este consume mucho tiempo para ofrecer la solución óptima,

[2] Según la Real Academia (consultado el 14 de marzo de 2016).
[3] Según la Real Academia, (consultado el 14 de marzo de 2016).
[4] Glosario: "Arte o técnica de la búsqueda o investigación. Método heurístico, por oposición al didáctico o de enseñanza." en Heurística.

existen restricciones de tiempo y/o además cuando puede ser un intermediario o elemento de entrada de otra técnica.

Para el matemático George Pólya (Polya, 1945) la base de la heurística está en la experiencia de resolver problemas y en ver cómo otros lo hacen. Por lo que se dice que hay "*búsquedas ciegas*", "*búsquedas heurísticas (basadas en la experiencia)*" y "*búsquedas racionales*".

Así mismo dentro de las heurísticas existen distintos métodos, en consecuencia algunos autores como (Silver, Victor, Vidal, & de Werra, 1980) proponen la siguiente clasificación de dichos métodos:

- *Métodos constructivos*, construyen la solución por lotes, es decir en partes y de forma sucesiva (por pasos).
- *Métodos de descomposición*, divide el problema en varios más pequeños (divide and conquer en inglés) y la solución final se obtiene con la solución de cada uno de estos.
- *Métodos de reducción*, buscan encontrar alguna característica de la solución que permita simplificar la manera de tratar el problema.
- *Métodos de manipulación del modelo*, a partir de la solución de un problema simplificado (con menos restricciones) se obtiene una solución al problema planteado.
- *Métodos de búsqueda por entornos*, a partir de una solución inicial (generalmente aleatoria) se realizan modificaciones en sucesivas iteraciones para obtener una solución final. Con cada iteración existe un conjunto de soluciones vecinas que pueden ser la nueva solución en el proceso. En este grupo se encuentran las técnicas metaheurísticas, que serán explicadas en el siguiente apartado.

Como ya se dijo uno de los inconvenientes existentes con las heurísticas es que no garantizan la mejor solución, la Figura 7 muestra un ejemplo de dicho caso, en el que una heurística puede caer en cualquiera de las colinas inferiores y asumir que ha encontrado la mejor solución, cuando no es realmente cierto ya que solo es un algún óptimo local.

Figura 7 – Ejemplo de espacio de soluciones.

En seguida se presenta el concepto de metaheurística, las características más comunes que comparten y una clasificación general.

3. Metaheurística.

Día a día se presentan y resuelven problemas de optimización. Muchos de los pequeños problemas se pueden resolver utilizando el razonamiento y posiblemente algunas sencillas fórmulas matemáticas. Pero si los problemas presentan una complejidad mayor (problemas tipo NP-completos o de explosión combinatoria) hay que apoyarse de otras técnicas.

Etimológicamente la palabra metaheurística combina el prefijo griego “meta” que significa más allá (en este caso “nivel superior”) y la palabra *heurística* (descrita en el apartado anterior). El concepto fue introducido en el área de inteligencia artificial por Glover, pero hoy en día es ampliamente conocido y utilizado.

En el libro “Handbook of Metaheuristics” de Glover y sus colegas (Glover & Kochenberger, 2003) describen que “*Las metaheurísticas son métodos de solución que orquestan una interacción entre procedimientos de mejora local y estrategias de más alto nivel para crear un proceso capaz de escapar de óptimos locales y desarrollar una búsqueda robusta del espacio de solución*”. Otra definición interesante dice que “*las metaheurísticas son procedimientos iterativos que guían una heurística subordinada combinando de forma inteligente distintos conceptos para explorar y explotar adecuadamente el espacio de búsqueda*” (Herrera, 2006).

Las técnicas metaheurísticas son capaces de proporcionar soluciones aceptables (aunque no necesariamente la óptima (global) pero si muy cercana) en tiempo y con recursos computacionales razonables.

También Herrera (Herrera, 2006) dice que con el fin de obtener buenas soluciones, cualquier algoritmo de búsqueda debe establecer el balance adecuado entre dos características contradictorias del proceso.

- ***Intensificación***: Cuando el esfuerzo de búsqueda se centra en el entorno local y el vecindario (explotación).
- ***Diversificación***: Cuando se prueba en otras áreas del espacio de soluciones (exploración).

Lo anterior es representado de forma gráfica en la Figura 8.

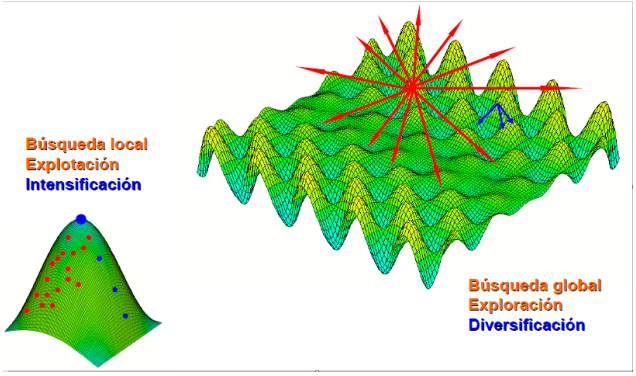


Figura 8 - Funcionamiento de las Metaheurísticas (Herrera, 2006).

El equilibrio entre exploración y explotación es necesario para poder identificar de forma rápida regiones del espacio con buenas soluciones y para no destinar mucho tiempo en regiones del espacio no prometedoras o ya exploradas.

Características de las Metaheurísticas.

Con base a lo que establece (Sait & Youssef, 1999) las técnicas metaheurísticas se caracterizan por las siguientes propiedades:

- Son ciegas, no saben si llegan a la solución óptima. Por lo que es necesario establecer un criterio de paro, ya sea en base a un número de iteraciones o un valor encontrado.
- Son algoritmos aproximativos, pueden o no garantizar la obtención de la solución óptima, sin embargo como ya se dijo, sí son capaces de obtener una solución confiable en un tiempo razonable.
- Aceptan ocasionalmente malos movimientos, ósea que algunas veces aceptan, incluso, soluciones no factibles como paso intermedio para acceder a nuevas regiones no exploradas.
- Son relativamente sencillas; básicamente todo lo que se necesita es una forma de representar el espacio de soluciones, una solución inicial (o un conjunto de ellas) y un mecanismo para explorar y evaluar el campo de soluciones.
- Son generales. Esto es, que se pueden aplicar en la resolución de prácticamente cualquier problema de optimización combinatoria.
- "*En cada iteración, la nueva solución depende de la solución de partida y de la trayectoria seguida hasta ese momento, de forma que el proceso de búsqueda*

puede pasar varias veces por la misma solución, eligiendo en cada una de las ocasiones una nueva solución distinta" (García Sánchez, 2007).

En seguida se presenta una clasificación de las metaheurísticas más importantes.

Clasificación de las metaheurísticas.

Las técnicas metaheurísticas se pueden clasificar atendiendo a diferentes características. Según la característica seleccionada, se puede definir una clasificación u otra, resultado de un punto de vista específico.

Gómez (Gómez, 2014) presenta en su artículo, una tabla (Tabla 2) resumida con distintos enfoques de clasificación con base a los autores que él consultó (Blum & Roli, 2003; Duarte, Pantrigo, & Gallego, 2007; Herrera, 2006; Vega, Batista, & Pérez, 2003).

Tabla 2 - Formas de clasificación de la metaheurísticas (Gómez, 2014).

Criterio	Tipo de Metaheurística
Fuente de inspiración	*Fenómenos naturales* *Sin inspiración*
Cantidad de soluciones	*Poblacional* *Trayectorial*
Función de Adaptabilidad	*Estática* *Dinámica*
Cantidad de vecindades	*Una vecindad* *Varias vecindades*
Uso de memoria	*Sin Memoria* *Con memoria*
Estrategia seguida	*Método constructivo* *Basada en trayectorias* *Basada en poblaciones*
Tipo de procedimientos referidos	*Para métodos de relajación* *Para procesos constructivos* *Para búsquedas por entorno* *Para procesos evolutivos* *De descomposición* *De memoria a largo plazo*

Tal como se puede apreciar en la tabla anterior no existe una clasificación rigurosa ni que sea completamente aceptada, ya que las técnicas no son meramente puristas, ósea que pueden tener elementos para pertenecer a una u otra categoría (o incluso en varias a la vez).

La clasificación que más se usa es la que se basa en si la técnica utiliza un único punto del espacio de búsqueda o trabaja sobre un conjunto o población. Con base a ello esta clasificación divide las metaheurísticas en basadas en trayectoria y basadas en población (Ruiz-Rodríguez, 2012).

En seguida se enumeran algunas de las técnicas más importantes que están en la clasificación que se menciona en el párrafo anterior, si se desea saber más sobre cada una de ellas se puede consultar la referencia correspondiente.

Basadas en trayectoria.

Los algoritmos basados en trayectorias efectúan un estudio local del espacio de búsqueda, analizan el entorno de la solución actual para decidir cómo continuar el recorrido de la búsqueda (Herrera, 2006).

- Búsqueda Local (BL) (Blum & Roli, 2003).
    - BL con Multiarranque aleatorio (Boender, Kan, Timmer, & Stougie, 1982; Kan & Timmer, 1989).
    - BL iterativa (Lourenço, Martin, & Stützle, 2003).
    - Búsqueda de vecindario variable (Hansen & Mladenović, 2002; Mladenović & Hansen, 1997).
    - Descenso de vecindario variable (Hansen & Mladenović, 2002).
    - GRASP (Feo & Resende, 1995; Resende & Ribeiro, 2010).
    - BL guiada (Mills & Tsang, 1999).
- Enfriamiento simulado (Kirkpatrick, Vecchi, & others, 1983).
- Búsqueda tabú (Glover & Laguna, 1997).

Basadas en Poblaciones.

Las técnicas metaheurísticas basadas en población trabajan con un conjunto de individuos que representan otras tantas soluciones. Su eficiencia y resultado depende fundamentalmente de la forma con la que se manipula la población en cada iteración.

- Algoritmos Evolutivos.
    - Programación Evolutiva (D. B. Fogel, 1995; L. J. Fogel, 1966).
    - Estrategias de Evolución (Beyer & Schwefel, 2002; Rechenberg, 1973).
    - **Algoritmos Genéticos** (Goldberg & others, 1989; Holland, 1975).

    - Evolución Diferencial (Lampinen, Price, & Storn, 2005; Storn & Price, 1997).
    - **Algoritmos de Estimación de Distribución** (Larranaga & Lozano, 2001).
- Algoritmos de Optimización Basados en Enjambres.
    - Algoritmos Basados en Colonia de Hormigas (Dorigo, Birattari, & Stützle, 2006; Dorigo, Maniezzo, & Colorni, 1996).
    - Algoritmos Basados en Nubes de Partículas (Eberhart, Kennedy, & others, 1995; Kennedy, Kennedy, & Eberhart, 2001).

En lo anterior de forma intencional han sido resaltados dos técnicas, los Algoritmos Genéticos y los Algoritmos de Estimación de Distribución por el hecho de que son de interés debido a que son precisamente estás las técnicas base que se utilizan para atacar el problema que se está describiendo en este documento. Por lo tanto en los siguientes apartados serán descritos con mayor detalle.

Algoritmos Genéticos.

En este apartado se explican a detalle los antecedentes, componentes y la forma general en cómo trabaja el algoritmo genético.

La técnica de los Algoritmos Genéticos (AG's; en inglés: GA 'Genetic Algorithms') fue planteada por Holland (Holland, 1975) de la Universidad de Michigan, dicha técnica está inspirada en los procesos naturales de evolución de los seres vivos.

Pero no fue hasta la publicación del trabajo de Goldberg (Golberg, 1989) que se popularizaron como una herramienta para resolver problemas de búsqueda y optimización. A partir de entonces se encuentran bien descritos y se han hecho varios trabajos al respecto (Davis & others, 1991; Michalewicz, Algorithms, & Structures, 1996; Reeves, 1993).

"*Los organismos vivos poseen una consumada destreza en la resolución de problemas, esta habilidad la obtienen a través de la evolución*" (García Martínez, 2008). Los AG's están inspirados en los postulados de la "Teoría de la Evolución" de Charles Darwin (Darwin, 1859), en la que a través de generaciones (millones de años) las poblaciones evolucionan conforme a los principios de selección natural y supervivencia de los más fuertes. De forma análoga los mecanismos que usa el AG para la búsqueda de la solución óptima pueden verse como una metáfora de los procesos biológico-evolutivos.

De forma natural, los individuos de una población compiten entre sí por la búsqueda y obtención de recursos básicos de supervivencia (agua, comida, etc.). Incluso compiten en la búsqueda de un compañero para reproducirse. Aquellos con más éxito en la supervivencia y atrayendo compañeros tienen mayor probabilidad de generar un gran número de descendientes, por el contrario, los individuos poco favorecidos genéticamente producirán un número menor de descendientes (incluso están condenados a desaparecer).

Lo anterior significa que los genes de los individuos mejor adaptados estarán presentes en generaciones posteriores y de forma creciente. La combinación de buenas características que provienen de diferentes ancestros, en ocasiones llega a producir descendientes "superdotados", cuyo éxito de supervivencia será mejor que cualquiera de sus ancestros; de esta manera se va dando la evolución donde los individuos se adaptan cada vez mejor al entorno que los rodea.

Los AG's usan esa analogía que emula el comportamiento natural. Trabajan con poblaciones de individuos, donde cada uno representa una solución factible a un problema específico. Normalmente, a cada individuo se le asigna un valor o puntaje, relacionado con la bondad de dicha solución. Cuanto mayor sea dicho valor, será más la probabilidad de generar buenas soluciones. En la naturaleza esto equivaldría al grado de aptitud de un organismo para competir por determinados recursos.

Cuanto mayor sea la adaptación de un individuo al problema, mayor será la posibilidad de reproducirse, cruzando su material genético con otro individuo (ambos seleccionados en igualdad de condiciones). Dicho cruce producirá nuevos individuos con las mejores características de los padres. Así, a través de varias generaciones, las buenas características serán heredadas en cada población. Esto favorece el cruce de los individuos mejor adaptados y con ello que se exploren las áreas más prometedoras del espacio de búsqueda. Si el algoritmo genético se diseña correctamente, la población debería converger hacia una solución óptima del problema.

Aunque no es garantía que el AG encuentre la solución óptima global del problema, existe evidencia empírica de que encuentran buenas soluciones en un tiempo y recursos computacionales aceptables, en comparación con el resto de algoritmos de optimización combinatoria. Incluso se ha demostrado teóricamente que los AG llegan a converger, se pueden consultar los siguientes trabajos de diferentes autores (Kenneth A. De Jong, Spears, & Gordon, 1995; De Silva & Suzuki, 2005; Ding & Yu, 2005; Jose, Davis, & Principe, 1993; Meiyi, Zixing, & Guoyun, 2004; Nehab & Pacheco, 2004; Paszynska, 2005; Schmitt & Rothlauf, 2001).

Es posible que para un problema en particular exista una técnica especializada para resolver dicho problema, en tal situación lo más probable es que un AG sea superado por dicha técnica tanto en tiempo como en eficiencia, porque como ya se dijo las metaheurísticas son aproximadas. El gran campo de aplicación de los AG's involucra aquellos problemas en los cuales no existen técnicas especializadas. O aunque existan y funcionen bien, pueden efectuarse mejoras de las mismas, lo que podría convertirse en una hibridación, de hecho la técnica que se aplica para la solución al problema de este trabajo es un AG hibrido, concepto que será detallado más adelante.

En el siguiente apartado se presenta y describe el esquema general de un AG simple, además de los principales componentes de los AG's

Componentes de un Algoritmo Genético Simple.

En seguida se muestra el Algoritmo Genético Simple (SGA por sus siglas en inglés) de Goldberg (Goldberg & others, 1989) el cual muestra los principales componentes que el AG debe tener. La Figura 9 muestra el seudocódigo del SGA.

```
BEGIN /* Algoritmo Genetico Simple */
        Generar una poblacion inicial.
        Computar la funcion de evaluacion de cada individuo.
        WHILE NOT Terminado DO
        BEGIN /* Producir nueva generacion */
                FOR Tamaño poblacion/2 DO
                BEGIN /*Ciclo Reproductivo */
                        Seleccionar dos individuos de la anterior generacion,
                                para el cruce (probabilidad de selección
                                proporcional
                                a la funcion de evaluacion del individuo).
                        Cruzar con cierta probabilidad los dos
                                individuos obteniendo dos descendientes.
                        Mutar los dos descendientes con cierta probabilidad.
                                Computar la funcion de evaluacion de los dos
                                descendientes mutados.
                        Insertar los dos descendientes mutados en la nueva
                                generacion.
                END
                IF la poblacion ha convergido THEN Terminado := TRUE
        END
END
```

Figura 9- Algoritmo Genético Simple (Golberg, 1989).

De la figura anterior se tiene que el SGA requiere tener de forma indispensable los siguientes elementos:

- Un sistema de codificación o representación del problema.
- Poblaciones de individuos.
- Una función de adaptabilidad o ajuste al problema, conocida como función objetivo, a fin de evaluar cada solución codificada.
- Un mecanismo de selección.
- Un operador de cruzamiento.
- Un operador de mutación.

El algoritmo itera hasta cumplir con un criterio de paro. Este puede ser al alcanzar una cantidad límite de generaciones, encontrar un individuo que satisface un criterio que esté dentro de un umbral de aceptación, si el algoritmo "se estanca", al paso de las generaciones no produce mejores resultados o incluso una combinación de criterios.

Tipos de Representación o Codificación.

La representación de los cromosomas depende del problema a tratar e influye directamente sobre los resultados del AG. Existen distintas formas generales de llevar a cabo la codificación y destacan los siguientes:

- La *codificación binaria*: Es la más antigua de todas las existentes. Cada gen es un valor entre [0, 1].
  En la Figura 10 se puede observar un ejemplo de una representación binaria de un individuo de 4 bits (genes).

  0 0 0 1

  Figura 10 - Representación simbólica binaria.

- La *codificación real*: Aquí, cada variable del problema se asocia a un único gen que toma un valor real dentro del intervalo especificado, por lo que no existen diferencias entre el genotipo y el fenotipo, a menudo "*está intuitivamente más cerca del espacio de problemas*" (Fleming & Purshouse, 2002), (Figura 11).

  3.5 2.7 4.9 1.56

  Figura 11 - Representación simbólica real.

- *La codificación entera:* En este caso cada gen es representado por un valor entero (Figura 12).

  5 7 9 1 6

  Figura 12 - Representación simbólica entera.

- *La codificación gramatical:* (Marczyk, 2004; Mitchell, 1998): consiste en representar los individuos como cadenas precisamente, donde cada letra representa un gen.

Los anteriores no son los únicos y pueden existir más formas de representación ya que ello depende de las características que se quieran integrar según el problema.

La flexibilidad de los métodos de representación o codificación de soluciones, es que facilitan las operaciones que causan los cambios aleatorios en los individuos que fueron seleccionados: cambiar un 0 por un 1 o viceversa, sumar o restar al valor de un número una cantidad elegida al azar, o cambiar una letra por otra.

Población.

Primero se describen conceptos básicos, relacionados con los elementos que comprenden la genética de poblaciones.

- **Población**. Conjunto de individuos que representan las soluciones a optimizar.
- **Individuo**. Se denomina individuos o cromosoma a cada miembro o cadena de la población.
- **Gen**. A su vez, se conoce con el nombre de gen a cada elemento del cromosoma que tiene significado por sí mismo.
- **Generación**. Se denomina generación a cada iteración del algoritmo en donde se crean los nuevos individuos (Figura 13).

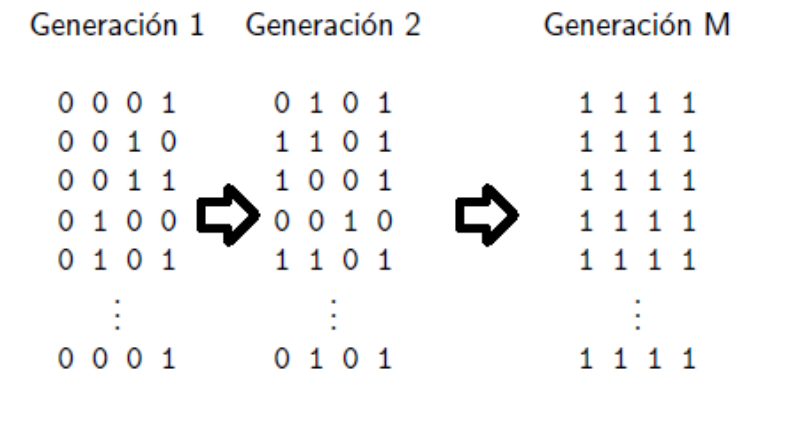


Figura 13 – Ejemplo de Generaciones

Tamaño de la población.

Una pregunta importante que surge al momento de aplicar un AG a un problema que se desea resolver con dicha técnica es: ¿Cuál es el tamaño ideal de la población?, incluso en principio suele pasar desapercibido ya que puede parecer trivial, sin embargo las poblaciones pequeñas pueden verse limitadas en el principio de exploración y con ello no cubrir más áreas del espacio de búsqueda, por el contrario trabajar con poblaciones excesivamente grandes puede repercutir en el tiempo y costo computacional. "*Para cada*

problema deberá determinarse empíricamente el tamaño adecuado y relacionarse el tamaño de la población a la magnitud y complejidad del problema" (Sarria Cerro, 2010).

Goldberg en 1989 (Goldberg & others, 1989), con base a un estudio demostró que el tamaño óptimo de la población para cadenas de longitud l, con codificación binaria, crece exponencialmente con el tamaño del cromosoma (individuo).

Posteriormente Alander, basándose en evidencia empírica, sugiere que un tamaño de población comprendida entre l y 2l es suficiente para atacar con éxito los problemas considerados por él (Alander, 1992).

Población inicial.

Aunque existen diversas formas de establecer una población inicial, la más común es generando soluciones (individuos) al azar, otra manera sería crear poblaciones iniciales a partir de otras heurísticas o búsquedas locales, sin embargo, algunos estudios demuestran que al no crear dicha población de forma aleatoria se puede acelerar la convergencia a un óptimo sin la garantía que este sea precisamente el óptimo global, lo que es denominado convergencia prematura, dicho de otro modo se estaría limitando el espacio de búsqueda y diversidad.

Función de Adaptabilidad.

Por cada individuo (solución) que se crea o se genera debe existir una forma de compararlo con respecto a otros individuos, para ello se establece una métrica que permita evaluar de manera cuantitativa, la cual es denominada *función de adaptabilidad* (también en la literatura es mencionada como función de adaptabilidad, función de aptitud y en inglés como fitness).

Al hacer mención de la mejor solución, se hace referencia a aquella que optimice el valor de la función de adaptabilidad para el problema que se está tratando. A pesar de que existen varios tipos de AG's, todos tienen en común que para cada iteración, todos los miembros de la población se evalúan de acuerdo a dicha función, así mismo una buena función de adaptabilidad, debe ser capaz de distinguir entre diversas soluciones que estén cercanas a la mejor.

La evolución de la población de individuos se desarrolla respecto al valor de la función de adaptabilidad para cada individuo como una solución al problema. Los individuos con mejor

aptitud tendrán mayor probabilidad de producir descendientes. Análogamente esta idea corresponde con el principio de la evolución natural: *supervivencia* de los más aptos.

Mecanismos de Selección.

La aplicación de los operadores de selección permite obtener el conjunto de individuos de la población actual que serán candidatos a ser parte de la siguiente generación de una población dada.

En términos biológicos, los individuos mejor adaptados tienen más probabilidad de sobrevivir y diversificarse.

Si bien la técnica metaheurística, selecciona aquellos individuos cuya función de adaptabilidad sea mayor, se corre el riesgo de converger en torno a un óptimo local de forma prematura (poca diversificación) por lo que se debe establecer un balance al elegir los individuos; tanto conservar los mejores e incluir una proporción de los "peores" con ello darles una oportunidad de reproducirse y en tal caso puedan contener información útil para las siguientes generaciones.

La probabilidad de que este operador seleccione a cada individuo es mayor cuanto mejor sea su valor en la función de adaptabilidad. Algunas de las formas de las que se dispone son las siguientes:

- *Selección proporcional*. Es el criterio de selección de la propuesta original de Holland (Holland, 1975), con este método la probabilidad que tiene un individuo para reproducirse es proporcional a su valor en la función de adaptabilidad. El comportamiento es similar al de una ruleta. Dentro de este están las técnicas:
    - *La ruleta* (Kenneth Alan De Jong, 1975).
    - *Sobrante Estocástico* (Booker, 1982; Brindle, 1981).
    - *Universal Estocástica* (Baker, 1987).
    - *Muestreo determinístico* (Kuri-Morales, 2004).

  El algoritmo propuesto para llevar a cabo la creación de cierta proporción de la población (a excepción de la inicial) en cada generación se apoya de este método de selección, por lo que enseguida se ilustra un ejemplo.

Con los porcentajes de la cuarta columna de la Tabla 3, se elabora una ruleta (Figura 14) en la que se puede observar (y se espera) que el cromosoma 3 tenga más oportunidad de ser seleccionado dado que cuenta con la mayor proporción.

Tabla 3 - Tabla de ejemplo para ilustrar la selección por ruleta.

Cromosoma No.	Cadena	Objetivo	% del Total
1	11010110	254	24.5
2	10100111	47	4.5
3	00110110	457	44.1
4	01110010	194	18.7
5	11110010	85	8.2
Total		1037	100

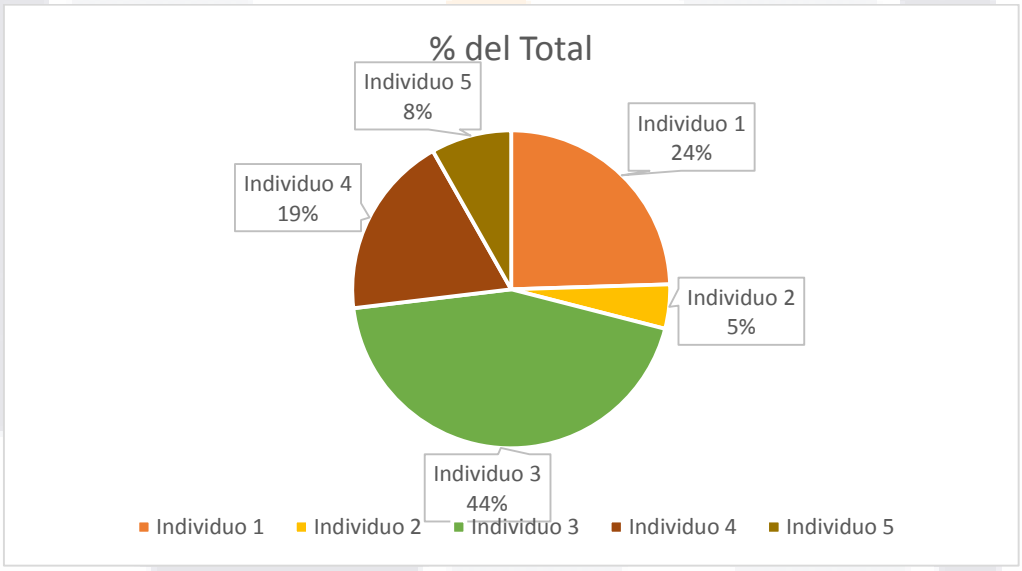


Figura 14 - Ruleta que representa los porcentajes de la Tabla 3.

- *Selección elitista.* Se fuerza a que el mejor individuo de la población sea seleccionado como padre. Este tipo de selección es utilizado en uno de los algoritmos propuestos.
- *Selección por torneo*. Consiste en seleccionar de forma aleatoria un conjunto de individuos de la población, dependiendo del tamaño del torneo, luego dichos individuos compiten entre sí y se elige aquél con mejor valor de aptitud. Dicho proceso se realiza

tantas veces como elementos existan en la población. Para el caso de un torneo binario, la competencia es llevada a cabo entre dos individuos aplicando el mismo principio.

- *Selección por ranking*. Propuesto por Whitley (Whitley & others, 1989), a cada individuo de la población se le asigna un rango numérico basado en el valor de su función de adaptabilidad, y para llevar a cabo se utiliza este ranking, en lugar de las diferencias absolutas en aptitud.
- *Selección por truncamiento*. Los individuos de la población son ordenados según el valor de su función de adaptabilidad y una proporción p (por ejemplo =1/2, 1/3, 1/4,...) de los individuos con mejor valor es seleccionada y reproducida 1/p veces. De igual manera uno de los algoritmos hace uso de esta técnica.

Cruzamiento.

En biología este proceso se refiere a que los individuos al juntarse y crear descendencia, estos heredan características de sus progenitores, por tanto, análogamente y siendo el operador más importante de esta técnica metaheurística al aplicar el operador de cruzamiento se consigue combinar información de varias soluciones para que estas creen diversidad y con ello tratar de encontrar una mejor solución.

La forma en que se realiza el cruce depende del tipo de representación que se elija. Para ilustrar algunas posibilidades en seguida se hace mención y descripción de las principales técnicas (aunque no son las únicas) a la vez que se ilustra con un simple ejemplo, para todos los casos la representación es de tipo binario y tiene 8 genes:

- *Cruzamiento simple o de un punto*. Se selecciona un punto de corte en la cadena de cada uno de los padres y se generan nuevos individuos combinando las partes que generan los cortes anteriores (Figura 15). El Algoritmo Genético Canónico, utiliza el cruce basado en un punto.

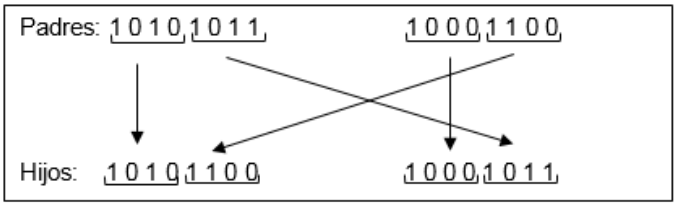


Figura 15 - Cruzamiento simple o de un solo punto.

- *Operador con dos puntos de corte*. Es análogo al anterior, salvo que se seleccionan dos puntos de corte y los padres intercambian los elementos de la cadena que quedan entre dichos puntos para generar los descendientes (Figura 16).

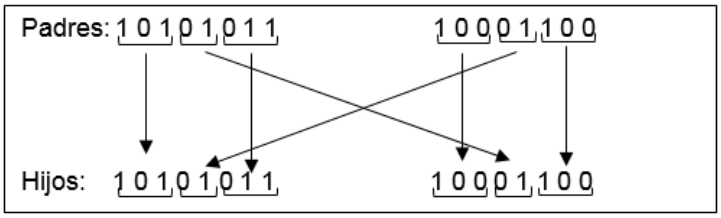


Figura 16 - Cruzamiento con dos puntos de corte.

Aunque se han hecho investigaciones teniendo en consideración más de un punto de cruce, De Jong en 1975 (Kenneth Alan De Jong, 1975) llegó a la conclusión que el cruce basado en dos puntos, representaba una mejora a la solución pero al añadir más puntos de cruce se corre el riesgo de romper las buenas soluciones.

- *Operador de cruce conforme a una máscara* (Syswerda, 1991). En este caso, el individuo resultante se obtiene de acuerdo a un criterio establecido por una máscara. En el ejemplo de la Figura 17 la máscara, es una cadena de ceros y unos. Para cada posición de la descendencia se tomará el gen del padre 1 si el valor de la máscara para dicha posición es 1 y del padre 2 si el valor del gen es 0.

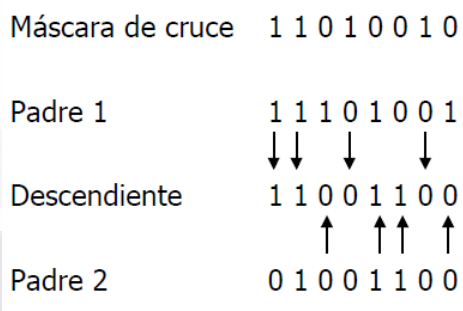


Figura 17 - Operador de cruce con mascara.

Existen otros operadores de cruce específicos para un determinado problema como son, por ejemplo, los definidos para el TSP, la idea de que el cruce debería de ser más probable en algunas posiciones ha sido descrita por varios autores (Davis & others, 1991; Holland, 1975; Levenick, 1991; Schaffer & Morishima, 1987)

Mutación.

La mutación de un individuo provoca que alguno de sus genes, generalmente uno sólo, varíe su valor de forma aleatoria.

La mutación se suele aplicar de manera conjunta con el operador de cruzamiento, una vez que se lleva a cabo el cruce de forma exitosa, entonces uno o ambos de los descendientes se muta con cierta probabilidad Pm, de esta manera se imita el comportamiento que se da en la naturaleza, pues cuando se copia el material genético en ciertas ocasiones ocurre algún tipo de error (o cambio), por lo general sin mayor transcendencia.

La probabilidad de mutación es muy baja, generalmente menor al 1%. Esto se debe sobre todo a que los individuos suelen tener un ajuste menor después de mutados, sin embargo se realizan mutaciones para garantizar que ningún punto del espacio de búsqueda quede sin ser explorado.

Al igual que en el cruzamiento, existen diferentes técnicas para llevar a cabo la mutación dependiendo del tipo de representación, si se trabaja con codificaciones binarias se puede simplemente de forma aleatoria negar un bit, otra técnica consiste en intercambiar los valores de dos (o incluso más) genes del cromosoma. Con otro tipo de codificaciones no binarias existen otras opciones, se puede consultar el trabajo de sarria (Sarria Cerro, 2010) en dicho trabajo se mencionan y explican otras técnicas.

Para el caso del trabajo que se está presentando se llevará a cabo una mutación inteligente, que será explicada más adelante.

Algoritmos de Estimación de Distribución.

Durante los últimos años se ha venido trabajando en una nueva familia de Algoritmos Evolutivos que siguen un esquema diferente y son llamados Algoritmos de Estimación de Distribución (AED's) conocidos ampliamente en inglés como: *Estimation of Distribution Algorithms* (EDA's) (Larranaga & Lozano, 2001).

"*Estos métodos se basan principalmente en sustituir el cruce y la mutación por la estimación y posterior muestreo de una distribución de probabilidad aprendida a partir de los individuos seleccionados. Este conjunto de algoritmos ha sido objeto de gran atención por parte de la comunidad científica alrededor de la computación evolutiva y los modelos gráficos probabilísticos*" (Larranaga, Lozano, Mühlenbein, Informationstechnik, & Germany, 2003), otro hecho que hay que agregar es que en la mayoría de los EDA's en cada iteración, el modelo se vuelve a estimar hasta que se consigue una criterio de paro.

En muchos problemas de optimización, las variables implicadas en la solución pueden o no interactuar entre sí para proporcionar un efecto positivo en su función de adaptabilidad. Por ejemplo, en el famoso problema del *OneMax* (o *BitCounting*) (Schaffer & Morishima, 1987) que es un simple problema que consiste en maximizar el número de 1's en un cromosoma, dicho de otro modo la función de adaptabilidad corresponde a la suma de todos los bits 1, al tratar de resolverlo mediante AG, cada gen del cromosoma contribuye de manera independiente e individual a la función de adaptabilidad.

En consecuencia, para resolver problemas de programación de operaciones es deseable tener el conocimiento de la interacción entre las variables. Según Shakya y sus colegas (Shakya, McCall, & Brown, 2006) esto puede ser usado para dirigir la búsqueda más eficientemente.

Por otra parte un AG busca encontrar soluciones prometedoras en la población aplicando los operadores de cruzamiento y mutación, asumiendo que estos producirán mejores resultados, sin embargo ni el operador de cruzamiento ni el de mutación, intentan aprovechar la interacción que pudiera existir entre las variables (genes), sino que simplemente se llevan a cabo de forma aleatoria definidos con cierta probabilidad, dicha aleatoriedad puede algunas veces alterar los valores que ofrece la interacción entre variables y con ello no obtener efectos positivos en la función de adaptabilidad. Otra

consecuencia que puede ocasionar es que se requiera un mayor tiempo computacional para converger a una buena solución.

Al notar este hecho, los investigadores han llevado a cabo estudios para descubrir y aprovechar la interacción entre variables asociadas al problema en estudio. Esta tarea ha originado dos enfoques principalmente. Ricardo y su colega (Pérez Rodríguez & Hernández Aguirre, 2015) en su documento de reporte describen dichos enfoques y que en su momento motivaron a (Larranaga & Lozano, 2001) a la creación de esta metaheurística.

El primer enfoque está basado en el cambio de la representación del problema. La idea es manipular la representación de la solución para evitar una importante alteración en las variables que interactúan en la cadena de soluciones. Los algoritmos genéticos de (Goldberg & others, 1989), (Georges Raif Harik, 1997) y (Kargupta & Buescher, 1996) caen en esta categoría.

El segundo enfoque está basado en cambiar el proceso de variación. Lo que se pretende es aprovechar la relación entre las variables, mediante la estimación de una distribución de la población y con ella muestrear la siguiente población de tal manera que sean sustituidos los operadores de cruzamiento y mutación involucrados en el proceso de exploración, o dicho de otra manera ahora la exploración se hace mediante la estimación y muestreo.

En seguida se explica la metodología general de la metaheurística a la que se ha hecho mención.

Metodología General.

La forma de operar de un EDA es en principio igual a un AG, es decir se genera una población inicial P que consta de M soluciones, a partir de aquí un EDA está fundamentado en los siguientes tres pasos básicos los cuales iteran de forma continua hasta que se establezca un criterio de paro, previamente establecido (Larrañaga & Lozano, 2002):

1. *Seleccionar algunos individuos de la población.*
2. *Estimar el modelo probabilístico subyacente a dichos individuos seleccionados.*
3. *Muestreo de la distribución de probabilidad aprendida, con el fin de obtener una nueva población de individuos.*

Con el proposito de entender mejor la técnica, en el esquema de la Figura 18 que fue tomado del articulo de (Abdelmalik Moujahid, Inza, & Larranaga, 2015; Larranaga & Lozano,

2001) se puede observar un esquema de la aproximación EDA, iniciando con M individuos generados de forma aleatoria y que constituyen la población inicial D_0, Luego para individuos se calcula su valor de adapabilidad, a partir de aquí, en un primer paso, un número N (N <= M) de individuos es seleccionado (habitualmente aquellos con mejor valor en su función de adaptabilidad), como segundo paso en seguida se lleva a cabo la inducción del modelo probabilistico n-dimensional que mejor refleja la relacion (llamado también interdepencia) entre las *n* variables, en un tercer paso, M nuevos individuos (la nueva población) son generados por medio de la simulación de la distribución de probabilidad aprendida en el paso anterior. Estos tres pasos se repiten hasta que se cumpla un criterio de paro previamente establecido (máximo de poblaciones, no mejora la solución, etc).

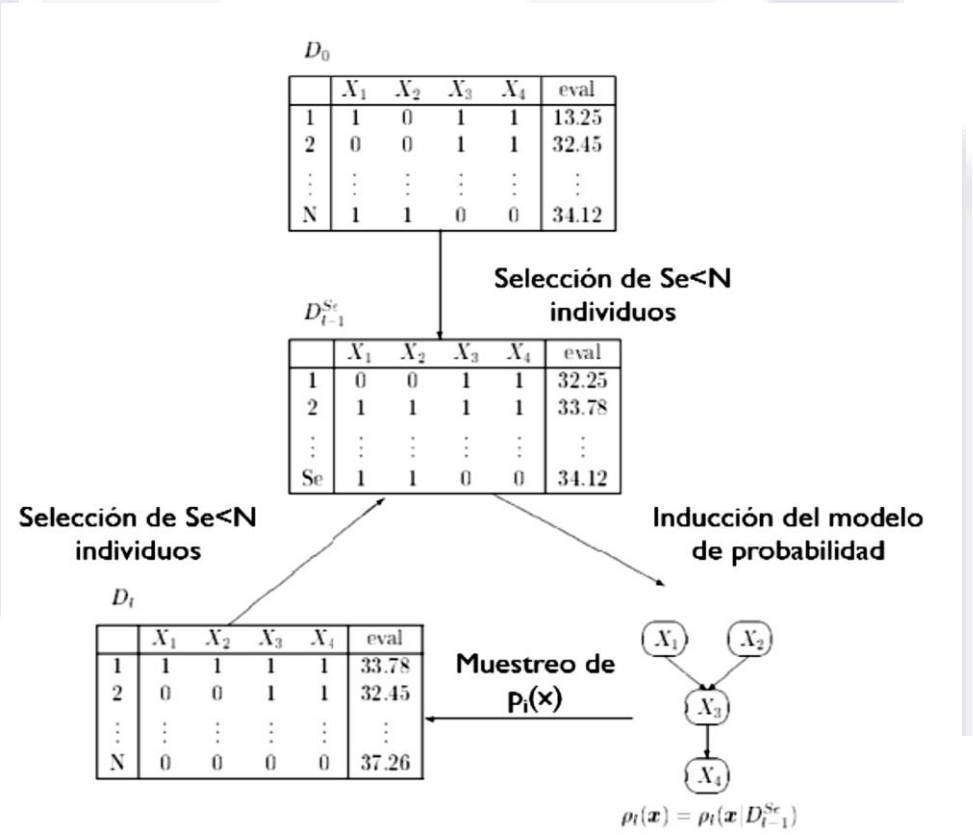


Figura 18 - Metodología general de un EDA (Abdelmalik Moujahid et al., 2015) a partir de (Larranaga & Lozano, 2001).

Del esquema y descripción anterior se presenta el pseudocódigo de aproximación del EDA (Figura 19).

EDA

$D_0 \leftarrow$ Generar M individuos (la poblacion inicial) al azar

Repeat for $l = 1, 2, \ldots$ hasta que se verifique el criterio de parada

$D_{l-1}^{Se} \leftarrow$ Seleccionar $N \leq M$ individuos de D_{l-1} de acorde con el metodo de seleccion

$p_l(\boldsymbol{x}) = p(\boldsymbol{x}|D_{l-1}^{Se}) \leftarrow$ Estimar la distribucion de probabilidad de que un individuo se encuentre en los individuos seleccionados

$D_l \leftarrow$ Muestrear M individuos (la nueva poblacion) de $p_l(\boldsymbol{x})$

Figura 19 - Pseudocódigo de la aproximación de un EDA (Larranaga & Lozano, 2001).

Como se ha hecho mención, los EDA's son una familia de técnicas dentro de los Algoritmos Evolutivos, por tanto dichas técnicas se pueden clasificar por el tipo de variables involucradas; discretos o continuos. *"Aunque el objetivo real es clasificarlos según el tipo de vínculo utilizado por su modelo de distribución*" (Pérez Rodríguez & Hernández Aguirre, 2015). Según el tipo de conexión utilizado en su modelo de distribución, los EDA's pueden clasificarse en univariados, bivariados y multivariados.

- Univaridados. En esta categoría los EDA's asumen que no hay dependencia entre las variables, ósea que no se consideran interacciones entre las variables, por lo tanto, la distribución de probabilidad $p_i(x)$ llega a ser simplemente el producto de las probabilidades marginales univariadas de todas las variables de la solución representada, como se puede ver en la siguiente ecuación (Ecuación 1):

$$p_l(x) = \prod_{i=1}^{n} p_i(x)$$

Ecuación 1

Debido a que el modelo de distribución utilizado en esta categoría es simple, los algoritmos son computacionalmente económicos y se ejecutan muy bien en problemas en donde la interacción entre variables es despreciable. Algoritmos conocidos en esta categoría son: *Univariate Marginal Distribution Algorithm (UMDA)* propuesto por Mühlenbein (Mühlenbein, 1997; Mühlenbein & Paass, 1996), Population Based Incremental Learning (PBIL) introducido por Baluja (Baluja, 1994) y *Compact Genetic Algorithm (cGA)* de Harik y sus colegas (Georges R. Harik, Lobo, & Goldberg, 1999).

- Bivaridados. La estimación de la distribución de probabilidad se puede llevar a cabo considerando la interacción entre pares de variables, esto es, que es suficiente considerar estadísticos de orden dos. Obviamente estos algoritmos se desempeñan mejor donde existe tal interacción. Algunos de los algoritmos más populares son *Mutual Information Maximization for Input Clustering (MIMIC)* desarrollado por Bonet (De Bonet, Isbell, Viola, & others, 1997), *Combining Optimizers with Mutual Information Trees (COMIT)* de Baluja y Davies (Baluja & Davies, 1997) y el *Bivariate Marginal Distribution Algorithm)* introducido por Pelikan & Mühlenbein (Pelikan & Mühlenbein, 1999).
- Multivariados. Si existe una interacción entre dos o más variables el algoritmo cae en esta categoría, evidentemente llegan a ser más complejos por la cantidad de combinaciones que en algún momento puedan existir, la mayoría de estos algoritmos utilizan redes bayesianas para codificar las distribuciones de probabilidad en cada paso. Dentro de esta categoría destacan *Extended Compact Genetic Algorithm (EcGA)* de Harik (G. Harik, 1999), *Factorised Distribution Algorithm (FDA)* de (Mühlenbein & Mahnig, 1999), *Bayesian Optimization Algorithm (BOA)* de Pelikan y sus colegas (Pelikan, Goldberg, & Cantú-Paz, 2000a, 2000b; Pelikan, Goldberg, & Cantu-Paz, 2000), *Learning Factorised Distribution Algorithm (LFDA)* (Mühlenbein & Mahnig, 1999) y el Estimation of Bayesian Network Algorithm (EBNA) de Etxeberria y Larrañaga (Etxeberria & Larranaga, 1999).

Es importante señalar que la técnica utilizada para dar solución al problema que se ha planteado en este documento de tesis es un UMDA.

4. Hibridación.

En secciones anteriores de este documento se ha hecho mención del termino hibridación[5] (o sus variantes idiomáticas y derivados), así mismo se hizo mención que una de las técnicas desarrolladas utiliza este concepto es por eso que en este punto se explica el concepto teórico de hibridación tanto de forma general y como es aplicado en las técnicas metaheurísticas y una clasificación general.

1. Antecedentes.

Primeramente hay que definir el concepto de ***hibrido***, dicho término puede adquirir distintos significados según la ciencia o área en la que se aplique. Al consultar el Diccionario de la Real Academia (RAE) una definición nos dice que híbrido[6] es "*adj. Dicho de una cosa: Que es producto de elementos de distinta naturaleza.*", así mismo al consultar en el diccionario en línea Merriam-Webmaster[7] dice que es "*algo que está formado por la combinación de dos o más cosas*" por tanto de forma general se puede concluir que es la combinación de dos o más elementos (dígase cosas, entes biológicos, técnicas, etc.) para dar pie a una nueva "especie" mejorada.

Dicho lo anterior y al aplicarlo al ámbito de las metaheurísticas, existen varias percepciones de lo que una metaheurística hibrida es, pero se pude afirmar por tanto que una metaheurística híbrida consiste en aprovechar lo mejor de dos o más técnicas para crear una nueva de tal manera que se optimice el proceso de búsqueda.

Blum y sus colegas (Blum, Roli, & Sampels, 2008) en su capítulo del libro "Hybrid metaheuristics: an emerging approach to optimization" dicen que en el contexto de la optimización combinatoria, los algoritmos pueden ser clasificados ya sea en *completos* o *aproximados*. Los algoritmos completos garantizan encontrar para cada instancia de tamaño finito de un problema de optimización combinatoria una solución óptima en un tiempo razonable (Nemhauser & Wolsey, 1988; Papadimitriou & Steiglitz, 1982); pero para

[5] Según el Servicio de Consultas Lingüísticas de la RAE (2012) son correctas en español las formas hibridar, hibridación y sus derivados, pero no es correcta la forma hibridización ni tampoco sus derivados.
[6] Según la Real Academia (Consultado el 03/04/2016).
[7] Según el diccionario Merriam-Webmaster (Consultado el 04/04/2016)

problemas que son NP-completos (Garey & Johnson, 1979, 2002), no existe un algoritmo de tiempo polinómico, asumiendo que P≠NP.

Por lo tanto, los métodos completos podrían necesitar un tiempo computacional exponencial muy alto por lo que puede llegar a ser ineficiente para fines prácticos. Debido a su intratabilidad, se han diseñado una gran cantidad de métodos aproximados, los cuales encuentran buenas soluciones en tiempos computacionales razonables.

En esta clase de problemas, la búsqueda de una solución requiere un balance organizado entre la exploración y explotación del espacio de búsqueda: una búsqueda no conducida es muy ineficiente.

En la década de los sesenta se diseñaron diversos métodos aproximados, denominados heurísticos, capaces de encontrar buenas soluciones o incluso en muchos casos la solución óptima o la mejor conocida. Muchos de estos métodos fueron inspirados en la resolución de algún problema en particular que aunque de fácil representación la solución es muy complicada, como lo son el problema del vendedor viajero, el problema de la mochila, etc.

Sin embargo, los métodos eran útiles para el problema que habían sido inspirados, por lo que de ello surgió la necesidad de crear una nueva clase de algoritmos, que se basan en la combinación o conducción de métodos heurísticos básicos en un marco de alto nivel para aumentar las capacidades de exploración y explotación del espacio de búsqueda, estos métodos son denominados metaheurísticas (Bäck, 1996; Blum & Roli, 2003; Goldberg & others, 1989), con el surgimiento de estas técnicas se permitió extender su aplicación a una amplia gama de problemas de CO representativos del mundo real

Este hecho, junto con su notable eficacia, fueron los principales factores que llamaron la atención de la comunidad científica. Como mecanismo para mejorar la eficiencia de las metaheurísticas, se han propuesto en los últimos años combinaciones de las mismas (en parte o en su totalidad) o incluso con otras técnicas de optimización para establecer referencia a las ***metaheurísticas hibridas***.

2. Clasificación.

De hecho la idea de hibridar metaheurísticas no es nueva, se remonta a los orígenes mismos de las metaheurísticas. La principal motivación para hibridar algoritmos es obtener técnicas con mejor desempeño que aprovechen y unan las ventajas de estrategias purista de forma sinérgica.

Sin embargo tales híbridos no eran tan populares por el hecho que en la comunidad de investigadores defendían, diferían e incluso competían entre sí sobre cuál técnica metaheurística era su favorita y "mejor" a las demás.

Esta situación cambió radicalmente gracias al teorema del "No Free Lunch" (Wolpert & Macready, 1997), que establece que no puede haber una estrategia de optimización general que sea globalmente mejor que otra, esto es, que para cualquier algoritmo, cualquier rendimiento elevado sobre una sola clase de problemas se paga exactamente en el rendimiento en otra clase.

Torres (Torres, 2010) establece en su tesis doctoral citando a Raidl (Raidl, 2006a) que actualmente, se recomienda la hibridación de técnicas heurísticas porque está probado que producen mejores soluciones que los mecanismos puristas en la resolución de problemas NP-completos.

En cuanto a la clasificación de las técnicas han surgido varias propuestas de diversos autores (Cotta-Porras, 1998; El-Abd & Kamel, 2005; Talbi, 2002) pero Raidl (Raidl, 2006b) unificó dichas clasificaciones y taxonomías propuestas por los autores citados para generalizar en la Figura 20, creada a partir de Raidl (Raidl, 2006b).

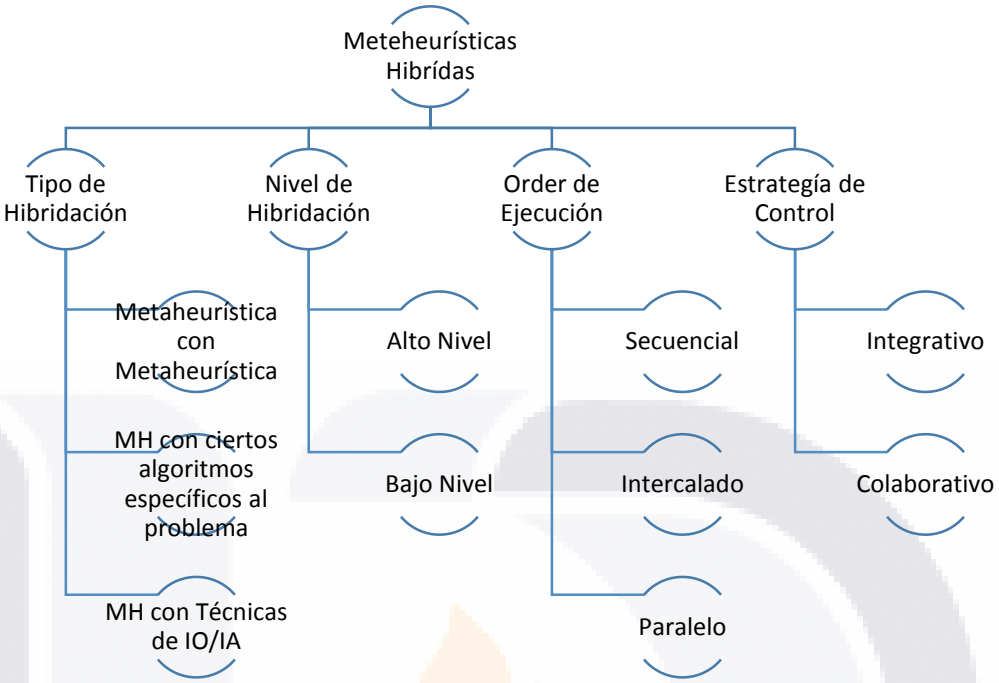


Figura 20 - Clasificación de las Metaheurísticas Híbridas.

5. Fundamentos básicos de Bioinformática y Proteómica.

A partir de que Hooke (Hooke, 1961) en su obra “Micrographia” acuño el término “célula”, hoy en día es posible afirmar que todos los seres vivos están constituidos por células como elemento básico; dichas unidades comparten una estructura compleja para llevar a cabo todas las funciones biológicas esenciales. “*Los seres vivos, aunque infinitamente diversos por fuera, son muy similares por dentro*” (Romero-Zaliz, 2005).

En este apartado se explican tanto los antecedentes como los conceptos básicos de bioinformática y proteómica, para permitir que el lector tenga nociones y se contextualice sobre estos temas.

1. Bioinformática.

Las células vivas, al igual que las computadoras, contienen información, y se estima que han estado evolucionando y diversificándose durante miles de años. Dichas células, sin que se conozca excepción alguna almacenan su información genética en forma de moléculas de ADN (ácido desoxirribonucleico) (Berg, Stryer, & Tymoczko, 2007; Bray et al., 2006).

El ADN de todos los organismos está compuesto por los mismos elementos tanto físicos como químicos, que se denominan bases, ordenados de lado a lado en una estructura de doble hélice. El orden de estas bases contiene la información pertinente para crear un organismo con todas sus particularidades.

Por otra parte un *gen*[8] se define como un fragmento de la secuencia de ADN que corresponde a una sola proteína, así mismo el *genoma*[9] de una célula es la totalidad de la información genética incluida en su secuencia completa de ADN, su tamaño suele expresarse como el número total de pares bases, o el número total de genes.

[8] A partir de https://www.genome.gov/GlossaryS/index.cfm?id=70 consultado (06/04/2016)
[9] A partir de https://www.genome.gov/GlossaryS/index.cfm?id=90 consultado (06/04/2016)

El tamaño del genoma de un organismo puede variar a razón de cientos, hasta billones (quizá más) de pares bases, por ejemplo se estima que el genoma humano contiene alrededor de 3 billones de pares bases (Morton, 1991; Zanlungo, Arrese, & Rigotti, 1999) organizados en 46 cromosomas[10]. Como puede observarse al tratar de procesarlo, almacenarlo y analizarlo se genera una enorme cantidad de información.

En las últimas décadas, los avances en la Biología Molecular junto con el aumento en el poder de computo disponible para la investigación en este campo han permitido la rápida secuenciación de grandes porciones de genomas de diversas especies incluido el *Genoma Humano* (Collins, Morgan, & Patrinos, 2003), diseñado con el fin de secuenciar los 23 pares (46 cromosomas) contenidos.

Ello trajo como consecuencia un aumento exponencial en la cantidad de información generada, en este sentido, con la genómica, la biología se ha convertido en una ciencia de la información tanto en la recopilación de datos genéticos, como en su almacenamiento, análisis e integración; dichos volúmenes de información son almacenadas en grandes bases de datos disponibles para la comunidad científica, por citar algunas *GenBank* (Benson, Karsch-Mizrachi, Lipman, Ostell, & Wheeler, 2003) y EMBL (Kanz et al., 2005).

Pelta (Pelta, 2013) textualiza en su tesis doctoral que "*La magnitud de la información que genera la investigación genómica es tal que, probablemente, supera la magnitud de información que genera la investigación en otras disciplinas científicas*".

Esto dio pie al surgimiento de la **bioinformática** como un área en la frontera entre la biología y las ciencias de la computación cuyo principal objetivo es el desarrollo y uso de técnicas matemáticas y computacionales para ayudar en el tratamiento masivo de datos y en la resolución de problemas de la biología molecular.

Una definición más formal de Luscombe (Luscombe, Greenbaum, Gerstein, & others, 2001) dice que "*La bioinformática es una conceptualización de la biología en términos de moléculas (en el sentido de química física) y la aplicación de técnicas informáticas (derivadas de disciplinas como matemática aplicada, estadística, ciencia de la computación) para entender y organizar la información asociada con dichas moléculas en gran escala. En*

[10] https://www.genome.gov/GlossaryS/index.cfm?id=33 (07/04/2016)

breve, bioinformática es un sistema de manejo de información para la biología molecular que tiene un gran número de aplicaciones prácticas".

Esta área es de interés y especialmente prometedora debido a que los modelos que se aplican a una buena parte de los problemas de biología, resultan ser NP-completos, por lo que deben ser abordados mediante técnicas no deterministas (heurísticas/metaheurísticas). Por lo que, la resolución a dichos problemas y la obtención de nuevos mecanismos de solución causan impacto tanto en la Informática como en la Biología.

Dentro del amplio catálogo de problemas que abarca la bioinformática, Luscombe (Luscombe et al., 2001) menciona como ejemplos: la construcción de árboles filogenéticos para detectar antecesores comunes, el alineamiento simple y múltiple de secuencias, la construcción de mapas de genomas, la predicción de estructuras de proteínas, la comparación de moléculas, el agrupamiento y clasificación de estructuras proteicas, el análisis de perfiles de expresión génica, y un largo etcétera.

Si bien los genes atraen mucho la atención, realmente son las proteínas las que llevan a cabo la mayor parte de las funciones de la vida y generan la mayor parte de las estructuras celulares. En el siguiente apartado se describe un área que se apoya bastante de la bioinformática y cuyo objeto de estudio es precisamente todo lo relacionado con las proteínas.

2. Proteómica.

Esta sección comprende los antecedentes, definición y mención de algunos de los estudios que lleva a cabo la proteómica haciendo un poco más de énfasis a la alineación de secuencias.

Preliminares.

Las proteínas[11] son los pilares fundamentales de la vida, son macromoléculas complejas, constituidas por subunidades más simples que son llamados aminoácidos[12], existen 20 aminoácidos diferentes que se combinan entre sí de distintas maneras para formar cada tipo de proteína. "*La secuencia de aminoácidos y las características químicas de los mismos causan que la proteína se pliegue en una estructura tridimensional que define su funcionalidad en la célula*" (Pelta, 2013).

Las proteínas llevan a cabo una labor fundamental en los seres vivos, siendo las biomoléculas más diversas y versátiles, realizan diversas funciones, entre ellas estructurales, enzimáticas, transportadoras, etcétera. Ósea que cada una cumple una función específica según su secuencia de aminoácidos, determinada genéticamente.

El conjunto de todas las proteínas contenidas en una célula es llamado "*proteoma*". Se puede decir que el proteoma es un elemento altamente dinámico púes cambia constantemente en respuesta a las reacciones tanto al interior como al exterior de la célula. Pelta (Pelta, 2013) también escribe en su tesis que "*La química de una proteína y su comportamiento está especificada por la secuencia de un gen, pero también por el número y la identidad de otras proteínas fabricadas en la célula al mismo tiempo y con las cuales ésta se asocia y reacciona*".

San Miguel y sus colegas (San Miguel Hernández, San Miguel, & Martín-Gil, 2010) escriben que aunque el término proteoma se suele usar para cualquier conjunto de proteínas, su descripción más acertada es al conjunto de proteínas que constituyen una célula o tipo

[11] A partir de http://proteinas.org.es/que-son-las-proteinas (08/04/2016)
[12] Puede consultar https://www.genome.gov/GlossaryS/index.cfm?id=5 (08/04/2016)

celular, bajo condiciones específicas. "*El proteoma de una célula es la expresión de su fenotipo característico*" (San Miguel Hernández et al., 2010).

Definición.

López y González (López-Muñoz & González, 2007) en su libro "Historia de la psicofarmacología, Volumen 3" refieren que el término "proteoma" fue utilizado por primera vez en 1995, para hacer mención al conjunto de proteínas que se expresan a partir de un genoma. Y la ciencia que lo estudia se denominó "proteómica".

La proteómica, es el área que estudia las proteínas, reacciones e interacción entre sí, será objeto de investigación durante mucho tiempo, debido a que se perfila como una potente herramienta para el estudio y descubrimiento de aspectos fisiopatológicos en los seres humanos y ayudará a dilucidar las bases de la salud y enfermedad.

Esto, en conjunto con el progresivo y rápido avance biotecnológico, ha contribuido a la creación de un gran volumen bibliográfico sobre la utilización de técnicas proteómicas en el abordaje de algunas enfermedades y su presencia en el laboratorio clínico. Puede consultar los trabajo de (González-Buitrago, 2006; González-Buitrago, Ferreira, & Lorenzo, 2007; Master, 2005; Muñiz, Brugés, & Vaca, 2005; Righetti et al., 2005).

Otra iniciativa interesante es la creación de la HUPO (Human Proteome Organization) (disponible en: http://www.hupo.org), creada en el año 2001 para impulsar un mayor conocimiento de la importancia de la Proteómica y las oportunidades que ofrece en el diagnóstico, el pronóstico y el tratamiento de las enfermedades. Se han constituido posteriormente varios grupos: HPPP (Human Plasma Proteome Project), HLPP (Human Liver Proteome Project), PSI (Proteome Standards Initiative), HBPP (Human Brain Proteome Project) y MRPP (Mouse and Rat Proteome Project).

Según (Mojica, Sánchez, & Bobadilla, 2003) los grandes tópicos de la proteómica son los siguientes:

- Interacción existente entre genoma y proteoma.
- Fuente y manejo de las proteínas.
- Separación de las proteínas y sus componentes.
- Identificación de las proteínas.

- Función de las proteínas (localización celular, interacciones proteína-proteína, determinación de la estructura terciaria, etc.).
- Investigaciones que se puedan aplicar en el área del diagnóstico y tratamiento de enfermedades humanas.
- "*Aplicaciones en informática que generan muchas anotaciones basadas en la homología de las secuencias, la construcción de bases de datos, generación de algoritmos para el análisis y, por último, la estandarización*" (Mojica et al., 2003).
- Otras áreas que surjan de la colaboración de la comunidad internacional científica, con apego a consideraciones éticas y legales.

Aunque el objetivo de esta investigación no es ahondar en las técnicas, en seguida se hace una breve descripción sobre la "semejanza entre secuencias" y sus técnicas, debido a que el insumo principal para los algoritmos fue obtenido mediante una técnica de semejanza.

Comparación de Secuencias.

La bioinformática es fuente medular en el procesamiento de la gran cantidad de datos que se generan con el estudio de las proteínas, quizá la tarea más recurrente en esta área sea la comparación de secuencias, si se dispone de una secuencia obtenida, la primer pregunta sería ¿Cuál es la función de esta nueva secuencia?

Para ello se puede comparar con todas las secuencias conocidas y almacenadas en bases de datos. Al momento de llevar a cabo la comparación si se encuentran secuencias similares a la secuencia desconocida, se puede inferir por su parecido que comparten funciones biológicas similares. A este mecanismo se le conoce como "*asignación de función por homología*" (Droit, Poirier, & Hunter, 2005).

Existen diferentes tipos de alineamiento y cada uno tiene distintas técnicas para cumplir dicha alineación, sin entrar a detalle en seguida se muestra un resumen de ellas tomando como fuente el artículo enciclopédico ("Alineamiento de secuencias", 2015). Sin embargo si se quiere consultar más información sobre algunas de las técnicas, puede referirse al trabajo académico del Dr. Trelles[13] (Carazo & Trelles, 2007).

[13] Disponible en http://www.bioscripts.net/col/index.php , sección "Apuntes – Bioinformática – Tema 3" (12/04/2016).

Las aproximaciones computacionales al alineamiento de secuencias se dividen en dos categorías: alineamiento global y alineamiento local. En un alineamiento global se "fuerza" al alineamiento a ocupar la longitud total de todas las secuencias (secuencias problema). Mientras que los alineamientos locales identifican regiones similares dentro de largas secuencias que normalmente son muy diversas entre sí. Se aplican gran variedad de algoritmos computacionales al problema de alineamiento de secuencias.

- **Alineamiento de pares:** Los métodos de alineamiento de pares, o emparejamientos, se utilizan para encontrar la mejor coincidencia en bloque (local) o alineamiento global de dos secuencias. Los alineamientos de pares sólo pueden utilizarse con dos secuencias a la vez. Algunas técnicas son:
    - *Métodos de matriz de puntos.*
    - *Programación dinámica*.
    - *Métodos de palabra corta*. Los métodos de palabra corta son más conocidos por su implementación en las herramientas de búsqueda en bases de datos FASTA y la familia BLAST (Altschul, Gish, Miller, Myers, & Lipman, 1990).
- **Alineamiento múltiple de secuencias:** Es una extensión del alineamiento de pares, lo que permite usar más de dos secuencias de manera simultánea. Los métodos de alineamiento múltiple intentan alinear todas las secuencias de un conjunto dado.
    - *Programación dinámica.*
    - *Métodos progresivos.*
    - *Métodos iterativos.*
    - *Descubrimiento de motivos.*
    - *Técnicas inspiradas por las ciencias de la computación.*
- **Alineamiento estructural:** Los alineamientos estructurales, que son específicos de las proteínas y, algunas veces, de secuencias de ARN (ácido ribonucleico), usan información sobre la estructura secundaria y terciaria de la proteína o molécula de ARN para alinear las secuencias. Estos métodos pueden usarse para dos o más secuencias, y producen típicamente alineamientos locales.
    - *DALI.* El método DALI (del inglés, Distance matrix ALIgnment, alineamiento de matriz de distancias).
    - *SSAP.* SSAP (del inglés Sequential Structure Alignment Program, programa de alineamiento de estructura secuencial).

- *Extensión combinatoria*.

El Dr. Trelles (Carazo & Trelles, 2007) comenta que los alineamientos múltiples, son la base para la construcción de árboles filogenéticos. Hoy en día es raro no encontrar un estudio bioinformático en el que de alguna manera no esté involucrada una técnica de análisis múltiple de datos.

Capítulo III: Metodología.

En el capítulo del marco teórico, se describió la teoría base, los componentes y el comportamiento general de los principales algoritmos utilizados, en este capítulo se describen el desarrollo y como fueron aplicados dichos algoritmos a la problemática que se le dio solución, ósea la metodología utilizada para el desarrollo de los algoritmos para llevar a cabo la clusterización de hongos mediante su información proteómica.

Primero se describe a grandes rasgos los antecedentes que dieron lugar al desarrollo del trabajo de tesis propuesto, de donde y como fueron obtenidos los datos y como se discriminaron.

Luego se plantea el diseño de la investigación; en la cual se hace hincapié en lo novedoso y aportaciones, se plantea y describen de forma general cada técnica utilizada para atacar el problema, es decir se hace mención a los algoritmos que se desarrollaron e implementaron; la teoría base de dichos algoritmos fueron descritos en el **capítulo II – "Marco Teórico"** y son el Algoritmo Genético (AG) y el Algoritmo de Estimación de Distribución (EDA).

Posteriormente se describe a detalle la función de adaptabilidad utilizada en cada técnica, la cual es fundamental ya que sirve para poder calificar a los individuos (soluciones), finalmente se hace una descripción del esquema de trabajo de la investigación.

Dicho de otro modo, dentro del presente capítulo se detalla la forma en cómo cada uno de los algoritmos propuestos trabaja para generar los clústeres de hongos.

1. Antecedentes y Caso de Estudio.

Teniendo como insumo principal el resultado (matriz de semejanzas) de la investigación: "Aprendizaje heurístico de modelos probabilísticos para identificación de secuencias genómicas"[14] realizada en 2014-2015 por el grupo de investigadores de bioinformática del Centro de Ciencias Básicas (CCB) de la Universidad Autónoma de Aguascalientes (UAA); que constituye la información proteómica de 33 hongos, a los que se les realizó un análisis de los mejores aciertos bidireccionales (BBH, por sus siglas en inglés de Bidirectional Best Hit), con ayuda del software BLAST (Basic Local Allignment Search Tool, por sus siglas en inglés).

El mejor acierto de un gen particular hacía un genoma objetivo, es el gen en ese genoma que representa una mejor semejanza. La semejanza es bidireccional si los dos genes son mejores aciertos, ósea de un gen hacia el otro. Un BBH representa una fuerte similitud entre dos genes, y es considerado evidencia de que los genes podrían ser ortólogos derivados de un ancestro común; puede ampliar la información consultando el trabajo de Overbeek (Overbeek, Fonstein, D'Souza, Pusch, & Maltsev, 1999).

Como resultado del procesamiento mencionado y a manera de ejemplo se obtuvo un resultado como el que presenta la Tabla 4.

Tabla 4 - Ejemplo de una Matriz de Semejanzas.

	H1	H2	…	H33
H1	**1**	0.5	…	0.2
H2	0.5	**1**	…	0.56
…	…	…	**1**	…
H33	0.2	0.56	…	**1**

La tabla corresponde a una matriz cuadrada simétrica de 1089 celdas; al ser una matriz simétrica dividida por una diagonal de 1's, los valores por encima de dicha diagonal están reflejados en su correspondiente parte inferior, por lo tanto el punto de intersección (celda) tanto de (fila, columna), como de (columna, fila) representa el valor de semejanzas entre un

[14] Financiado por la Universidad Autónoma de Aguascalientes bajo el Proyecto PIINF 12-8.

hongo y otro, ejemplo H1 tiene una semejanza de 0.5 con respecto a H2 (lo mismo que H2 contra H1), como se hizo mención dicha tabla posee una diagonal principal con valores de 1's, debido a que cada hongo comparado consigo mismo exhibe un 100% de semejanza.

Tal cómo se puede observar, el número de combinaciones generadas es 2^{33} por lo que, para la identificación de los grupos factibles es un problema de explosión combinatoria por lo tanto el desarrollo del presente se enfocará a atacar dicha problemática mediante técnicas metaheurísticas híbridas.

La siguiente tarea consistió en diseñar y desarrollar un mecanismo de clusterización que parte de éste insumo.

La lista de hongos con los que se realizó este trabajo se muestra en la Tabla 5.

Tabla 5 - Lista de Hongos considerados en la investigación.

No.	Nombre		No.	Nombre
1	Ashbya gossypii.		18	Fusarium oxysporum
2	Aspergillus fumigatus.		19	Fusarium verticilloides.
3	Aspergillus nidulans.		20	Histoplasma capsulatum.
4	Aspergillus terreus.		21	Kluyveromyces.
5	Batrachochytrium dendrobatidis		22	Loderomyces elongisporus.
6	Botrytis cinerea.		23	Magnaporthe grisea.
7	Candida albicans.		24	Neurospora crassa.
8	Candida glabrata.		25	Puccinia graminis.
9	Candida guilliermondii.		26	Rhizopus oryzae.
10	Candida lusitaniae.		27	Saccharomyces cerevisiae.
11	Candida tropicalis.		28	Sacharomices japonicus.
12	Chaetomium globosum.		29	Sclerotinia sclerotiorum.
13	Coprinus cinereus.		30	Stagonospora nodorumm.
14	Coccidiodes immitis.		31	Uncinocarpus reesii.
15	Cryptococcus neoformans serotype A.		32	Ustilago maydis.
16	Debaryomyces hansenii		33	Yarrowia lipolytica.
17	Fusarium graminearum.			

La condición para **incluir** o **excluir** un hongo en esta investigación se basó en tener al alcance tanto la información del proteoma como el resultado de la comparación de los mejores aciertos bidireccionales de cada hongo contra los demás (matriz de semejanzas).

2. Diseño de la Investigación.

En este apartado se describen las aportaciones al trabajo de tesis, el diseño propuesto para llevar a cabo la aplicación de los algoritmos y la función de adaptabilidad que permite evaluar las soluciones.

Aportaciones, y Generalidades de los Algoritmos.

Las principales aportaciones de este trabajo de tesis, son **las técnicas metaheurísticas introducidas**, en combinación con los mecanismo diseñados para su funcionamiento, entre los que se encuentran: **la función de adaptabilidad** para cada una de las técnicas y **la mutación inteligente** (parte hibridada); además del **marco de trabajo** robusto y flexible, y finalmente el análisis de **los clústeres obtenidos** (a veces llamados grupos) a partir de la aplicación de las técnicas metaheurísticas mencionadas las cuales aportan información de valor científico a la comunidad.

Cabe resaltar que si bien el problema en cuestión consiste en crear clústeres de hongos a partir de su información proteómica, la novedad de las técnicas radica en que no están limitadas al contexto de los hongos, siempre que se provea de una matriz de semejanzas que cumpla los requisitos básicos de la Tabla 4, cada algoritmo será capaz de presentar un resultado.

Para el caso de la primera técnica se tomó como base el Algoritmo Genético (AG) descrito a detalle en el capítulo de “Marco Teórico” el cual está fundamentado en la teoría de la evolución de las especies (Darwin, 1859), que postula la supervivencia de los individuos más aptos; análogamente se diseñó una forma de representación, un mecanismo de selección de los mejores individuos y la aplicación de un mecanismo de mutación, pero en este caso no se llevó a cabo de manera purista sino que la mutación es inducida (**mutación inteligente**) con el propósito de robustecer uno de los principios fundamentales de la clusterización que consiste en juntar los individuos que más se parecen, lo anterior para dar pie a la creación de un AG híbrido, el cual fue denominado: **AGH-CHIP** por sus siglas de “**A**lgoritmo **G**enético **H**íbrido para la **C**lusterización de **H**ongos mediante su **I**nformación **P**roteómica”.

La segunda técnica se fundamentó en la familia de algoritmos evolutivos denominados Algoritmos de Estimación de Distribución (EDA, por sus siglas en inglés) propuesta por Larrañaga (Larranaga & Lozano, 2001), (Larrañaga & Lozano, 2002). Familia que al igual que el AG se basa en poblaciones, pero con la diferencia de que sustituye el cruce y mutación por la estimación basada en probabilidades. Dentro del conjunto de algoritmos para estimar y muestrear la población fue seleccionado el Univariate Marginal Distribution Algorithm (UMDA, por sus siglas en inglés) por su sencillez y fácil aplicación con lo que se creó el **UMDA-CHIP** por sus siglas de "**U**nivariate **M**arginal **D**istribution **A**lgorithm para la **C**lusterización de **H**ongos mediante su **I**nformación **P**roteómica".

Cabe destacar que tanto el AGH-CHIP como el UMDA-CHIP se desarrollaron completamente desde cero en un lenguaje de programación de nivel medio con características de bajo nivel pero con la disponibilidad de poder utilizar estructuras de alto nivel. Dicho lenguaje de programación es el C++ que es una extensión del C que aunque tiene sus orígenes entre 1969 y 1972 en los laboratorios Bell, luego publicado y liberado en 1978 (Kemighan & Ritchie, 1978) aún sigue siendo bastante apreciado por la eficiencia del código que produce ya que se necesitan pocas instrucciones para crear aplicaciones robustas.

El entorno de IDE (Siglas en inglés de Integrated Development Enviroment) sobre el que se desarrollaron los algoritmos es el WxDev-C++ en su versión 7.4.2.569.

Otro punto a resaltar es que se implementó el uso de estructuras dinámicas de tal manera que es posible variar en caso de ser requerido la longitud del cromosoma (en ocasiones llamado individuo) dependiendo del tamaño de la matriz de semejanzas que se lea, de igual manera es posible redefinir el número de individuos de la población en tiempo de ejecución, de hecho esto permitió la implementación de un diseño factorial de experimentos descrito más adelante.

En resumen y como ya se dijo el algoritmo es flexible ya que permite un tamaño nxm de las poblaciones con las que trabaja.

Esquema de Trabajo.

La Figura 21 representa a manera de esquema, la metodología propuesta de principio a fin sobre el desarrollo de la investigación.

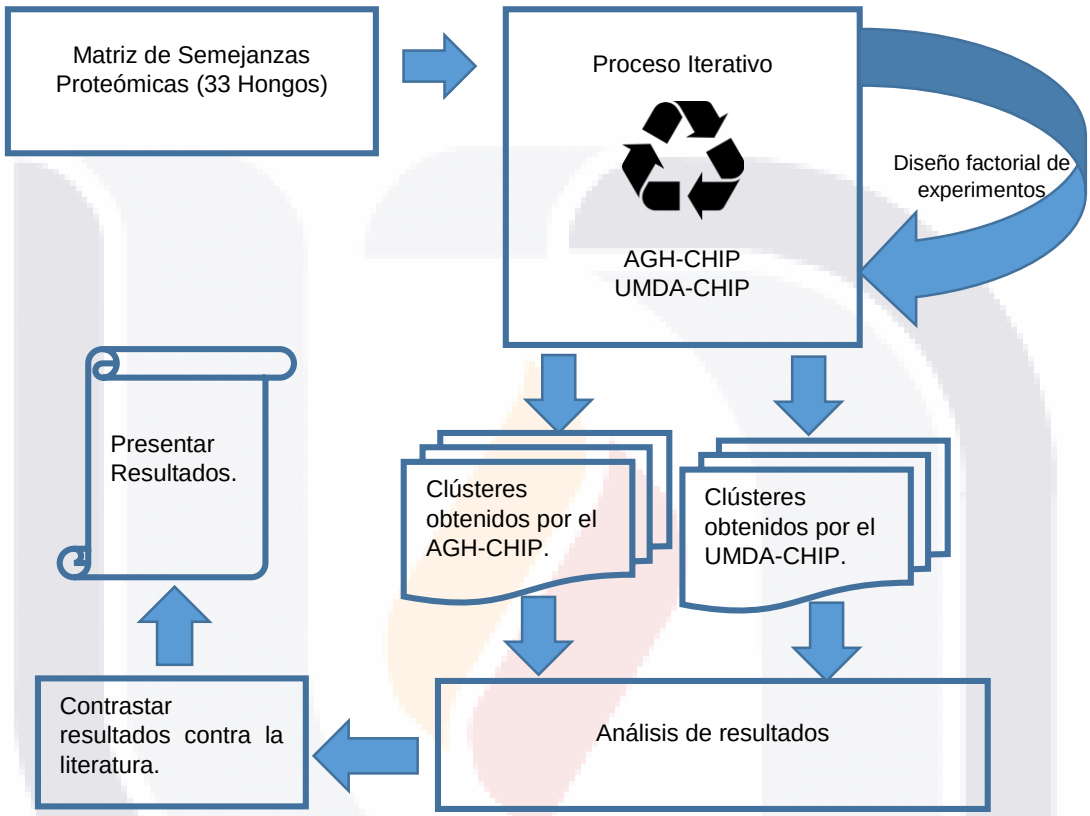


Figura 21 - Marco de Trabajo de la Investigación.

A continuación se describen sus principales. Más adelante se presentan los algoritmos diseñados detalladamente, pues son parte del resultado de este trabajo.

- *Matriz de semejanzas Proteómicas (33 Hongos):* En la sección “Antecedentes y Caso de Estudio” del presente capítulo se describió a detalle en que consiste y de donde se obtuvo tal matriz, y que para el caso de las técnicas que se proponen corresponde al insumo que permite establecer un valor de adaptabilidad a cada solución propuesta.

- *Proceso Iterativo:* En esta parte del proceso se lleva a cabo el desarrollo, implementación, ajuste de parámetros de cada uno de los algoritmos. De manera paralela se realizó un diseño factorial de experimentos para obtener la mejor combinación de parámetros.
- *Clústeres Generados (AGH-CHIP y UMDA-CHIP):* Cada algoritmo una vez concluida su ejecución produce resultados que corresponden a propuestas de clústeres de hongos, en esta parte únicamente se almacenan los resultados en archivos de texto para su análisis posterior.
- *Análisis de resultados:* En esta etapa se lleva a cabo un análisis empírico con los resultados obtenidos y con base a ello se determina si se ajustan los parámetros del experimento, una vez se tiene un resultado favorable o el algoritmo (cualquiera de los dos) ha convergido; posteriormente se llevan a cabo las respectivas pruebas estadísticas para determinar si los algoritmos son sensibles a la variabilidad de los parámetros en términos de las respuestas (adaptabilidad y tiempo de ejecución).
- *Contrastar los resultados con la literatura:* Una vez que se tiene la cadena con el mejor valor de adaptabilidad (de todo el experimento de cada algoritmo), e identificados los clústeres que fueron creados, se procede a “darle un nombre” a cada objeto en su respectivo clúster, con base a lista de hongos presentada en la sección de “Antecedentes y Caso de Estudio” de este mismo capítulo (Tabla 5), el siguiente paso es llevar a cabo un análisis minucioso y exhaustivo a la luz de la teoría, para inferir si los clústeres generados por cada uno de los algoritmos empataran (o que tanto se parece) con una clasificación reconocida.
- *Presentar resultados:* se reportan los resultados que fueron obtenidos.

En los capítulos siguientes se describe a detalle el proceso de construcción e implementación de los algoritmos de los que se ha hecho énfasis en este capítulo, que son el AGH-CHIP y el UMDA-CHIP, por sí mismos corresponden a una parte de los resultados obtenidos durante el desarrollo de esta investigación.

Capítulo IV: Metaheurística Evolutiva AGH-CHIP.

En este capítulo primeramente se describen los mecanismo diseñados tanto para calificar las soluciones como para mejorarlas para luego describir componentes básicos del AGH-CHIP, es decir la metodología desarrollada para crear clústeres; luego explicar la forma como se puso a punto el algoritmo mediante la implementación de un experimento factorial además de definir las condiciones y parámetros del experimento, posteriormente se presentan los resultados obtenidos para con ellos aplicar un modelo estadístico que corrobora la interpretación empírica de los resultados. Adicionalmente se anexa una sección que corresponde a un conjunto de tablas con información biológica básica de la lista de hongos que están considerados en la investigación además su relación con la clasificación obtenida.

1. Mecanismos Diseñados.

Aquí se reportan los principales mecanismos que fueron diseñados y aplicados a la metaheurística y que son presentados como resultados de la misma. En primera instancia se describe la función de adaptabilidad que fue usada para calificar cada una de las soluciones, seguido de un mecanismo de mejora que proporciona un empuje a la técnica.

Función de Adaptabilidad.

Para garantizar que los mejores individuos sean seleccionados, es necesario contar con un mecanismo de evaluación de su calidad conocido como función de adaptabilidad, este instrumento es importante porque conduce la búsqueda en el proceso evolutivo.

En el caso del AGH-CHIP se utilizó un **enfoque mono-objetivo** (un objetivo), el cual consiste en maximizar la cohesión entre los hongos que pertenecen al mismo clúster

generado, luego, el número de clústeres generados son promediados hasta alcanzar un único valor que representa la ponderación asignada al individuo.

El algoritmo no permite que todos los objetos (hongos) pertenezcan a un mismo grupo, dicho de otro modo el algoritmo crea al menos 2 grupos.

Además en el caso de que se haya generado un clúster que tenga un solo hongo, por ejemplo G1 {H1} la solución es castigada asignándole el valor de la menor semejanza existente en la matriz debido a que no hay otros objetos con que comparar.

La Ecuación 2 representa la forma general del cálculo de la función mono objetivo, que como ya se dijo pretende maximizar **la semejanza** entre clústeres.

$$Semejanza = \sum_{i=1}^{k} \frac{SIG_k}{k}$$

Ecuación 2

Donde *k* representa el número de clústeres creados y SIG_k corresponde a la Semejanza Interna del k-ésimo Grupo.

SIG. A continuación se presenta el cálculo del SIG, mediante un ejemplo paso a paso, suponiendo que se tiene la siguiente matriz de semejanzas (Tabla 6).

Tabla 6 - Ejemplo de Matriz de Semejanzas.

	H1	H2	H3	H4	H5	H6	H7
H1	**1**	0.46	0.43	0.42	0.39	0.31	0.70
H2	0.46	**1**	0.70	0.70	0.35	0.46	0.46
H3	0.43	0.70	**1**	0.68	0.34	0.45	0.44
H4	0.42	0.70	0.68	**1**	0.33	0.44	0.43
H5	0.39	0.35	0.34	0.33	**1**	0.26	0.37
H6	0.31	0.46	0.45	0.44	0.26	**1**	0.32
H7	0.70	0.46	0.44	0.43	0.37	0.32	**1**

Además suponga que se genera la siguiente solución (Figura 22)

H1	H2	H3	H4	H5	H6	H7
2	3	1	2	3	1	1

Figura 22 - Ejemplo de Solución (1).

En un primer paso se determina el número de clústeres generados, para este ejemplo se crearon 3 clústeres. G1 {H3, H6, H7}, G2 {H1, H4}, G3 {H2, H5}.

La SIG para cada clúster se obtiene del promedio de todas las **combinaciones de semejanzas** tomadas de dos en dos (ósea, el valor de la intersección en la posición (m, n) en la matriz). Y considerando NC como el número de combinaciones tomadas de dos en dos.

Por lo tanto, el siguiente paso consiste en obtener el promedio de las combinaciones del primer grupo:

$$SIG_1 = [(H3, H6) + (H3, H7) + (H6, H7)] / NC.$$

Para este primer SIG, fueron 3 las combinaciones de pares de hongos (NC = 3) que se obtuvieron [(H3, H6), (H3, H7) y (H6, H7)].

Obteniendo los valores de la matriz y efectuando los cálculos se tiene:

$$SIG_1 = (0.45 + 0.44 + 0.32) / 3 = 1.21 / 3 = 0.4033333333333333.$$

Se acumula el resultado obtenido para luego repetir el paso anterior hasta que no haya grupos por comparar.

$$SIG_2 = (H1, H4) / NC = (0.42) / 1 = 0.42$$

…

$$SIG_n$$

Una vez que se tiene la suma de las SIG's, se obtiene el promedio para obtener el valor de semejanza (Ecuación 2) de la solución que se generó, la suma de los SIG's en el ejemplo ilustrado fue 1.453333333, al dividirlo entre el número de clústeres generados (*k*) que como ya se dijo es 3, la semejanza de la solución es:

$$Semejanza = 1.453333333 / 3 = 0.484444444$$

Mutación Inteligente.

Como se mencionó en el apartado de aportaciones, se incorpora un mecanismo que permite llevar a cabo una mutación "inteligente", evidentemente se está hablando del AGH-CHIP por ser una metaheurística que aplica dicho operador.

La mutación es un elemento de exploración muy importante dentro del espacio de búsqueda ya que potencializa y se enfoca en las áreas donde encuentra una buena solución.

Si bien es cierto que las técnicas metaheurísticas garantizan buenos resultados con sus mecanismos puristas para explorar ampliamente el espacio de soluciones de un problema, también es cierto que muy probablemente dicha exploración está basada en la aleatoriedad como es el caso del Algoritmo Genético Simple, porque lo que la exploración se lleva a cabo a ciegas confiando totalmente en dicha aleatoriedad.

Para los problemas de explosión combinatoria como en el caso del problema que se describe en este trabajo, el impacto computacional crece de forma exponencial.

Con el objetivo de minimizar tal impacto y "darle un empuje" al AGH-CHIP, se propone un mecanismo de mejora para poder llevar a cabo una mutación conducida, denominado **Mecanismo de Mutación Inteligente (MMI)**. El MMI está inspirado en el mecanismo de mejora que utiliza la metaheurística de búsqueda dispersa (Scatter Search en inglés) (Glover, 1977). Esta metaheurística cuenta con un mecanismo de mejora, que a partir de una solución específica (que quizá ni es factible) trata de recomponerla para tratar de convertirla en solución prometedora.

El MMI, es por lo tanto, un mecanismo que permite saltar espacios de búsqueda que no son prometedores por lo que los recursos computaciones son mejor aprovechados y la técnica se concentra más en aquellas soluciones que prometen.

El MMI, puede ser descrito como un mecanismo de exploración potencializado por el "conocimiento" de que una de las condiciones al momento de crear clústeres es que los objetos contenidos en un clúster en particular tengan una alta similitud entre sí (similitud interna del clúster) y esto es lo que busca mantener precisamente el MMI, en lugar de mutar aleatoriamente un objeto y con ello posiblemente descomponer la solución, lo que se hace es que después de seleccionar el objeto que será mutado, se identifica a cual objeto se parece más y se envía dicho objeto al clúster donde se encuentra el objeto con el que es

más semejante. **Se busca juntar los objetos con la mayor semejanza.** En general la técnica consta de los siguientes pasos (en el contexto de los hongos):

- Determinar cuál será el gen del cromosoma que se le va aplicar el MMI aleatoriamente.
- Buscar en la matriz de semejanzas a cuál hongo se parece más.
- Determinar e identificar si al hongo que se parece más está en otro clúster de ser así.
- Se procede a juntar esos hongos en el mismo grupo.

Con el fin de entender mejor lo anterior, en seguida se explica con un ejemplo como funciona y como se aplica la técnica.

Suponga que se tiene la siguiente matriz de semejanzas (Tabla 7).

Tabla 7 - Ejemplo de Matriz de Semejanzas (2).

	H1	H2	H3	H4	H5	H6	H7
H1	**1**	0.36	0.43	0.42	0.39	0.31	0.70
H2	0.36	**1**	0.70	0.70	0.35	0.46	0.46
H3	0.43	0.70	**1**	0.68	0.34	0.45	0.44
H4	0.42	0.70	0.68	**1**	0.33	0.44	0.43
H5	0.39	0.35	0.34	0.33	**1**	0.26	0.37
H6	0.31	0.46	0.45	0.44	0.26	**1**	0.52
H7	0.70	0.46	0.44	0.43	0.37	0.52	**1**

Además de la siguiente solución (Figura 23) seleccionada previamente por ruleta. La cual consta de los grupos: G1 {H2, H5}, G2 {H3, H4, H7} y G3 {H1, H6}.

H1	H2	H3	H4	H5	H6	H7
3	1	2	2	1	3	2

Figura 23 - Ejemplo de Solución (3)

Una vez que se determina el número de elementos a mutar (número de veces que se aplica el MMI), continuando con el ejemplo, se tiene lo siguiente:

- **H5** se selecciona de forma aleatoria para aplicarle el MMI.
- Al verificar en la matriz de semejanzas se puede ver que su semejanza más alta es con respecto a **H1** pues se parece en 0.39.

- Al verificar la solución se puede apreciar que H1 se encuentra en **G3**, mientas que H5 está en **G1**, por lo tanto se procede a hacer el cambio de tal manera que ahora H5 pertenezca a **G3**.

Por lo tanto ahora la solución quedaría de la siguiente manera (Figura 24):

H1	H2	H3	H4	H5	H6	H7
3	1	2	2	1	3	2

H1	H2	H3	H4	H5	H6	H7
3	1	2	2	3	3	2

Figura 24 - Ejemplo de Solución con MMI.

Y los grupos quedarían: G1 {H2}, G2 {H3, H4, H7} y G3 {H1, **H5**, H6}.

Como se hizo mención el AGH-CHIP tiene una parte híbrida, tal parte es precisamente este mecanismo inspirado en la búsqueda dispersa y diseñada para que los grupos se conformen de manera más eficiente.

2. Generalidades.

Como se hizo mención en el capítulo del marco teórico, el Algoritmo Genético, está basado en los postulados de la teoría de la evolución de Charles Darwin (Darwin, 1859), en los cuales se pretende emular la supervivencia del más apto por medio de la selección natural. Este mismo principio se puede aplicar en algoritmos computacionales para la solución de un gran número de problemas de distintas áreas, cómo en el caso de esta investigación.

Entrando en el tema computacional, la Figura 25 muestra el Algoritmo Genético Simple (SGA, por sus siglas en inglés) propuesto por (Golberg, 1989).

```
BEGIN /* Algoritmo Genetico Simple */
        Generar una poblacion inicial.
        Computar la funcion de evaluacion de cada individuo.
        WHILE NOT Terminado DO
        BEGIN /* Producir nueva generacion */
                FOR Tamaño poblacion/2 DO
                BEGIN /*Ciclo Reproductivo */
                        Seleccionar dos individuos de la anterior generacion,
                                para el cruce (probabilidad de selección
                                proporcional
                                a la funcion de evaluacion del individuo).
                        Cruzar con cierta probabilidad los dos
                                individuos obteniendo dos descendientes.
                        Mutar los dos descendientes con cierta probabilidad.
                                Computar la funcion de evaluacion de los dos
                                descendientes mutados.
                        Insertar los dos descendientes mutados en la nueva
                                generacion.
                END
                IF la poblacion ha convergido THEN Terminado := TRUE
        END
END
```

Figura 25- Algoritmo Genético Simple (Golberg, 1989)

Del cual podemos hacer mención de sus principales elementos para su implementación y ejecución.

1. *Una población inicial.*
2. *Mecanismo para evaluar la población inicial.*
3. *Mecanismo para generar las poblaciones futuras.*
4. *Establecer un Criterio de Paro.*

Como quedó establecido el SGA es la base del AGH-CHIP así que después de un minucioso análisis y de varias refinaciones, se obtuvo siguiente pseudocódigo simplificado (Figura 26).

```
INICIO
        Establecer Parámetros
        Generar P0
        Evaluar P0
        Ordenar P0
        Repetir
                Aplicar Elitismo (2 mejores)
                Aplicar MMI a un % de la población (% es definido el usuario)
                Generar resto de individuos de forma aleatoria
                Evaluar Pn
                Ordenar Pn.
        Hasta que no haya poblaciones por generar
FIN
```

Figura 26 - Pseudocódigo Simplificado del AGH-CHIP.

De lo anterior se pueden definir los siguientes pasos básicos que lleva a cabo el algoritmo desde que arranca hasta que se cumple el criterio de paro para crear clústeres de hongos, los cuales son:

1. Establecer parámetros iniciales (punto de arranque).
2. Mecanismo de representación (clústeres).
3. Generar una población inicial.
4. Un mecanismo de evaluación para ponderar la calidad de los clústeres obtenidos.
5. Un método de ordenamiento.
6. Mecanismo para generar las siguientes poblaciones.
7. Un criterio de paro.

En seguida se explica a detalle cada una de las fases (pasos anteriores) que componen y conlleva la ejecución el AGH-CHIP.

1. Establecer parámetros iniciales.

En primera instancia se establecen variables, se crean los archivos de salida que contienen las poblaciones creadas, también se lee y extrae la información de la matriz de semejanzas.

Los principales parámetros que de forma obligada se tiene que definir de forma correcta a fin de que el algoritmo se ejecute de forma correcta se ilustran en la Tabla 8.

Tabla 8 - Descripción de parámetros del AGH-CHIP.

Parámetro	Descripción
Nombre de archivo	Parámetro de entrada que corresponde al nombre del archivo, el cual corresponde a la matriz de semejanzas que es el insumo principal del algoritmo. La estructura de dicho archivo debe ser en texto plano y debe contener algún carácter que funja como separador de valores (tabulador por ejemplo). El número de genes se establece de forma automática al leer la estructura del archivo, al ser una matriz cuadrada la dimensión ya sea de filas o columnas corresponde al número de genes, para el caso de este algoritmo se está trabajando con una matriz de 33x33 por lo tanto la longitud será de 33 genes.
Tamaño de la población	Es el número de individuos (soluciones) que contendrá la población con los que va a trabajar el algoritmo.
Número de clústeres a crear.	Valor que se define para establecer hasta cuantos grupos (clústeres) tiene el algoritmo la posibilidad de crear. Como se hizo mención el algoritmo se ajusta de forma automática para que al menos sean dos clústeres.
Número de bits a mutar.	Este parámetro define hasta cuantos genes (bits) se les puede aplicar el MMI en un mismo individuo.
Porcentaje de la población a aplicar mutación	En líneas anteriores quedó establecido que el MMI es aplicado a un porcentaje de la población, el cual es definido por el usuario por lo tanto este parámetro permite que se establezca dicho porcentaje, el valor requerido es un valor entre 0 y 100.
Número de generaciones	Es el número máximo de generaciones a la será sometido cada corrimiento del algoritmo, además de que funge como principal y único criterio de paro del algoritmo.

2. Mecanismo de Representación.

Como ya se ha hecho mención a través del presente trabajo, el principal objetivo es la creación de clústeres de hongos basados en su información proteómica.

Aquí se explica como el algoritmo representa e identifica los clústeres creados en cada solución el número que se genera es una “etiqueta” para identificar al clúster, así, por ejemplo suponer que el tamaño de un individuo es de 10 genes (bits) y el número máximo de grupos a crear es 4, y suponer que de forma aleatoria se crean los 10 genes con un número comprendido entre 1 y 4; siguiendo esto y tomando en consideración que en el orden de izquierda a derecha y de manera ascendiente se va identificando a cada hongo (H1, H2 … Hn, donde n es el número máximo de bits), lo anterior queda representado de forma gráfica en la Figura 27.

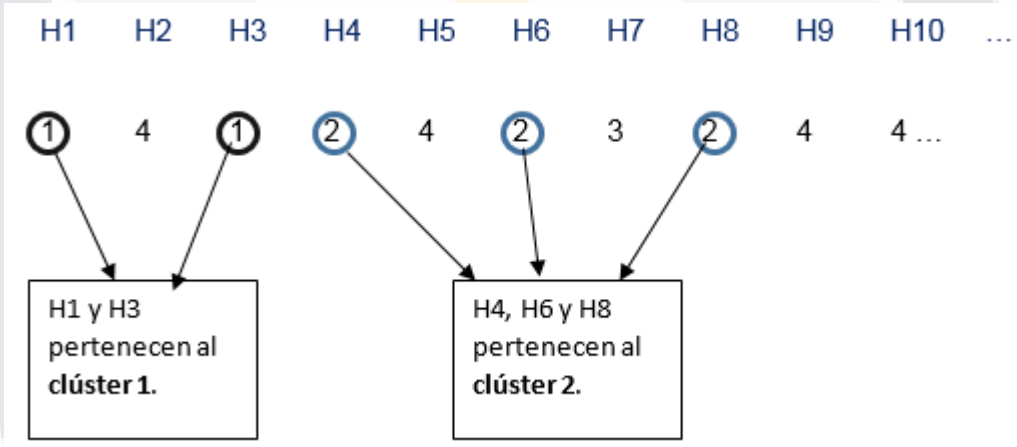


Figura 27 - Ejemplo de Representación de Clústeres en un Cromosoma.

3. Generar Población Inicial.

Una vez que se establecen de forma correcta los parámetros iniciales, el algoritmo está listo para iniciar.

En primera instancia se debe generar la población inicial definida en el pseudocódigo general del AGH-CHIP como P_0 y que corresponde a la generación inicial (Generación 0), el algoritmo ejecuta lo siguiente de forma iterativa hasta completar el tamaño de la población:

1. Para cada individuo, con base al valor del número máximo de clústeres que se pueden crear, aplicar la técnica de selección por ruleta (descrito en el capítulo de marco teórico) para seleccionar un número entre 2 y el máximo, para definir realmente cual será el máximo real de clústeres que se pueden crear. Ejemplo si define que el algoritmo puede crear un máximo de 8 clústeres, al aplicar la ruleta suponga que el número seleccionado corresponde a un 5, significa que para ese individuo se pueden crear hasta 5 clústeres.
2. Generar de forma aleatoria cada gen hasta completar el tamaño del individuo, estableciendo como valor máximo el establecido del paso anterior. Se ilustra en seguida un ejemplo tomando en consideración que la longitud del individuo es de 10 y se van a crear hasta 5 clústeres (grupos).

   2 5 2 3 1 3 4 3 1 5

   Lo anterior significa que se crearon los siguientes grupos: G1 {5, 9}, G2 {1, 3}, G3 {4, 6, 8}, G4 {7}, G5 {2, 10}.

La Tabla 9 muestra un ejemplo de una población de 15 individuos cada individuo tiene un tamaño de 10 genes y como se puede apreciar en cada individuo el máximo de clústeres es variable.

Tabla 9 - Ejemplo de Población.

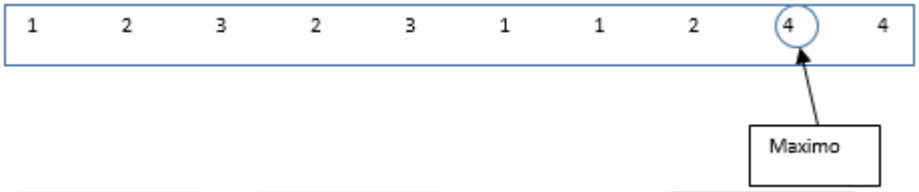
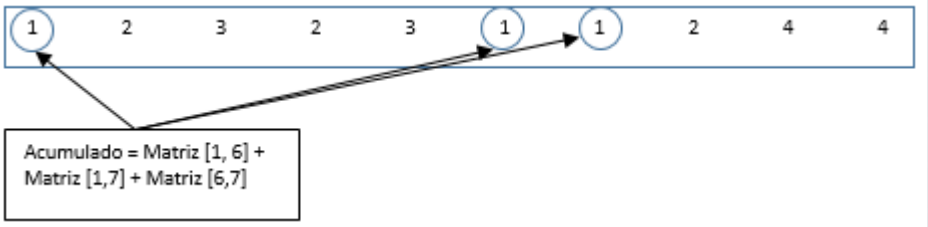
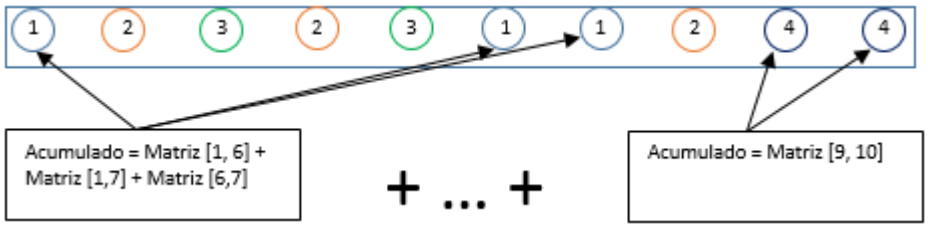
4	4	4	1	2	4	4	5	2	4
1	4	4	1	1	2	2	3	3	4
3	5	2	4	5	2	5	3	4	4
1	4	5	5	2	2	5	4	3	2
1	3	5	2	1	5	5	4	4	4
2	2	4	2	1	4	3	3	4	1
2	4	1	5	3	5	3	3	3	5
3	2	5	3	5	4	2	1	5	3
5	2	5	4	2	1	2	3	3	5
4	3	2	2	5	4	3	4	2	4
2	4	1	5	2	2	4	2	3	5
3	2	5	2	1	2	3	5	5	2
5	4	1	3	1	3	2	3	2	5
5	5	5	5	4	5	5	3	2	4
1	1	4	1	2	4	2	3	2	2

4. Mecanismo para Calificar cada Individuo.

En la sección “Función de Adaptabilidad” de este mismo capítulo se explicó a detalle la función que aplica el AGH-CHIP para calificar cada una de las soluciones generadas, por lo que aquí se describe un ejemplo adicional a fin de entender mejor el mecanismo. Recordando que la función de adaptabilidad del AGH-CHIP es una función mono-objetivo que busca maximizar la semejanza interna (**Semejanza intraclúster**) de los clústeres de un individuo. Cualquier detalle favor de referirse a la sección correspondiente.

La Tabla 10 explica el proceso que se efectuó para obtener la semejanza entre clústeres de un individuo.

Tabla 10 – Ejemplo de cálculo del Valor de Adaptabilidad del AGH-CHIP.

Suponga que se creó la solución mostrada. Determinar el número clústeres que fueron creados (máximo).   
Empezando con el clúster de menor valor, se procede a acumular el valor obtenido de la matriz de semejanzas en la posición (j, k) de ese mismo clúster (lo correspondiente a la SIG explicada en la función de adaptabilidad).   
Una vez terminado con los elementos del j-ésimo clúster, el valor acumulado se divide entre el número de combinaciones contabilizadas y dicho valor se acumula. En este caso se contabilizaron 3 combinaciones por lo tanto: Acumulado = (Matriz [1, 6] + Matriz [1, 7] + Matriz [6, 7]) / 3.
Se almacena dicho valor y se procede a repetir lo anterior hasta que ya no haya clústeres por comparar.   

Una vez acumulado el valor de cada uno de los clústeres que se crearon, dicho valor se divide entre el número máximo de clústeres creados que en este caso es 4 y con ello se está computando íntegramente la función de adaptabilidad del AGH-CHIP.

Lo anterior se explica de forma gráfica en el diagrama DFD de la Figura 28.

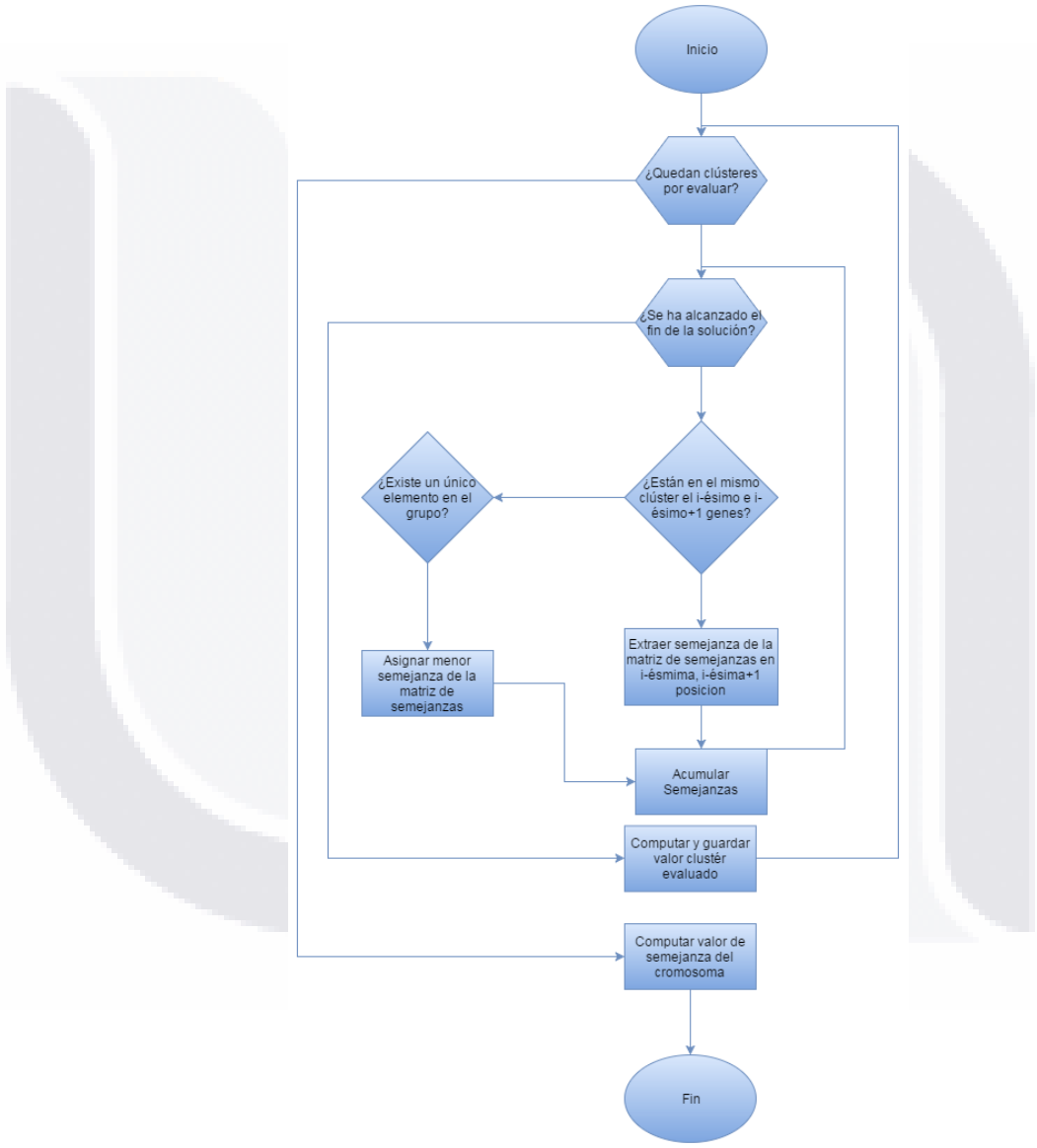
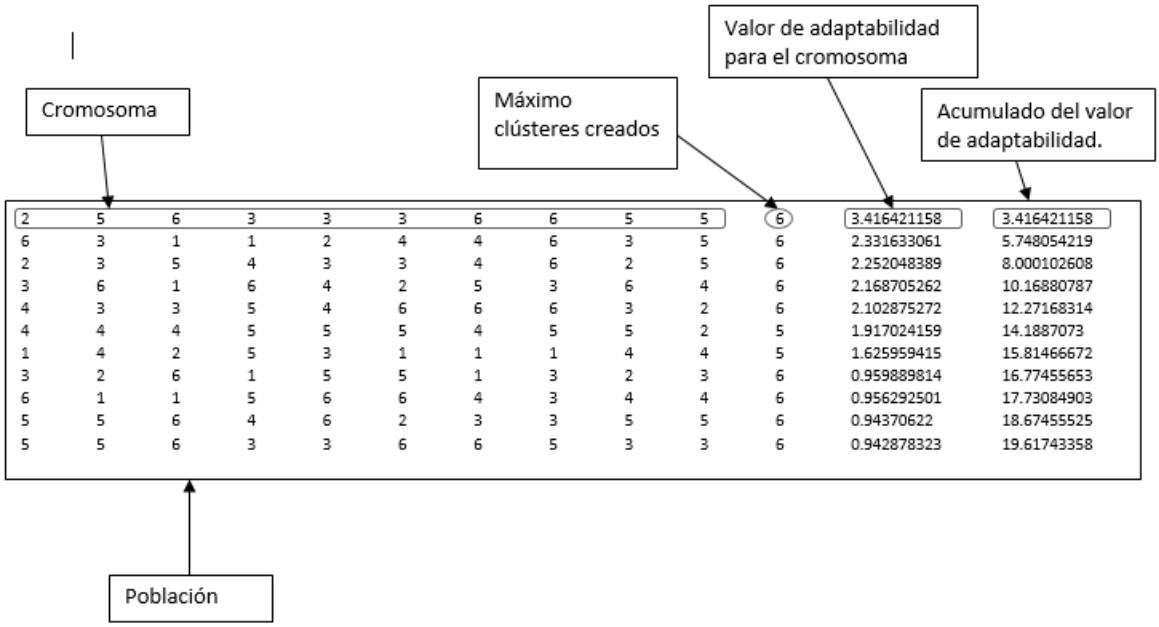


Figura 28 – DFD de la función de adaptabilidad del AGH (semejanza)

Lo anterior es únicamente para calcular el valor de adaptabilidad de un cromosoma, por lo tanto se repite la serie de pasos tantas veces como individuos haya en la población, adicionalmente se agrega a la población el número máximo de clústeres creados para cada individuo, el valor de adaptabilidad de dicho individuo así como el valor de adaptabilidad acumulado hasta ese momento. Lo anterior se muestra en la Figura 29, la cual es solamente un ejemplo ilustrativo.



2	5	6	3	3	3	6	6	5	5	6	3.416421158	3.416421158
6	3	1	1	2	4	4	6	3	5	6	2.331633061	5.748054219
2	3	5	4	3	3	4	6	2	5	6	2.252048389	8.000102608
3	6	1	6	4	2	5	3	6	4	6	2.168705262	10.16880787
4	3	3	5	4	6	6	6	3	2	6	2.102875272	12.27168314
4	4	4	5	5	5	4	5	5	2	5	1.917024159	14.1887073
1	4	2	5	3	1	1	1	4	4	5	1.625959415	15.81466672
3	2	6	1	5	5	1	3	2	3	6	0.959889814	16.77455653
6	1	1	5	6	6	4	3	4	4	6	0.956292501	17.73084903
5	5	6	4	6	2	3	3	5	5	6	0.94370622	18.67455525
5	5	6	3	3	6	6	5	3	3	6	0.942878323	19.61743358

Figura 29 - Ejemplo de población completa.

Una vez que se completa y evalúa la población, el siguiente paso es ordenar dicha población de forma descendente por el valor de adaptabilidad computado para cada individuo. El método utilizado es “ordenamiento de la burbuja”. Se explica de mejor manera en el siguiente punto.

5. Método de Ordenamiento.

Como ya se hizo mención, el siguiente paso una vez que se tiene la población completa, es el ordenar dicha población para ello se usa el método de ordenamiento de la burbuja, debido a su facilidad de implementación.

Según la enciclopedia en línea Wikipedia "*Funciona revisando cada elemento de la lista que va a ser ordenada con el siguiente, intercambiándolos de posición si están en el orden equivocado. Es necesario revisar varias veces toda la lista hasta que no se necesiten más intercambios, lo cual significa que la lista está ordenada, dado que solo usa comparaciones para operar elementos, se lo considera un algoritmo de comparación, siendo el más sencillo de implementar*" ("Ordenamiento de burbuja", 2015).

La Figura 30 muestra el pseudocódigo general para el algoritmo de la burbuja

```
procedimiento DeLaBurbuja (a_0, a_1, a_2, ..., a_(n-1))
  para i ← 1 hasta n hacer
    para j ← 0 hasta n − i hacer
      si a_(j) > a_(j+1) entonces
        aux ← a_(j)
        a_(j) ← a_(j+1)
        a_(j+1) ← aux
      fin si
    fin para
  fin para
fin procedimiento
```

Figura 30 - Pseudocódigo del Algoritmo de la Burbuja[15].

En base a ello y al implementarlo a la problemática en cuestión se tiene en consideración lo siguiente:

- El ordenamiento se hace de forma descendente en función al valor de adaptabilidad.
- Cada cadena del cromosoma se considera como un único elemento al momento de hacer el intercambio.

Lo anterior se ilustra con la Figura 31

[15] Tomado de https://es.wikipedia.org/wiki/Ordenamiento_de_burbuja 24/04/2016

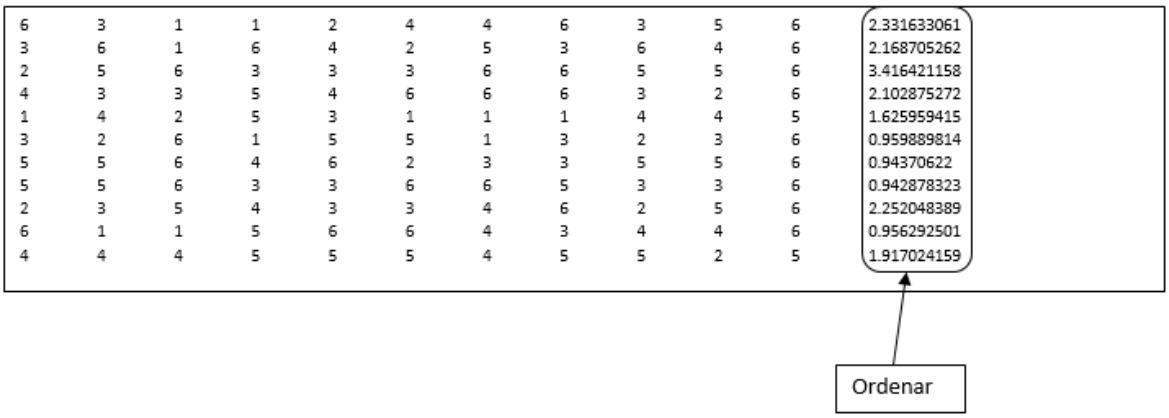


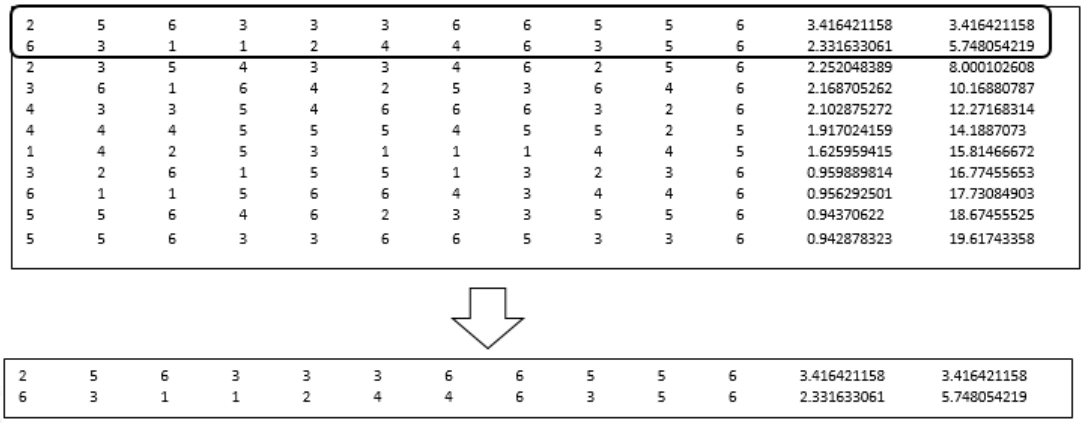
Figura 31 - Elementos a ordenar.

6. Mecanismos para generar las siguientes poblaciones.

Dado que el algoritmo está basado en la teoría de la evolución de Darwin, a excepción de la primera generación las generaciones futuras son consideradas para emular la aplicación de los mecanismos básicos de supervivencia que son selección, mutación y cruce, mismos que son explicados a detalle en el capítulo de marco teórico, lo que aquí se enuncia es la forma como se aplicó.

Como ya se hizo mención a partir de la segunda generación se aplican una serie de mecanismos de forma iterativa en cada generación como parte del algoritmo de trabajo con el fin de conservar las mejores soluciones y que estás a su vez vayan mejorando a través de generaciones futuras. Los siguientes puntos comprenden la forma y procedimiento en cómo se crean las poblaciones futuras.

1. **Elitismo**. "*En el modelo de selección elitista a que el mejor individuo de la población en el tiempo t, sea seleccionado como padre*" (Moujahid, Inza, & Larrañaga, 2008). Tomando en consideración el hecho que la población ya fue ordenada de mayor a menor por su valor de adaptabilidad, se seleccionan directamente (elitismo) los dos mejores individuos para formar parte de la siguiente generación. La Figura 32 muestra un ejemplo general de ello.



2	5	6	3	3	3	6	6	5	5	6	3.416421158	3.416421158
6	3	1	1	2	4	4	6	3	5	6	2.331633061	5.748054219
2	3	5	4	3	3	4	6	2	5	6	2.252048389	8.000102608
3	6	1	6	4	2	5	3	6	4	6	2.168705262	10.16880787
4	3	3	5	4	6	6	6	3	2	6	2.102875272	12.27168314
4	4	4	5	5	5	4	5	5	2	5	1.917024159	14.1887073
1	4	2	5	3	1	1	1	4	4	5	1.625959415	15.81466672
3	2	6	1	5	5	1	3	2	3	6	0.959889814	16.77455653
6	1	1	5	6	6	4	3	4	4	6	0.956292501	17.73084903
5	5	6	4	6	2	3	3	5	5	6	0.94370622	18.67455525
5	5	6	3	3	6	6	5	3	3	6	0.942878323	19.61743358

2	5	6	3	3	3	6	6	5	5	6	3.416421158	3.416421158
6	3	1	1	2	4	4	6	3	5	6	2.331633061	5.748054219

Figura 32 - Ejemplo de Selección Elitista.

2. **Mutación.** "*La mutación se considera un operador básico, que proporciona un pequeño elemento de aleatoriedad en la vecindad (entorno) de los individuos de la población*" (Davis & others, 1991).

   Dentro de lo novedoso de la técnica y como ya se ha resaltado en secciones anteriores lo que se lleva a cabo es una mutación inteligente definida como MMI (Mecanismo de Mutación Inteligente) y explicada en la sección de "Mecanismos diseñados" de este mismo capítulo, lo cual permite tener una técnica más robusta y a su vez híbrida, dicho de otra manera en lugar de tener la tradicional mutación arbitraria (como ocurre en la naturaleza) la implementación del método de Scatter Search permite que sea una mutación conducida, en donde se busca que el gen seleccionado se reubique con el clúster que más se parezca a fin de dar un empuje de mejora a la solución.

   Dentro de la sección "Establecer parámetros iniciales" de este mismo capítulo el usuario define de forma manual los valores que serán considerados para la técnica (consultar sección en caso necesario).

   Primero hay que establecer la cantidad de cromosomas de la generación anterior que se les va a aplicar mutación, otro elemento es el número de genes a mutar, que para el caso serán las etiquetas asignadas a cada clúster que pueden mutar (cambiar). Dicho esto, en seguida se enuncia la forma en cómo se aplica el MMI para completar la población.

- Tomando en consideración la generación antecesora se selecciona un cromosoma utilizando el método de selección por ruleta y teniendo en cuenta que los elementos mejor calificados tienen más posibilidad de ser seleccionados (mayor porción en la ruleta).
- Una vez seleccionado el cromosoma, el siguiente paso es aleatoriamente establecer cuántos elementos van a mutar, tomando como límite el valor que el usuario definió en los parámetros de entrada.
- Considerando lo anterior, enseguida se determina cuáles serán los genes que se les va a aplicar el MMI de forma aleatoria con el objetivo de mejorar dichos genes.
- Aplicar MMI, como ya se dijo el objetivo es buscar que el gen seleccionado se reubique en el clúster que más se parezca.
- Continuar aplicando MMI hasta completar el número de mutaciones establecido para el cromosoma en cuestión.
- El nuevo cromosoma mutado se incluye en la población de la siguiente generación.
- Todos los pasos anteriores se repiten hasta que se complete el porcentaje de la población (número de individuos) definido por el usuario.

3. **Generar el resto de forma aleatoria.** La última parte para completar el resto de la población se lleva a cabo aleatoriamente, esto es, considerando los dos individuos que se heredan de forma directa por elitismo + número de individuos que sufrieron mutación y pasaron a la siguiente generación + resto aleatoriamente para completar el 100% de la población.

7. Establecer Criterio de Paro.

Para esta implementación se optó por número máximo de generaciones, esto es: el criterio de paro está determinado por el número de generaciones establecido antes de la ejecución del algoritmo, ósea el algoritmo continua mientras $P_n < N$, donde P es la población actual en su n-ésima generación y N el número máximo de generaciones a crear.

Una vez que se cumple el criterio de paro el AGH-CHIP es capaz de presentar los resultados.

3. Experimentación.

En esta parte, se hace énfasis en las condiciones y la forma en la que se llevó a cabo la experimentación del AGH-CHIP.

1. Condiciones Generales.

En la sección de “Antecedentes y Caso de Estudio” del Capítulo – Metodología quedó establecido que el insumo principal es una matriz de semejanzas proteómicas de hongos de 33 x 33 elementos que corresponde a una lista de 33 hongos y que el algoritmo fue probado con los todos elementos sin excluir ninguno ya que el interés es crear clústeres con todos ellos.

Debido a la complejidad y tamaño de los valores de la matriz de semejanzas utilizada, no se incluye en esta sección pero puede ser consultada en el ANEXO A.

2. Diseño de Experimentos.

Con el fin de darle mayor flexibilidad a la investigación, tanto el AGH-CHIP como el UMDA-CHIP fueron sometidos a un diseño factorial de experimentos (Montgomery, 1991).

El software de colección de datos para el experimento se diseñó de tal manera que fuese interactivo con el usuario, lo que permite en una sola ejecución llevar a cabo todas las combinaciones correspondientes de los niveles establecidos del experimento sin necesidad de estar ejecutando el programa cada vez que se desee modificar los parámetros principales.

Para automatizar dicha combinación primeramente se definieron los siguientes factores (Tabla 11) donde cada uno corresponde a un nivel en el experimento que se lleva cabo.

Tabla 11 - Factores considerados para el experimento.

Factor	Descripción
Generaciones	Cantidad de veces que varía el valor del tamaño de la generación.
Cromosomas	Número de veces que cambia el valor del tamaño de la población.
Grupos de Crear	Número de veces que se puede determinar el máximo (no el real) de clústeres a crear en cada ejecución del experimento.
Bitas a mutar	Número de veces que se puede determinar el máximo (no el real) de bits a aplicar MMI en cada ejecución del experimento.

Se destaca la flexibilidad y aportación del diseño factorial ya que permite variar dinámicamente el número de veces que se modifica cada factor y con ello ir creando la profundidad de los niveles.

El diseño factorial consta de los siguientes niveles para el AGH-CHIP (Tabla 12)

Tabla 12 - Diseño factorial sobre el AGH-CHIP.

Factor	Niveles	Valores Probados
Generaciones	3	50, 100, 150
Cromosomas	3	500, 750, 1000
Grupos de Crear	2	5, 10
Bitas a mutar	2	2, 5

Para cada combinación fueron consideradas 10 réplicas por lo tanto, 3x3x2x2x10 = 360 veces fueron las que se ejecutó el algoritmo, variando dinámicamente los parámetros que se enuncian.

En cada réplica, para sus correspondientes poblaciones creadas, se crea de forma automática conforme a los parámetros, un archivo de texto plano con el siguiente formato de ejemplo para su nombramiento "*V2-50gen-500crom-5gr-2nbm-case4.txt*" de tal manera que permita más fácilmente consultar los resultados de un caso en particular. Lo anterior se describe a detalle en la Tabla 13.

Tabla 13 - Elementos que conforman el archivo de salida de cada replica.

Elemento	Descripción
V2	Se refiere a la versión del AGH-CHIP que en este caso es la 2.
50gen	Se refiere al valor del tamaño de generaciones con las que se está ejecutando la réplica, para este caso 50.
500crom	Valor del tamaño de las poblaciones, para este caso 500.
5gr	Máximo de grupos que se pueden llegar a crear en la réplica. 5 para este caso.
2nbm	Máximo de bits a los que se les puede aplicar el MMI. 2 en este caso.
Case4	Se refiere a la réplica ejecutada. En este caso es la 4ta replica que se ejecuta con los parámetros anteriores.

El proceso se ejecutó en una laptop HP ® modelo 2000 con sistema operativo Windows ® 8, procesador Intel Core i3 ®, 4 GB de memoria RAM.

En cuanto a los principales parámetros de salida (computados) se describen en la Tabla 14. Cabe destacar que para concentrar y resumir el total de ejecuciones el AGH-CHIP crea dos archivos en texto plano (además de los mencionados para cada ejecución), que además de contener las condiciones del experimento, contiene los valores de los parámetros de salida. Puede consultar el ANEXO B para ver los resultados generados por el algoritmo a fin de comprender mejor la tabla.

Tabla 14 - Descripción de los parámetros de salida.

Nombre del archivo de Salida	Principales Variables de Salida
Statistics.txt Presenta en forma de secciones los mejores valores encontrados de cada réplica del experimento bajo las condiciones definidas por los parámetros de entrada.	*Max_Fitness*: valor de adaptabilidad máximo encontrado en la réplica. *Time (ms)*: Tiempo que tardó en ejecutarse dicha replica expresado en milisegundos.
Resume.txt Resume las instancias del archivo anterior de tal manera que se muestran promedios de todos los casos de cada variación de parámetros en el experimento.	*#Max_Groups_Created:* Máximo número de clústeres creados al que le corresponde el valor de adaptabilidad señalado. *Max_Fitness_Found*: Máximo valor de adaptabilidad que se encontró durante todos los casos de una combinación de parámetros en particular. *Ocurr_Max_Fitness_Found*: De lo anterior, esta variable representa la cantidad de veces que se encontró dicho máximo expresando en porcentaje. *Avg_Max_Fitness*: Promedio de máximos valores de adaptabilidad. *Avg_Time (ms)*: Promedio de los tiempos de ejecución.

3. Experimentos.

En la (Figura 33) puede apreciarse una captura de pantalla del algoritmo en ejecución puesto a punto con un diseño factorial, en el cual se pude ver que de forma automática se van creando los archivos mencionados en la sección anterior, también el tiempo en milisegundos de cada instancia por lo que el usuario solo tiene que esperar a que se complete el experimento.

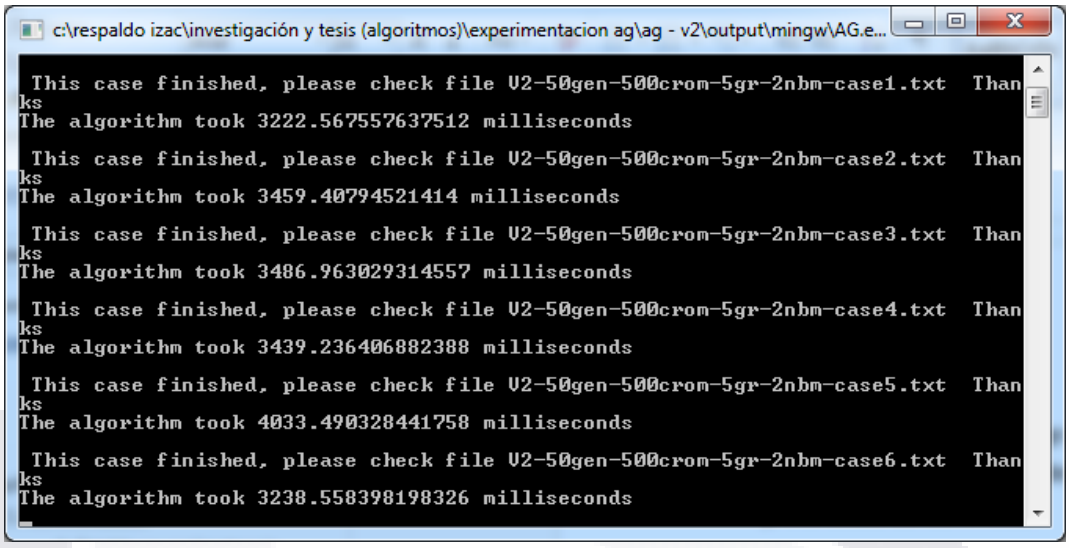


Figura 33 - Captura de pantalla del AGH-CHIP en ejecución.

Una vez fueron ejecutados todos los 360 casos mencionados del experimento, se tomaron los resultados del archivo “Resume.txt” para crear la siguiente tabla de concentración (Tabla 15) con el fin de visualizar mejor dichos resultados.

Tabla 15 – Resumen de Resultados del Experimento AGH-CHIP.

#Cases	#Generations	#population	#Max_Groups_Created	#Max_Bits_Mutated	Max_Fitness_Found	Ocurr_Max_Fitness_Found	Avg_Max_Fitness	Avg_Time (ms)
10	50	500	2	2	3.963737152088470	30.00%	3.896135174844610	3358.2012182856
10	50	500	2	5	3.963737152088470	20.00%	3.893674006911820	3108.1659949045
10	50	500	2	2	3.941706315118290	10.00%	3.851496313793590	3108.8893058038
10	50	500	2	5	3.945363551206310	10.00%	3.851189130970570	3852.3649749865
10	50	750	2	2	3.948337826704610	10.00%	3.860852857723570	6419.5569407714
10	50	750	2	5	3.941706315118290	20.00%	3.871964899333020	6059.5416362533
10	50	750	2	2	3.920799356075520	10.00%	3.843791654699190	6173.2373735674
10	50	750	2	5	3.945363551206310	10.00%	3.853946337439860	6110.2791104003
10	50	1000	2	2	3.963737152088470	10.00%	3.873619882445820	10328.0702431399
10	50	1000	2	5	3.963737152088470	10.00%	3.907376847302590	10299.1345639896
10	50	1000	2	2	3.948337826704610	10.00%	3.855731012484240	10549.7426394218
10	50	1000	2	5	3.941706315118290	20.00%	3.871548731980830	10457.6771781227
10	100	500	2	2	3.963737152088470	30.00%	3.894611647919870	5949.3687707368
10	100	500	2	5	3.948337826704610	10.00%	3.847660742954150	5955.2361731808

10	100	500	2	2	3.963737152088470	20.00%	3.886477453554780	6061.8680625745
10	100	500	2	5	3.963737152088470	10.00%	3.910292743720380	5933.0042259847
10	100	750	2	2	3.963737152088470	10.00%	3.879646780314050	12147.8110187603
10	100	750	2	5	3.963737152088470	20.00%	3.935183239850680	12052.8761245413
10	100	750	2	2	3.963737152088470	10.00%	3.842877995218960	12256.0426572387
10	100	750	2	5	3.924859427723470	20.00%	3.866713871380230	12263.4471210018
10	100	1000	2	2	3.943159625953310	10.00%	3.914344418869040	20537.6178320625
10	100	1000	2	5	3.963737152088470	20.00%	3.936210400473220	20268.1805194176
10	100	1000	2	2	3.963737152088470	20.00%	3.906837905653120	20663.3959280458
10	100	1000	2	5	3.963737152088470	10.00%	3.888830665375070	20637.3706526014
10	150	500	2	2	3.963737152088470	10.00%	3.902308438350950	9015.9729412180
10	150	500	2	5	3.963737152088470	10.00%	3.935200997085230	8891.3330777107
10	150	500	2	2	3.963737152088470	20.00%	3.880911869071310	8965.8404224835
10	150	500	2	5	3.963737152088470	10.00%	3.818535239472000	8911.9062828167
10	150	750	2	2	3.963737152088470	10.00%	3.920292671816700	18179.8569470935
10	150	750	2	5	3.963737152088470	20.00%	3.935115987669680	18168.7777568864
10	150	750	2	2	3.963737152088470	20.00%	3.889997249585320	18442.0503127820
10	150	750	2	5	3.963737152088470	20.00%	3.903956000151760	18242.2119790696
10	150	1000	2	2	3.943159625953310	10.00%	3.892393846405700	31014.2165753513
10	150	1000	2	5	3.963737152088470	20.00%	3.920288131147720	30680.4421150121
10	150	1000	2	2	3.963737152088470	10.00%	3.932009885830640	31331.2736823869
10	150	1000	2	5	3.948337826704610	10.00%	3.893081829671880	30947.6988212073

Como se puede observar la tabla anterior contiene además de la fila de encabezados 36 filas, donde cada fila concentra un resumen de las 10 réplicas de cada combinación en particular del experimento (36x10=360). Las columnas sin resaltar representan los parámetros de entrada mientras que las columnas que están resaltadas corresponden a parámetros de salida que se explicó su significado en la Tabla 14.

Al llevar a cabo un análisis de los resultados, respecto al valor de adaptabilidad se puede observar que el mayor valor encontrado es **3.96373715208847** el cual aparece en 23 de las 36 filas lo que significa que en esas 23 combinaciones de parámetros aparece por lo menos una vez este máximo valor.

Con respecto al tiempo este es relativo y como se puede notar el número de generaciones influye de manera exponencial en el tiempo de ejecución del algoritmo sin embargo se puede decir de forma empírica que la mejor combinación de parámetros que satisfacen el máximo que se ha hecho mención es la siguiente: Número de Generaciones: 50, Tamaño de la Población: 500, Máximo Número de Bits a Mutar: 5, Máximo Número de Grupos: 5.

Otro aspecto importante que se pude observar es en la columna “#Max_Groups_Created” que representa el máximo de clústeres que se crearon para su correspondiente valor de adaptabilidad, esto es, que a pesar que se probó con la posibilidad de crear hasta 5 clústeres el AGH-CHIP tiene una tendencia a discriminar y agrupar los hongos en **2 grupos**.

4. Pruebas estadísticas.

Con el fin de comprobar si el algoritmo es sensible a la variabilidad de los parámetros en términos de las respuestas (valor de adaptabilidad y tiempo de ejecución) y establecer soporte estadístico a los resultados y conclusiones, se llevaron a cabo una serie de pruebas con los valores del archivo “Statistics.txt”.

Primeramente se determinó si los valores de adaptabilidad y tiempo seguían una distribución normal, para ello se aplicaron las pruebas de Kolmogorov-Smirnov y Shapiro-Wilk, además cuando fue necesario se ejecutó una prueba de homocedasticidad empleando el estadístico de Levene. En seguida se muestran las gráficas de histograma resultantes, para normalidad del valor de adaptabilidad (Figura 34) y normalidad del tiempo (Figura 35) respectivamente.

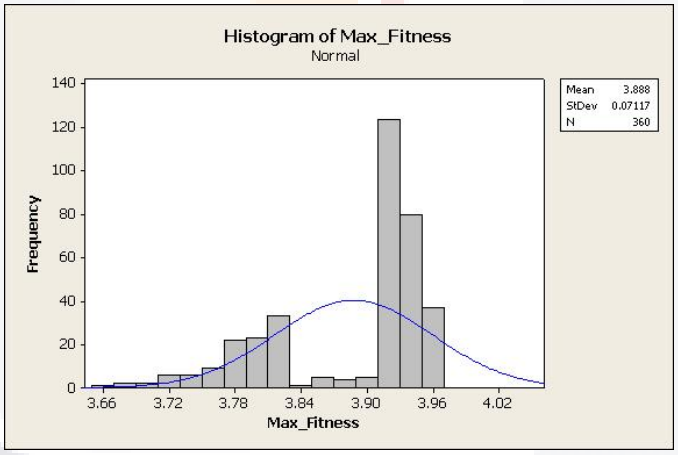


Figura 34 - Prueba de Normalidad del Valor de Adaptabilidad.

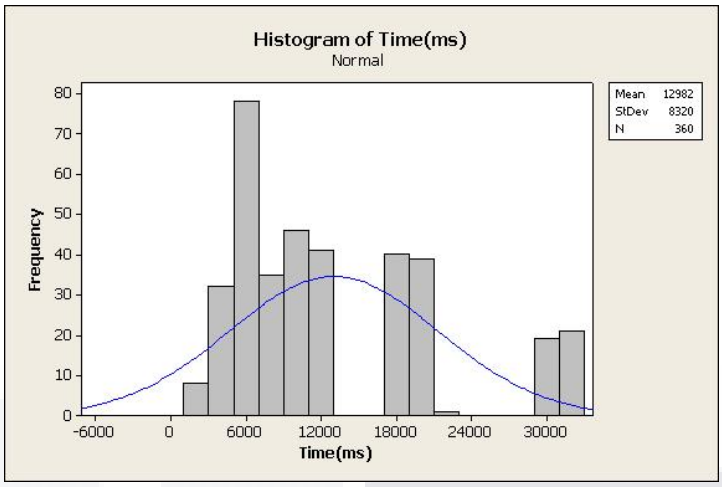


Figura 35 - Prueba de Normalidad del Tiempo.

Tal como se puede observar de forma gráfica los datos no siguen una distribución normal. En la Tabla 16 se presentan los resultados de la prueba de homogeneidad de varianzas.

Tabla 16 - Prueba de homogeneidad de varianzas.

	Estadístico de Levene	gl1	gl2	Sig.
Fitness	3.273	35	324	.000
Tiempo	5.540	35	324	.000

De acuerdo a la prueba de Levene las varianzas de los grupos para las variables de interés (adaptabilidad y tiempo) se dice que son estadísticamente diferente pues el valor de p-value (Sig.) para ambos casos es menor a 0.05.

Como no se cumplen los supuestos de la estadística paramétrica, por lo que se procedió a usar estadística no paramétrica y se aplicó la prueba de Kruskal-Wallis con un 95% de confianza para la comparación de las 360 réplicas. Para llevar a cabo dichas pruebas se utilizó el paquete estadístico SPSS. La tabla (Tabla 17) muestra el resumen de dicha prueba.

Tabla 17 - Resultados de la prueba de Kruskal-Wallis.

Estadísticos de contraste[a,b]

	Fitness	Tiempo
Chi-cuadrado	65.271	355.311
gl	35	35
Sig. asintót.	.001	.000

a. Prueba de Kruskal-Wallis
b. Variable de agrupación: Grupo

Al observar el resultado de la prueba, tanto el valor de p = 0.001 para adaptabilidad y p = 000 para el tiempo, están por debajo de 0.05, por lo tanto se concluye que estadísticamente hay evidencia suficiente para afirmar que existe diferencia significativa entre los grupos de ejecuciones del modelo, por lo que el algoritmo sí es sensible al cambio de parámetros.

Dicho lo anterior se dice que la mejor combinación de parámetros con el mejor valor de adaptabilidad es la que se hace referencia en el punto anterior: Número de Generaciones: 50, Tamaño de la Población: 500, Máximo Número de Bits a Mutar: 5, Máximo Número de Grupos: 5, con un tiempo promedio de las 10 réplicas de 3108.1659949045 milisegundos

Luego se aplicó la prueba de U de Mann Whitney los grupos 22 y 25 concluyendo que no hay diferencia en cuanto al valor de adaptabilidad pero sí respecto al tiempo, luego se aplicó para los grupos 18 y 30 que tampoco difieren en adaptabilidad por lo tanto la mejor combinación de parámetros en cuanto a tiempo es: Número de Generaciones: 150, Tamaño de la Población: 500, Máximo Número de Bits a Mutar: 5, Máximo Número de Grupos: 5, con un tiempo promedio de las 10 réplicas de 8891.3330777107 milisegundos.

En la siguiente sección se le da un sentido, interpretación y soporte científico a los resultados que se obtuvieron.

Si se desea consultar las tablas y figuras con los resultados de las pruebas estadísticas llevabas a cabo, se pueden consultar dichos resultados en el ANEXO C.

5. Interpretación de Resultados.

Retomando el mejor valor de adaptabilidad que se obtuvo hasta el momento de esta investigación que corresponde a **3.96373715208847**, al llevar a cabo una búsqueda sobre el conjunto de archivos resultantes para ver los clústeres que formó el algoritmo, se tiene la siguiente imagen (Figura 36) que corresponde a una búsqueda parcial en el conjunto de archivos.

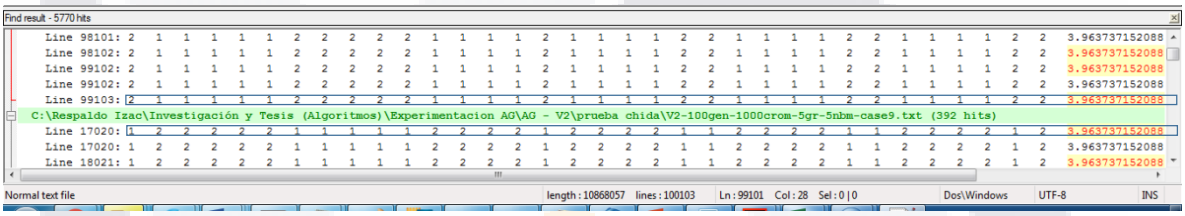


Figura 36 - Ejemplo de archivos donde se encontró el mejor valor de adaptabilidad.

Como se dijo y como se puede ver, la tendencia del algoritmo es crear dos grupos, de igual forma en la figura anterior se señala con un recuadro los dos casos en la cadena resultante donde aparece el valor de adaptabilidad, que básicamente es su homologo como se muestra en seguida:

2 1 1 1 1 1 2 2 2 2 2 1 1 1 1 2 1 1 1 1 2 2 1 1 1 1 2 2 1 1 1 1 2
1 2 2 2 2 2 1 1 1 1 1 2 2 2 2 1 2 2 2 2 1 1 2 2 2 2 1 1 2 2 2 2 1

Tomando de izquierda a derecha el orden de la cadena resultante, y teniendo en consideración que el número asignado al clúster es meramente una etiqueta (antes de clasificar) además al consultar la lista de hongos (Tabla 5) de la que se hace mención en el punto “Antecedentes y Caso de Estudio” del Capítulo “Metodología” se creó la Tabla 18 que corresponde a la clasificación construida por el AGH-CHIP.

Tabla 18 - Clasificación propuesta por el AGH-CHIP.

Grupo	Hongos
1	1 Ashbya gossypii 7 Candida albicans 8 Candida glabrata 9 Candida guilliermondii 10 Candida lusitaniae 11 Candida tropicalis 16 Debaryomyces hansenii 21 Kluyveromyces lactis 22 Loderomyces elongisporus 27 Saccharomyces cerevisiae 28 Sacharomices japonicus 33 Yarrowia lipolytica
2	2 Aspergillus fumigatus 3 Aspergillus nidulans 4 Aspergillus terreus 5 Batrachochytrium dendrobatidis 6 Botrytis cinerea 12 Chaetomium globosum 13 Coprinus cinereus 14 Coccidiodes immitis 15 Cryptococcus neoformans serotype A 17 Fusarium graminearum 18 Fusarium oxysporum 19 Fusarium verticilloides 20 Histoplasma capsulatum 23 Magnaporthe grisea 24 Neurospora crassa 25 Puccinia graminis 26 Rhizopus oryzae 29 Sclerotinia sclerotiorum 30 Stagonospora nodorumm 31 Uncinocarpus reesii 32 Ustilago maydis

Antes de dar interpretación a la tabla anterior, se hizo una minuciosa revisión de la literatura relacionada con clasificación de hongos, a fin de observar si los clústeres encontrados empatan con alguna de las clasificaciones ya existente; también se consultó al Dr. Onésimo Moreno Rico[16] (11 de Junio de 2014) del departamento de microbiología, del Centro de Ciencias Básicas de la Universidad Autónoma de Aguascalientes experto del área para que

[16] http://www.uaa.mx/investigacion/investigacion/desarrollo/academias/curriculums/basicas/Onesimo%20Moreno%20Rico.pdf

diera su opinión científica al respecto con el fin de fundamentar si la clasificación tiene un soporte válido.

El doctor Moreno Rico concluye que el algoritmo tiende a realizar una **clasificación correcta al 100%** en dos grupos los cuales corresponden a **hongos levaduras** y **hongos mohos**, dicha opinión se soporta en la literatura por lo que se crearon una serie de tablas con cada una de las principales características de los hongos en cuestión según su clasificación, dichas tablas se consideran un **aporte adicional a esta investigación** y puede ser consultada en el siguiente punto.

Los hongos se pueden clasificar en dos formas morfológicas básicas: levaduras y mohos. Según la enciclopedia en línea Wikipedia proporciona las siguientes definiciones:

- Levadura: "*Se denomina levadura o fermento a cualquiera de los diversos organismos eucariotas, clasificados como hongos microscópicos unicelulares, que son importantes por su capacidad para realizar la descomposición mediante fermentación de diversos cuerpos orgánicos, principalmente los azúcares o hidratos de carbono, produciendo distintas sustancias*" ("Levadura", 2016).
- Mohos: "*El moho es un hongo que se encuentra tanto al aire libre como en lugares húmedos y con baja luminosidad. Existen muchas especies de mohos que son especies microscópicas del reino fungi, que crecen en formas de filamentos pluricelulares o unicelulares. El moho crece mejor en condiciones cálidas y húmedas; se reproducen y propagan mediante esporas. Las esporas del moho pueden sobrevivir en variadas condiciones ambientales, incluso en extrema sequedad, si bien ésta no favorece su crecimiento normal*" ("Moho", 2016).

6. Aportes.

Las tablas (Tabla 19) que en seguida se presentan fueron creadas a partir de la información de la base de datos en línea de taxonomía y clasificación del Centro Nacional para la Información Biotecnológica (NCBI, por sus siglas en inglés) que puede ser consultada en http://www.ncbi.nlm.nih.gov/taxonomy; del proyecto "A database for eukaryotic genome and EST sequencing projects" que puede ser consultado en http://www.diark.org/diark; así como de otros autores a los que se les da su respectiva cita, en muchos casos se respeta el texto original de la fuente consultada.

Tabla 19 - Conjunto de tablas con información biológica básica de los hongos de estudio.

Grupo	1
Hongo	1 Ashbya gossypii
Filogenia	cellular organisms » Eukaryota » Opisthokonta » Fungi » Dikarya » Ascomycota » saccharomyceta » Saccharomycotina » Saccharomycetes » Saccharomycetales » Saccharomycetaceae » Eremothecium » Eremothecium gossypii
Descripción	Eremothecium gossypii is a pathogen that attacks cotton and some citrus fruits but also produces vitamin B-2.
Notas	Levadura (Carrillo et al., 2007), Pathogenic in cotton.
Correctamente clasificado	Sí

Grupo	1
Hongo	7 Candida albicans
Filogenia	cellular organisms » Eukaryota » Opisthokonta » Fungi » Dikarya » Ascomycota » saccharomyceta » Saccharomycotina » Saccharomycetes » Saccharomycetales » mitosporic Saccharomycetales » Candida » Candida albicans
Descripción	Candida albicans is one of the most commonly encountered human pathogens, causing a wide variety of infections ranging from mucosal infections in generally healthy persons to life-threatening systemic infections in individuals with impaired immunity. Candida albicans differs in various respects from other genome sequences in that both copies of the genome are explicitly represented. Similar to many other Candida species, a CUG codon in Candida albicans corresponds to a serine residue instead of the universal leucine.
Notas	Levadura (Carrillo et al., 2007), Pathogenic in humans.
Correctamente clasificado	Sí

Grupo	1
Hongo	8 Candida glabrata
Filogenia	Cellular organisms » Eukaryota » Opisthokonta » Fungi » Dikarya » Ascomycota » saccharomyceta » Saccharomycotina » Saccharomycetes » Saccharomycetales » Saccharomycetaceae » Nakaseomyces » mitosporic Nakaseomyces » Candida glabrata.
Descripción	Candida glabrata is a pathogenic hemiascomycete yeast that is the second most frequent causative agent of human candidasis, after Candida albicans. Despite its name, it is phylogenetically more closely related to Saccharomyces cerevisiae than to other Candida species. Candida glabrata is haploid and has no known sexual cycle.
Notas	Levadura (Carrillo et al., 2007), Pathogenic in humans.
Correctamente clasificado	Sí

Grupo	1
Hongo	9 Candida guilliermondii
Filogenia	Cellular organisms » Eukaryota » Opisthokonta » Fungi » Dikarya » Ascomycota » saccharomyceta » Saccharomycotina » Saccharomycetes » Saccharomycetales » Debaryomycetaceae » Meyerozyma » Meyerozyma guilliermondii.
Descripción	Pichia guilliermondii (anamorph Candida guilliermondii) is a nonpathogenic yeast that is closely related to the pathogenic species Candida albicans and Candida tropicalis. Unlike Candida albicans and Candida tropicalis, Pichia guilliermondii is haploid.
Notas	Levadura (Carrillo et al., 2007).
Correctamente clasificado	Sí

Grupo	1
Hongo	10 Candida lusitaniae
Filogenia	Cellular organisms » Eukaryota » Opisthokonta » Fungi » Dikarya » Ascomycota » saccharomyceta » Saccharomycotina » Saccharomycetes » Saccharomycetales » Metschnikowiaceae » Clavispora » Clavispora lusitaniae.
Descripción	The pathogenic yeast Clavispora lusitaniae, also known as Candida lusitaniae, causes approximately 1 percent of invasive candidiasis cases and has emerged as a causative agent of candidemia.
Notas	Levadura (Carrillo et al., 2007), Pathogenic in humans.
Correctamente clasificado	Sí

Grupo	1
Hongo	11 Candida tropicalis
Filogenia	Cellular organisms » Eukaryota » Opisthokonta » Fungi » Dikarya » Ascomycota » saccharomyceta » Saccharomycotina » Saccharomycetes » Saccharomycetales » mitosporic Saccharomycetales » Candida » Candida tropicalis.
Descripción	The yeast Candida tropicalis is the second most pathogenic Candida species after Candida albicans and is more often associated with deep fungal infections than normal mucosa. Candida tropicalis is an asexual diploid organism. Similar to many other Candida species, a CUG codon in Candida tropicalis corresponds to a serine residue instead of the universal leucine.
Notas	Levadura (Carrillo et al., 2007), Pathogenic in humans.
Correctamente clasificado	Sí

Grupo	1
Hongo	16 Debaryomyces hansenii
Filogenia	Cellular organisms » Eukaryota » Opisthokonta » Fungi » Dikarya » Ascomycota » saccharomyceta » Saccharomycotina » Saccharomycetes » Saccharomycetales » Debaryomycetaceae » Debaryomyces » Debaryomyces hansenii » Debaryomyces hansenii var. Hansenii.
Descripción	Anamorph: Candida famata var. Flareri Debaryomyces hansenii is a salt-tolerant marine yeast that is often found on fish and cheese. It is generally considered nonpathogenic, but is closely related to the pathogenic Candida albicans.
Notas	Levadura (Carrillo et al., 2007), Pathogenic in humans.
Correctamente clasificado	Sí

Grupo	1
Hongo	21 Kluyveromyces lactis
Filogenia	Cellular organisms » Eukaryota » Opisthokonta » Fungi » Dikarya » Ascomycota » saccharomyceta » Saccharomycotina » Saccharomycetes » Saccharomycetales » Saccharomycetaceae » Kluyveromyces » Kluyveromyces lactis.
Descripción	Anamorph: Candida sphaerica. Kluyveromyces lactis is a petite-negative hemiascomycete yeast. Compared to Saccharomyces cerevisiae, it can use a wider variety of carbon sources, and many of its strains were originally isolated from milk-derived products in which the major carbon source is lactose.
Notas	Levadura (Carrillo et al., 2007), Pathogenic in humans.
Correctamente clasificado	Sí

Grupo	1
Hongo	22 Loderomyces elongisporus
Filogenia	Cellular organisms » Eukaryota » Opisthokonta » Fungi » Dikarya » Ascomycota » saccharomyceta » Saccharomycotina » Saccharomycetes » Saccharomycetales » Debaryomycetaceae » Lodderomyces » Lodderomyces elongisporus.
Descripción	The yeast Lodderomyces elongisporus, a member of the Candida clade, is the closest sexual relative of Candida albicans. It is the only species in the Candida clade known to form ascospores.
Notas	Levadura (Carrillo et al., 2007).
Correctamente clasificado	Sí

Grupo	1
Hongo	27 Saccharomyces cerevisiae
Filogenia	Cellular organisms » Eukaryota » Opisthokonta » Fungi » Dikarya » Ascomycota » saccharomyceta » Saccharomycotina » Saccharomycetes » Saccharomycetales » Saccharomycetaceae » Saccharomyces » Saccharomyces cerevisiae.
Descripción	Anamorph: Candida robusta The budding yeast Saccharomyces cerevisiae is one of the major model organisms for understanding cellular and molecular processes in eukaryotes.
Notas	Levadura (Carrillo et al., 2007).
Correctamente clasificado	Sí

Grupo	1
Hongo	28 Sacharomices japonicus
Filogenia	Cellular organisms » Eukaryota » Opisthokonta » Fungi » Dikarya » Ascomycota » Taphrinomycotina » Schizosaccharomycetes » Schizosaccharomycetales » Schizosaccharomycetaceae » Schizosaccharomyces » Schizosaccharomyces japonicus.
Descripción	Schizosaccharomyces japonicus is a dimorphic fission yeast, capable of forming true mycelia. It is highly invasive, making it a potential model for invasive fungal growth.
Notas	Levadura (Carrillo et al., 2007).
Correctamente clasificado	Sí

Grupo	1
Hongo	33 Yarrowia lipolytica
Filogenia	Cellular organisms » Eukaryota » Opisthokonta » Fungi » Dikarya » Ascomycota » saccharomyceta » Saccharomycotina » Saccharomycetes » Saccharomycetales » Dipodascaceae » Yarrowia » Yarrowia lipolytica.
Descripción	Anamorph: Candida lipolytica Yarrowia lipolytica is a nonpathogenic yeast that can use hydrocarbons and various fats as carbon sources. This genome project was originally submitted in error as Yarrowia lipolytica CLIB99. The correct strain for this project is CLIB122.
Notas	Levadura (Carrillo et al., 2007).
Correctamente clasificado	Sí

Grupo	2
Hongo	2 Aspergillus fumigatus
Filogenia	Cellular organisms » Eukaryota » Opisthokonta » Fungi » Dikarya » Ascomycota » saccharomyceta » Pezizomycotina » leotiomyceta » Eurotiomycetes » Eurotiomycetidae » Eurotiales » Aspergillaceae » Aspergillus » Aspergillus fumigatus.
Descripción	Filamentous fungi, growing on the nonliving organic materials in the soil and causing more infections worldwide than any other mold.
Notas	Parasitic fungi in grains such corn and wheat.
Correctamente clasificado	Sí (Castañón Olivares, García Yáñez, & Gutiérrez Quiroz, 2013).

Grupo	2
Hongo	3 Aspergillus nidulans
Filogenia	Cellular organisms » Eukaryota » Opisthokonta » Fungi » Dikarya » Ascomycota » saccharomyceta » Pezizomycotina » leotiomyceta » Eurotiomycetes » Eurotiomycetidae » Eurotiales » Aspergillaceae » Aspergillus » Aspergillus nidulans.
Descripción	Anamorph: Aspergillus nidulans FGSC A4 Emericella nidulans is a filamentous Ascomycete that is normally haploid but can be induced to grow as a heterokaryon or a diploid. It produces both sexual and asexual spores. In contrast, most other Aspergillus fungi are asexual.
Notas	Parasitic fungi en grains and seeds.
Correctamente clasificado	Sí (Castañón Olivares et al., 2013).

Grupo	2
Hongo	4 Aspergillus terreus.
Filogenia	Cellular organisms » Eukaryota » Opisthokonta » Fungi » Dikarya » Ascomycota » saccharomyceta » Pezizomycotina » leotiomyceta » Eurotiomycetes » Eurotiomycetidae » Eurotiales » Aspergillaceae » Aspergillus » Aspergillus terreus.
Descripción	Aspergillus terreus is a filamentous fungus that produces clinically relevant secondary metabolites -- statins. Statins are cholesterol-lowering drugs that have been widely used in the past decade for the prevention of heart disease. Of the five statins currently prescribed by physicians, three (pravastatin, simvastatin, and lovastatin) are derived by fermentation by Aspergillus terreus. In addition, this fungus can produce the toxins patulin and citrinin, which may cause toxicoses in humans and other animals and is associated with aspergillosis of the lungs and/or disseminated aspergillosis.
Notas	Parasitic fungus in humans and toxic.
Correctamente clasificado	Sí (Castañón Olivares et al., 2013).

Grupo	2
Hongo	5 Batrachochytrium dendrobatidis
Filogenia	Cellular organisms » Eukaryota » Opisthokonta » Fungi » Chytridiomycota » Chytridiomycetes » Rhizophydiales » Rhizophydiales incertae sedis » Batrachochytrium » Batrachochytrium dendrobatidis.
Descripción	Batrachochytrium dendrobatidis is a non-filamentous, aquatic chytrid fungus. Chytrids are unique among true fungi in that they produce flagellated zoospores. No resting spore or sexual stages are known for Batrachochytrium dendrobatidis, which is cultured as a diploid. Batrachochytrium dendrobatidis, known as the amphibian chytrid, is an amphibian pathogen that causes chytridiomycosis; this disease is implicated as the primary cause of the recent declines of frog populations around the world. Batrachochytrium dendrobatidis infects the skin of frogs, causing thickening of the keratinized layer. It is the only chytrid species known to parasitize vertebrates.
Notas	Parasitic fungus in amphibians like frogs and toads (“Batrachochytrium dendrobatidis”, 2015) .
Correctamente clasificado	Sí.

Grupo	2
Hongo	6 Botrytis cinérea.
Filogenia	Cellular organisms » Eukaryota » Opisthokonta » Fungi » Dikarya » Ascomycota » saccharomyceta » Pezizomycotina » leotiomyceta » sordariomyceta » Leotiomycetes » Helotiales » Sclerotiniaceae » Botrytis » Botrytis cinérea.
Descripción	The widespread fungus Botryotinia fuckeliana is a pathogen of most vegetable and fruit crops, and many varieties of trees, shrubs, flowers, and weeds. It causes gray mold rot, or Botrytis blight, which is responsible for significant crop and economic losses. Under certain environmental conditions, Botryotinia fuckeliana causes Noble rot of grapes, which is necessary for the production of some rare dessert wines.
Notas	Parasitic fungi in plants.
Correctamente clasificado	Sí.

Grupo	2
Hongo	12 Chaetomium globosum
Filogenia	Cellular organisms » Eukaryota » Opisthokonta » Fungi » Dikarya » Ascomycota » saccharomyceta » Pezizomycotina » leotiomyceta » sordariomyceta » Sordariomycetes » Sordariomycetidae » Sordariales » Chaetomiaceae » Chaetomium » Chaetomium globosum.
Descripción	Chaetomium globosum is a filamentous fungus that can infect the skin and nails of healthy people, and can cause a fatal systemic infection in immunocompromised patients. It produces mycotoxins in buildings that it has infested.
Notas	Parasitic fungus in humans.
Correctamente clasificado	Sí.

Grupo	2
Hongo	13 Coprinus cinereus
Filogenia	Cellular organisms » Eukaryota » Opisthokonta » Fungi » Dikarya » Basidiomycota » Agaricomycotina » Agaricomycetes » Agaricomycetidae » Agaricales » Psathyrellaceae » Coprinopsis » Coprinopsis cinérea.
Descripción	Coprinopsis cinerea is a multicellular basidiomycete that undergoes a complete sexual cycle, including typical mushroom formation.
Notas	Saprophyte, causes no problems.
Correctamente clasificado	Sí.

Grupo	2
Hongo	14 Coccidiodes immitis
Filogenia	Cellular organisms » Eukaryota » Opisthokonta » Fungi » Dikarya » Ascomycota » saccharomyceta » Pezizomycotina » leotiomyceta » Eurotiomycetes » Eurotiomycetidae » Onygenales » mitosporic Onygenales » Coccidioides » Coccidioides immitis.
Descripción	Coccidioides immitis is a dimorphic soil fungus that is morphologically identical to, but genetically distinct from, Coccidioides posadasii. Both fungi cause the disease coccidioidomycosis, also known as valley fever.
Notas	Parasitic fungus in humans.
Correctamente clasificado	Sí.

Grupo	2
Hongo	15 Cryptococcus neoformans serotype A
Filogenia	Cellular organisms » Eukaryota » Opisthokonta » Fungi » Dikarya » Basidiomycota » Agaricomycotina » Tremellomycetes » Tremellales » Tremellaceae » Filobasidiella » Filobasidiella/Cryptococcus neoformans species complex » Cryptococcus neoformans » Cryptococcus neoformans var. Grubii.
Descripción	Cryptococcus gattii, formerly known as Cryptococcus neoformans var gattii, is an encapsulated yeast found primarily in tropical and subtropical climates. Its teleomorph is Filobasidiella bacillispora, a filamentous fungus belonging to the class Tremellomycetes. Cryptococcus gattii causes the human diseases of pulmonary cryptococcosis (lung infection), basal meningitis, and cerebral cryptococcomas. Occasionally, the fungus is associated with skin, soft tissue, lymph node, bone, and joint infections. In recent years, it has appeared in British Columbia, Canada and the Pacific Northwest.
Notas	Parasitic fungus in humans.
Correctamente clasificado	Sí (Castañón Olivares et al., 2013).

Grupo	2
Hongo	17 Fusarium graminearum
Filogenia	Cellular organisms » Eukaryota » Opisthokonta » Fungi » Dikarya » Ascomycota » saccharomyceta » Pezizomycotina » leotiomyceta » sordariomyceta » Sordariomycetes » Hypocreomycetidae » Hypocreales » Nectriaceae » Fusarium » Fusarium sambucinum species complex » Fusarium graminearum.
Descripción	Anamorph: Fusarium graminearum Gibberella zeae is the cause agency of head blight (scab) of wheat and barley.
Notas	Parasitic fungus of barley and wheat.
Correctamente clasificado	Sí (Castañón Olivares et al., 2013).

Grupo	2
Hongo	18 Fusarium oxysporum
Filogenia	Cellular organisms » Eukaryota » Opisthokonta » Fungi » Dikarya » Ascomycota » saccharomyceta » Pezizomycotina » leotiomyceta » sordariomyceta » Sordariomycetes » Hypocreomycetidae » Hypocreales » Nectriaceae » Fusarium » Fusarium oxysporum species complex » Fusarium oxysporum.
Descripción	Plant pathogen.
Notas	Parasitic fungi in plants.
Correctamente clasificado	Sí (Castañón Olivares et al., 2013).

Grupo	2
Hongo	19 Fusarium verticilloides
Filogenia	Cellular organisms » Eukaryota » Opisthokonta » Fungi » Dikarya » Ascomycota » saccharomyceta » Pezizomycotina » leotiomyceta » sordariomyceta » Sordariomycetes » Hypocreomycetidae » Hypocreales » Nectriaceae » Fusarium » Fusarium fujikuroi species complex » Fusarium verticillioides.
Descripción	Gibberella moniliformis is primarily a pathogen of maize, but it can also cause disease in other crop species.
Notas	Parasitic fungi in corn.
Correctamente clasificado	Sí (Castañón Olivares et al., 2013).

Grupo	2
Hongo	20 Histoplasma capsulatum
Filogenia	Cellular organisms » Eukaryota » Opisthokonta » Fungi » Dikarya » Ascomycota » saccharomyceta » Pezizomycotina » leotiomyceta » Eurotiomycetes » Eurotiomycetidae » Onygenales » Ajellomycetaceae » Ajellomyces » Ajellomyces capsulatus.

Descripción	Ajellomyces capsulatus is the causative agent of histoplasmosis, an infection that chiefly affects the lungs but occasionally spreads to other organs. Ajellomyces capsulatus is a dimorphic fungus, existing as a saprophytic mycelial form in the soil and as a pathogenic yeast form in the lungs of its host.
Notas	Parasitic fungi in humans.
Correctamente clasificado	Sí

Grupo	**2**
Hongo	23 Magnaporthe grisea
Filogenia	Cellular organisms » Eukaryota » Opisthokonta » Fungi » Dikarya » Ascomycota » saccharomyceta » Pezizomycotina » leotiomyceta » sordariomyceta » Sordariomycetes » Sordariomycetidae » Magnaporthales » Magnaporthaceae » Magnaporthe » Magnaporthe grisea.
Descripción	Magnaporthe grisea, a haploid filamentous Ascomycete, is the causal agent of rice blast disease.
Notas	Parasitic fungi rice
Correctamente clasificado	Sí

Grupo	**2**
Hongo	24 Neurospora crassa
Filogenia	Cellular organisms » Eukaryota » Opisthokonta » Fungi » Dikarya » Ascomycota » saccharomyceta » Pezizomycotina » leotiomyceta » sordariomyceta » Sordariomycetes » Sordariomycetidae » Sordariales » Sordariaceae » Neurospora » Neurospora crassa.
Descripción	Neurospora crassa is the best-characterized of the filamentous fungi, a group of organisms critically important to agriculture, medicine, and the environment. This orange bread mold is an important model organism for genetic and biochemical studies.
Notas	No cause diseases.
Correctamente clasificado	Sí.

Grupo	**2**
Hongo	25 Puccinia graminis
Filogenia	Cellular organisms » Eukaryota » Opisthokonta » Fungi » Dikarya » Basidiomycota » Pucciniomycotina » Pucciniomycetes » Pucciniales » Pucciniaceae » Puccinia » Puccinia graminis.
Descripción	The stem, black or cereal rusts are caused by the fungus Puccinia graminis and are a significant disease affecting cereal crops. Crop species which are affected by the disease include bread wheat, durum wheat, barley and triticale.

Notas	Parasitic fungi in wheat.
Correctamente clasificado	Sí.

Grupo	**2**
Hongo	26 Rhizopus oryzae.
Filogenia	Cellular organisms » Eukaryota » Opisthokonta » Fungi » Fungi incertae sedis » Early diverging fungal lineages » Mucoromycotina » Mucorales » Mucorineae » Rhizopodaceae » Rhizopus » Rhizopus oryzae.
Descripción	Rhizopus arrhizus is a filamentous fungus that is the most common cause of mucormycosis, also referred to as zygomycosis. An opportunistic pathogen, Rhizopus oryzae causes disease primarily in immunocompromised people, such as those with diabetes mellitus, cancer, or AIDS. Rhizopus oryzae is found in soil, decaying fruit and vegetables, old bread, and animal dung.
Notas	Parasitic mushroom rice and human with low defenses.
Correctamente clasificado	Sí

Grupo	**2**
Hongo	29 Sclerotinia sclerotiorum
Filogenia	Cellular organisms » Eukaryota » Opisthokonta » Fungi » Dikarya » Ascomycota » saccharomyceta » Pezizomycotina » leotiomyceta » sordariomyceta » Leotiomycetes » Helotiales » Sclerotiniaceae » Sclerotinia » Sclerotinia sclerotiorum.
Descripción	Sclerotinia sclerotiorum has a wide host range, which can include crops such as broccoli, cabbage, cauliflower, carrots, celery, beans, tomato, peppers, potatoes, stocks, and sunflower.
Notas	Parasitic fungi in plants.
Correctamente clasificado	Sí.

Grupo	**2**
Hongo	30 Stagonospora nodorumm
Filogenia	Cellular organisms » Eukaryota » Opisthokonta » Fungi » Dikarya » Ascomycota » saccharomyceta » Pezizomycotina » leotiomyceta » dothideomyceta » Dothideomycetes » Pleosporomycetidae » Pleosporales » Pleosporineae » Phaeosphaeriaceae » Parastagonospora » Parastagonospora nodorum.
Descripción	The fungus Phaeosphaeria nodorum is a major pathogen of wheat, causing the economically important disease leaf and glume blotch.
Notas	Parasitic fungi in wheat.
Correctamente clasificado	Sí.

Grupo	2
Hongo	31 Uncinocarpus reesii
Filogenia	Cellular organisms » Eukaryota » Opisthokonta » Fungi » Dikarya » Ascomycota » saccharomyceta » Pezizomycotina » leotiomyceta » Eurotiomycetes » Eurotiomycetidae » Onygenales » Onygenaceae » Uncinocarpus » Uncinocarpus reesii.
Descripción	The non-pathogenic fungus Uncinocarpus reesii is the closest known relative of the pathogenic species Coccidioides immitis and Coccidioides posadasii, with an estimated evolutionary distance to Coccidioides immitis of 20-30 million years. Uncinocarpus reesii is an environmental saprophyte. It has an identified sexual stage.
Notas	Saprophyte, no sick humans or plants.
Correctamente clasificado	Sí.

Grupo	**2**
Hongo	32 Ustilago maydis
Filogenia	Cellular organisms » Eukaryota » Opisthokonta » Fungi » Dikarya » Basidiomycota » Ustilaginomycotina » Ustilaginomycetes » Ustilaginales » Ustilaginaceae » Ustilago » Ustilago maydis.
Descripción	Ustilago maydis is a basidiomycete fungus that usually exists as a filamentous mycelium in nature. Unlike typical basidiomycetes, it does not have a real fruiting body. Ustilago maydis is the causal agent of corn smut disease.
Notas	Parasitic fungi in corn.
Correctamente clasificado	Sí.

7. Conclusiones del AGH-CHIP.

Las técnicas de inteligencia artificial, representan un poderoso instrumento para resolver problemas de cualquier área del conocimiento humano.

El algoritmo diseñado y referenciado como AGH-CHIP, implementado y puesto a punto, fue capaz de agrupar hongos partiendo de una matriz de semejanzas entre sus proteomas.

Contrastando los resultados del algoritmo con la literatura, se pudo comprobar que la clasificación realizada por el algoritmo corresponde con la de los **hongos levaduras** y **hongos mohos**;

Además de esto, se logró construir una tabla comparativa de hongos en base a su información biológica.

Capítulo V: Metaheurística Evolutiva UMDA - CHIP.

Nota: antes de empezar a explicar este capítulo punto cabe hacer mención que el algoritmo en cuestión tiene algunos elementos en común con respecto al AGH-CHIP, por ejemplo: la forma de generar la población inicial, el método de ordenamiento, el criterio de paro, algunas condiciones del experimento debido a que se está trabajando con la misma matriz de semejanzas, etc. Sin embargo se explica en su correspondiente sección con un enfoque al algoritmo que será descrito en este capítulo.

Este capítulo describe los componentes básicos y forma de trabajo del UMDA-CHIP para crear clústeres; luego se explica la forma en cómo se puso a punto el algoritmo mediante la implementación de un experimento factorial además de definir las condiciones y parámetros del experimento, posteriormente se presentan los resultados obtenidos para con ellos aplicar un modelo estadístico que corrobora la interpretación empírica de los resultados.

1. Mecanismos Diseñados.

Función de Adaptabilidad.

Con la intención de mejorar el desempeño se utilizó un **enfoque multi-objetivo**, de manera que, dada una población de individuos (soluciones), para cada clúster creado dentro del mismo individuo, las funciones de optimización serán: (1) calcular la **semejanza** entre los elementos del mismo clúster, (2) calcular la **diferencia** con respecto a otros clústeres. Los valores obtenidos son acumulados y promediados para obtener un único valor para cada individuo. En otras palabras se busca maximizar la semejanza entre objetos que pertenecen al mismo clúster y por otro lado maximizar la diferencia con los objetos de otros clústeres.

Al igual que en el AGH-CHIP el UMDA-CHIP no permite que todos los objetos (hongos) pertenezcan a un mismo grupo, ósea que el algoritmo crea al menos 2 grupos y en el caso

de que se haya generado un clúster que tenga un solo hongo, por ejemplo G1 {H1} la solución es castigada asignándole el valor de la menor semejanza existente en la matriz debido a que no hay otros objetos con que comparar.

En seguida se define y explica la función multi-objetivo partiendo de lo general a lo particular.

La Ecuación 3 corresponde a la función multi-objetivo, la cual promedia la semejanza y diferencia entre clústeres generados.

$$Función\ de\ Adaptabilidad = 0.5 * Semejanza + 0.5 * Diferencia$$

Ecuación 3

Tal como puede verse, tanto la semejanza como la diferencia tienen un peso del 50% sobre la función de adaptabilidad. En seguida se explica cómo se lleva a cabo la obtención de cada valor.

Semejanza.

El cálculo de la semejanza se realiza exactamente igual que en el AGH-CHIP, favor de referirse a la **Ecuación 2** junto con el ejemplo que se presenta del mismo apartado.

Diferencia.

Para efectuar el cálculo de la diferencia, se calcula la sumatoria de las combinaciones de las Diferencias Entre Grupos (**DEG**) tomando de 2 en 2 grupos y se divide entre el número total de grupos. Como lo expresa la Ecuación 4.

$$Diferencia\ = \frac{[DEG1(G_1, G_2) + DEG2(G_1, G_3) + \cdots + DEGn(G_i, G_j)\ ]}{k}$$

Ecuación 4

Donde *k* es el número de grupos creados, *n* es un contador de diferencias, DEG_n (G_i, G_j) es la Diferencia *n* entre el Grupo *i* y Grupo *j*.

Con base a lo anterior se presenta un ejemplo, suponer que se genera la siguiente solución (Figura 37) y utilizando la matriz de semejanzas del ejemplo en el AGH-CHIP (Tabla 6):

H1	H2	H3	H4	H5	H6	H7
3	1	2	3	1	2	2

Figura 37 - Ejemplo de Solución (2)

En donde se crearon 3 grupos: G1 {H2, H5}, G2 {H3, H6, H7} y G3 {H1, H5}. Aplicando la Ecuación 4 se tiene:

$$Diferencia = [DEG1\ (G_1, G_2) + DEG2\ (G_1, G_3) + DEG3\ (G_2, G_3)] / k.$$

Al efectuar los cálculos y sustituir con los valores se tiene:

$$Diferencia = (0.568333333 + 0.53 + 0.501666667) / 3 = 0.533333333$$

DEG. La Ecuación 5 corresponde al cálculo de la diferencia entre los elementos (hongos) del Grupo *1* y el Grupo *k*, expresado por DEG_k (G_i, G_j).

$$DEG_k = \frac{\sum_{i=1}^{|Gk|} \sum_{j=1}^{|Gl|} d\ (hi, hj)}{|Gk| * |Gl|}$$

Ecuación 5

Donde *d* (h_i, h_j) es la diferencia entre el hongo $_i$ y el hongo $_j$, entiéndase para este trabajo dicha diferencia como el complemento del valor de intersección en la matriz de semejanzas, por ejemplo, en la matriz de semejanzas del ejemplo anterior el par de hongos (H1, H3) tienen una semejanza de 0.43, por lo tanto su diferencia sería igual a 1 – 0.43 = 0.57.

Para entender mejor el cálculo, se retoma la solución de la Figura 37, para el primer DEG1 (G_1, G_2) y extrayendo los valores de la matriz (Tabla 6), se crea la Figura 38.

DEG1 (G_1, G_2) :

1 – (H2, H3) = 1 – 0.70 = 0.30
1 – (H2, H6) = 1 – 0.46 = 0.54
1 – (H2, H7) = 1 – 0.46 = 0.54
1 – (H5, H3) = 1 – 0.34 = 0.66
1 – (H5, H6) = 1 – 0.26 = 0.74
1 – (H5, H7) = 1 – 0.37 = 0.63

Por lo tanto al efectuar el promedio se tiene = 3.41 / 6 = 0.5683333333333333

Figura 38 - Ejemplo de Cálculo de DEG.

Se acumula el valor DEG_1 (G_1, G_2) y se repite hasta acumular todos los valores de los DEG's.

2. Generalidades.

En el capítulo de marco teórico se dedicó un apartado para explicar a detalle la familia de algoritmos de estimaciones y de ahí uno en particular es la base para la creación del UMDA-CHIP. Según Larrañaga (Larranaga et al., 2003) los **EDA's** (Estimation Distrubution Algorithm, por sus siglas inglés) se basan principalmente en sustituir el cruce y la mutación por la estimación y posterior muestreo de una distribución de probabilidad aprendida a partir de los individuos seleccionados.

También el mismo autor (Larrañaga & Lozano, 2002) enuncia de forma general que un EDA está fundamentado en los siguientes tres pasos básicos los cuales iteran de forma continua hasta que se establezca un criterio de paro, previamente establecido.

1. *Seleccionar algunos individuos de la población.*
2. *Estimar el modelo probabilístico subyacente a dichos individuos seleccionados.*
3. *Muestreo de la distribución de probabilidad aprendida, con el fin de obtener una nueva población de individuos.*

El pseudocódigo general que corresponde al EDA se presenta en la Figura 39.

EDA

$D_0 \leftarrow$ Generar M individuos (la poblacion inicial) al azar

Repeat for $l = 1, 2, \ldots$ hasta que se verifique el criterio de parada

$D_{l-1}^{Se} \leftarrow$ Seleccionar $N \leq M$ individuos de D_{l-1} de acorde con el metodo de seleccion

$p_l(\boldsymbol{x}) = p(\boldsymbol{x}|D_{l-1}^{Se}) \leftarrow$ Estimar la distribucion de probabilidad de que un individuo se encuentre en los individuos seleccionados

$D_l \leftarrow$ Muestrear M individuos (la nueva poblacion) de $p_l(\boldsymbol{x})$

Figura 39 - Pseudocódigo de la aproximación de un EDA (Larranaga & Lozano, 2001)

El mayor problema con los EDA's es como estimar la distribución de probabilidad pi (x). Obviamente el cálculo de todos los parámetros necesarios para especificar la distribución de probabilidad conjunta no es práctica. Este problema trae como consecuencia la

aproximación de la distribución de probabilidad conjunta por medio de distintas factorizaciones –más o menos complejas– de la misma.

La dependencia de los EDA's con respecto al problema se limita a la función de adaptabilidad. En este caso, la distribución de probabilidad conjunta *n*-dimensional se factoriza como el producto de *n* distribuciones de probabilidad univariantes e independientes.

De cualquier manera y como se hizo mención en el capítulo de marco teórico referente a la clasificación en función de su complejidad y la forma de estimar la distribución de probabilidad, dentro de la familia de los EDA's existe uno denominado **UMDA** (*Univariate Marginal Distribution Algorithm*, por sus siglas en inglés) el cual fue introducido por (Mühlenbein, 1997) que es uno de los más simples y fáciles de implementar, y se adecua para atacar el problema en cuestión. En concreto, en cada generación la distribución de probabilidad conjunta, pi (x), que sirve para estimar el comportamiento de los individuos seleccionados, se factoriza como un producto de distribuciones marginales univariantes e independientes según lo muestra la Ecuación 6 de (Abdelmalik Moujahid et al., 2015):

$$p_l(x) = p\left(x \middle| D_{l-1}^{Se}\right) = \prod_{i=1}^{n} p_l(x)$$

Ecuación 6

La Figura 40 corresponde al pseudocódigo del algoritmo UMDA.

UMDA

$D_0 \leftarrow$ Generar M individuos (la poblacion inicial) al azar

Repeat for $l = 1, 2, \ldots$ hasta que se verifique el criterio de parada

$D_{l-1}^{Se} \leftarrow$ Seleccionar $N \leq M$ individuos de D_{l-1} de acorde al metodo de seleccion

$p_l(\boldsymbol{x}) = p(\boldsymbol{x}|D_{l-1}^{Se}) = \prod_{i=1}^{n} p_l(x_i) = \prod_{i=1}^{n} \frac{\sum_{j=1}^{N} \delta_j(X_i = x_i | D_{l-1}^{Se})}{N} \leftarrow$ Estimar la distribucion de probabilidad conjunta

$D_l \leftarrow$ Muestrar M individuos (la nueva poblacion) de $p_l(\boldsymbol{x})$

Figura 40 - Pseudocódigo del UMDA (Mühlenbein, 1997; Mühlenbein & Paass, 1996).

En base a lo establecido de que el UMDA es el algoritmo que utilizó como base para el desarrollo del algoritmo que se enuncia en este capítulo y después de un minucioso análisis y de varias refinaciones del modelo, se obtuvo el siguiente modelo global del **UMDA-CHIP** (Figura 41).

```
INICIO
        Establecer Parámetros
        Generar P0
        Evaluar P0
        Ordenar P0
        Repetir
                Seleccionar Mejores Individuos (Definido por el usuario)
                Estimar Matriz de Probabilidades
                Generar Resto de la Población en base a la Matriz de Probabilidades
                Evaluar Pn
                Ordenar Pn.
        Hasta que ya no haya poblaciones por generar
FIN
```

Figura 41 - Seudocódigo Simplificado del UMDA-CHIP.

En algunos componentes es similar al AGH-CHIP de cualquier forma se establecen los siguientes componentes estructurales del UMDA-CHIP, los cuales son:

1. Establecer parámetros iniciales (punto de arranque).
2. Mecanismo de representación (clústeres).
3. Generar población inicial.
4. Un mecanismo de evaluación para ponderar la calidad de los clústeres obtenidos.
5. Un método de ordenamiento.
6. Mecanismo para estimar la distribución de probabilidad de la población.
7. Mecanismo para generar las siguientes poblaciones.
8. Un criterio de paro.

En los siguientes apartados se explica a detalle cada uno de los puntos anteriores, a fin de mostrar en forma clara cómo trabaja el algoritmo en mención y como da solución a la problemática, que precisamente es la aportación al trabajo de tesis.

1. Establecer parámetros iniciales.

Se establecen variables, se crean los archivos de salida que contendrán la información resultante, también se lee y extrae la información de la matriz de semejanzas.

Los principales parámetros que de forma obligada se tiene que establecer a fin de que el algoritmo se ejecute de forma adecuada se ilustran en la Tabla 20.

Tabla 20 - Descripción de parámetros del UMDA-CHIP.

Parámetro	Descripción
Nombre de archivo	Parámetro de entrada que corresponde a un identificador de archivo, el cual corresponde a la matriz de semejanzas que es el insumo principal del algoritmo. La estructura de dicho archivo debe ser en texto plano y debe contener algún carácter que funja como separador de valores (tabulador por ejemplo). El número de genes se establece de forma automática al leer la estructura del archivo, al ser una matriz cuadrada la dimensión ya sea de filas o columnas corresponde al número de genes, para el caso de este algoritmo se está trabajando con una matriz de 33x33 por lo tanto la longitud será de 33 genes.
Tamaño de la población	Es el número de individuos (cromosomas) que contendrá la población para trabajar con el algoritmo.
Número de clústeres a crear	Valor que se define para establecer hasta cuantos grupos (clústeres) tiene el algoritmo la posibilidad de crear. Como se hizo mención el algoritmo se ajusta de forma automática para que al menos sean dos clústeres.
Población que pasa directamente a la siguiente generación (truncamiento)	Parámetro definido por el usuario, el cual hace alusión a la cantidad (porcentaje) de individuos que pasaran directamente a formar parte de la siguiente generación, dichos individuos son los que fueron mejor evaluados.

Número de generaciones	Es el número máximo de generaciones a la será sometido cada corrimiento del algoritmo, además de que funge como principal y único criterio de paro del algoritmo.

2. Mecanismo de Representación.

El principal objetivo es la creación de clústeres de hongos basados en su información proteómica.

Este punto es básicamente igual al AGH-CHIP ya que la forma de representar e identificar cada clúster es la misma, sin embargo se muestra otro ejemplo para reforzar el entendimiento del mecanismo; para cada solución el número que se genera es una "etiqueta" para identificar al clúster, así, por ejemplo sí el tamaño de un individuo es de 10 genes (bits) y el número máximo de clústeres a crear es 4, por lo tanto de forma aleatoria se crean los 10 genes con un número comprendido entre 1 y 4, siguiendo esto y tomando en consideración que en el orden de izquierda a derecha y de manera ascendiente se va identificando a cada hongo en particular (H1, H2 … Hn, donde n es el número máximo de bits), lo anterior queda representado de forma gráfica en la Figura 42.

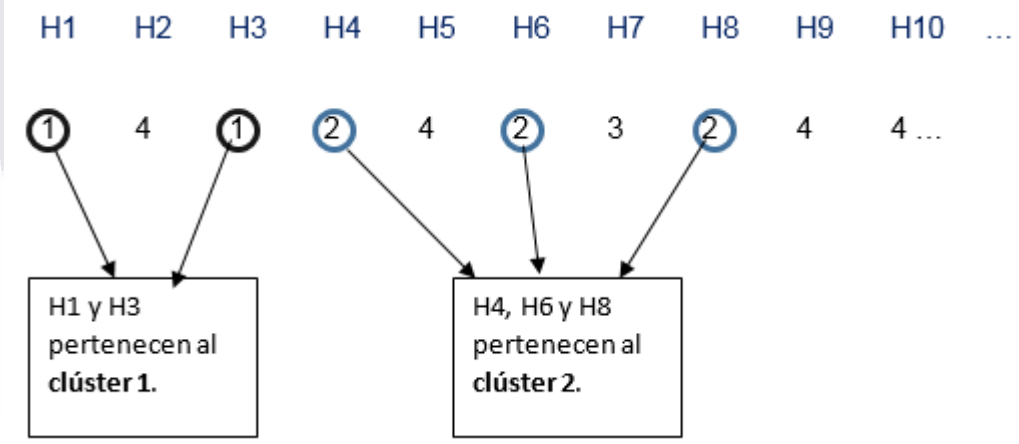


Figura 42 - Ejemplo de Clústeres creados en un individuo.

3. Generar Población Inicial.

Del mismo modo y al igual que el AGH-CHIP una vez que se establecen de forma correcta los parámetros iniciales, el algoritmo está listo para iniciar.

En primera instancia se debe generar la población inicial definida en el pseudocódigo general del UMDA-CHIP como P_0 y que corresponde a la generación inicial (Generación 0), para ello se siguen los siguientes pasos de forma iterativa hasta completar el tamaño de la población:

1. Para cada individuo, con base al valor del número máximo de clústeres que se pueden crear, aplicar la técnica de selección por ruleta (descrito en el capítulo de marco teórico) para seleccionar un número entre 2 y el máximo, para definir realmente cual será el máximo real de clústeres que se pueden crear. Ejemplo, si define que el algoritmo puede crear un máximo de 8 clústeres, al aplicar la ruleta suponga que el número seleccionado corresponde a un 4, significa que para ese individuo se pueden crear hasta 4 clústeres.
2. Generar de forma aleatoria cada gen hasta completar el tamaño del individuo, estableciendo como valor máximo el establecido del paso anterior. Se ilustra en seguida un ejemplo tomando en consideración que la longitud del individuo es de 10 y se van a crear hasta 4 clústeres (grupos).

2	3	2	3	4	3	1	1	1	1

Lo anterior significa que se crearon los siguientes grupos: G1 {7, 8, 9, 10}, G2 {1, 3}, G3 {2, 4, 6}, G4 {5}.

Los pasos anteriores se repiten hasta que se complete el tamaño de la población (parámetro previamente establecido), la Tabla 21 muestra un Ejemplo de una población de 15 individuos cada individuo tiene un tamaño de 10 genes.

Tabla 21 - Ejemplo de Población de Clústeres.

1	2	4	5	2	1	4	5	2	4
2	4	2	4	1	1	2	2	3	3
3	5	2	4	5	2	5	3	4	4
3	2	1	1	2	2	5	4	3	2
4	3	5	2	5	5	5	4	4	4
2	2	4	2	2	4	3	3	4	3
2	4	2	5	3	2	3	3	3	5
3	2	5	3	5	2	2	2	5	3
5	2	5	4	1	1	1	3	3	5
4	3	2	2	5	4	3	4	2	4
2	4	2	5	2	2	4	2	3	5
3	2	5	2	4	2	3	5	5	2
5	4	4	2	2	1	2	3	2	5
5	2	1	4	4	5	5	3	2	4
2	2	4	3	2	4	2	3	2	2

4. Mecanismo para calificar cada individuo (función de adaptabilidad).

En la sección de “Función de Adaptabilidad” de este mismo capítulo la función que aplica el UMDA-CHIP para calificar cada solución que es generada, en este punto se describe un ejemplo adicional con el objetivo de entender mejor el mecanismo.

Recordando, la función de adaptabilidad multi-objetivo del algoritmo busca maximizar la semejanza entre los clústeres por un lado, y por el otro maximizar la diferencia con respecto a otros clústeres, por lo tanto una vez ha sido generado el cromosoma (solución), hay que considerar lo siguiente: éste deberá ser calificado en base a dos características que se evalúan por separado:

La primer parte asigna una calificación a cada uno de los conjuntos de hongos dentro del mismo clúster basándose en su semejanza proteómica (**Semejanza intraclúster**), mientras que la segunda parte asigna una calificación basándose en la diferencia con respecto a los otros clústeres (**Diferencia entre clúster**) para finalmente promediar ambos valores en una

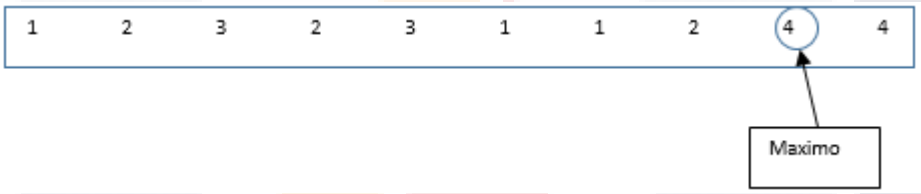
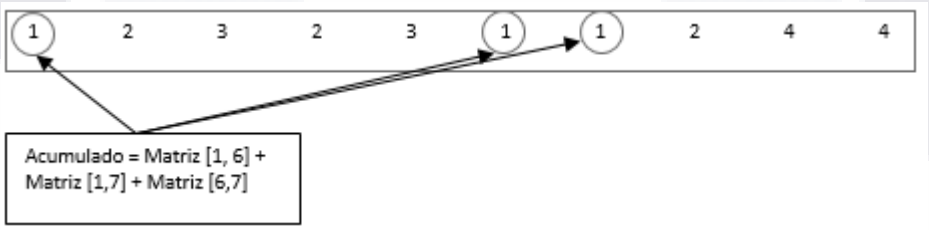
sola. En seguida se explica con un ejemplo más detallado la manera en cómo se aplica el mecanismo de evaluación a la problemática que plantea esta investigación.

1. **Calcular semejanza entre clústeres (interna).**

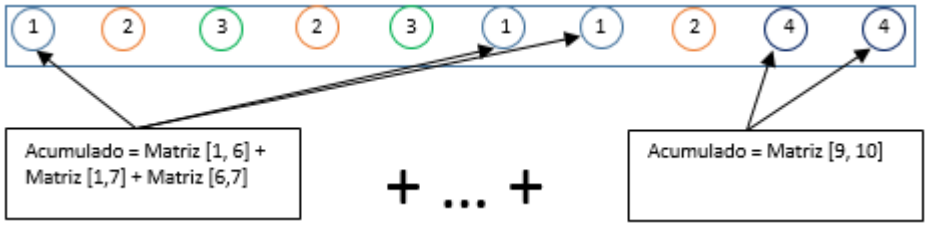
   Se hace mención que la forma de calcular la semejanza en el UMDA-CHIP es la misma que en el AGH-CHIP.

   La Tabla 22 explica el proceso que se efectuó para obtener la semejanza entre clústeres de un individuo.

Tabla 22 - Pasos del EDA-UMDA (cálculo de semejanza).

Suponga que se creó la solución mostrada Determinar el número clústeres que fueron creados (máximo). 
Empezando con el clúster de menor valor, se procede a acumular el valor obtenido de la matriz de semejanzas en la posición (j, k) de ese mismo clúster (lo correspondiente a la SIG explicada en la función de adaptabilidad). 
Una vez terminado con los elementos del j-ésimo clúster, el valor acumulado se divide entre el número de combinaciones contabilizadas y dicho valor se acumula. En este caso se contabilizaron 3 combinaciones por lo tanto: Acumulado = (Matriz [1, 6] + Matriz [1, 7] + Matriz [6, 7]) / 3.

Se almacena dicho valor y se procede a repetir lo anterior hasta que ya no haya clústeres por comparar.



Una vez acumulado el valor de cada uno de los clústeres que se crearon, dicho valor se divide entre el número máximo de clústeres creados que en este caso es 4.

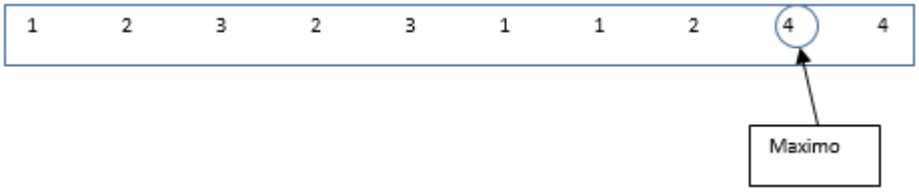
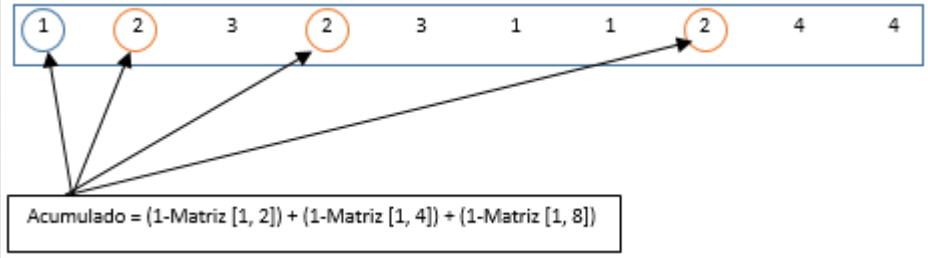
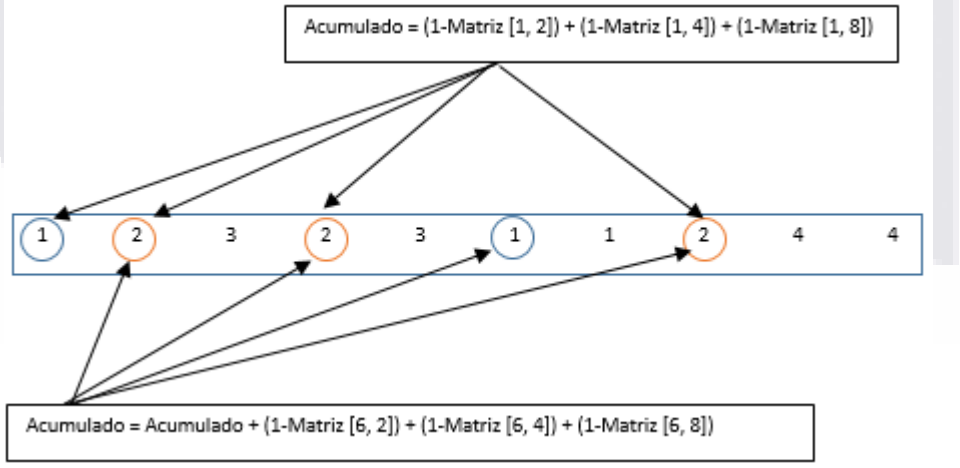
Finalmente dicho valor se multiplica por **0.5** para obtener la primera mitad del valor de la función de adaptabilidad y con ello se está computando la primer parte de la función de adaptabilidad del UMDA-CHIP que corresponde a la semejanza entre clústeres.

Suponiendo que el la semejanza fue 6.8789798789789787 se tiene que Fitness1= 6.8789798789789787 * 0.5 = 3.439489939489489 que corresponde al primer objetivo de la función de adaptabilidad y le otorga ese valor como peso.

2. **Calcular diferencia con respecto a otros clústeres (entre clústeres).**
   Como ya se hizo mención en el punto anterior, el mecanismo para evaluar el clúster consta de dos partes, la primera fue calcular la semejanza entre clústeres del cromosoma, la segunda parte corresponde a la diferencia con respecto a otros clústeres dentro del mismo cromosoma.

   La Tabla 23 muestra los pasos que sigue el algoritmo para el cálculo de la otra parte de la función multi-objetivo.

Tabla 23 - Pasos del EDA-UMDA (calculo diferencias).

Determinar el número clústeres que fueron creados (máximo). 
Empezando con el i-ésimo elemento del j-ésimo clúster, se procede a acumular el valor del complemento obtenido de la matriz de semejanzas en la posición (j, k) de ese mismo clúster (lo correspondiente al DEG descrito en la función de adaptabilidad). 
Continuar con el paso anterior hasta que cada uno de los elementos del clúster en cuestión se haya comparado (por pares) con los elementos del j-ésimo clúster. 
Una vez terminado con los elementos del j-ésimo clúster, el valor acumulado se divide entre el número de combinaciones contabilizadas y dicho valor se almacena.

Continuando con el ejemplo, suponga que DEG (G1 vs G2), donde G1 y G2 son los clústeres con etiqueta 1 y 2 respectivamente fue de 2.98273827398732 y que el número de comparaciones corresponde a 6 por lo tanto el valor éste DEG sería 0.4971230456645533.

Se procede a repetir los pasos anteriores hasta que ya no haya clústeres por comparar.

Acumulado = DIG_1 (G1 Vs G2) + DIG_2 (G1 Vs G3) + DIG_3 (G1 Vs G4) + DIG_4 (G2 Vs G3) + DIG_5 (G2 Vs G4) + DIG_6 (G3 Vs G4).

Una vez obtenido el valor de comparación con respecto a la diferencia con respecto a los otros clústeres, dicho valor se divide entre el número clústeres para ponderar el valor. Para este caso se crearon 4 grupos y suponga que el valor de la suma de DEG's corresponde a 12.923892183920 por lo tanto el valor de la diferencia de este individuo sería 3.23097304598.

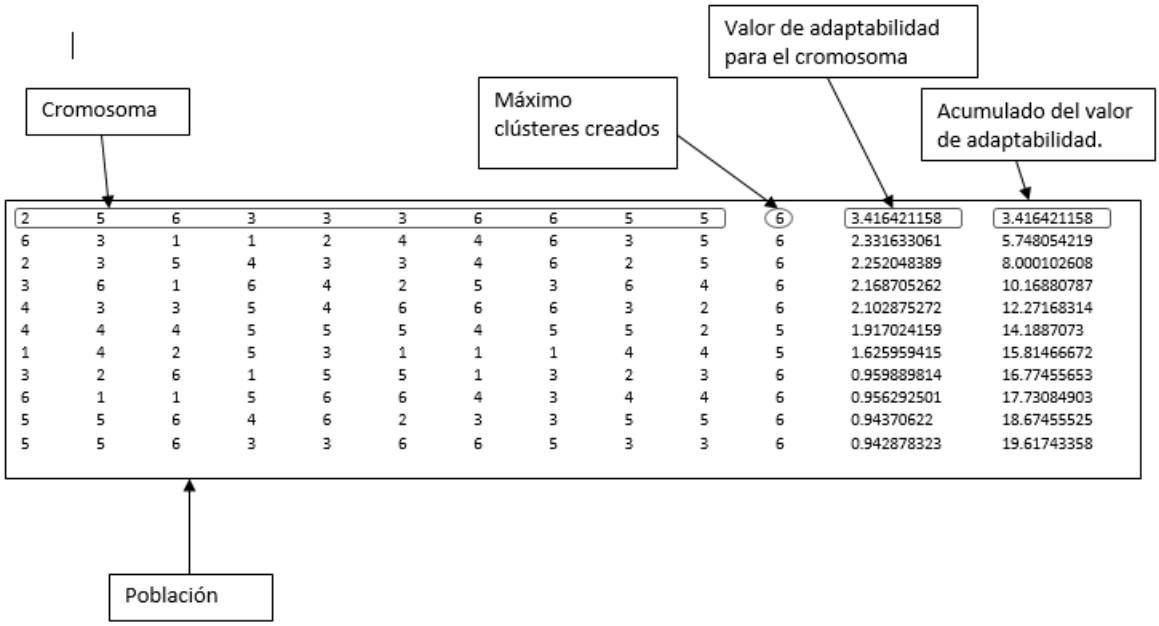
Finalmente dicho valor se multiplica por **0.5** para obtener el complemento al valor de la función de adaptabilidad.

Fitness2 = 3.23097304598 * 0.5 = 1.61548652299

Finalmente los dos valores se suman tal como lo expresa la Ecuación 3 de la función de adaptabilidad, que para la explicación y ejemplo se denominaron Fitness1 y Fitness2, como si fuesen justamente objetivos separados para luego integrarlos en la función multio-bjetivo a la que se hace alusión en el apartado del mismo nombre.

Completando con los valores hipotéticos de lo anterior por lo tanto se tiene que el valor de adaptabilidad corresponde a 3.439489939489489 + 1.61548652299 = 5.054976462479489.

Lo anterior es únicamente para calcular el valor de un sólo individuo, por lo tanto se repiten los pasos anteriores tantas veces como individuos haya en la población, adicionalmente se agrega a la población el número máximo de clústeres creados para cada individuo, el valor de adaptabilidad y el valor de adaptabilidad acumulado hasta ese momento. Lo anterior se muestra en la Figura 43, la cual es solamente un ejemplo ilustrativo.



2	5	6	3	3	3	6	6	5	5	6	3.416421158	3.416421158
6	3	1	1	2	4	4	6	3	5	6	2.331633061	5.748054219
2	3	5	4	3	3	4	6	2	5	6	2.252048389	8.000102608
3	6	1	6	4	2	5	3	6	4	6	2.168705262	10.16880787
4	3	3	5	4	6	6	6	3	2	6	2.102875272	12.27168314
4	4	4	5	5	5	4	5	5	2	5	1.917024159	14.1887073
1	4	2	5	3	1	1	1	4	4	5	1.625959415	15.81466672
3	2	6	1	5	5	1	3	2	3	6	0.959889814	16.77455653
6	1	1	5	6	6	4	3	4	4	6	0.956292501	17.73084903
5	5	6	4	6	2	3	3	5	5	6	0.94370622	18.67455525
5	5	6	3	3	6	6	5	3	3	6	0.942878323	19.61743358

Figura 43 - Ejemplo de población completa

Una vez que se completa y evalúa la población, el siguiente paso es ordenar dicha población de forma descendente por el valor de adaptabilidad computado para cada individuo. El método utilizado es “ordenamiento de la burbuja”. Se explica de mejor manera en el siguiente punto.

5. Método de Ordenamiento.

Cómo ya se hizo mención, el siguiente paso una vez que se tiene la población completa, es el ordenar dicha población para ello se usó el algoritmo de la burbuja, debido a que es fácil de implementar.

De nueva cuenta se hace mención que la forma de implementar es igual al AGH-CHIP. Por lo tanto puede consultar dicho punto.

6. Mecanismo para Estimar la Distribución de Probabilidad de la Población.

Como se ya dijo, los operadores básicos evolutivos son remplazados por un mecanismo que estima de forma probabilística los componentes de la siguiente generación.

Antes de generar cada nueva población se debe estimar el modelo de probabilidad.

Acorde a la forma en cómo se adaptó el algoritmo a la problemática, se creó una matriz denominada “matriz de probabilidades” la cual contiene la probabilidad de que un individuo sea seleccionado (para este caso un miembro de un clúster). En la Tabla 24 se explican con un ejemplo ilustrativo la forma en cómo se estima dicha matriz de probabilidades.

Tabla 24 - Ejemplo de estimación de la matriz de probabilidades.

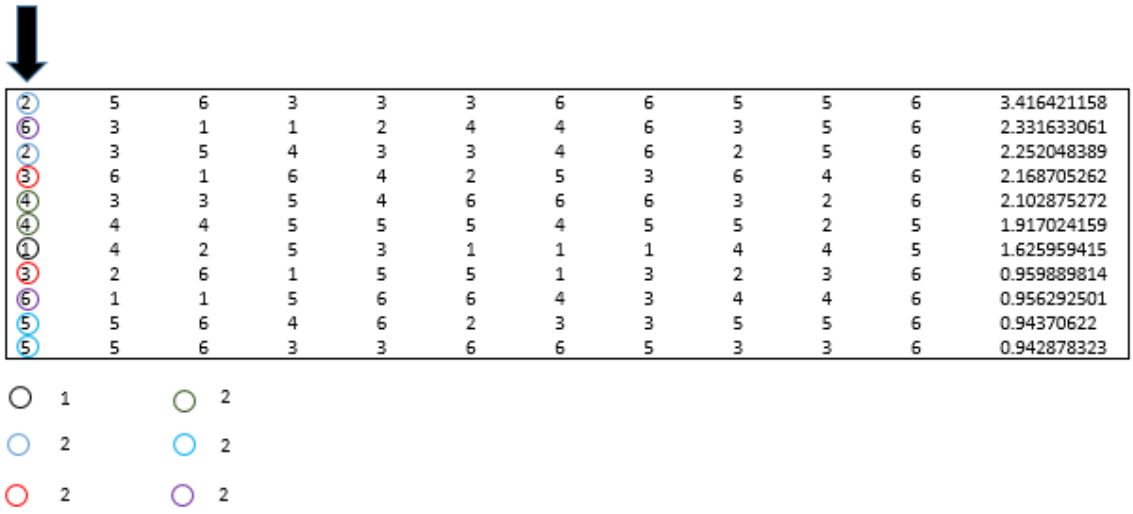
Partiendo de una población ya ordenada de mayor a menor, se establece la proporción de ocurrencia de cada individuo de cada cromosoma.

11

2	5	6	3	3	3	6	6	5	5	6	3.416421158
6	3	1	1	2	4	4	6	3	5	6	2.331633061
2	3	5	4	3	3	4	6	2	5	6	2.252048389
3	6	1	6	4	2	5	3	6	4	6	2.168705262
4	3	3	5	4	6	6	6	3	2	6	2.102875272
4	4	4	5	5	5	4	5	5	2	5	1.917024159
1	4	2	5	3	1	1	1	4	4	5	1.625959415
3	2	6	1	5	5	1	3	2	3	6	0.959889814
6	1	1	5	6	6	4	3	4	4	6	0.956292501
5	5	6	4	6	2	3	3	5	5	6	0.94370622
5	5	6	3	3	6	6	5	3	3	6	0.942878323

Ejemplo: Son 11 cromosomas, por lo tanto 100/11 = 9.1 (aproximadamente), de forma individual cada gen (visto en forma de columna) tendría un peso de **9.1%** de ser seleccionado

Empezando por la primera columna (de izquierda a derecha), se cuentan cada uno de los elementos que pertenecen al mismo clúster.



2	5	6	3	3	3	6	6	5	5	6	3.416421158
6	3	1	1	2	4	4	6	3	5	6	2.331633061
2	3	5	4	3	3	4	6	2	5	6	2.252048389
3	6	1	6	4	2	5	3	6	4	6	2.168705262
4	3	3	5	4	6	6	6	3	2	6	2.102875272
4	4	4	5	5	5	4	5	5	2	5	1.917024159
1	4	2	5	3	1	1	1	4	4	5	1.625959415
3	2	6	1	5	5	1	3	2	3	6	0.959889814
6	1	1	5	6	6	4	3	4	4	6	0.956292501
5	5	6	4	6	2	3	3	5	5	6	0.94370622
5	5	6	3	3	6	6	5	3	3	6	0.942878323

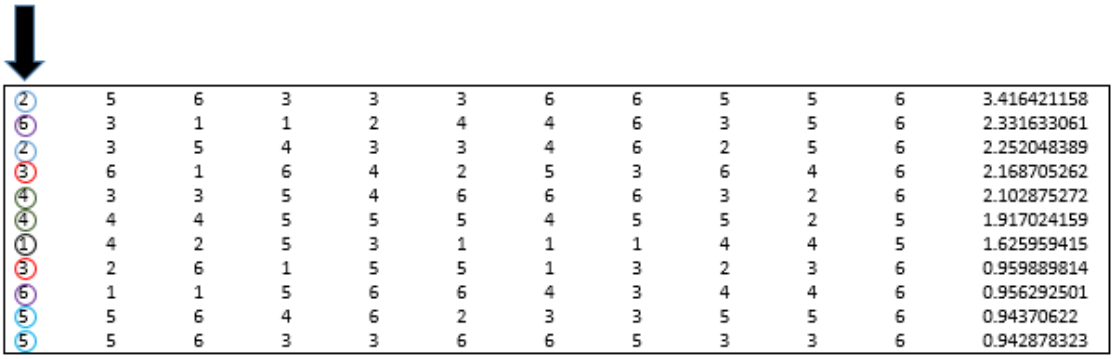
1 2

2 2

2 2

Existen 1 elementos dentro del clúster etiquetado cómo “1”, 2 elementos dentro del clúster “2”, etcétera.

Una vez contados todos los elementos de cada grupo, cada grupo se multiplica por su respectiva proporción (se puede comprobar al sumar todas las proporciones resultantes que será prácticamente 100%, dependiendo de decimales usados) para conocer la probabilidad acumulada de cada clúster.



2	5	6	3	3	3	6	6	5	5	6	3.416421158
6	3	1	1	2	4	4	6	3	5	6	2.331633061
2	3	5	4	3	3	4	6	2	5	6	2.252048389
3	6	1	6	4	2	5	3	6	4	6	2.168705262
4	3	3	5	4	6	6	6	3	2	6	2.102875272
4	4	4	5	5	5	4	5	5	2	5	1.917024159
1	4	2	5	3	1	1	1	4	4	5	1.625959415
3	2	6	1	5	5	1	3	2	3	6	0.959889814
6	1	1	5	6	6	4	3	4	4	6	0.956292501
5	5	6	4	6	2	3	3	5	5	6	0.94370622
5	5	6	3	3	6	6	5	3	3	6	0.942878323

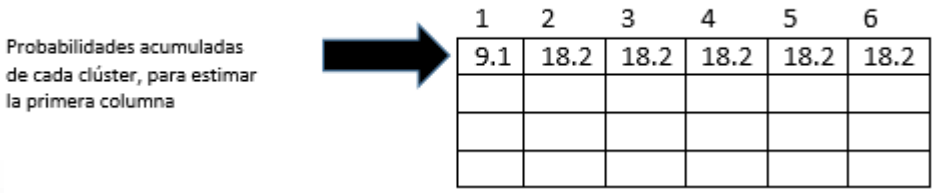
1 * 9.1 = 9.1 2 * 9.1 = 18.2

2 * 9.1 = 18.2 2 * 9.1 = 18.2

2 * 9.1 = 18.2 2 * 9.1 = 18.2

Esto es, que, finalmente la probabilidad de pasar a la siguiente generación el clúster “1” es del **9.1%** (aprox.), de que sea el clúster 2 a 6 es de **18.2%** (por tener el mismo número de ocurrencias) aproximadamente.

Almacenar en la matriz de probabilidades cada valor. Se hace mención que aunque el cálculo se hace por columna para fines prácticos y de conveniencia se almacena en fila pero respetando el mismo principio.

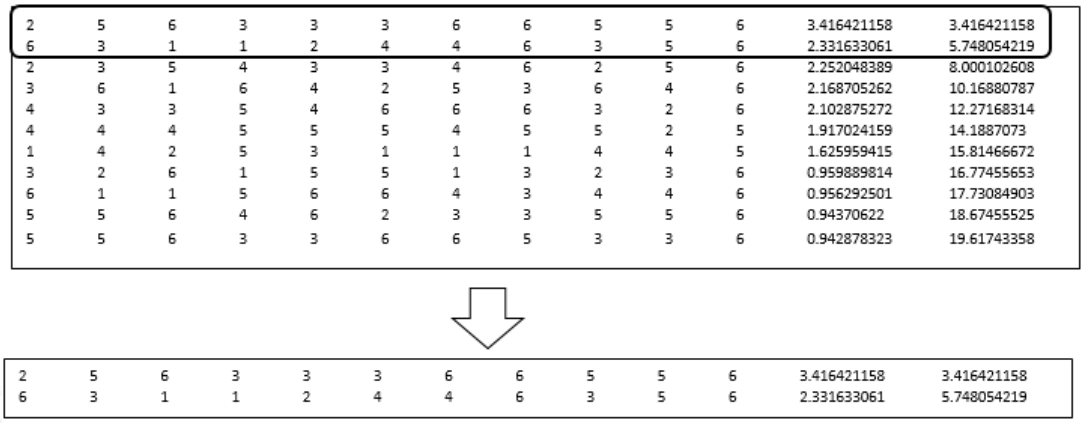


Se continúa con los tres pasos anteriores hasta recorrer todas las columnas y completar la matriz.

7. Mecanismos para generar las siguientes poblaciones.

A partir de la segunda generación se aplican un par de mecanismos de forma iterativa en cada generación como parte del algoritmo de trabajo con el fin de conservar las mejores soluciones y que estás a su vez vayan mejorando a través de generaciones futuras. Los siguientes puntos comprenden la forma y procedimiento en cómo se crean las poblaciones futuras.

1. **Truncamiento**. Primeramente las soluciones candidatas son ordenadas según su función de adaptabilidad, y una proporción **p** (por ejemplo =1/2, 1/3, 1/4,...) de los individuos con mejor desempeño es seleccionada y reproducida 1/p veces. Tomando en consideración el hecho que la población ya fue ordenada de mayor a menor en base a su función de adaptabilidad, en base a lo establecido por el usuario se selecciona la parte proporcional de los mejores individuos para formar parte de la siguiente generación. La Figura 44 muestra un ejemplo general de ello, en donde 2/11 elementos con mejor desempeño fueron directamente seleccionados para formar parte de la siguiente población.



2	5	6	3	3	3	6	6	5	5	6	3.416421158	3.416421158
6	3	1	1	2	4	4	6	3	5	6	2.331633061	5.748054219
2	3	5	4	3	3	4	6	2	5	6	2.252048389	8.000102608
3	6	1	6	4	2	5	3	6	4	6	2.168705262	10.16880787
4	3	3	5	4	6	6	6	3	2	6	2.102875272	12.27168314
4	4	4	5	5	5	4	5	5	2	5	1.917024159	14.1887073
1	4	2	5	3	1	1	1	4	4	5	1.625959415	15.81466672
3	2	6	1	5	5	1	3	2	3	6	0.959889814	16.77455653
6	1	1	5	6	6	4	3	4	4	6	0.956292501	17.73084903
5	5	6	4	6	2	3	3	5	5	6	0.94370622	18.67455525
5	5	6	3	3	6	6	5	3	3	6	0.942878323	19.61743358

2	5	6	3	3	3	6	6	5	5	6	3.416421158	3.416421158
6	3	1	1	2	4	4	6	3	5	6	2.331633061	5.748054219

Figura 44 - Ejemplo de Selección por Truncamiento.

2. **Muestrear la población.** La otra parte para completar la población de las generaciones siguientes, se aplica el principio del UMDA, y así poder estimar probabilísticamente los individuos.

   En el punto 6 (anterior) se describió el mecanismo para obtener la matriz de probabilidades, una vez que se tiene la matriz de probabilidades, se procede a muestrear el resto de la población (recordar que una parte fue creada por truncamiento) de la siguiente generación basándose precisamente en dicha matriz. Para explicar y entender la forma en cómo se generan dichos elementos, se describe en la Tabla 25.

Tabla 25 - Ejemplo de estimación de individuos.

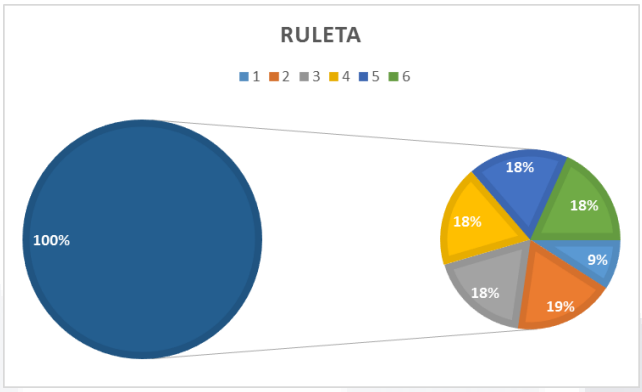
Suponiendo que se generó la siguiente matriz de probabilidades:

Probabilidades acumuladas de cada clúster.

1	2	3	4	5	6
9.1	18.2	18.2	18.2	18.2	18.2
20	20	20	20	10	10
10	15	15	10	40	10
5	15	15	15	10	40

El primer paso es generar un número aleatorio entre 0 y 100. Suponer que se generó un 79.

Utilizando el principio de la ruleta se establece en que sección cae el número generado, el cual en base a la etiqueta que le corresponda será el elegido para aparecer en la siguiente generación.



0 – 9.1	1
9.1 – 27.3	2
27.3 – 45.5	3
45.5 – 63.7	4
63.7 – 81.9	5
81.9 – 100	6

79

Por lo tanto la probabilidad acumulada del número 79 cae en el rango que le corresponde al clúster 5, por lo tanto el gen de ese clúster es 5.

Se repiten los dos pasos anteriores, pero ahora cambiando en su correspondiente fila de la matriz de probabilidades hasta completar el cromosoma. Ósea para el segundo gen del cromosoma se utiliza la segunda fila de la matriz de probabilidades.

Los pasos 1 a 3 son repetidos hasta completar el individuo.

Lo anterior fue únicamente para generar un individuo, por lo tanto se repiten los pasos anteriores, es decir volver a empezar en la primer fila de la matriz para estimar el primer elemento del siguiente individuo y en general en forma sucesiva hasta completar el resto de la población.

8. Establecer Criterio de Paro.

Para esta implementación se optó por número máximo de generaciones, esto es: el criterio de paro está determinado por el número de generaciones establecido antes de la ejecución del algoritmo, es decir el algoritmo continua mientras $P_n < N$, donde P es la población actual en su n-ésima generación y N el número máximo de generaciones a crear.

Una vez que se cumple el criterio de paro el UMDA-CHIP es capaz de presentar los resultados.

3. Experimentación.

En esta parte, se hace énfasis en las condiciones y la forma en la que se llevó a cabo la experimentación del UMDA-CHIP.

1. Condiciones Generales.

En la sección de “Antecedentes y Caso de Estudio” del Capítulo – Metodología quedó establecido que el insumo principal es una matriz de semejanzas proteómicas de hongos de 33 x 33 elementos que corresponde a una lista de 33 hongos y que el algoritmo fue probado con los todos elementos sin excluir ninguno ya que el interés es crear clústeres con todos ellos.

Como se comentó en el AGH-CHIP y siendo la misma matriz de semejanzas con la que se trabajó, debido a la complejidad y tamaño de los valores de dicha matriz, se puede consultar el ANEXO A para ver la lista completa.

2. Diseño de Experimentos.

Con el fin de darle mayor flexibilidad a la investigación, tanto el AGH-CHIP como el UMDA-CHIP fueron sometidos a un diseño factorial de experimentos (Montgomery, 1991).

El software de colección de datos para el experimento se diseñó de tal manera que fuese interactivo con el usuario, lo que permite en una sola ejecución llevar a cabo todas las combinaciones correspondientes de los niveles establecidos del experimento sin necesidad de estar ejecutando el programa cada vez que se desee modificar los parámetros principales.

Para automatizar dicha combinación primeramente se definieron los siguientes factores (Tabla 26) donde cada uno corresponde a un nivel en el experimento que se lleva cabo.

Tabla 26 - Factores considerados para el experimento.

Factor	Descripción
Generaciones	Cantidad de veces que varía el valor del tamaño de la generación.
Cromosomas	Número de veces que cambia el valor del tamaño de la población.
Grupos a Crear	Número de veces que se puede determinar el máximo (no el real) de clústeres a crear en cada ejecución del experimento.
%Truncamiento	Número de veces que se puede modificar el porcentaje de la población que pasará por truncamiento a la siguiente generación.

Se destaca la flexibilidad y aportación del diseño factorial ya que permite variar dinámicamente el número de veces que se modifica cada factor y con ello ir creando la profundidad de los niveles.

Los niveles para el caso del experimento factorial usado en el UMDA-CHIP son los mostrados en la Tabla 27.

Tabla 27 - Diseño factorial sobre el UMDA-CHIP.

Factor	Niveles	Valores Probados
Generaciones	3	50, 100, 150
Cromosomas	3	1000, 1500, 2000
Grupos a crear	2	2, 5
%Truncamiento	2	30, 50

Para cada combinación fueron consideradas 10 réplicas por lo tanto, 3x3x2x2x10 = 360, ósea que el algoritmo fue ejecutado 360 veces, variando dinámicamente los parámetros que se enuncian.

En cada réplica, para sus correspondientes poblaciones creadas, se crea de forma automática conforme a los parámetros, un archivo de texto plano con el siguiente formato de ejemplo para su nombramiento "*V2-50gen-500crom-5gr-2nbm-case4.txt*" de tal manera que permita más fácilmente consultar los resultados de un caso en particular. Lo anterior se describe en la Tabla 28.

Tabla 28 - Elementos que conforman el archivo de salida de cada replica.

Elemento	Descripción
V2	Se refiere a la versión del AGH-CHIP que en este caso es la 2.
50gen	Se refiere al valor del tamaño de generaciones con las que se está ejecutando la réplica, para este caso 50.
500crom	Valor del tamaño de las poblaciones, para este caso 500.
5gr	Máximo de grupos que se pueden llegar a crear en la réplica. 5 para este caso.
2nbm	Máximo de bits a los que se les puede aplicar el MMI. 2 en este caso.
Case4	Se refiere a la réplica ejecutada. En este caso es la 4ta replica que se ejecuta con los parámetros anteriores.

El proceso se ejecutó en una laptop HP ® modelo 2000 con sistema operativo Windows ® 7 Ultimate, procesador Intel Core i3 ®, 8 GB de memoria RAM.

En cuanto a los principales parámetros de salida (computados) se describen en la Tabla 29. Cabe destacar que para concentrar y resumir el total de ejecuciones el UMDA-CHIP también crea dos archivos en texto plano (además de los mencionados para cada ejecución), que además de contener las condiciones del experimento, contiene los valores de los parámetros de salida. Puede consultarse el ANEXO D para ver los resultados generados por el algoritmo a fin de comprender mejor la tabla.

Tabla 29 - Descripción de los parámetros de salida.

Nombre del archivo de Salida	Principales Variables de Salida
Statistics.txt Presenta en forma de secciones los mejores valores encontrados de cada réplica del experimento bajo las condiciones definidas por los parámetros de entrada.	*Max_Fitness*: valor de adaptabilidad máximo encontrado en la réplica. *Time (ms)*: Tiempo que tardó en ejecutarse dicha replica expresado en milisegundos.
Resume.txt Resume las instancias del archivo anterior de tal manera que se muestran promedios de todos los	*#Max_Groups_Created:* Máximo número de clústeres creados al que le corresponde el valor de adaptabilidad señalado.

casos de cada variación de parámetros en el experimento.	*Max_Fitness_Found*: Máximo valor de adaptabilidad que se encontró durante todos los casos de una combinación de parámetros en particular. *Ocurr_Max_Fitness_Found*: De lo anterior, esta variable representa la cantidad de veces que se encontró dicho máximo expresando en porcentaje. *Avg_Max_Fitness*: Promedio de máximos valores de adaptabilidad. *Avg_Time (ms)*: Promedio de los tiempos de ejecución.

3. Experimentos.

Una vez fueron ejecutados todos los 360 casos mencionados del experimento, se tomaron los resultados del archivo “Resume.txt” para crear la siguiente tabla de concentración (Tabla 30) con el fin de visualizar mejor dichos resultados.

Tabla 30 – Resumen de Resultados del Experimento.

#Cases	#Generations	#Chromosomes	#Max_Groups_Created	%Population Truncated	Max_Fitness_Found	Ocurr_Max_Fitness_Found	Avg_Max_Fitness	Avg_Time (ms)
10	50	1000	2	30	0.44184862 6391628	20.00%	0.43750617 3554110	5362.0949 559671
10	50	1000	2	50	0.43598095 9820314	20.00%	0.43208750 5460700	4646.4457 606773
10	50	1000	5	30	8.42349847 5441320	10.00%	6.78963949 4282930	6717.6180 299092
10	50	1000	5	50	5.83909182 1104490	10.00%	4.78012349 0354020	5413.6346 692896
10	50	1500	2	30	0.44184862 6391628	30.00%	0.43796988 9709688	11121.121 0357698
10	50	1500	2	50	0.44184862 6391628	30.00%	0.43752501 4934997	9750.7759 324244
10	50	1500	5	30	7.96255512 2232610	10.00%	6.70874091 8708420	14936.261 6732652
10	50	1500	5	50	6.04034907 0650540	10.00%	5.16343874 4769950	11546.780 6665933
10	50	2000	2	30	0.44113357 5261640	10.00%	0.43710407 9009714	20500.809 8828035
10	50	2000	2	50	0.44184862 6391628	20.00%	0.43693697 4893693	15500.681 5528337
10	50	2000	5	30	8.16179134 5226350	10.00%	6.80760068 9041520	24035.755 3733556

10	50	2000	5	50	5.879991061399490	10.00%	5.316766171666000	18548.5616862377
10	100	1000	2	30	0.441848626391628	70.00%	0.439925261511442	9115.7831083933
10	100	1000	2	50	0.441848626391628	10.00%	0.436187241690264	8094.6114138958
10	100	1000	5	30	8.992077237941800	70.00%	8.925682726207980	11958.6362457039
10	100	1000	5	50	8.931081606042660	10.00%	8.724303513881420	9502.3195046775
10	100	1500	2	30	0.441848626391628	70.00%	0.439816354847259	14232.3694490925
10	100	1500	2	50	0.441848626391628	30.00%	0.437471715746888	13745.7184343579
10	100	1500	5	30	8.992077237941800	80.00%	8.961753613769420	22671.6168673037
10	100	1500	5	50	8.992077237941800	30.00%	8.867074228406650	18509.6426876549
10	100	2000	2	30	0.445503296065505	10.00%	0.438845125229098	24095.6704029873
10	100	2000	2	50	0.441848626391628	20.00%	0.436937073256187	19069.3317235298
10	100	2000	5	30	8.992077237941800	80.00%	8.970199851833290	36233.4756895581
10	100	2000	5	50	8.992077237941800	30.00%	8.838203407945130	29969.0005123346
10	150	1000	2	30	0.441848626391628	40.00%	0.437751938078439	11404.0990464336
10	150	1000	2	50	0.441848626391628	10.00%	0.435655080997687	10880.5254549843
10	150	1000	5	30	8.992077237941800	30.00%	8.908754577403140	14249.6095911668
10	150	1000	5	50	8.992077237941800	50.00%	8.833657928394220	13490.5731661458
10	150	1500	2	30	0.445503296065505	10.00%	0.439471240474142	19119.7649500720
10	150	1500	2	50	0.441848626391628	30.00%	0.437469484943721	16842.0480576442
10	150	1500	5	30	8.992077237941800	60.00%	8.946065505714650	26647.1805585433
10	150	1500	5	50	8.992077237941800	10.00%	8.792739018611700	29134.5626353705
10	150	2000	2	30	0.450916233029988	10.00%	0.441876810016167	31897.7872317017
10	150	2000	2	50	0.450916233029988	10.00%	0.438995451105647	25776.8819676277
10	150	2000	5	30	8.992077237941800	90.00%	8.978900655748930	40933.9105171460
10	150	2000	5	50	8.992077237941800	80.00%	8.958510140368120	38391.8587237285

La tabla anterior contiene además de la fila de encabezados, 36 filas y en cada una de ellas se resumen las 10 réplicas de cada combinación en particular del experimento (36x10=360). Las columnas que están sin resaltar representan los parámetros con los que se llevó a cabo el experimento mientras que las columnas que están resaltadas corresponden a parámetros de salida que se explicó su significado en la Tabla 14.

Al llevar a cabo un análisis de los resultados, respecto al valor de adaptabilidad se puede observar que el mayor valor encontrado es **8.9920772379418** el cual aparece en 11 de las 36 filas lo que significa que en esas 11 combinaciones de parámetros aparece por lo menos una vez este máximo valor.

Con respecto al tiempo este es relativo y como se puede notar el número de generaciones influye de manera exponencial en el tiempo de ejecución del algoritmo sin embargo se puede decir de forma empírica que la mejor combinación de parámetros que satisfacen el máximo que se ha hecho mención es la siguiente: Número de Generaciones: 100, Tamaño de la Población: 1000, %Truncamiento 30: 5, Máximo Número de Grupos: 5.

Otro aspecto importante que se pude observar es en la columna “#Max_Groups_Created” que representa el máximo de clústeres que se crearon para su correspondiente valor de adaptabilidad, tiene una tendencia a discriminar y agrupar los hongos en **5 grupos**.

4. Pruebas Estadísticas.

Con el fin de comprobar si el algoritmo es sensible a la variabilidad de los parámetros en términos de las respuestas (valor de adaptabilidad y tiempo de ejecución) y establecer soporte estadístico a los resultados y conclusiones, se llevaron a cabo una serie de pruebas con los valores del archivo “Statistics.txt”.

En primer lugar se determinó si los valores de adaptabilidad y tiempo seguían una distribución normal, para ello se aplicaron las pruebas de Kolmogorov-Smirnov y Shapiro-Wilk, además cuando fue necesario se ejecutó una prueba de homocedasticidad empleando el estadístico de Levene. En seguida se muestran las gráficas de histograma resultantes, para los datos del valor de adaptabilidad (Figura 45) y valores del tiempo (Figura 46) respectivamente.

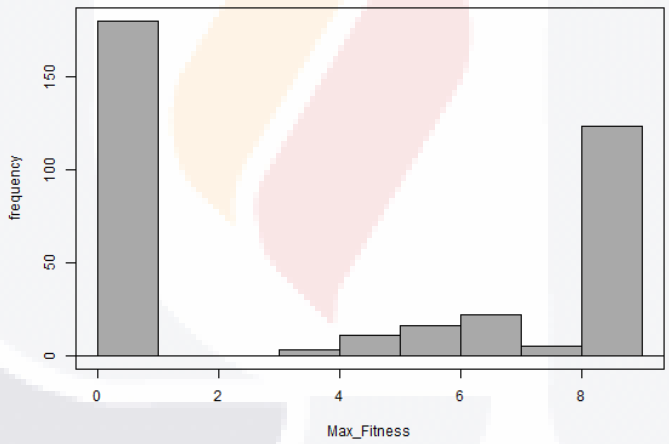


Figura 45 - Prueba de Normalidad del Valor de Adaptabilidad.

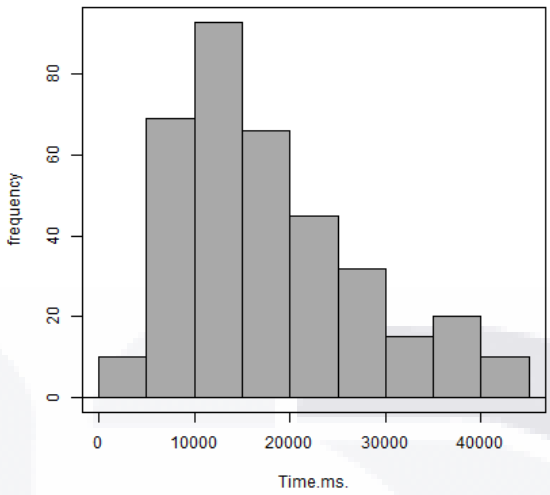


Figura 46 - Prueba de Normalidad del Tiempo.

Tal como se puede observar de forma gráfica los datos no siguen una distribución normal, de hecho la prueba de Shapiro-Wilk muestra que para el valor de adaptabilidad p-value < 2.2e-16 y para tiempo p-value = 1.445e-11 ósea que estadísticamente no siguen una distribución normal, por lo que se procedió a usar estadística no paramétrica y se aplicó la prueba de Kruskal-Wallis con un 95% de confianza para la comparación de las 360 réplicas. Para llevar a cabo la prueba de Kruskal-Wallis se usó el programa estadístico SPSS.

La Tabla 31 muestra el resultado de la prueba mencionada.

Tabla 31 - Prueba de Kruskal-Wallis UMDA-CHIP.

Estadísticos de contraste[a,b]

	Tiempo	MilFitness
Chi-cuadrado	353.909	314.014
gl	35	35
Sig. asintót.	.000	.000

Al observar el resultado de la prueba el valor de p (Sig. Asintót) es muy pequeño tanto para adaptabilidad como para tiempo, incluso se muestra que ambos tienen un valor de .000, evidentemente los cuales están por debajo de 0.05, por lo tanto se concluye que estadísticamente hay evidencia suficiente para afirmar que existe diferencia significativa

entre los grupos de ejecuciones del modelo, por lo que el algoritmo sí es sensible al cambio de parámetros.

Dicho lo anterior se dice que la mejor combinación de parámetros en el menor tiempo es la que se hace referencia en el punto anterior: Número de Generaciones: 100, Tamaño de la Población: 1000, %Truncamiento: 30, Máximo Número de Grupos: 5, con un tiempo promedio de las 10 réplicas de **11958.63624570393** milisegundos, lo cual es muy bueno considerando la magnitud del espacio de soluciones del problema.

En la siguiente sección se le da un sentido, interpretación y soporte científico a los resultados que se obtuvieron.

Si se desea consultar las tablas y figuras con los resultados de las pruebas estadísticas llevabas a cabo, se pueden consultar dichos resultados en su correspondiente ANEXO E.

5. Interpretación de Resultados.

Retomando el mejor valor de adaptabilidad que se obtuvo hasta el momento de esta investigación que corresponde a **8.9920772379418**, al llevar a cabo una búsqueda sobre el conjunto de archivos resultantes para ver los clústeres que formó el algoritmo, se tiene la siguiente imagen (Figura 47) que corresponde a una búsqueda parcial en el conjunto de archivos.

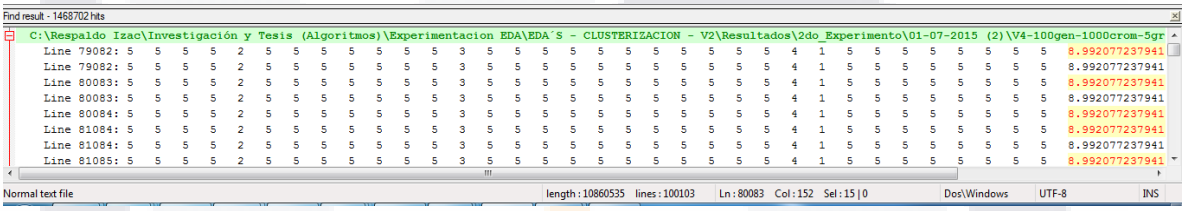


Figura 47 - Captura de pantalla de donde se encontró el valor máximo.

Como se dijo y como se puede ver, el UMDA-CHIP crea 5 grupos para el valor señalado los casos en la cadena resultante donde aparece el valor de adaptabilidad, que básicamente es su complemento o inverso tal como se muestra en seguida:

5 5 5 5 1 5 5 5 5 5 5 2 5 5 5 5 5 5 5 5 5 5 4 3 5 5 5 5 5 5 5 5

5 5 5 5 1 5 5 5 5 5 5 3 5 5 5 5 5 5 5 5 5 5 4 2 5 5 5 5 5 5 5 5

5 5 5 5 2 5 5 5 5 5 5 1 5 5 5 5 5 5 5 5 5 5 4 3 5 5 5 5 5 5 5 5

5 5 5 5 2 5 5 5 5 5 5 3 5 5 5 5 5 5 5 5 5 5 4 1 5 5 5 5 5 5 5 5

Para fines explicativos se utilizó la cadena resaltada.

Antes de dar interpretación a la cadena resultante y ver cuantos grupos fueron creados, se hizo una minuciosa revisión de la literatura relacionada con clasificación de hongos, a fin de observar si los clústeres encontrados empatan con alguna de las clasificaciones ya existentes.

Después de las últimas modificaciones hechas en el Congreso Internacional de Micología de 1994, donde se introdujeron muchos cambios, se propuso la siguiente clasificación basada en filos para el reino fungi: (Hibbett, 2005) a partir de (Lutzoni et al., 2004);

(Blackwell, Hibbett, Taylor, & Spatafora, 2006); (Bruns et al., 1992); (Popoff, 2007) a partir de (Alexopoulos, Mims, & Blackwell, 1996); (Ebersberger et al., 2009) y (Montes et al., 2003).

Reino fungi:

- Phylum Ascomycota (1)
- Phylum Basidiomycota (2)
- Phylum Chtridiomycota (3)
- Phylum Zygomycota (4)

Tomando como referencia la clasificación a cuatro filos (en el orden presentado) y la lista de hongos (Tabla 5) que son objeto de estudio de esta investigación, se creó la Tabla 32 para ilustrar a que filo corresponde cada hongo de la lista. Si lo desea puede consultar el ANEXO F, correspondiente para el árbol filogenético que Ebersberger y sus colegas (Ebersberger et al., 2009) incluyen en su publicación.

Tabla 32 - Clasificación de los hongos basada en cuatro familias.

No.	Nombre	Fam.	No.	Nombre	Fam.
1	Ashbya gossypii.	1	18	Fusarium oxysporum	1
2	Aspergillus fumigatus.	1	19	Fusarium verticilloides.	1
3	Aspergillus nidulans.	1	20	Histoplasma capsulatum.	1
4	Aspergillus terreus.	1	21	Kluyveromyces.	1
5	Batrachochytrium dendrobatidis	3	22	Loderomyces elongisporus.	1
6	Botrytis cinerea.	1	23	Magnaporthe grisea.	1
7	Candida albicans.	1	24	Neurospora crassa.	1
8	Candida glabrata.	1	25	Puccinia graminis.	2
9	Candida guilliermondii.	1	26	Rhizopus oryzae.	4
10	Candida lusitaniae.	1	27	Saccharomyces cerevisiae.	1
11	Candida tropicalis.	1	28	Sacharomices japonicus.	1
12	Chaetomium globosum.	1	29	Sclerotinia sclerotiorum.	1
13	Coprinus cinereus.	2	30	Stagonospora nodorumm.	1
14	Coccidiodes immitis.	1	31	Uncinocarpus reesii.	1
15	Cryptococcus neoformans serotype A.	2	32	Ustilago maydis.	2
16	Debaryomyces hansenii	1	33	Yarrowia lipolytica.	1
17	Fusarium graminearum.	1			

Ahora, en relación a la tabla anterior y los clústeres creados, tomando de izquierda a derecha el orden en la cadena resultante, y teniendo en consideración que el número asignado al clúster es meramente una etiqueta (antes de clasificar) y para fines de conveniencia a partir de las etiquetas de los clústeres se propone la siguiente relación:

- Familia 1 => Chtridiomycota
- Familia 2 => Basidiomycota
- Familia 3 => Zygomycota
- Familia 4 => Sin clase
- Familia 5 => Ascomycota.

Con base a la explicación anterior se presenta la Tabla 33 con la clasificación propuesta por el UMDA-CHIP como su interpretación:

Tabla 33 - Clasificación propuesta por el UMDA-CHIP.

Grupo / Familia	Hongos
1 (Chtridiomycota)	5 Batrachochytrium dendrobatidis
2 (Basidiomycota)	13 Coprinus cinereus
3 (Zygomycota)	26 Rhizopus oryzae
4 (Sin clase)	**25 Puccinia graminis**
5 (Ascomycota)	1 Ashbya gossypii 2 Aspergillus fumigatus 3 Aspergillus nidulans 4 Aspergillus terreus 6 Botrytis cinerea 7 Candida albicans 8 Candida glabrata 9 Candida guilliermondii 10 Candida lusitaniae 11 Candida tropicalis 12 Chaetomium globosum 14 Coccidiodes immitis **15 Cryptococcus neoformans serotype A** 16 Debaryomyces hansenii 17 Fusarium graminearum 18 Fusarium oxysporum 19 Fusarium verticilloides 20 Histoplasma capsulatum 21 Kluyveromyces lactis 22 Loderomyces elongisporus 23 Magnaporthe grisea 24 Neurospora crassa 27 Saccharomyces cerevisiae 28 Sacharomices japonicus 29 Sclerotinia sclerotiorum 30 Stagonospora nodorumm 31 Uncinocarpus reesii **32 Ustilago maydis** 33 Yarrowia lipolytica

De la figura anterior los elementos que están resaltados en negritas corresponden a los hongos que fueron incorrectamente clasificados, ósea que quedaron en una familia que no

les corresponde, sin embargo los otros **30 hongos empatan perfectamente con la clasificación de la literatura**. Dicho de otra manera de la lista de 33 hongos, **30 (90.90%)** fueron correctamente clasificados en su respectiva familia filogenética

Si bien la literatura reporta una clasificación basada en 4 familias y el UMDA-CHIP se probó para 5 grupos, como puede observarse el algoritmo fue capaz de llevar a cabo un muy buen acomodo.

6. Conclusiones del UMDA-CHIP.

El algoritmo diseñado y referenciado como UMDA-CHIP, implementado y puesto a punto, fue capaz de agrupar hongos partiendo de una matriz de semejanzas entre sus proteomas.

Al contrastar los resultados del algoritmo con la literatura, se pudo comprobar que el algoritmo fue capaz de discernir de tal forma que logró por sí mismo acomodar con poco más del 90% de asertividad la lista de hongos dentro de las familias: Ascomycota, Basidiomycota, Chtridiomycota, Zygomycota.

En resumen, con la aplicación de una técnica metaheurística fue posible llevar a cabo una clasificación correcta en más del 90% con un tiempo y costo computacional razonable en relación a la magnitud del problema.

Capítulo VI: Conclusiones.

En seguida se presentan las conclusiones de la tesis, las aportaciones que se lograron con el desarrollo de la investigación titulada “Clusterización de Hongos Mediante Metaheurísticas Híbridas a Partir de su Información Proteómica”, los objetivos que se cumplieron, un análisis comparativo general entre las dos técnicas propuestas y finalizar con un apartado sobre el rumbo que el trabajo pudiera tomar en un futuro próximo y cercano.

1. Conclusiones Generales.

Las técnicas de inteligencia artificial, representan un poderoso instrumento para resolver problemas de cualquier área del conocimiento humano y en particular los métodos metaheurísticos han demostrado obtener muy buenos resultados en un tiempo y costo computacional aceptable como lo es en el caso de la presente investigación.

Dichas técnicas, apoyando a ciencias como la bioinformática en áreas como la proteómica, tienen un futuro prometedor debido a que esta se perfila como una potente herramienta para el estudio y descubrimiento de aspectos fisiopatológicos en los seres humanos y ayudará a dilucidar las bases de criterios importantes relativos a la salud y la enfermedad.

Durante el desarrollo de esta investigación se logró consolidar un marco de trabajo robusto y flexible que permite la aplicación de técnicas metaheurísticas en problemas normalmente intratables como la clusterización (NP-completos). Además, se logró desarrollar y poner a punto dos técnicas metaheurísticas que a partir de una matriz de semejanzas proteómicas llevan a cabo la creación de clústeres.

Cabe destacar que una de las técnicas se combinó con un operador de mejora MMI (Mecanismo de Mutación Inteligente), que es una aportación valiosa al área de conocimientos para la tarea de clusterización y que reportó muy buenos resultados.

Los resultados empíricos de los experimentos fueron confirmados estadísticamente y se lograron identificar los mejores parámetros para cada algoritmo. Los resultados dentro del área de la aplicación fueron confirmados por expertos en el área de bioinformática y de biología de nuestra universidad.

Otro hecho a resaltar son los resultados mismos ya que al llevar una minuciosa investigación en la literatura existente se logró comprobar que cada uno de los algoritmos empató sus resultados de una forma muy aceptable con una clasificación utilizada por investigadores.

De manera general se puede concluir que en base a la información de los mejores hits bidireccionales de los proteomas, de los hongos de estudio y con la aplicación de métodos heurísticos, es posible hacer una clusterización con un grado de certidumbre de alta confiabilidad, lo que justifica el uso e importancia de éstas técnicas que si bien es cierto aún son nuevas y poco exploradas la tendencia apunta que tienen un futuro prometedor.

2. Conclusiones de los Algoritmos.

Cada técnica tiene algo que aportar, en seguida se presentan los elementos más sobre salientes.

Metaheurística AGH-CHIP.

Esta técnica está inspirada en los postulados de la teoría de la evolución de Charles Darwin (Darwin, 1859) en la cual se simula la evolución natural.

Primeramente se planteó un marco de trabajo flexible para trabajar con la técnica y así poder atacar el problema que se plantea.

Es sabido que los algoritmos genéticos tienden a converger muy rápido hacia óptimos locales, tratando de superar esto se incorporó a la técnica un componente que diera un empuje y permitiera explorar otras áreas del espacio de soluciones, el cual fue referenciado como Mecanismo de Mutación Inteligente (MMI), lo que dio pie a trabajar con una técnica hibrida.

Una vez se desarrolló y puso a punto el algoritmo por medio de un diseño factorial de experimentos se llevaron a cabo una serie experimentos y a partir de los resultados obtenidos se realizó una minuciosa revisión de la literatura, también se consultó a un experto en el área de micología, para concluir que el AGH-CHIP tuvo una convergencia hacia la creación de dos grupos los cuales corresponden a hongos levaduras y hongos mohos respectivamente.

Los resultados estadísticos obtenidos muestran que el algoritmo es sensible a la variabilidad de los parámetros en términos de las respuestas (valor de adaptabilidad y tiempo de ejecución).

Dicho lo anterior la mejor combinación de parámetros en el menor tiempo es la que corresponde al experimento con los parámetros: Número de Generaciones: 50, Tamaño de la Población: 500, Máximo Número de Bits a Mutar: 5, Máximo Número de Grupos: 5, en donde se obtuvo el mejor valor de adaptabilidad que corresponde a 3.96373715208847 en un tiempo promedio de las réplicas de 3108.1659949045 milisegundos lo cual es muy bueno considerando la magnitud del espacio de soluciones del problema.

Además se logró la creación de una serie de tablas con información biológica básica de la lista de hongos que están considerados en la investigación además de su relación con la clasificación obtenida lo cual aporta valor y conocimiento con respecto a los resultados.

Metaheurística UMDA-CHIP.

Esta técnica fue inspirada en una nueva familia de algoritmos evolutivos llamados EDA propuesto Larrañaga (Larrañaga & Lozano, 2002) al igual que el Algoritmo Genético trabaja con poblaciones, pero con la diferencia que sustituye el cruce y mutación por la estimación basada en probabilidades. Dentro del conjunto de algoritmos para estimar y muestrear la población fue seleccionado el Univariate Marginal Distribution Algorithm (UMDA, por sus siglas en inglés).

Se diseñó un marco de trabajo robusto y flexible para acoplar la técnica a la solución del problema de clusterización de hongos mediante su información proteómica.

Después de que algoritmo fue desarrollado y puesto a punto mediante un diseño factorial de experimentos se procedió a realizar una serie de experimentos y a partir de los resultados que se obtuvieron se revisó la literatura existente para contrastar la clasificación que se propone. Y se logró establecer que a partir de una clasificación basada en 4 filos (Ascomycota, Basidiomycota, Chtridiomycota, Zygomycota) el UMDA-CHIP fue capaz de acertar en 30 de los 33 hongos.

Con el propósito de fundamentar los resultados de forma estadística se llevaron a cabo una serie de pruebas y los resultados de dichas pruebas muestran que el algoritmo es sensible a la variabilidad de los parámetros en términos de las respuestas (valor de adaptabilidad y tiempo de ejecución), por lo que se establece que la mejor combinación de parámetros donde se obtuvo el mejor valor de adaptabilidad que fue de 8.9920772379418 en el menor tiempo promedio posible que corresponde a 11958.63624570393 milisegundos, es la siguiente: Número de Generaciones: 100, Tamaño de la Población: 1000, %Truncamiento: 30, Máximo Número de Grupos: 5, con un tiempo promedio.

3. Principales Contribuciones de la Tesis.

Las principales aportaciones al área de conocimiento de este trabajo de tesis fueron las siguientes:

1. Un marco de trabajo sólido y flexible para poder atacar el problema de la clusterización de hongos mediante su información proteómica.
2. Un Mecanismo de Mutación Inteligente (MMI) el cual aporta una ayuda al operador de mutación en el Algoritmo Genético y permite explorar mejor el espacio de soluciones.
3. La flexibilidad de ambos algoritmos de trabajar con una matriz de semejanzas proteómicas de cualquier ente biológico.
4. La metaheurística evolutiva “Algoritmo Genético Hibrido para la Clusterización de Hongos mediante su Información Proteómica (AGH-CHIP)” que en su parte hibrida integra el MMI para robustecer la búsqueda de soluciones.
5. La metaheurística evolutiva “Univariate Marginal Distribution Algorithm para la Clusterización de Hongos mediante su Información Proteómica (UMDA-CHIP)”.
6. La flexibilidad de ambos algoritmos de trabajar con una matriz de semejanzas proteómicas de cualquier ente biológico.
7. Resultados veraces a partir de una matriz de semejanzas proteómicas de hongos real los cuales fueron contrastados con la literatura existente.

4. Conclusiones de los Objetivos de la Investigación.

En seguida se enuncian los objetivos que se propusieron y la forma en como se les dio cumplimiento junto con las publicaciones surgidas, las cuales pueden ser consultadas en el ANEXO G correspondiente.

Objetivo General.

El objetivo general de esta investigación es buscar una alternativa rápida, eficiente y confiable para clusterizar hongos a partir de su información proteómica mediante el diseño, desarrollo y ajuste de los parámetros de dos algoritmos metaheurísticos (una de ellas híbrida).

En el caso de este objetivo se diseñaron dos técnicas metaheurísticas una de ellas hibrida (AGH) y otra basada en un EDA para atacar el problema de la clusterización de hongos mediante su información proteómica.

Objetivos Específicos.

- **Diseño e implementación de un Algoritmo Genético Híbrido.** Se dio cumplimiento con el diseño, implementación y ajuste de parámetros de un algoritmo denominado AGH-CHIP, posteriormente los resultados fueron presentados en
    - "Quinto Congreso Internacional: La investigación en el posgrado" en la Universidad Autónoma de Aguascalientes bajo el título "Aplicación de un Algoritmo Evolutivo Hibrido para la clasificación de hongos" (Rincón Miranda et al., 2014).
    - Posteriormente fue modificado y remitido en co-participación al "Seventh International Workshop on Hybrid Intelligent Systems" dentro de MICAI 2014 bajo el título "Fungi clustering using a Hybrid Evolutionary Algorithm" (Torres Soto, Torres Soto, & Rincón Miranda, 2014).
    - Nuevamente se le dio un enfoque diferente y se envió artículo (estado «aceptado para participación») al "17vo Seminario de Investigación" de la

Universidad Autónoma de Aguascalientes con el título “Algoritmo Genético Híbrido para la Clusterización de Hongos mediante su Información Proteómica.”.

- **Diseño e implementación de un Algoritmo de Estimación de Distribución (EDA).** Se dio cumplimiento con el diseño, implementación y ajuste de los parámetros de un algoritmo denominado UMDA-CHIP, posteriormente los resultados fueron presentados en:
    - “Sexto Congreso Internacional: La investigación en el posgrado” en la Universidad Autónoma de Aguascalientes bajo el título “Clusterización de Hongos por Medio de un EDA” (Rincón Miranda, Torres Soto, & Torres Soto, 2015).
    - Luego fue enviado y aceptado en CISCI 2016 (estado «Aceptado para participación») con el título “Clusterización de Hongos en base a su Información Proteómica por Medio de un EDA-UMDA”.
- **Realización del diseño de experimentos factoriales para el ajuste de los parámetros de ambos algoritmos.** Esto se logró con el diseño factorial de experimentos en cada algoritmo.
- **Análisis de resultados de los algoritmos metaheurísticos.** Se revisó a detalle la literatura contra los resultados obtenidos, además se consultó a un experto en el área para concluir que el AGH-CHIP tuvo una asertividad del 100% para clasificar la lista de hongos en dos grupos denominados levaduras y mohos; mientras que el UMDA-CHIP empató 30 de los 33 hongos con una clasificación basada en 4 filos.

5. Limitaciones.

Algunas de las limitaciones de la investigación fueron:

- Aunque los algoritmos están diseñados para ejecutarse sobre una matriz de semejanzas que cumpla con las especificaciones señaladas en el trabajo, sólo se probó con una matriz en particular, sería interesante probar con una matriz distinta (otros entes biológicos) o incluso de mayor dimensión para analizar los resultados aunque esto se ve planteado como trabajo futuro.
- Si bien cada una de las técnicas reportó muy buenos resultados, estos fueron distintos entre sí por lo que no se pudo llevar a cabo un análisis y comparativa entre las técnicas propuestas.
- Otra limitante fue encontrar una clasificación que hiciera sentido con los resultados reportados por los algoritmos.

6. Trabajo Futuro.

Algunas de las premisas que se espera que la investigación tome en un futuro próximo son las siguientes:

- Se pretende trabajar con una función multi-objetivo para el caso del AGH-CHIP.
- Mejorar la función de adaptabilidad de ambas técnicas con el fin de tratar de encontrar una clasificación más profunda, por ejemplo si el UMDA-CHIP se basó en 4 filos la idea es que empate más con las sub-clasificaciones del árbol filogenético o mejor aún una clasificación que permite diferenciar entre hongos patógenos y no.
- Incorporar un mecanismo de mejora en el UMDA-CHIP para poder hibridarla.
- Aplicar las técnicas (en su estado actual y mejorado) a una matriz de semejanzas de otros entres biológicos.
- Llevar a cabo un análisis y comparativa entre los dos algoritmos.
- Buscar la incorporación de al menos otra técnica metaheurística.

Glosario.

AG.	Son las siglas de "*Algoritmo Genético*".
AGH-CHIP.	Son las siglas de la técnica propuesta "Algoritmo Genético Híbrido para la Clusterización de Hongos mediante su Información Proteómica".
BBH	Siglas en inglés de "Bidirectional Best Hits"
Bioinformática.	Informática aplicada a la biología.
BLAST.	Son las siglas en inglés de "*Basic Local Allignment Search Tool, por sus siglas en inglés*".
Clúster.	Representa la división de datos en grupo de objetos similares (Mitra & Acharya, 2003)
Clusterización.	En inglés clustering. La meta principal es encontrar grupos que son diferentes de los otros, y que sus miembros sean similares entre sí.
Cromosoma.	Se denomina cromosoma, individuo o solución a cada miembro o cadena de la población.
Cruzamiento.	En biología este proceso se refiere a que los individuos al juntarse y crear descendencia, análogamente en el AG se refiere a combinar varias soluciones.
EDA.	Siglas en inglés de "*Estimation Distribution Algorithms*", Algoritmos de Estimación de Distribución.
Elitismo.	El mejor individuo de la población.
Función de Adaptabilidad.	Mecanismo que permite asignar una medida del nivel de desempeño de un individuo o posible solución.
Gen.	Se conoce con el nombre de gen a cada elemento del cromosoma que tiene significado po sí mismo.

Generación.	Se denomina generación a cada iteración del algoritmo en donde se crean los nuevos individuos
Hibridación.	Llevar a cabo la acción de combinar elementos de distinta naturaleza.
Matriz de Semejanzas Proteómicas.	Matriz cuadrada simétrica en donde sus celdas corresponden a la intersección de la semejanza (similitud) a nivel proteínico que existe entre ambos entes.
Metaheurística.	Métodos aproximados de solución de problemas.
MMI.	Siglas Mecanismo de Mutación Inteligente.
Mutación.	Cualquier cambio en la secuencia del cromosoma (solución).
Población.	Conjunto de individuos que representan las soluciones a optimizar.
Proteómica.	Es el área que estudia las proteínas, reacciones e interacción entre sí
Scatter Search.	Metaheurística, Búsqueda Dispersa
UMDA.	Son las siglas en inglés de "*Univariate Marginal Distribution Algorithm*". Algoritmo de Distribución Marginal Univariada.
UMDA-CHIP.	Siglas de la técnica propuesta "*Univariate Marginal Distribution Algorithm para la Clusterización de Hongos mediante su Información Proteómica"*.

Bibliografía.

Abdelmalik Moujahid, I. I., Inza, I., & Larranaga, P. (2015). Tema 3. Algoritmos de Estimación de Distribuciones.

Abraham, A., Jain, L. C., & Goldberg, R. (Eds.). (2005). Evolutionary multiobjective optimization: theoretical advances and applications. New York: Springer.

Agrawal, R., Gehrke, J., Gunopulos, D., & Raghavan, P. (1998). Automatic subspace clustering of high dimensional data for data mining applications (Vol. 27). ACM. Recuperado a partir de http://dl.acm.org/citation.cfm?id=276314

Alander, J. T. (1992). On optimal population size of genetic algorithms. En CompEuro'92.'Computer Systems and Software Engineering', Proceedings. (pp. 65–70). IEEE. Recuperado a partir de http://ieeexplore.ieee.org/xpls/abs_all.jsp?arnumber=218485

Alexopoulos, C. J., Mims, C. W., & Blackwell, M. (1996). Introductory mycology (4th ed). New York: Wiley.

Alineamiento de secuencias. (2015, diciembre 15). En Wikipedia, la enciclopedia libre. Recuperado a partir de https://es.wikipedia.org/w/index.php?title=Alineamiento_de_secuencias&oldid=87800997

Altamiranda, J., Aguilar, J., & Hernández, L. (2008). Sistema de Reconocimiento de Patrones en Bioinformática. En IFMBE PROCEEDINGS (Vol. 18, p. 573). SPRINGER SCIENCE+ BUSINESS MEDIA. Recuperado a partir de http://www.ing.ula.ve/~aguilar/publicaciones/objetos/congreso/BIO.pdf

Altschul, S. F., Gish, W., Miller, W., Myers, E. W., & Lipman, D. J. (1990). Basic local alignment search tool. Journal of Molecular Biology, 215(3), 403–410. http://doi.org/10.1016/S0022-2836(05)80360-2

Ankerst, M., Breunig, M. M., Kriegel, H.-P., & Sander, J. (1999). OPTICS: ordering points to identify the clustering structure. En ACM Sigmod Record (Vol. 28, pp. 49–60). ACM. Recuperado a partir de http://dl.acm.org/citation.cfm?id=304187

Babitsch Soler, I. (2010, septiembre). Bioinformática: identificar genes en una interfaz gráfica vía web para la comparación de genomas [info:eu-repo/semantics/bachelorThesis].

Recuperado el 6 de diciembre de 2014, a partir de http://www.recercat.net/handle/2072/114651

Bäck, T. (1996). Evolutionary algorithms in theory and practice. Recuperado a partir de http://158.69.150.236:1080/jspui/handle/961944/14020

Baker, J. E. (1987). Reducing bias and inefficiency in the selection algorithm. En Proceedings of the second international conference on genetic algorithms (pp. 14–21). Recuperado a partir de http://books.google.es/books?hl=es&lr=&id=MYJ_AAAAQBAJ&oi=fnd&pg=PA14&dq=Reducing+bias+and+ine%0Eciency+in+the+selection+algorithm.&ots=XvoNrr1xGw&sig=dUqIg-crW3mest2lpBRiF7NA_Ts

Baluja, S. (1994). Population-Based Incremental Learning. A Method for Integrating Genetic Search Based Function Optimization and Competitive Learning.

Baluja, S., & Davies, S. (1997). Using Optimal Dependency-Trees for Combinatorial Optimization: Learning the Structure of the Search Space. DTIC Document. Recuperado a partir de http://oai.dtic.mil/oai/oai?verb=getRecord&metadataPrefix=html&identifier=ADA322735

Batrachochytrium dendrobatidis. (2015, mayo 24). En Wikipedia, la enciclopedia libre. Recuperado a partir de https://es.wikipedia.org/w/index.php?title=Batrachochytrium_dendrobatidis&oldid=82689314

Benson, D. A., Karsch-Mizrachi, I., Lipman, D. J., Ostell, J., & Wheeler, D. L. (2003). GenBank. Nucleic acids research, 31(1), 23.

Berg, J. M., Stryer, L., & Tymoczko, J. L. (2007). Bioquímica. Reverté. Recuperado a partir de https://books.google.es/books?hl=es&lr=&id=HRr4MNH2YssC&oi=fnd&pg=PA1&dq=J.+Berg,+J.+Tymoczko,+y+Lubert+L.+Stryer.+Bioquimica.+Quinta+Edici%C3%B3n.+Reverte,+2003.&ots=LUuIL5Bn8B&sig=WU3b_q6J8y9niMNqHapx_bUm1Z4

Beyer, H.-G., & Schwefel, H.-P. (2002). Evolution strategies–A comprehensive introduction. Natural computing, 1(1), 3–52.

Blackwell, M., Hibbett, D. S., Taylor, J. W., & Spatafora, J. W. (2006). Research coordination networks: a phylogeny for kingdom Fungi (Deep Hypha). Mycologia, 98(6), 829–837.

Blum, C., & Roli, A. (2003). Metaheuristics in combinatorial optimization: Overview and conceptual comparison. ACM Computing Surveys (CSUR), 35(3), 268–308.

Blum, C., Roli, A., & Sampels, M. (2008). Hybrid metaheuristics: an emerging approach to optimization (Vol. 114). Springer Science & Business Media. Recuperado a partir de http://books.google.es/books?hl=es&lr=&id=SfSgnS5XzPwC&oi=fnd&pg=PA1&dq=%22First+International+Workshop+on+Hybrid+Metaheuristics%22+blum+roli+2004&ots=LcJF2dqEJC&sig=I6NcZ6lRfvBfXYeJPMEuGPBjSe8

Boender, C. G. E., Kan, A. R., Timmer, G. T., & Stougie, L. (1982). A stochastic method for global optimization. Mathematical programming, 22(1), 125–140.

Booker, L. (1982). Intelligent Behavior as a Adaptation to the Task Environment. Recuperado a partir de http://www.citeulike.org/group/664/article/431772

Bray, D., Hopkin, K., Johnson, A., Lewis, J., Raff, M., Roberts, K., & Walter, P. (2006). Introducción a la biología celular. Ed. Médica Panamericana, Buenos Aires, Madrid.

Brindle, A. (1981). Genetic algorithms for function optimization. Recuperado a partir de http://www.citeulike.org/group/712/article/431776

Bruns, T. D., Vilgalys, R., Barns, S. M., Gonzalez, D., Hibbett, D. S., Lane, D. J., … others. (1992). Evolutionary relationships within the fungi: analyses of nuclear small subunit rRNA sequences. Molecular phylogenetics and evolution, 1(3), 231–241.

Cagnina, M. L. C. (2010). Optimización Mono y Multiobjetivo a través de una Heurıstica de Inteligencia Colectiva. Recuperado a partir de http://delta.cs.cinvestav.mx/~ccoello/tesis/tesis-cagnina.pdf.gz

Carazo, J. M., & Trelles, O. (2007). Curso Bioinformática Clásica. Campus Virtual Andaluz. Recuperado a partir de http://www.bioscripts.net/col/index.php

Carrillo, L., Audisio, M. C., DEL VALLE, B., GÓMEZ, M., ANCASI, E., & BENÍTEZ, A. (2007). Manual de Microbiología de los Alimentos. Jujuy, 10, 102–116.

Castañón Olivares, L. R., García Yáñez, Y., & Gutiérrez Quiroz, M. (2013). Generalidades e Importancia de la Micología Médica. Departamento de Microbiología y Parasitología Facultad de Medicina, UNAM . Recuperado a partir de http://www.facmed.unam.mx/deptos/microbiologia/pdf/Generalidades_micol_med_2013.pdf

Charniak, E., Riesbeck, C. K., McDermott, D. V., & Meehan, J. R. (2014). Artificial intelligence programming. Psychology Press. Recuperado a partir de http://books.google.es/books?hl=es&lr=&id=jTelAgAAQBAJ&oi=fnd&pg=PP1&dq=Eugene+Charniak+and+Drew+McDermott+-+Introduction+to+Artificial+Intelligence&ots=2BYJdP6EAF&sig=kN0GUpUYow55mYO43X0zlgtqdek

Chire. (2011). Deutsch: DBSCAN-Clusteranalyse auf einem Datensatz mit Gauss-verteilten Clustern. Selbst mit sorgfältig gewählten Parametern minPts und ist DBSCAN nicht in der lage, alle Cluster zur gleichen Zeit korrekt zu erfassen, da die Dichte-Unterschiede der Cluster zu groß und die Trennung der Daten gering ist. OPTICS, eine DBSCAN-Erweiterung, ist überraschend gut in der Lage, diesen Datensatz zu trennen: siehe Beispiel. Visualisiert mit ELKI. Recuperado a partir de https://commons.wikimedia.org/wiki/File:DBSCAN-Gaussian-data.svg

Coello, C. A. C., Van Veldhuizen, D. A., & Lamont, G. B. (2002). Evolutionary algorithms for solving multi-objective problems (Vol. 242). Springer. Recuperado a partir de http://link.springer.com/content/pdf/10.1007/978-0-387-36797-2.pdf

Collins, F. S., Morgan, M., & Patrinos, A. (2003). The Human Genome Project: Lessons from Large-Scale Biology. Science, 300(5617), 286–290. http://doi.org/10.1126/science.1084564

Cooke, W. B. (1958). The ecology of the fungi. The Botanical Review, 24(6), 341–429. http://doi.org/10.1007/BF02872436

Cotta-Porras, C. (1998). A study of hybridisation techniques and their application to the design of evolutionary algorithms. AI Communications, 11(3, 4), 223–224.

Darwin, C. (1859). On the origin of species by means of natural selection, or the preservation of favoured races in the struggle for life. Murray, London.

Davis, L., & others. (1991). Handbook of genetic algorithms (Vol. 115). Van Nostrand Reinhold New York. Recuperado a partir de http://tocs.ulb.tu-darmstadt.de/28323289.pdf

De Bonet, J. S., Isbell, C. L., Viola, P., & others. (1997). MIMIC: Finding optima by estimating probability densities. Advances in neural information processing systems, 424–430.

De Jong, K. A. (1975). Analysis of the behavior of a class of genetic adaptive systems. Recuperado a partir de http://deepblue.lib.umich.edu/handle/2027.42/4507

De Jong, K. A., Spears, W. M., & Gordon, D. F. (1995). Using Markov chains to analyze GAFOs. Foundations of genetic algorithms, 3, 115–137.

De Silva, U. C., & Suzuki, J. (2005). On the stationary distribution of GAs with fixed crossover probability. En Proceedings of the 7th annual conference on Genetic and evolutionary computation (pp. 1147–1151). ACM. Recuperado a partir de http://dl.acm.org/citation.cfm?id=1068200

Deb, K. (2001). Multi-objective optimization using evolutionary algorithms (Vol. 16). John Wiley & Sons. Recuperado a partir de https://books.google.com.mx/books?hl=es&lr=&id=OSTn4GSy2uQC&oi=fnd&pg=PR15&dq=Multi-objective+optimization+using+evolutionary+algorithms.+Chichester,+New+York:+John+Wiley+%26+Sons.+2001.&ots=tEloqvMpb5&sig=AJEoNIG4Oc8Kf44sCsRMFbs-uh8

Ding, L., & Yu, J. (2005). Some theoretical results about the computation time of evolutionary algorithms. En Proceedings of the 7th annual conference on Genetic and evolutionary computation (pp. 1409–1415). ACM. Recuperado a partir de http://dl.acm.org/citation.cfm?id=1068234

Dorigo, M., Birattari, M., & Stützle, T. (2006). Ant colony optimization. Computational Intelligence Magazine, IEEE, 1(4), 28–39.

Dorigo, M., Maniezzo, V., & Colorni, A. (1996). Ant system: optimization by a colony of cooperating agents. Systems, Man, and Cybernetics, Part B: Cybernetics, IEEE Transactions on, 26(1), 29–41.

Droit, A., Poirier, G. G., & Hunter, J. M. (2005). Experimental and bioinformatic approaches for interrogating protein–protein interactions to determine protein function. Journal of Molecular Endocrinology, 34(2), 263–280. http://doi.org/10.1677/jme.1.01693

Duarte, A., Pantrigo, J. J., & Gallego, M. (2007). Metaheurísticas. Madrid: Dykinson.

Eberhart, R. C., Kennedy, J., & others. (1995). A new optimizer using particle swarm theory. En Proceedings of the sixth international symposium on micro machine and human science (Vol. 1, pp. 39–43). New York, NY. Recuperado a partir de http://www.ppgia.pucpr.br/~alceu/mestrado/aula3/PSO_2.pdf

Ebersberger, I., Gube, M., Strauss, S., Kupczok, A., Eckart, M., Voigt, K., … von Haeseler, A. (2009). A stable backbone for the fungi. Nature precedings: hdl, 10101. Recuperado a partir de http://www.researchgate.net/profile/Martin_Eckart/publication/36789883_A_stable_backbone_for_the_fungi/links/09e41505b447636461000000.pdf

Edna, H. V. (2006, agosto). Algoritmo de clustering basado en entropıa para descubrir grupos en atributos de tipo mixto. CENTRO DE INVESTIGACIÓN Y DE ESTUDIOS AVANZADOS (CINVESTAV) DEL INSTITUTO POLITÉCNICO NACIONAL, Mexico D. F. (Mexico). Recuperado a partir de http://webserver.cs.cinvestav.mx/TesisGraduados/2006/tesisEdnaHernandez.pdf.

El-Abd, M., & Kamel, M. (2005). A taxonomy of cooperative search algorithms. En Hybrid Metaheuristics (pp. 32–41). Springer. Recuperado a partir de http://link.springer.com/chapter/10.1007/11546245_4

Ester, M., Kriegel, H.-P., Sander, J., & Xu, X. (1996). A density-based algorithm for discovering clusters in large spatial databases with noise. En Kdd (Vol. 96, pp. 226–231). Recuperado a partir de http://www.aaai.org/Papers/KDD/1996/KDD96-037

Etxeberria, R., & Larranaga, P. (1999). Global optimization using Bayesian networks. En Second Symposium on Artificial Intelligence (CIMAF-99) (pp. 332–339). Habana, Cuba.

Feo, T. A., & Resende, M. G. (1995). Greedy randomized adaptive search procedures. Journal of global optimization, 6(2), 109–133.

Fleming, P. J., & Purshouse, R. C. (2002). Evolutionary algorithms in control systems engineering: a survey. Control engineering practice, 10(11), 1223–1241.

Fogel, D. B. (1995). Evolutionary Computation: Toward a New Philosophy of Machine Intelligence (Piscataway, NJ: IEEE).

Fogel, L. J. (1966). Artificial Intelligence Through Simulated Evolution.[By] Lawrence J. Fogel... Alvin J. Owens... Michael J. Walsh. John Wiley & Sons.

Franco, M. L., Cediel, J. F., & Payán, C. (2008). Brief history of bioinformatics. Colombia Médica, 39(1), 117–120.

García Martínez, C. (2008). Algoritmos genéticos locales. Universidad de Granada. Recuperado a partir de https://dialnet.unirioja.es/servlet/tesis?codigo=21362

García Sánchez, Á. (2007). Programación del transporte de hidrocarburos por oleoductos mediante la combinación de técnicas metaheurísticas y simulación. Industriales. Recuperado a partir de http://oa.upm.es/1262

Garey, M. R., & Johnson, D. S. (1979). A Guide to the Theory of NP-Completeness. WH Freemann, New York.

Garey, M. R., & Johnson, D. S. (2002). Computers and intractability (Vol. 29). wh freeman New York. Recuperado a partir de http://www.c.csce.kyushu-u.ac.jp/~makiyama/rinkou/NPcomplete/NP_shiryou.pdf

Gil-Garcia, R. J., & Badía, J. M. (2002). Algoritmos de Agrupamiento. Castellón: Departamento de Ingeniería y Ciencia de Computadores/Universidad Jaime I.

Glover, F. (1977). Heuristics for integer programming using surrogate constraints. Decision Sciences, 8(1), 156–166.

Glover, F., & Kochenberger, G. A. (2003). Handbook of Metaheuristics. Springer Science & Business Media.

Glover, F., & Laguna, M. (1997). Tabu search, 1997. Kluwer Academic Publishers.

Golberg, D. E. (1989). Genetic algorithms in search, optimization, and machine learning. Addion wesley, 1989.

Goldberg, D. E., & others. (1989). Genetic algorithms in search optimization and machine learning (Vol. 412). Addison-wesley Reading Menlo Park. Recuperado a partir de https://pdfs.semanticscholar.org/146b/b2ea1fbdd86f81cd0dae7d3fd63decac9f5c.pdf

Gómez, M. M. (2014). Las metaheurísticas: tendencias actuales y su aplicabilidad en la ergonomía. Ingeniería Industrial. Actualidad y Nuevas Tendencias, (12), 108–120.

González-Buitrago, J. M. (2006). Multiplexed testing in the autoimmunity laboratory. Clinical Chemical Laboratory Medicine, 44(10), 1169–1174.

González-Buitrago, J. M., Ferreira, L., & Lorenzo, I. (2007). Urinary proteomics. Clinica Chimica Acta, 375(1), 49–56.

Guarro, J. (2012). Taxonomía y biología de los hongos causantes de infección en humanos. Enfermedades Infecciosas y Microbiología Clínica, 30(1), 33–39. http://doi.org/10.1016/j.eimc.2011.09.006

Guarro, J., Gené, J., & Stchigel, A. M. (1999). Developments in fungal taxonomy. Clinical microbiology reviews, 12(3), 454–500.

Guha, S., Rastogi, R., & Shim, K. (1998). CURE: an efficient clustering algorithm for large databases. En ACM SIGMOD Record (Vol. 27, pp. 73–84). ACM. Recuperado a partir de http://dl.acm.org/citation.cfm?id=276312

Guha, S., Rastogi, R., & Shim, K. (1999). ROCK: A robust clustering algorithm for categorical attributes. En Data Engineering, 1999. Proceedings., 15th International Conference on (pp. 512–521). IEEE. Recuperado a partir de http://ieeexplore.ieee.org/xpls/abs_all.jsp?arnumber=754967

Han, J., & Kamber, M. (2000). Data Mining: Concepts and Techniques (The Morgan Kaufmann Series in Data Management Systems). Morgan Kaufmann. Recuperado a partir de http://www.amazon.ca/exec/obidos/redirect?tag=citeulike09-20&path=ASIN/1558604898

Hansen, P., & Mladenović, N. (2002). Developments of variable neighborhood search. Springer. Recuperado a partir de http://link.springer.com/chapter/10.1007/978-1-4615-1507-4_19

Harik, G. (1999). Linkage learning via probabilistic modeling in the ECGA. Urbana, 51(61), 801.

Harik, G. R. (1997). Learning gene linkage to efficiently solve problems of bounded difficulty using genetic algorithms. The University of Michigan. Recuperado a partir de http://www.leg.ufpr.br/~leonardo/artigos/harik97learning.pdf

Harik, G. R., Lobo, F. G., & Goldberg, D. E. (1999). The compact genetic algorithm. Evolutionary Computation, IEEE Transactions on, 3(4), 287–297.

Haugeland, J. (1989). Artificial intelligence: The very idea. MIT press. Recuperado a partir de http://books.google.es/books?hl=es&lr=&id=zLFSPdIuqKsC&oi=fnd&pg=PA15&dq=John+Haugeland+-+Artificial+Intelligence:+The+Very+Idea&ots=iLCUxdQDCg&sig=oa6o-YgowgfF2vgAlgyoGS4GGzI

Hauser, M., Mayer, C. E., & Söding, J. (2013). kClust: fast and sensitive clustering of large protein sequence databases. BMC Bioinformatics, 14, 248. http://doi.org/10.1186/1471-2105-14-248

Herrera, F. (2006). Introducción a los algoritmos metaheurísticos. Grupo de Investigación “Soft Computing and Intelligent Information Systems” Dpto. Ciencias de la Computación e IA Universidad de Granada. Recuperado a partir de http://sci2s.ugr.es/sites/default/files/files/Teaching/OtherPostGraduateCourses/Metaheuristicas/Int-Metaheuristicas-CAEPIA-2009.pdf

Hertz, J., Krogh, A., & Palmer, R. G. (1991). Introduction to the theory of neural computation (Vol. 1). Basic Books.

Heurística. (2016, marzo 13). En Wikipedia, la enciclopedia libre. Recuperado a partir de https://es.wikipedia.org/w/index.php?title=Heur%C3%ADstica&oldid=89799579

Hibbett, D. (2005, agosto 31). Teaching the Fungal Tree of Life-Home. Recuperado el 9 de febrero de 2016, a partir de http://www.clarku.edu/faculty/dhibbett/tftol/content/1introprogress.html

Hinneburg, A., & Keim, D. A. (1998). An efficient approach to clustering in large multimedia databases with noise. En KDD (Vol. 98, pp. 58–65). Recuperado a partir de http://www.aaai.org/Papers/KDD/1998/KDD98-009.pdf

Holland, J. H. (1975). Adaptation in natural and artificial systems: an introductory analysis with applications to biology, control, and artificial intelligence. U Michigan Press. Recuperado a partir de http://psycnet.apa.org/psycinfo/1975-26618-000

Hooke, R. (1961). 1665. Micrographia.

Huang, Z. (1997). A Fast Clustering Algorithm to Cluster Very Large Categorical Data Sets in Data Mining. En DMKD (p. 0). Recuperado a partir de http://grid.cs.gsu.edu/~wkim/index_files/papers/fastclusteringHuang.pdf

Huang, Z. (1998). Extensions to the k-means algorithm for clustering large data sets with categorical values. Data mining and knowledge discovery, 2(3), 283–304.

Jain, A. K., & Dubes, R. C. (1988). Algorithms for clustering data. Prentice-Hall, Inc. Recuperado a partir de http://dl.acm.org/citation.cfm?id=SERIES10022.42779

Jain, A. K., Murty, M. N., & Flynn, P. J. (1999). Data Clustering: A Review. ACM Comput. Surv., 31(3), 264–323. http://doi.org/10.1145/331499.331504

Jose, J. C. P., Davis, T. E., & Principe, J. C. (1993). A Markov Framework for the Simple Genetic Algorithm. En Evolutionary Computation. Citeseer. Recuperado a partir de http://citeseerx.ist.psu.edu/viewdoc/summary?doi=10.1.1.28.9951

Justel, A. (2012). TÉCNICAS DE ANÁLISIS MULTIVARIANTE PARA AGRUPACIÓN. Universidad Autónoma de Madrid.

Kan, A. R., & Timmer, G. T. (1989). Chapter IX Global optimization. Handbooks in operations research and management science, 1, 631–662.

Kanz, C., Aldebert, P., Althorpe, N., Baker, W., Baldwin, A., Bates, K., … others. (2005). The EMBL nucleotide sequence database. Nucleic Acids Research, 33(suppl 1), D29–D33.

Kargupta, H., & Buescher, K. (1996). The gene expression messy genetic algorithm for financial applications. En , Proceedings of the IEEE/IAFE 1996 Conference on

Computational Intelligence for Financial Engineering, 1996 (pp. 155–161). http://doi.org/10.1109/CIFER.1996.501840

Karypis, G., Han, E.-H., & Kumar, V. (1999). Chameleon: Hierarchical clustering using dynamic modeling. Computer, 32(8), 68–75.

Kemighan, B. W., & Ritchie, D. M. (1978). The C programming language. Bell Telephone Laboratories, Incorporated, Englewood Cliffs, NJ.

Kennedy, J., Kennedy, J. F., & Eberhart, R. C. (2001). Swarm intelligence. Morgan Kaufmann. Recuperado a partir de http://books.google.es/books?hl=es&lr=&id=vOx-QV3sRQsC&oi=fnd&pg=PR13&dq=%E2%80%9CSwarm+Intelligence%E2%80%9D&ots=-P43a49gru&sig=ZssJi7h8Pdl3ng4GFAkuDoLlMwc

Kirkpatrick, S., Vecchi, M. P., & others. (1983). Optimization by simmulated annealing. science, 220(4598), 671–680.

Kuri-Morales, A. (2004). Pattern recognition via vasconcelos' genetic algorithm. En Progress in Pattern Recognition, Image Analysis and Applications (pp. 328–335). Springer. Recuperado a partir de http://link.springer.com/chapter/10.1007/978-3-540-30463-0_40

Lampinen, J., Price, K., & Storn, R. (2005). Differential Evolution—a Practical Approach to Global Optimization. Springer, Berlin.

Larranaga, P., & Lozano, J. A. (2001). Estimation of distribution algorithms: a new tool for evolutionary optimization. Kluwer Academic Publishers, Boston, 51, 52.

Larranaga, P., Lozano, J. A., Mühlenbein, H., Informationstechnik, G. F., & Germany, S. A. (2003). Algoritmos de Estimación de Distribuciones en Problemas de Optimización Combinatoria. Inteligencia Artificial, Revista Iberoamericana de Inteligencia Artificial, 7(19), 149–168.

Larrañaga, P., & Lozano, J. A. (2002). Estimation of distribution algorithms: A new tool for evolutionary computation (Vol. 2). Springer Science & Business Media. Recuperado a partir de http://books.google.es/books?hl=es&lr=&id=o0llxS4u93wC&oi=fnd&pg=PR11&dq=P.+Larra%C3%B1aga+and+J.+A.+Lozano.+Estimation+of+distribution+algorithms.+A+new+tool&ots=KxtZvvllVk&sig=By1LAOGmNhjipM3eElRTrboo7dc

Ledesma Tamayo, Y., Tamayo, Y. L., & Pérez, O. M. (2014). Inferencia filogenética molecular en ambiente de alto Procesamiento computacional. Serie Científica, 7(2). Recuperado a partir de http://publicaciones.uci.cu/index.php/SC/article/view/1618

Levadura. (2016, abril 25). En Wikipedia, la enciclopedia libre. Recuperado a partir de https://es.wikipedia.org/w/index.php?title=Levadura&oldid=90684022

Levenick, J. R. (1991). Inserting Introns Improves Genetic Algorithm Success Rate: Taking a Cue from Biology. En Proceedings of the Fourth International Conference on Genetic Algorithms. Citeseer. Recuperado a partir de http://citeseerx.ist.psu.edu/viewdoc/summary?doi=10.1.1.46.9813

López, J. G. (2009). Optimización de sistemas de detección de intrusos en red utilizando técnicas computacionales avanzadas. Universidad Almería.

López-Muñoz, F., & González, C. Á. (2007). Historia de la Psicofarmacología (Vol. 2). Ed. Médica Panamericana. Recuperado a partir de https://books.google.es/books?hl=es&lr=&id=_DSADxtczAIC&oi=fnd&pg=PR5&dq=Historia+de+la+psicofarmacolog%C3%ADa&ots=31RWb1nQuA&sig=QWWQms9TRbil73REMx7_QfGKdB4

Lourenço, H. R., Martin, O. C., & Stützle, T. (2003). Iterated local search. Springer. Recuperado a partir de http://link.springer.com/chapter/10.1007/0-306-48056-5_11

Luger, G. F., & Stubblefield, W. A. (1993). Artificial intelligence: its roots and scope. Artificial intelligence: structures and strategies for, 1–34.

Luscombe, N. M., Greenbaum, D., Gerstein, M., & others. (2001). What is bioinformatics? A proposed definition and overview of the field. Methods of information in medicine, 40(4), 346–358.

Lutzoni, F., Kauff, F., Cox, C. J., McLaughlin, D., Celio, G., Dentinger, B., … Vilgalys, R. (2004). Assembling the fungal tree of life: progress, classification, and evolution of subcellular traits. American Journal of Botany, 91(10), 1446–1480. http://doi.org/10.3732/ajb.91.10.1446

Marczyk, A. (2004). Algoritmos genéticos y computación evolutiva. Departamento de Informática, universidad de Colorado. Recuperado a partir de http://the-geek.org/docs/algen/

Master, S. R. (2005). Diagnostic proteomics: back to basics? Clinical chemistry, 51(8), 1333–1334.

Meiyi, L., Zixing, C., & Guoyun, S. (2004). Genetic Algorithms with Fitness-& Diversity-guided Adaptive Operating Probabilities and Analyses of its Convergence. Recuperado a partir de http://citeseerx.ist.psu.edu/viewdoc/summary?doi=10.1.1.90.8388

Mendoza, A. (2011). Classifying Parasitic Fungi by their Proteome using a Univariate Estimation of Distribution Algorithm with a Simplified Design. En Springer-Verlag. Springer.

Michalewicz, Z., Algorithms, G., & Structures, D. (1996). Evolution Programs. Springer-Verlag, AI Series, New York.

Milenova, B. L., & Campos, M. M. (1997). Clustering large databases with numeric and nominal values using orthogonal projections. MA, USA. Recuperado a partir de http://citeseerx.ist.psu.edu/viewdoc/download?doi=10.1.1.125.5218&rep=rep1&type=pdf

Mills, P., & Tsang, E. (1999). Guided local search applied to the satisfiability (SAT) problem. En Proceedings of the 15th National Conference of the Australian Society for Operations Research (ASOR'99) (pp. 872–883).

Mitchell, M. (1998). An introduction to genetic algorithms. MIT press. Recuperado a partir de http://books.google.es/books?hl=es&lr=&id=0eznlz0TF-IC&oi=fnd&pg=PP9&dq=An+Introduction+to+Genetic+Algorithms&ots=sgkJ2X_cJe&sig=m9IclnZ9lf1WAIPnQ_wnNIDrg6w

Mitra, S., & Acharya, T. (2003). Data Mining: Concepts and Algorithms From Multimedia to Bioinformatics. John Wiley & Sons, Inc. Recuperado a partir de http://dl.acm.org/citation.cfm?id=861364

Mladenović, N., & Hansen, P. (1997). Variable neighborhood search. Computers & Operations Research, 24(11), 1097–1100.

Moho. (2016, mayo 6). En Wikipedia, la enciclopedia libre. Recuperado a partir de https://es.wikipedia.org/w/index.php?title=Moho&oldid=90895179

Mojica, T., Sánchez, O., & Bobadilla, L. (2003). La Proteómica, otra cara de la genómica. Nova, 1(1). Recuperado a partir de http://www.unicolmayor.edu.co/publicaciones/index.php/nova/article/view/3

Mollá Santiago, S. (2014). Generalització de mètodes de density-based clustering a dades mixtes. Recuperado a partir de http://upcommons.upc.edu/handle/2099.1/21766

Montes, B., Restrepo, A., & McEwen, J. G. (2003). Nuevos aspectos sobre la clasificación de los hongos y su posible aplicación médica. Biomédica, 23(2), 213–24. http://doi.org/10.7705/biomedica.v23i2.1214

Montgomery, D. C. (1991). Diseño y análisis de experimentos. Recuperado a partir de http://dialnet.unirioja.es/servlet/libro?codigo=368608

Morton, N. E. (1991). Parameters of the human genome. Proceedings of the National Academy of Sciences, 88(17), 7474–7476.

Moujahid, A., Inza, I., & Larrañaga, P. (2008). Tema 2. Algoritmos Genéticos. Departamento de Ciencias de la Computación e Inteligencia Artificial Universidad del País Vasco. Recuperado a partir de http://www.sc.ehu.es/ccwbayes/docencia/mmcc/docs/t2geneticos.pdf

Mühlenbein, H. (1997). The equation for response to selection and its use for prediction. Evolutionary Computation, 5(3), 303–346.

Mühlenbein, H., & Mahnig, T. (1999). FDA-A scalable evolutionary algorithm for the optimization of additively decomposed functions. Evolutionary computation, 7(4), 353–376.

Mühlenbein, H., & Paass, G. (1996). From recombination of genes to the estimation of distributions I. Binary parameters. En Parallel Problem Solving from Nature—PPSN IV (pp. 178–187). Springer. Recuperado a partir de http://link.springer.com/chapter/10.1007/3-540-61723-X_982

Muñiz, E. Z., Brugés, J. M., & Vaca, F. B. (2005). Diagnóstico precoz del cáncer mediante análisis proteómicos del suero:?` ficción o realidad? Medicina clínica, 124(5), 181–185.

Nehab, D. F., & Pacheco, M. A. C. (2004). Schemata theory for the real coding and arithmetical operators. En Proceedings of the 2004 ACM symposium on Applied computing (pp. 1006–1012). ACM. Recuperado a partir de http://dl.acm.org/citation.cfm?id=968105

Nemhauser, G. L., & Wolsey, L. A. (1988). The Scope of Integer and Combinatorial Optimization. Wiley-Interscience, New York, NY, USA, 0–471–82819–X. http://doi.org/10.2307/2583737

Newell, A., Shaw, J. C., & Simon, H. A. (1959). The processes of creative thinking. Rand Corporation Santa Monica, CA. Recuperado a partir de http://shelf1.library.cmu.edu/IMLS/BACKUP/MindModels.pre_Oct1/creativethinking.pdf

Ng, R. T., & Han, J. (2002). CLARANS: A method for clustering objects for spatial data mining. Knowledge and Data Engineering, IEEE Transactions on, 14(5), 1003–1016.

Nieto, J. J. (2005). Comparación de secuencias. Recuperado a partir de http://mathgene.usc.es/cursoverano/cv2005/materiales/Nieto/ComparacionSecuencias.pdf

Ordenamiento de burbuja. (2015, noviembre 10). En Wikipedia, la enciclopedia libre. Recuperado a partir de https://es.wikipedia.org/w/index.php?title=Ordenamiento_de_burbuja&oldid=86749059

Overbeek, R., Fonstein, M., D'Souza, M., Pusch, G. D., & Maltsev, N. (1999). The use of gene clusters to infer functional coupling. Proceedings of the National Academy of Sciences of the United States of America, 96(6), 2896–2901.

Pandre, A. (2011, agosto 7). Cluster Analysis: see it 1st. Recuperado a partir de https://apandre.wordpress.com/visible-data/cluster-analysis/

Papadimitriou, C. H., & Steiglitz, K. (1982). Combinatorial optimization: algorithms and complexity. Courier Corporation. Recuperado a partir de https://books.google.es/books?hl=es&lr=&id=cDY-joeCGoIC&oi=fnd&pg=PP1&dq=C.+H.+Papadimitriou+and+K.+Steiglitz.+Combinatorial+Optimization&ots=XkJ2urdbl8&sig=ok2Ks-z09_kmHjshBi7wOHAw6qw

Pardalos, P. M., & Resende, M. G. (2001). Handbook of applied optimization. Oxford university press.

Paszynska, A. (2005). An extension of vose's markov chain model for genetic algorithms. En Proceedings of the 7th annual conference on Genetic and evolutionary computation (pp. 1553–1554). ACM. Recuperado a partir de http://dl.acm.org/citation.cfm?id=1068255

Pelikan, M., Goldberg, D. E., & Cantú-Paz, E. (2000a). Bayesian Optimization Algorithm, Population Sizing, and Time to Convergence. En GECCO (pp. 275–282). Recuperado a partir de https://e-reports-ext.llnl.gov/pdf/238344.pdf

Pelikan, M., Goldberg, D. E., & Cantú-Paz, E. (2000b). Hierarchical Problem Solving and the Bayesian Optimization Algorithm. En GECCO (pp. 267–274). Recuperado a partir de http://martinpelikan.net/files/2000002.pdf

Pelikan, M., Goldberg, D. E., & Cantu-Paz, E. (2000). Linkage problem, distribution estimation, and Bayesian networks. Evolutionary computation, 8(3), 311–340.

Pelikan, M., & Mühlenbein, H. (1999). The bivariate marginal distribution algorithm. En Advances in Soft Computing (pp. 521–535). Springer. Recuperado a partir de http://link.springer.com/chapter/10.1007/978-1-4471-0819-1_39

Pelta, D. A. (2013). Algoritmos heurísticos en bioinformática. Recuperado a partir de http://digibug.ugr.es/handle/10481/24513

Pérez Rodríguez, R., & Hernández Aguirre, A. (2015, enero). UN ALGORITMO DE ESTIMACIÓN DE DISTRIBUCIONES PARA EL PROBLEMA DE SECUENCIAMIENTO EN CONFIGURACIÓN JOBSHOP FLEXIBLE. Comunicaciones CIMAT. Recuperado a partir de http://www.cimat.mx/reportes/enlinea/I-15-01.pdf

Petricoin III, E. F., Ardekani, A. M., Hitt, B. A., Levine, P. J., Fusaro, V. A., Steinberg, S. M., … Liotta, L. A. (2002). Use of proteomic patterns in serum to identify ovarian cancer. The Lancet, 359(9306), 572–577. http://doi.org/10.1016/S0140-6736(02)07746-2

Polya, G. (1945). How to solve it: A new aspect of mathematical model. Princeton University Press Princeton.

Ponce de León Sentí, E. E., Díaz Díaz, E., Martínez Guerra, J. J., Torres Soto, A., & Torres Soto, M. D. (2014). ÁRBOL FILOGENÉTICO DE HONGOS CONSTRUIDO POR MEDIO DE UN AGRUPAMIENTO JERARQUICO (Vol. 15). Presentado en 15 Seminario de Investigación, Unidad de Estudios Avanzados y Edificio Polivalente, Ciudad Universitaria, Aguascalientes, Ags.: Publicación Electrónica de Abstracts. Recuperado a partir de https://investigacion.uaa.mx/seminario/Memoria_Electronica/15seminario/ponencias/m_ing/EUNICE_ESTHER_PONCE_DE_LEON_SENTI.pdf

Popoff, O. (2007). Reino Fungi: Clasificación. Recuperado el 9 de febrero de 2016, a partir de http://www.biologia.edu.ar/fungi/fungiclas.htm

Raidl, G. R. (2006a). A unified view on hybrid metaheuristics. En Hybrid Metaheuristics (pp. 1–12). Springer. Recuperado a partir de http://link.springer.com/chapter/10.1007/11890584_1

Raidl, G. R. (2006b). A Unified View on Hybrid Metaheuristics. En F. Almeida, M. J. B. Aguilera, C. Blum, J. M. M. Vega, M. P. Pérez, A. Roli, & M. Sampels (Eds.), Hybrid Metaheuristics (pp. 1–12). Springer Berlin Heidelberg. Recuperado a partir de http://link.springer.com.dibpxy.uaa.mx/chapter/10.1007/11890584_1

Rechenberg, I. (1973). Evolution Strategy: Optimization of Technical systems by means of biological evolution. Fromman-Holzboog, Stuttgart, 104.

Reeves, C. R. (1993). Modern heuristic techniques for combinatorial problems. John Wiley & Sons, Inc. Recuperado a partir de http://dl.acm.org/citation.cfm?id=166648

Resende, M. G., & Ribeiro, C. C. (2010). Greedy randomized adaptive search procedures: Advances, hybridizations, and applications. En Handbook of metaheuristics (pp. 283–319). Springer. Recuperado a partir de http://link.springer.com/chapter/10.1007/978-1-4419-1665-5_10

Rich, E., & Knight, K. (1991). Artificial Intelligence (Edición: 2nd Revised edition). New York: McGraw-Hill Publishing Co.

Righetti, P. G., Castagna, A., Antonucci, F., Piubelli, C., Cecconi, D., Campostrini, N., … others. (2005). Proteome analysis in the clinical chemistry laboratory: myth or reality? Clinica Chimica Acta, 357(2), 123–139.

Rincón Miranda, J. I., Torres Soto, M. D., & Torres Soto, A. (2014). APLICACIÓN DE UN ALGORITMO EVOLUTIVO HÍBRIDO PARA LA CLASIFICACIÓN HONGOS. Presentado en Quinto Congreso Internacional: La investigación en el posgrado, Aguascalientes, Aguascalientes, México. Recuperado a partir de http://posgrados.dgip.uaa.mx/congreso/ciip-2014/index.php?option=com_content&view=article&id=109&Itemid=180

Rincón Miranda, J. I., Torres Soto, M. D., & Torres Soto, A. (2015). Clusterización de Hongos por Medio de un EDA. Presentado en Sexto Congreso Internacional: La investigación en el posgrado, Aguascalientes, Aguascalientes, México.

Romero-Zaliz, R. (2005). Reconocimiento de perfiles de regulación genética mediante algoritmos evolutivos multiobjeto. Recuperado a partir de http://digibug.ugr.es/handle/10481/716

Rousseeuw, P. J., & Kaufman, L. (1990). Finding Groups in Data. Wiley Online Library. Recuperado a partir de https://leseprobe.buch.de/images-adb/5c/cc/5ccc031f-49c1-452f-a0ac-22babc5e252e.pdf

Ruiz-Rodríguez, F.-J. (2012). Aplicación de flujos de carga probabilistas en sistemas fotovoltaicos. Recuperado a partir de http://ruja.ujaen.es/handle/10953/386

Russell, S. J., & Norvig, P. (1996). Inteligencia Artificial: un enfoque moderno. Recuperado a partir de http://dialnet.unirioja.es/servlet/libro?codigo=372367

Sait, S. M., & Youssef, H. (1999). Iterative computer algorithms with applications in engineering: solving combinatorial optimization problems. IEEE Computer Society Press. Recuperado a partir de http://dl.acm.org/citation.cfm?id=554663

San Miguel Hernández, Á., San Miguel, R., & Martín-Gil, F. J. (2010). Importancia de las aplicaciones clínicas de la Proteómica. Revista del Laboratorio Clínico, 3(1), 40–48. http://doi.org/10.1016/j.labcli.2009.06.004

Santiago, N. A. R. (2006). Una nueva propuesta para optimización multiobjetivo basada en búsqueda dispersa (Scatter Search). Recuperado a partir de http://www.cs.cinvestav.mx/TesisGraduados/2006/tesisNoel.pdf

Sarria Cerro, V. M. (2010). Metaheurísticas aplicadas al problema QAP. Estudio y experiencia computacional. Recuperado a partir de http://upcommons.upc.edu/handle/2099.1/10505

Sawicki, M. P., Samara, G., Hurwitz, M., & Passaro Jr., E. (1993). Human Genome Project. The American Journal of Surgery, 165(2), 258–264. http://doi.org/10.1016/S0002-9610(05)80522-7

Schaffer, J. D., & Morishima, A. (1987). An adaptive crossover distribution mechanism for genetic algorithms. En Genetic Algorithms and their Applications: Proceedings of the Second International Conference on Genetic Algorithms (pp. 36–40). Hillsdale, NJ: Lawrence Erlbaum Associates, Inc. Recuperado a partir de http://books.google.es/books?hl=es&lr=&id=MYJ_AAAAQBAJ&oi=fnd&pg=PA36&dq=An+adaptive+crossover+distribution+mechanism+for+genetic+algorithms,+en+Genetic+Algorithms+and+their+applications:+&ots=XvpFpj-CCA&sig=tBgQIhOeqXdqEa4sLDECFQ-lSPE

Schmitt, F., & Rothlauf, F. (2001). On the mean of the second largest eigenvalue on the convergence rate of genetic algorithms. Citeseer. Recuperado a partir de http://citeseerx.ist.psu.edu/viewdoc/download?doi=10.1.1.12.3026&rep=rep1&type=pdf

Shakya, S. K., McCall, J. A., & Brown, D. F. (2006). Solving the Ising spin glass problem using a bivariate EDA based on Markov random fields. En Evolutionary Computation, 2006. CEC 2006. IEEE Congress on (pp. 908–915). IEEE. Recuperado a partir de http://ieeexplore.ieee.org/xpls/abs_all.jsp?arnumber=1688408

Sheikholeslami, G., Chatterjee, S., & Zhang, A. (2000). WaveCluster: a wavelet-based clustering approach for spatial data in very large databases. The VLDB Journal, 8(3–4), 289–304.

Silver, E. A., Victor, R., Vidal, V., & de Werra, D. (1980). A tutorial on heuristic methods. European Journal of Operational Research, 5(3), 153–162.

Storn, R., & Price, K. (1997). Differential evolution–a simple and efficient heuristic for global optimization over continuous spaces. Journal of global optimization, 11(4), 341–359.

Syswerda, G. (1991). Schedule optimization using genetic algorithms. Handbook of genetic algorithms. Recuperado a partir de http://ci.nii.ac.jp/naid/10000000876/

Talbi, E.-G. (2002). A taxonomy of hybrid metaheuristics. Journal of heuristics, 8(5), 541–564.

Torres, M. D. (2010). Metaheurísticas híbridas en selección de subconjuntos de características para aprendizaje no supervisado. Tesis Doctoral. Universidad Autónoma de Aguascalientes, México (May 2010).

Torres Soto, M. D., Torres Soto, A., & Rincón Miranda, J. I. (2014). Fungi clustering using a Hybrid Evolutionary Algorithm. Presentado en 13th Mexican International Conference on Artificial Intelligence, MICAI 2014, Tuxtla Gutiérrez, Chiapas, México.

Turing, A. (1950). Introducci Introducción a la Inteligencia Artificial. Intelligence, 59, 433–460.

Vega, J. M. M., Batista, M. B. M., & Pérez, J. A. M. (2003). Metaheurísticas: Una visión global. Inteligencia artificial: Revista Iberoamericana de Inteligencia Artificial, 7(19), 7–28.

Wang, W., Yang, J., Muntz, R., & others. (1997). STING: A statistical information grid approach to spatial data mining. En VLDB (Vol. 97, pp. 186–195). Recuperado a partir de http://suraj.lums.edu.pk/~cs536a04/handouts/STING.pdf

Weiss, S. M., & Indurkhya, N. (1998). Predictive data mining: a practical guide. Morgan Kaufmann. Recuperado a partir de https://books.google.es/books?hl=es&lr=&id=xzVD8C2YpnQC&oi=fnd&pg=PR11&dq=Weiss,+S.M.+y+Indurkhya,+N.+%E2%80%9C*+%3F%0C%3F%09+%3F%25%3F%3F%3F%3F%3F%3F%3F%3F.+%3F%3F*%3F%0C%3F%0C%3F%3F%3F+6%3F%3F+%E2%80%9D,+Morgan+Kaufmann+Publishers,+San+Francisco,+1998.+&ots=IN87lEIeMM&sig=dJpbWW-YbGioKp85zgnuyABT4so

Whitley, L. D., & others. (1989). The GENITOR Algorithm and Selection Pressure: Why Rank-Based Allocation of Reproductive Trials is Best. En ICGA (Vol. 89, pp. 116–123). Recuperado a partir de http://www.cse.unt.edu/~sweany/3110.fall09/ranking89.pdf

Wolpert, D. H., & Macready, W. G. (1997). No free lunch theorems for optimization. Evolutionary Computation, IEEE Transactions on, 1(1), 67–82.

Zadeh, L. A. (1965). Fuzzy sets. Information and control, 8(3), 338–353.

Zanakis, S. H., & Evans, J. R. (1981). Heuristic “optimization”: Why, when, and how to use it. Interfaces, 11(5), 84–91.

Zanlungo, S., Arrese, M., & Rigotti, A. (1999). Medicina molecular: presente y futuro. Revista médica de Chile, 127(8), 982–988.

Zapico Muñiz, E., Mora Brugés, J., & Blanco Vaca, F. (2005). Diagnóstico precoz del cáncer mediante análisis proteómicos del suero: ¿ficción o realidad? Medicina Clínica, 124(5), 181–185. http://doi.org/10.1157/13071481

Zhang, T., Ramakrishnan, R., & Livny, M. (1996). BIRCH: an efficient data clustering method for very large databases. En ACM Sigmod Record (Vol. 25, pp. 103–114). ACM. Recuperado a partir de http://dl.acm.org/citation.cfm?id=233324

ANEXOS.

Esta sección corresponde a los anexos que se hacen mención en el documento de tesis.

ANEXO A.

En este anexo se incluye la matriz de semejanzas completa, la cual está dividida debido a la magnitud del espacio que ocupa.

	H1	H2	H3	H4	H5	H6
H1	1	0.456145155768572	0.432165377364623	0.424755355725998	0.388150118203310	0.311365897569527
H2	0.456145155768572	1	0.699493966523939	0.695904991869117	0.352205292702486	0.461409651392906
H3	0.432165377364623	0.699493966523939	1	0.684732570831949	0.335677257095930	0.451615287942633
H4	0.424755355725998	0.695904991869117	0.684732570831949	1	0.331044527673741	0.441948124650121
H5	0.388150118203310	0.352205292702486	0.335677257095930	0.331044527673741	1	0.258975681358353
H6	0.311365897569527	0.461409651392906	0.451615287942633	0.441948124650121	0.258975681358353	1
H7	0.698298850574713	0.460608327100474	0.437878968018072	0.426734287266800	0.368881469115192	0.324314734320944
H8	0.848887546561965	0.439544431201165	0.415994962216625	0.406119966711478	0.373120501674624	0.298278096648769
H9	0.698439556307577	0.469412285696211	0.449562858004221	0.440156805096166	0.373456371285113	0.323905150387736
H10	0.698516988924348	0.456676989192947	0.438166375519547	0.428466527964753	0.367222448149654	0.318835386338186
H11	0.682397959183674	0.453391142768659	0.430183773562607	0.420547287566011	0.360573096312019	0.000000000000000
H12	0.401211968185835	0.546190090904764	0.531552618293634	0.529679516953089	0.315514993481095	0.436375531566896
H13	0.326470266126383	0.353378003499637	0.342765087364203	0.335699373695198	0.300330918522493	0.278755888150202
H14	0.435584843492586	0.577860794337397	0.553167313701354	0.549872980875234	0.336809338521401	0.413990911122700
H15	0.487188019966722	0.457152830298447	0.437246062225191	0.426586853399593	0.389702233250620	0.325102359545819
H16	0.691311044667935	0.469579168209305	0.452896844088554	0.441136023916293	0.371539935257977	0.331689272503083
H17	0.374743611064915	0.553429851775250	0.550237638622530	0.540987061153960	0.309499074032251	0.449074385728711
H18	0.298306906745499	0.476668485179123	0.475506667138259	0.468337259941458	0.258684628774692	0.395740800658882
H19	0.349177814201872	0.531849514159954	0.532904263877715	0.525263200682899	0.294268456959110	0.432906094690034
H20	0.446008388426815	0.575067581617800	0.548615968821825	0.542951151607188	0.343479936147961	0.405548216644650
H21	0.871876555500249	0.456158801104246	0.432841420710355	0.425220873323587	0.373983739837398	0.312580585743231
H22	0.699448354574853	0.454377351272078	0.429621529676204	0.424145167263301	0.368003284521692	0.313545188190219
H23	0.379259259259259	0.528896518332673	0.521598501936417	0.510198812290214	0.304110854503464	0.433113172033811
H24	0.456808803301238	0.585214876452382	0.564650754136770	0.554676685782084	0.345171673819743	0.464690397161497
H25	0.227012062487641	0.243186445130361	0.239818135245902	0.235495431504859	0.217934320231411	0.209546487715121
H26	0.260085643452783	0.271331432331652	0.265605004976539	0.259462562336311	0.267148563819669	0.221880907372401
H27	0.823086823086823	0.432481990831696	0.407028913260219	0.401418709227943	0.361101330892191	0.293496233694654
H28	0.585735963581184	0.439085957220672	0.418266801414856	0.412894111603416	0.390990346800143	0.306179254035999
H29	0.347920997920998	0.494161989430128	0.479374280382737	0.474165596919127	0.283461868037704	0.568664876581152
H30	0.321182266009852	0.490635855610935	0.489472525860172	0.479946672591934	0.270234113712375	0.408718850421391
H31	0.500159795461809	0.637602487984167	0.607485240751774	0.611404087013843	0.370365912373616	0.435109769711002
H32	0.504626334519573	0.472301785605460	0.449758538430209	0.443643667296786	0.397262059973924	0.329099559163720
H33	0.648915187376726	0.500643264105863	0.480556692591077	0.467640422752642	0.397797299069097	0.353472070098576

H7	H8	H9	H10	H11	H12	H13
0.698298850574713	0.848887546561965	0.698439556307577	0.698516988924348	0.682397959183674	0.401211968185835	0.326470266126383
0.460608327100474	0.439544431201165	0.469412285696211	0.456676989192947	0.453391142768659	0.546190090904764	0.353378003499637
0.437878968018072	0.415994962216625	0.449562858004221	0.438166375519547	0.430183773562607	0.531552618293634	0.342765087364203
0.426734287266800	0.406119966711478	0.440156805096166	0.428466527964753	0.420547287566011	0.529679516953089	0.335699373695198
0.368881469115192	0.373120501674624	0.373456371285113	0.367222448149654	0.360573096312019	0.315514993481095	0.300330918522493
0.324314734320944	0.298278096648769	0.323905150387736	0.318835386338186	0.000000000000000	0.436375531566896	0.278755888150202
1	0.667780513542033	0.785128757141674	0.775159182998429	0.850422875553766	0.399861119148197	0.325465712400386
0.667780513542033	1	0.664391558149978	0.658416285534930	0.650396583282489	0.385702919395312	0.317607548376779
0.785128757141674	0.664391558149978	1	0.794028340080972	0.775989489242897	0.408824219666745	0.329531442663379
0.775159182998429	0.658416285534930	0.794028340080972	1	0.766442512711169	0.399882766705744	0.325462012320329
0.850422875553766	0.650396583282489	0.775989489242897	0.766442512711169	1	0.397767805776090	0.000000000000000
0.399861119148197	0.385702919395312	0.408824219666745	0.399882766705744	0.397767805776090	1	0.319928652505270
0.325465712400386	0.317607548376779	0.329531442663379	0.325462012320329	0.000000000000000	0.319928652505270	1
0.434814012278801	0.415518121490556	0.436465775172498	0.427987555664003	0.424648519294047	0.491543487326815	0.335819340860797
0.473586447730143	0.463689382439882	0.480260172439873	0.473334340534824	0.461209439528024	0.393140128079887	0.418305670152547
0.780057710804745	0.656147043523496	0.785521692948770	0.764912280701754	0.761071797726008	0.410823826176690	0.329658158384937
0.388027518225691	0.366206301251618	0.393014916064654	0.381783247650205	0.380305429286480	0.547187563919002	0.324436999813884
0.316347569955817	0.292950094203216	0.320724243454607	0.310312606184166	0.309310315930613	0.465891688709453	0.280367231638418
0.364190251572327	0.340958268933539	0.368878946060154	0.358054741443545	0.356524715200704	0.520320707768869	0.308013987526587
0.440861601960531	0.426943147486954	0.448228436701814	0.442263657180242	0.432882680848337	0.492160406388902	0.329096230288735
0.687739463601533	0.819768544868146	0.687827865208500	0.682056290508745	0.672075960293483	0.397422649079083	0.327168671506544
0.818204634819711	0.665516301879938	0.779959030385797	0.767814524377770	0.803550688568110	0.396335697399527	0.323784901758014
0.384749065248302	0.363162852551671	0.388011945392491	0.381820119352089	0.376951283394447	0.547503756887627	0.320973612374886
0.459102572125915	0.439316352995943	0.461948926438826	0.452214748064475	0.449626865671642	0.625131289983768	0.352991526149106
0.229980541835055	0.216895508494298	0.225091554347416	0.226464928498661	0.224715750232992	0.227193840522546	0.245844449004720
0.264561462919065	0.253769508861652	0.267156967546073	0.261846771781396	0.264025289778714	0.246651044034836	0.245654767663087
0.645407279029463	0.817890167956218	0.646730956383261	0.638749006095945	0.634996993385448	0.374750106015630	0.306440534685898
0.535499823383963	0.564053168946253	0.535582213403085	0.533369359632532	0.522363238512035	0.392241114725922	0.323339212228101
0.359495139997098	0.334701322389421	0.361217102044810	0.353798025222407	0.352261790182868	0.469001013803322	0.585869714915301
0.332512964753450	0.309462681092976	0.336456899231692	0.328939777215639	0.327805731787355	0.449911619349951	0.296738661623702
0.491150125403081	0.475831860447245	0.495261699956262	0.486384156108927	0.479652817302220	0.536306944297643	0.353762533970574
0.486315955516997	0.479338842975207	0.489471146117987	0.484829025525767	0.475586854460094	0.409044542672560	0.405461975480913
0.627967918684984	0.612307956398592	0.633052767886047	0.626576139670223	0.614621080825587	0.440318906605923	0.351051051051051

H14	H15	H16	H17	H18	H19	H20
0.435584843492586	0.487188019966722	0.691311044667935	0.374743611064915	0.298306906745499	0.349177814201872	0.446008388426815
0.577860794337397	0.457152830298447	0.469579168209305	0.553429851775250	0.476668485179123	0.531849514159954	0.575067581617800
0.553167313701354	0.437246062225191	0.452896844088554	0.550237638622530	0.475506667138259	0.532904263877715	0.548615968821825
0.549872980875234	0.426586853399593	0.441136023916293	0.540987061153960	0.468337259941458	0.525263200682899	0.542951151607188
0.336809338521401	0.389702233250620	0.371539935257977	0.309499074032251	0.258684628774692	0.294268456959110	0.343479936147961
0.413990911122700	0.325102359545819	0.331689272503083	0.449074385728711	0.395740800658882	0.432906094690034	0.405548216644650
0.434814012278801	0.473586447730143	0.780057710804745	0.388027518225691	0.316347569955817	0.364190251572327	0.440861601960531
0.415518121490556	0.463689382439882	0.656147043523496	0.366206301251618	0.292950094203216	0.340958268933539	0.426943147486954
0.436465775172498	0.480260172439873	0.785521692948770	0.393014916064654	0.320724243454607	0.368878946060154	0.448228436701814
0.427987555664003	0.473334340534824	0.764912280701754	0.381783247650205	0.310312606184166	0.358054741443545	0.442263657180242
0.424648519294047	0.461209439528024	0.761071797726008	0.380305429286480	0.309310315930613	0.356524715200704	0.432882680848337
0.491543487326815	0.393140128079887	0.410823826176690	0.547187563919002	0.465891688709453	0.520320707768869	0.492160406388902
0.335819340860797	0.418305670152547	0.329658158384937	0.324436999813884	0.280367231638418	0.308013987526587	0.329096230288735
1	0.418041556393941	0.441106342393896	0.483556228446463	0.413112417601995	0.462842771377576	0.567807735029789
0.418041556393941	1	0.481315615593569	0.391795568055084	0.323002810116419	0.369260827092152	0.422557203771545
0.441106342393896	0.481315615593569	1	0.397556008146639	0.323651105445731	0.372331090962270	0.446259892775083
0.483556228446463	0.391795568055084	0.397556008146639	1	0.652979404442433	0.722706788777439	0.474812527569475
0.413112417601995	0.323002810116419	0.323651105445731	0.652979404442433	1	0.725277489545011	0.399599361946804
0.462842771377576	0.369260827092152	0.372331090962270	0.722706788777439	0.725277489545011	1	0.452004757050629
0.567807735029789	0.422557203771545	0.446259892775083	0.474812527569475	0.399599361946804	0.452004757050629	1
0.431196147997973	0.483332013619447	0.685213807315817	0.376662376662377	0.302071070416394	0.355803708636410	0.439356772962660
0.427859472097459	0.467704993128722	0.765497317375155	0.377778940210284	0.306785164929072	0.356160272122455	0.435127104655002
0.476104598737601	0.388298400715208	0.394026421596783	0.546247084464497	0.468462549277267	0.522366522366522	0.468959920652811
0.533359633118004	0.449661294090166	0.465150858063317	0.571663137881865	0.478381334305505	0.542782195944539	0.532575243857910
0.242908715832904	0.276364419247192	0.232016662947259	0.233829084041549	0.207413228552718	0.224613083251827	0.235726701430673
0.262856324308838	0.287375348217530	0.267216009417304	0.255424191243342	0.222152530292231	0.247552270860969	0.263797732696897
0.408207070707071	0.457390618841151	0.000000000000000	0.357998289136014	0.287416815275543	0.335887220349372	0.414607656801521
0.418970814132104	0.486165690913465	0.528295316036914	0.365750757247945	0.295499451152580	0.344695795888854	0.435106089831910
0.449257376196005	0.358504398826979	0.368888249124322	0.477462917070718	0.414379084967320	0.458195493958283	0.445142641699133
0.298957640275006	0.338424201849450	0.342293593995462	0.479243264924126	0.422979096623301	0.464146531566641	0.425036614507053
0.681895371131197	0.466490066225166	0.496847772189559	0.520857995170226	0.432811146973156	0.495157550129587	0.626815186329970
0.431356381412333	0.580005787037037	0.492640760065415	0.400745854961447	0.326067136344799	0.378819327122653	0.439669838069435
0.471437873675487	0.501819769981074	0.628929831438652	0.423647314875740	0.344451838296456	0.397557074305657	0.478302185619259

H21	H22	H23	H24	H25	H26	H27
0.871876555500249	0.699448354574853	0.379259259259259	0.456808803301238	0.227012062487641	0.260085643452783	0.823086823086823
0.456158801104246	0.454377351272078	0.528896518332673	0.585214876452382	0.243186445130361	0.271331432331652	0.432481990831696
0.432841420710355	0.429621529676204	0.521598501936417	0.564650754136770	0.239818135245902	0.265605004976539	0.407028913260219
0.425220873323587	0.424145167263301	0.510198812290214	0.554676685782084	0.235495431504859	0.259462562336311	0.401418709227943
0.373983739837398	0.368003284521692	0.504110854503464	0.545171673819743	0.217934320231411	0.267148563819669	0.361101330892191
0.312580585743231	0.313545188190219	0.433113172033811	0.464690397161497	0.209546487715121	0.221880907372401	0.293496233694654
0.687739463601533	0.818204634819711	0.384749065248302	0.459102572125915	0.229980541835055	0.264561462919065	0.645407279029463
0.819768544868146	0.665516301879938	0.363162852551671	0.439316352995943	0.216895508494298	0.253769508861652	0.817890167956218
0.687827865208500	0.779959030385797	0.388011945392491	0.461948926438826	0.225091554347416	0.267156967546073	0.646730956383261
0.682056290508745	0.767814524377770	0.381820119352089	0.452214748064475	0.226464928498661	0.261846771781396	0.638749006095945
0.672075960293483	0.803550688568110	0.376951283394447	0.449626865671642	0.224715750232992	0.264025289778714	0.634996993385448
0.397422649079083	0.396335697399527	0.547503756887627	0.625131289983768	0.227193840522546	0.246651044034836	0.374750106015630
0.327168671506544	0.323784901758014	0.320973612374886	0.352991526149106	0.245844449004720	0.245654767663087	0.306440534685898
0.431196147997973	0.427859472097459	0.476104598737601	0.533359633118004	0.242908715832904	0.262856324308838	0.408207070707071
0.483332013619447	0.467704993128722	0.388298400715208	0.449661294090166	0.276364419247192	0.287375348217530	0.457390618841151
0.685213807315817	0.765497317375155	0.394026421596783	0.465150858063317	0.232016662947259	0.267216009417304	0.000000000000000
0.376662376662377	0.377778940210284	0.546247084464497	0.571663137881865	0.233829084041549	0.255424191243342	0.357998289136014
0.302071070416394	0.306785164929072	0.468462549277267	0.478381334305505	0.207413228552718	0.222152530292231	0.287416815275543
0.355803708636410	0.356160272122455	0.522366522366522	0.542782195944539	0.224613083251827	0.247552270860969	0.335887220349372
0.439356772962660	0.435127104655002	0.468959926652811	0.532575243857910	0.235726701430673	0.263797732696897	0.414607656801521
1	0.683089094668704	0.379205903408778	0.456003696613638	0.225303159032981	0.260682635781346	0.795331465919701
0.683089094668704	1	0.378033068499034	0.451786400307338	0.224936463983613	0.263422602415854	0.645674926200912
0.379205903408778	0.378033068499034	1	0.578176039551514	0.226593610587143	0.247862965774448	0.358276146033489
0.456003696613638	0.451786400307338	0.578176039551514	1	0.245417749843693	0.275202462530690	0.429595527786912
0.225303159032981	0.224936463983613	0.226593610587143	0.245417749843693	1	0.192143871273071	0.215260115606936
0.260682635781346	0.263422602415854	0.247862965774448	0.275202462530690	0.192143871273071	1	0.248840262582057
0.795331465919701	0.645674926200912	0.358276146033489	0.429595527786912	0.215260115606936	0.248840262582057	1
0.563750714694111	0.536167107543556	0.368131562864604	0.446060444325839	0.227403435144167	0.265087920827074	0.541421800947867
0.350546627034107	0.352790628998917	0.464429333918257	0.506408149852120	0.220581948758870	0.241958173122011	0.332278322029641
0.327038861521620	0.323226008127540	0.812755716004813	0.471327453726485	0.221666128511463	0.234441052137154	0.304094631483167
0.494323809523809	0.488597910843019	0.513523994183228	0.585811577752554	0.242481931958399	0.274609143083317	0.469160154768227
0.494725293273694	0.485793148238350	0.393407047638731	0.466470876162506	0.277529624570859	0.287798574346576	0.465854682906342
0.635042081101760	0.622956180510137	0.421216524807972	0.501537704514700	0.243380365144614	0.287411621972137	0.597004822743041

H28	H29	H30	H31	H32	H33
0.585735963581184	0.347920997920998	0.321182266009852	0.500159795461809	0.504626334519573	0.648915187376726
0.439085957220672	0.494161989430128	0.490635855610935	0.637602487984167	0.472301785605460	0.500643264105863
0.418266801414856	0.479374280382737	0.489472525860172	0.607485240751774	0.449758538430209	0.480556692591077
0.412894111603416	0.474165596919127	0.479946672591934	0.611404087013843	0.443643667296786	0.467640422752642
0.390990346800143	0.283461868037704	0.270234113712375	0.370365912373616	0.397262059973924	0.397797299069097
0.306179254035999	0.568664876581152	0.408718850421391	0.435109769711002	0.329099559163720	0.353472070098576
0.535499823383963	0.359495139997098	0.332512964753450	0.491150125403081	0.486315955516997	0.627967918684984
0.564053168946253	0.334701322389421	0.309462681092976	0.475831860447245	0.479338842975207	0.612307956398592
0.535582213403085	0.361217102044810	0.336456899231692	0.495261699956262	0.489471146117987	0.633052767886047
0.533369359632532	0.353798025222407	0.328939777215639	0.486384156108927	0.484829025525767	0.626576139670223
0.522363238512035	0.352261790182868	0.327805731787355	0.479652817302220	0.475586854460094	0.614621080825587
0.392241114725922	0.469001013803322	0.449911619349951	0.536306944297643	0.409044542672560	0.440318906605923
0.323339212228101	0.585869714915301	0.296738661623702	0.353762533970574	0.405461975480913	0.351051051051051
0.418970814132104	0.449257376196005	0.298957640275006	0.681895371131197	0.431356381412333	0.471437873675487
0.486165690913465	0.358504398826979	0.338424201849450	0.466490066225166	0.580005787037037	0.501819769981074
0.528295316036914	0.368888249124322	0.342293593995462	0.496847772189559	0.492640760065415	0.628929831438652
0.365750757247945	0.477462917070718	0.479243264924126	0.520857995170226	0.400745854961447	0.423647314875740
0.295499451152580	0.414379084967320	0.422979096623301	0.432811146973156	0.326067136344799	0.344451838296456
0.344695795888854	0.458195493958283	0.464146531566641	0.495157550129587	0.378819327122653	0.397557074305657
0.435106089831910	0.445142641699133	0.425036614507053	0.626815186329970	0.439669838069435	0.478302185619259
0.563750714694111	0.350546627034107	0.327038861521620	0.494323809523809	0.494725293273694	0.635042081101760
0.536167107543556	0.352790628998917	0.323226008127540	0.488597910843019	0.485793148238350	0.622956180510137
0.368131562864604	0.464429333918257	0.812755716004813	0.513523994183228	0.393407047638731	0.421216524807972
0.446060444325839	0.506408149852120	0.471327453726485	0.585811577752554	0.466470876162506	0.501537704514700
0.227403435144167	0.220581948758870	0.221666128511463	0.242481931958399	0.277529624570859	0.243380365144614
0.265087920827074	0.241958173122011	0.234441052137154	0.274609143083317	0.287798574346576	0.287411621972137
0.541421800947867	0.332278322029641	0.304094631483167	0.469160154768227	0.465854682906342	0.597004822743041
1	0.340189953781299	0.312902040066164	0.481912842267644	0.497219608178629	0.557614410066362
0.340189953781299	1	0.428612744625470	0.733232337143644	0.363048850028512	0.390113560454242
0.312902040066164	0.428612744625470	1	0.463537610166018	0.344824603140274	0.365215126123388
0.481912842267644	0.733232337143644	0.463537610166018	1	0.487150837988827	0.532246733174090
0.497219608178629	0.363048850028512	0.344824603140274	0.487150837988827	1	0.515820342645470
0.557614410066362	0.390113560454242	0.365215126123388	0.532246733174090	0.515820342645470	1

ANEXO B.

Archivos con un concentrado de resultados que fueron generados por el AGH-CHIP y mencionados en el documento (statistics.txt y resume.txt respectivamente).

Statistics.txt

Case	#Generations	#Chromosomes	#Max_Groups_Created	#Max_Bits_Mutated		Max_Fitness	Time(ms)
1	50	500	2	2		3.8106324	2959.43441
2	50	500	2	2		3.78855944	3813.73441
3	50	500	2	2		3.94536355	3822.47334
4	50	500	2	2		3.94315963	2982.14897
5	50	500	2	2		3.96373715	3048.82379
6	50	500	2	2		3.94170632	3309.58109
7	50	500	2	2		3.94315963	3519.33222
8	50	500	2	2		3.96373715	3247.7308
9	50	500	2	2		3.69755933	3382.29397
10	50	500	2	2		3.96373715	3496.45919
---	---	---	---	---	---	---	---
			2			3.89613517	3358.20122
1	50	500	2	5		3.92485943	3094.57026
2	50	500	2	5		3.78743802	3074.87637
3	50	500	2	5		3.77019288	3259.62957
4	50	500	2	5		3.79298524	3647.40041
5	50	500	2	5		3.96373715	2984.20326
6	50	500	2	5		3.91855727	3061.02553
7	50	500	2	5		3.94833783	3012.11717
8	50	500	2	5		3.96373715	2985.44018
9	50	500	2	5		3.91855727	2970.46983
10	50	500	2	5		3.94833783	2991.92736
---	---	---	---	---	---	---	---
			2			3.89367401	3108.16599
1	50	500	2	2		3.81141149	3097.35242
2	50	500	2	2		3.71661043	3063.6172
3	50	500	2	2		3.85714395	2968.39297
4	50	500	2	2		3.82567496	3227.2179
5	50	500	2	2		3.89817725	3126.20849
6	50	500	2	2		3.7848052	3246.94792
7	50	500	2	2		3.94170632	3016.95361
8	50	500	2	2		3.92485943	3046.42959

9	50	500	2	2	3.82971469	3326.46327
10	50	500	2	2	3.92485943	2969.30968
			2		3.85149631	3108.88931
1	50	500	2	5	3.91219387	3445.15169
2	50	500	2	5	3.94170632	3726.28477
3	50	500	2	5	3.79778575	4450.87447
4	50	500	2	5	3.76883433	3650.00234
5	50	500	2	5	3.81577211	3763.37606
6	50	500	2	5	3.94170632	4111.76321
7	50	500	2	5	3.78262576	3497.86648
8	50	500	2	5	3.92485943	3773.21479
9	50	500	2	5	3.68104388	4382.58955
10	50	500	2	5	3.94536355	3722.52638
			2		3.85118913	3852.36497
1	50	750	2	2	3.94833783	7778.71835
2	50	750	2	2	3.92485943	7361.93256
3	50	750	2	2	3.76738718	6083.10337
4	50	750	2	2	3.91219387	6331.96271
5	50	750	2	2	3.92934705	6275.96205
6	50	750	2	2	3.7749281	6086.05384
7	50	750	2	2	3.86556414	6063.98586
8	50	750	2	2	3.78262576	6041.30003
9	50	750	2	2	3.92485943	6112.82895
10	50	750	2	2	3.77842579	6059.72169
			2		3.86085286	6419.55694
1	50	750	2	5	3.76540454	6020.22348
2	50	750	2	5	3.94170632	6032.25238
3	50	750	2	5	3.81651203	6035.10515
4	50	750	2	5	3.91855727	6156.13811
5	50	750	2	5	3.91855727	5974.39777
6	50	750	2	5	3.92485943	6218.37018
7	50	750	2	5	3.73930014	5952.76044
8	50	750	2	5	3.82818626	6003.91645
9	50	750	2	5	3.94170632	6079.1664
10	50	750	2	5	3.92485943	6123.08601

			2		3.8719649	6059.54164
1	50	750	2	2	3.89605192	6125.51675
2	50	750	2	2	3.92079936	6088.91769
3	50	750	2	2	3.72900702	6099.78111
4	50	750	2	2	3.78044463	6217.44896
5	50	750	2	2	3.85714395	6160.54924
6	50	750	2	2	3.8813952	6220.6946
7	50	750	2	2	3.82971469	6145.27756
8	50	750	2	2	3.90295188	6386.6583
9	50	750	2	2	3.87021502	6112.2538
10	50	750	2	2	3.77019288	6175.27573
			2		3.84379165	6173.23737
1	50	750	2	5	3.81577211	6107.62508
2	50	750	2	5	3.9154081	6125.61692
3	50	750	2	5	3.94536355	6010.34984
4	50	750	2	5	3.74264555	6094.27222
5	50	750	2	5	3.79808423	6157.90625
6	50	750	2	5	3.91855727	6117.62844
7	50	750	2	5	3.94315963	6030.67513
8	50	750	2	5	3.72011795	6153.00125
9	50	750	2	5	3.91855727	6122.40618
10	50	750	2	5	3.82179772	6183.30978
			2		3.85394634	6110.27911
1	50	1000	2	2	3.96373715	10386.2864
2	50	1000	2	2	3.88422596	10442.8187
3	50	1000	2	2	3.92485943	10305.0415
4	50	1000	2	2	3.90033982	10234.7889
5	50	1000	2	2	3.75715144	10345.2109
6	50	1000	2	2	3.92485943	10260.2306
7	50	1000	2	2	3.82222011	10317.2157
8	50	1000	2	2	3.91458203	10186.75
9	50	1000	2	2	3.81577211	10352.2827
10	50	1000	2	2	3.82845134	10450.0772
			2		3.87361988	10328.0702
1	50	1000	2	5	3.96373715	10213.3593

2	50	1000	2	5	3.91219387	10163.8831
3	50	1000	2	5	3.94315963	10124.9519
4	50	1000	2	5	3.92485943	10701.5607
5	50	1000	2	5	3.76738718	10423.1958
6	50	1000	2	5	3.94833783	10281.0978
7	50	1000	2	5	3.94170632	10597.3594
8	50	1000	2	5	3.94170632	10182.9255
9	50	1000	2	5	3.80582133	10224.0289
10	50	1000	2	5	3.92485943	10078.9833
			2		3.90737685	10299.1346
1	50	1000	2	2	3.8106324	10408.9775
2	50	1000	2	2	3.83396448	11107.7482
3	50	1000	2	2	3.94170632	10445.912
4	50	1000	2	2	3.81141149	10690.9391
5	50	1000	2	2	3.86225845	10340.324
6	50	1000	2	2	3.72664022	10523.1924
7	50	1000	2	2	3.94833783	10416.4085
8	50	1000	2	2	3.87328801	10516.2541
9	50	1000	2	2	3.85089368	10653.3104
10	50	1000	2	2	3.89817725	10394.3602
			2		3.85573101	10549.7426
1	50	1000	2	5	3.91219387	10456.7943
2	50	1000	2	5	3.92485943	10824.0485
3	50	1000	2	5	3.79298524	10434.4476
4	50	1000	2	5	3.94170632	10356.0312
5	50	1000	2	5	3.80810688	10300.4304
6	50	1000	2	5	3.79298524	10506.2536
7	50	1000	2	5	3.91219387	10628.6031
8	50	1000	2	5	3.94170632	10327.0134
9	50	1000	2	5	3.91855727	10320.3201
10	50	1000	2	5	3.77019288	10422.8296
			2		3.87154873	10457.6772
1	100	500	2	2	3.76167293	5981.75322
2	100	500	2	2	3.96373715	5934.44633
3	100	500	2	2	3.91855727	5960.86962
4	100	500	2	2	3.82666999	5931.1387

5	100	500	2	2	3.96373715	5904.14232
6	100	500	2	2	3.92485943	5923.04307
7	100	500	2	2	3.92485943	5906.03075
8	100	500	2	2	3.78609211	6018.20121
9	100	500	2	2	3.91219387	5990.0886
10	100	500	2	2	3.96373715	5943.97389
			2		3.89461165	5949.36877
1	100	500	2	5	3.78743802	5906.73481
2	100	500	2	5	3.94833783	5876.53261
3	100	500	2	5	3.76883433	6179.2554
4	100	500	2	5	3.91219387	5874.33792
5	100	500	2	5	3.91855727	5872.38832
6	100	500	2	5	3.78262576	6135.79681
7	100	500	2	5	3.92485943	5968.29198
8	100	500	2	5	3.72221842	5920.26707
9	100	500	2	5	3.91855727	5926.84621
10	100	500	2	5	3.79298524	5891.91061
			2		3.84766074	5955.23617
1	100	500	2	2	3.79298524	6040.0278
2	100	500	2	2	3.79808423	6018.37486
3	100	500	2	2	3.91855727	5970.03303
4	100	500	2	2	3.91219387	5993.69551
5	100	500	2	2	3.91855727	6122.77729
6	100	500	2	2	3.74302217	5996.74943
7	100	500	2	2	3.96373715	6462.9978
8	100	500	2	2	3.94170632	5973.06397
9	100	500	2	2	3.91219387	6019.29773
10	100	500	2	2	3.96373715	6021.6632
			2		3.88647745	6061.86806
1	100	500	2	5	3.82971469	5930.21173
2	100	500	2	5	3.94170632	5956.25116
3	100	500	2	5	3.96373715	5960.2764
4	100	500	2	5	3.91219387	5971.18251
5	100	500	2	5	3.91855727	5914.76311
6	100	500	2	5	3.94170632	5902.85244
7	100	500	2	5	3.94170632	5847.44622

8	100	500	2	5	3.94170632	5947.98024
9	100	500	2	5	3.77019288	5978.71325
10	100	500	2	5	3.94170632	5920.36519
			2		3.91029274	5933.00423
1	100	750	2	2	3.94833783	12135.8904
2	100	750	2	2	3.92485943	12355.3202
3	100	750	2	2	3.81141149	12129.2173
4	100	750	2	2	3.76883433	12163.7
5	100	750	2	2	3.96373715	12085.991
6	100	750	2	2	3.91855727	12028.9246
7	100	750	2	2	3.92485943	12094.9648
8	100	750	2	2	3.91855727	12155.6389
9	100	750	2	2	3.78743802	12099.9043
10	100	750	2	2	3.82987559	12228.5587
			2		3.87964678	12147.811
1	100	750	2	5	3.94170632	12132.3098
2	100	750	2	5	3.94170632	12072.7449
3	100	750	2	5	3.91219387	11880.0274
4	100	750	2	5	3.96373715	12275.0733
5	100	750	2	5	3.94170632	12228.8366
6	100	750	2	5	3.91219387	11985.0448
7	100	750	2	5	3.91949791	11931.5558
8	100	750	2	5	3.94315963	12001.6011
9	100	750	2	5	3.91219387	11861.8414
10	100	750	2	5	3.96373715	12159.7261
			2		3.93518324	12052.8761
1	100	750	2	2	3.79298524	12261.805
2	100	750	2	2	3.96373715	12171.0736
3	100	750	2	2	3.91219387	12290.3995
4	100	750	2	2	3.91219387	12287.5176
5	100	750	2	2	3.79298524	12261.6063
6	100	750	2	2	3.82971469	12342.5605
7	100	750	2	2	3.68293939	12172.9431
8	100	750	2	2	3.82567496	12178.5247
9	100	750	2	2	3.91219387	12314.2726
10	100	750	2	2	3.80416166	12279.7237

			2		3.842878	12256.0427
1	100	750	2	5	3.91855727	12722.3659
2	100	750	2	5	3.91219387	12344.1534
3	100	750	2	5	3.92485943	12071.3245
4	100	750	2	5	3.92485943	12146.6212
5	100	750	2	5	3.65901763	12190.8643
6	100	750	2	5	3.91219387	12189.6418
7	100	750	2	5	3.79808423	12139.3196
8	100	750	2	5	3.91219387	12192.7084
9	100	750	2	5	3.79298524	12501.1671
10	100	750	2	5	3.91219387	12136.305
			2		3.86671387	12263.4471
1	100	1000	2	2	3.92485943	20455.0282
2	100	1000	2	2	3.92485943	20534.1244
3	100	1000	2	2	3.79298524	20453.6135
4	100	1000	2	2	3.91855727	20345.3739
5	100	1000	2	2	3.92485943	20631.3877
6	100	1000	2	2	3.94170632	20517.7833
7	100	1000	2	2	3.94170632	20328.8829
8	100	1000	2	2	3.94315963	20978.0347
9	100	1000	2	2	3.91219387	20590.3599
10	100	1000	2	2	3.91855727	20541.5899
			2		3.91434442	20537.6178
1	100	1000	2	5	3.94170632	20283.6311
2	100	1000	2	5	3.94170632	20197.6106
3	100	1000	2	5	3.94170632	20158.2778
4	100	1000	2	5	3.91855727	20505.21
5	100	1000	2	5	3.96373715	20305.5768
6	100	1000	2	5	3.91219387	20266.3959
7	100	1000	2	5	3.91219387	20431.6038
8	100	1000	2	5	3.94170632	20340.0087
9	100	1000	2	5	3.96373715	20001.5493
10	100	1000	2	5	3.92485943	20191.9412
			2		3.9362104	20268.1805

1	100	1000	2	2	3.91219387	20673.226
2	100	1000	2	2	3.91219387	20566.7426
3	100	1000	2	2	3.91219387	20760.9026
4	100	1000	2	2	3.8106324	20578.6302
5	100	1000	2	2	3.96373715	20502.4324
6	100	1000	2	2	3.91219387	20962.7195
7	100	1000	2	2	3.94170632	20799.1344
8	100	1000	2	2	3.96373715	20469.2333
9	100	1000	2	2	3.94170632	20753.773
10	100	1000	2	2	3.79808423	20567.1654
			2		3.90683791	20663.3959
1	100	1000	2	5	3.77019288	20488.9145
2	100	1000	2	5	3.73746935	20492.4693
3	100	1000	2	5	3.94170632	20407.6199
4	100	1000	2	5	3.92485943	20595.3728
5	100	1000	2	5	3.94170632	20604.0268
6	100	1000	2	5	3.96373715	20772.4192
7	100	1000	2	5	3.82987559	20677.734
8	100	1000	2	5	3.91219387	21028.2477
9	100	1000	2	5	3.94170632	20589.0601
10	100	1000	2	5	3.92485943	20717.8422
			2		3.88883067	20637.3707
1	150	500	2	2	3.94170632	8937.83279
2	150	500	2	2	3.92485943	8898.66915
3	150	500	2	2	3.94315963	9421.44858
4	150	500	2	2	3.94170632	9177.54021
5	150	500	2	2	3.92485943	8904.9137
6	150	500	2	2	3.72900702	8921.37714
7	150	500	2	2	3.8106324	9165.62255
8	150	500	2	2	3.92485943	8986.35032
9	150	500	2	2	3.91855727	8807.97387
10	150	500	2	2	3.96373715	8938.0011
			2		3.90230844	9015.97294
1	150	500	2	5	3.94170632	8865.78578
2	150	500	2	5	3.96373715	9022.98469
3	150	500	2	5	3.91219387	8960.31499

4	150	500	2	5	3.94315963	8877.81797
5	150	500	2	5	3.91219387	8750.09873
6	150	500	2	5	3.94170632	8826.9674
7	150	500	2	5	3.94170632	8931.55005
8	150	500	2	5	3.94170632	8805.99512
9	150	500	2	5	3.91219387	8825.31625
10	150	500	2	5	3.94170632	9046.49979
			2		3.935201	8891.33308
1	150	500	2	2	3.91219387	8949.87524
2	150	500	2	2	3.91219387	8951.23573
3	150	500	2	2	3.94170632	8935.87374
4	150	500	2	2	3.76738718	9150.41944
5	150	500	2	2	3.91855727	8948.09354
6	150	500	2	2	3.82539381	8974.44088
7	150	500	2	2	3.6920182	8959.42373
8	150	500	2	2	3.96373715	8960.07319
9	150	500	2	2	3.91219387	8962.34054
10	150	500	2	2	3.96373715	8866.62819
			2		3.88091187	8965.84042
1	150	500	2	5	3.79808423	8892.77436
2	150	500	2	5	3.79298524	8932.74099
3	150	500	2	5	3.7894874	8866.88682
4	150	500	2	5	3.96373715	9034.45159
5	150	500	2	5	3.74758978	8875.02022
6	150	500	2	5	3.81141149	8963.44076
7	150	500	2	5	3.79808423	8881.33662
8	150	500	2	5	3.91219387	8931.05372
9	150	500	2	5	3.74178473	8806.75829
10	150	500	2	5	3.82999427	8934.59946
			2		3.81853524	8911.90628
1	150	750	2	2	3.94170632	18088.4801
2	150	750	2	2	3.94170632	18102.1618
3	150	750	2	2	3.96373715	18215.0954
4	150	750	2	2	3.91219387	18748.7345
5	150	750	2	2	3.94315963	18151.6354
6	150	750	2	2	3.81577211	18084.429

7	150	750	2	2	3.91219387	18068.1125
8	150	750	2	2	3.91855727	18034.1059
9	150	750	2	2	3.91219387	18079.4657
10	150	750	2	2	3.94170632	18226.3492
			2		3.92029267	18179.8569
1	150	750	2	5	3.94170632	18092.9364
2	150	750	2	5	3.91219387	18832.3544
3	150	750	2	5	3.96373715	17879.4309
4	150	750	2	5	3.94315963	18090.745
5	150	750	2	5	3.96373715	17804.8855
6	150	750	2	5	3.91219387	17753.5431
7	150	750	2	5	3.94170632	18519.7411
8	150	750	2	5	3.91219387	18100.3177
9	150	750	2	5	3.94833783	18319.6973
10	150	750	2	5	3.91219387	18294.1263
			2		3.93511599	18168.7778
1	150	750	2	2	3.7894874	18349.7682
2	150	750	2	2	3.91219387	18470.3385
3	150	750	2	2	3.77398087	18424.2997
4	150	750	2	2	3.94170632	18850.7202
5	150	750	2	2	3.91855727	18237.5817
6	150	750	2	2	3.91219387	18274.9082
7	150	750	2	2	3.96373715	18274.3154
8	150	750	2	2	3.80582133	18105.1229
9	150	750	2	2	3.96373715	18783.9485
10	150	750	2	2	3.91855727	18649.4999
			2		3.88999725	18442.0503
1	150	750	2	5	3.91219387	18192.3706
2	150	750	2	5	3.91855727	18066.5447
3	150	750	2	5	3.77143484	18330.0726
4	150	750	2	5	3.92485943	18324.1935
5	150	750	2	5	3.91855727	18268.2552
6	150	750	2	5	3.92485943	18351.5564
7	150	750	2	5	3.96373715	18177.8276
8	150	750	2	5	3.94170632	18188.9381
9	150	750	2	5	3.79991728	18202.33

10	150	750	2	5	3.96373715	18320.0311
			2		3.903956	18242.212
1	150	1000	2	2	3.94170632	30855.4306
2	150	1000	2	2	3.79097546	31097.1771
3	150	1000	2	2	3.81577211	31809.5129
4	150	1000	2	2	3.91219387	30800.2017
5	150	1000	2	2	3.94170632	30748.2349
6	150	1000	2	2	3.81651203	31184.0676
7	150	1000	2	2	3.92485943	30663.3433
8	150	1000	2	2	3.94315963	30784.9682
9	150	1000	2	2	3.92485943	30749.1479
10	150	1000	2	2	3.91219387	31450.0813
			2		3.89239385	31014.2166
1	150	1000	2	5	3.94170632	31064.3155
2	150	1000	2	5	3.96373715	31898.117
3	150	1000	2	5	3.91219387	30273.8306
4	150	1000	2	5	3.94170632	30508.8063
5	150	1000	2	5	3.96373715	30349.684
6	150	1000	2	5	3.91219387	31106.2334
7	150	1000	2	5	3.77618146	30377.639
8	150	1000	2	5	3.92485943	30493.8577
9	150	1000	2	5	3.94170632	30258.8061
10	150	1000	2	5	3.92485943	30473.1317
			2		3.92028813	30680.4421
1	150	1000	2	2	3.91219387	30959.2468
2	150	1000	2	2	3.91219387	31276.1599
3	150	1000	2	2	3.94833783	31149.1743
4	150	1000	2	2	3.96373715	31195.2742
5	150	1000	2	2	3.91219387	31179.7296
6	150	1000	2	2	3.92485943	31508.3637
7	150	1000	2	2	3.94315963	31680.2661
8	150	1000	2	2	3.91855727	31247.3186
9	150	1000	2	2	3.94170632	31333.3097
10	150	1000	2	2	3.94315963	31783.8939
			2		3.93200989	31331.2737

1	150	1000	2	5	3.91219387	31031.178
2	150	1000	2	5	3.94170632	31052.1388
3	150	1000	2	5	3.91855727	31200.0897
4	150	1000	2	5	3.8103031	30944.1418
5	150	1000	2	5	3.94170632	30858.6175
6	150	1000	2	5	3.8103031	31090.6633
7	150	1000	2	5	3.92485943	30899.2566
8	150	1000	2	5	3.94833783	30780.1026
9	150	1000	2	5	3.91219387	31123.0827
10	150	1000	2	5	3.81065721	30497.717

			2		3.89308183	30947.6988

Resume.txt

#Cases	#Generations	#Chromosomes	#Max_Groups_Created	#Max_Bits_Mutated	Max_Fitness_Found	Ocurr_Max_Fitness_Found	Avg_Max_Fitness	Avg_Time(ms)
10	50	500	2	2	3.963737152088470	30.00%	3.896135174844610	3358.2012182856
10	50	500	2	5	3.963737152088470	20.00%	3.893674006911820	3108.1659949045
10	50	500	2	2	3.941706315118290	10.00%	3.851496313793590	3108.8893058038
10	50	500	2	5	3.945363551206310	10.00%	3.851189130970570	3852.3649749865
10	50	750	2	2	3.948337826704610	10.00%	3.860852857723570	6419.5569407714
10	50	750	2	5	3.941706315118290	20.00%	3.871964899333020	6059.5416362533
10	50	750	2	2	3.920799356075520	10.00%	3.843791654699190	6173.2373735674
10	50	750	2	5	3.945363551206310	10.00%	3.853946337439860	6110.2791104003
10	50	1000	2	2	3.963737152088470	10.00%	3.873619882445820	10328.0702431399
10	50	1000	2	5	3.963737152088470	10.00%	3.907376847302590	10299.1345639896
10	50	1000	2	2	3.948337826704610	10.00%	3.855731012484240	10549.7426394218
10	50	1000	2	5	3.941706315118290	20.00%	3.871548731980830	10457.6771781227
10	100	500	2	2	3.963737152088470	30.00%	3.894611647919870	5949.3687707368
10	100	500	2	5	3.948337826704610	10.00%	3.847660742954150	5955.2361731808
10	100	500	2	2	3.963737152088470	20.00%	3.886477453554780	6061.8680625745
10	100	500	2	5	3.963737152088470	10.00%	3.910292743720380	5933.0042259847
10	100	750	2	2	3.963737152088470	10.00%	3.879646780314050	12147.8110187603
10	100	750	2	5	3.963737152088470	20.00%	3.935183239850680	12052.8761245413
10	100	750	2	2	3.963737152088470	10.00%	3.842877995218960	12256.0426572387
10	100	750	2	5	3.924859427723470	20.00%	3.866713871380230	12263.4471210018
10	100	1000	2	2	3.943159625953310	10.00%	3.914344418869040	20537.6178320625

10	100	1000	2	5	3.963737152088470	20.00%	3.936210400473220	20268.1805194176
10	100	1000	2	2	3.963737152088470	20.00%	3.906837905653120	20663.3959280458
10	100	1000	2	5	3.963737152088470	10.00%	3.888830665375070	20637.3706526014
10	150	500	2	2	3.963737152088470	10.00%	3.902308438350950	9015.9729412180
10	150	500	2	5	3.963737152088470	10.00%	3.935200997085230	8891.3330777107
10	150	500	2	2	3.963737152088470	20.00%	3.880911869071310	8965.8404224835
10	150	500	2	5	3.963737152088470	10.00%	3.818535239472000	8911.9062828167
10	150	750	2	2	3.963737152088470	10.00%	3.920292671816700	18179.8569470935
10	150	750	2	5	3.963737152088470	20.00%	3.935115987669680	18168.7777568864
10	150	750	2	2	3.963737152088470	20.00%	3.889997249585320	18442.0503127820
10	150	750	2	5	3.963737152088470	20.00%	3.903956000151760	18242.2119790696
10	150	1000	2	2	3.943159625953310	10.00%	3.892393846405700	31014.2165753513
10	150	1000	2	5	3.963737152088470	20.00%	3.920288131147720	30680.4421150121
10	150	1000	2	2	3.963737152088470	10.00%	3.932009885830640	31331.2736823869
10	150	1000	2	5	3.948337826704610	10.00%	3.893081829671880	30947.6988212073

ANEXO C.

Pruebas estadísticas del AGH-CHIP

Pruebas estadísticas sobre los datos

Pruebas de normalidad

	Kolmogorov-Smirnov[a]			Shapiro-Wilk		
	Estadístico	gl	Sig.	Estadístico	gl	Sig.
Fitness	.303	360	.000	.834	360	.000
Tiempo	.191	360	.000	.870	360	.000

a. Corrección de la significación de Lilliefors

Prueba de homogeneidad de varianzas

	Estadístico de Levene	gl1	gl2	Sig.
Fitness	3.273	35	324	.000
Tiempo	5.540	35	324	.000

Estadísticos de contraste[a,b]

	Fitness	Tiempo
Chi-cuadrado	65.271	355.311
Gl	35	35
Sig. asintót.	.001	.000

a. Prueba de Kruskal-Wallis
b. Variable de agrupación: Grupo

Estadísticos de contraste[a]

	Fitness	Tiempo
U de Mann-Whitney	42.000	.000
W de Wilcoxon	97.000	55.000
Z	-.617	-3.780
Sig. asintót. (bilateral)	.537	.000
Sig. exacta [2*(Sig. unilateral)]	.579[b]	.000[b]

a. Variable de agrupación: Grupo
b. No corregidos para los empates.

ANEXO D.

Archivos con un concentrado de resultados que fueron generados por el AGH-CHIP y mencionados en el documento (statistics.txt y resume.txt respectivamente).

Statistics.txt

Case	#Generations	#Chromosomes	#Max_Groups_Created	%Population Truncated	Max_Fitness	Time(ms)
1	50	1000	2	30	0.441848626391628	5077.5731267355
2	50	1000	2	30	0.435437410124341	5128.9030620204
3	50	1000	2	30	0.435437410124341	5174.0451609831
4	50	1000	2	30	0.434892876803426	5296.7004500991
5	50	1000	2	30	0.435437410124341	5796.3034728734
6	50	1000	2	30	0.435437410124341	5648.4510088230
7	50	1000	2	30	0.441848626391628	4975.1181079088
8	50	1000	2	30	0.439068119063384	5652.1001614182
9	50	1000	2	30	0.440216436269326	5349.2865904624
10	50	1000	2	30	0.435437410124341	5522.4684183475
1	50	1000	2	50	0.435437410124341	4489.0254475949
2	50	1000	2	50	0.435980959820314	4115.9579918453
3	50	1000	2	50	0.435437410124341	4433.3324575477
4	50	1000	2	50	0.435980959820314	4149.5257620567
5	50	1000	2	50	0.435437410124341	4672.2901685942
6	50	1000	2	50	0.402806316303232	4248.6354164443
7	50	1000	2	50	0.435437410124341	4577.1211392220
8	50	1000	2	50	0.433482357917095	5628.9248198208
9	50	1000	2	50	0.435437410124341	4819.1947314923
10	50	1000	2	50	0.435437410124341	5330.4496721551
1	50	1000	5	30	8.423498475441320	7270.9786248085
2	50	1000	5	30	6.409321062297800	7192.1829560065
3	50	1000	5	30	6.448421416978420	6448.1252986566
4	50	1000	5	30	8.254166301752200	6847.5833241650
5	50	1000	5	30	6.726870563149740	6590.5547335942
6	50	1000	5	30	6.545174880296790	6321.3762588735
7	50	1000	5	30	6.546840258472420	6758.7326779172
8	50	1000	5	30	6.133031503760710	6506.1968678620
9	50	1000	5	30	6.267452285570250	6498.9851833485
10	50	1000	5	30	6.141618195109620	6741.4643738598

1	50	1000	5	50	4.897873395340520	5086.3345413702
2	50	1000	5	50	5.839091821104490	5381.7459444281
3	50	1000	5	50	4.932157745862870	4981.3868175645
4	50	1000	5	50	4.352271331100740	5266.7354432105
5	50	1000	5	50	5.225613612653410	5239.8912438401
6	50	1000	5	50	3.951535507281160	6214.5746062659
7	50	1000	5	50	4.689225235055790	5789.5088812878
8	50	1000	5	50	5.332392149826670	5491.2792272943
9	50	1000	5	50	3.715842904808060	5292.7290357435
10	50	1000	5	50	4.865231200506460	5392.1609518915
1	50	1500	2	30	0.435980959820314	10064.1075114495
2	50	1500	2	30	0.435437410124341	11328.3863758401
3	50	1500	2	30	0.439354358392059	10306.4775024344
4	50	1500	2	30	0.441848626391628	10738.1592408252
5	50	1500	2	30	0.435437410124341	11544.1478085379
6	50	1500	2	30	0.441848626391628	12222.3807303067
7	50	1500	2	30	0.435980959820314	12046.2755572050
8	50	1500	2	30	0.435980959820314	10221.1298620473
9	50	1500	2	30	0.441848626391628	10919.2625501456
10	50	1500	2	30	0.435980959820314	11820.8832189065
1	50	1500	2	50	0.435437410124341	9546.7308289147
2	50	1500	2	50	0.433482357917095	10387.7297500562
3	50	1500	2	50	0.441848626391628	9538.6923479869
4	50	1500	2	50	0.441848626391628	9990.0501168353
5	50	1500	2	50	0.441848626391628	9767.2346410456
6	50	1500	2	50	0.435980959820314	9455.6177819524
7	50	1500	2	50	0.435980959820314	9210.3843823330
8	50	1500	2	50	0.435980959820314	8988.0093993698
9	50	1500	2	50	0.437404212548374	9973.3413577196
10	50	1500	2	50	0.435437410124341	10649.9687180304
1	50	1500	5	30	6.276012548399730	14913.4339227918
2	50	1500	5	30	7.962555122232610	16016.8803583715
3	50	1500	5	30	6.707817228090670	15422.4272837890
4	50	1500	5	30	6.178703214202120	14791.6481246418
5	50	1500	5	30	6.151869622515140	14608.5697817816
6	50	1500	5	30	7.409343480728010	15034.5854605346
7	50	1500	5	30	5.956536334155390	14669.5314437163
8	50	1500	5	30	7.415207227058230	14308.6327562453
9	50	1500	5	30	6.780030268567730	15004.1331610225
10	50	1500	5	30	6.249334141134550	14592.7744397572

1	50	1500	5	50	5.145812486167560	11403.4347766829
2	50	1500	5	50	4.528432526716680	11401.5328165103
3	50	1500	5	50	5.268864481624440	11313.0598528352
4	50	1500	5	50	4.494545140598810	11411.4802365278
5	50	1500	5	50	5.996965092715850	11655.6163040640
6	50	1500	5	50	4.710955130184280	11316.1589025530
7	50	1500	5	50	5.375625388293210	11495.9427860165
8	50	1500	5	50	4.602087192684420	11553.5984938676
9	50	1500	5	50	5.470750938063680	11986.3176277593
10	50	1500	5	50	6.040349070650540	11930.6648691166
1	50	2000	2	30	0.435437410124341	20253.9340566228
2	50	2000	2	30	0.433482357917095	21998.1037871709
3	50	2000	2	30	0.440216436269326	20959.2603661550
4	50	2000	2	30	0.440216436269326	16913.9343522005
5	50	2000	2	30	0.433482357917095	22204.1871039465
6	50	2000	2	30	0.435980959820314	18528.7215280708
7	50	2000	2	30	0.441133575261640	22079.0998674827
8	50	2000	2	30	0.435437410124341	20120.0726792638
9	50	2000	2	30	0.435437410124341	22189.6635669327
10	50	2000	2	30	0.440216436269326	19761.1215201888
1	50	2000	2	50	0.435980959820314	15020.7643310010
2	50	2000	2	50	0.441848626391628	17119.9716902280
3	50	2000	2	50	0.434892876803426	16431.7753379848
4	50	2000	2	50	0.435980959820314	15585.5824604213
5	50	2000	2	50	0.435437410124341	15515.8758048334
6	50	2000	2	50	0.435980959820314	15511.3994452992
7	50	2000	2	50	0.435980959820314	15569.7575606304
8	50	2000	2	50	0.435437410124341	15244.5309921393
9	50	2000	2	50	0.441848626391628	14417.0925174514
10	50	2000	2	50	0.435980959820314	14590.0653883480
1	50	2000	5	30	6.659062271532390	24516.5108862896
2	50	2000	5	30	6.349277495375650	23833.0877028196
3	50	2000	5	30	6.468502733677590	23642.6006236689
4	50	2000	5	30	6.687804089025600	24168.6972578603
5	50	2000	5	30	8.161791345226350	24519.3755264977
6	50	2000	5	30	7.504392633088100	24375.7461283431
7	50	2000	5	30	7.562527568416490	24050.4752232022
8	50	2000	5	30	6.178661684479840	23653.6264916409
9	50	2000	5	30	6.396630574152650	23687.0386730533
10	50	2000	5	30	6.107356495440490	23910.3952201807

1	50	2000	5	50	5.532629109431460	19042.5496365216
2	50	2000	5	50	3.910040469195950	18461.1922946187
3	50	2000	5	50	5.760535568448990	18004.0728139158
4	50	2000	5	50	5.642137824615220	19113.6114336009
5	50	2000	5	50	5.843559969216580	18654.4253723868
6	50	2000	5	50	5.879991061399490	18224.2576484826
7	50	2000	5	50	4.702370576472940	18771.9581363500
8	50	2000	5	50	4.727392391653580	18075.1350215197
9	50	2000	5	50	5.422995951502560	18246.0778485887
10	50	2000	5	50	5.746008794723200	18892.3366563926
1	100	1000	2	30	0.441848626391628	9901.8448151572
2	100	1000	2	30	0.441848626391628	8729.3428159027
3	100	1000	2	30	0.441848626391628	8753.7427521893
4	100	1000	2	30	0.441848626391628	8232.5802123890
5	100	1000	2	30	0.435437410124341	9827.6987472433
6	100	1000	2	30	0.441848626391628	9237.9613679991
7	100	1000	2	30	0.441848626391628	8016.5470124487
8	100	1000	2	30	0.435437410124341	9699.7850493533
9	100	1000	2	30	0.441848626391628	9175.5846279909
10	100	1000	2	30	0.435437410124341	9582.7436832590
1	100	1000	2	50	0.441848626391628	7510.5849646210
2	100	1000	2	50	0.435437410124341	8592.5215566434
3	100	1000	2	50	0.435437410124341	8974.6349205307
4	100	1000	2	50	0.435980959820314	7345.3566390849
5	100	1000	2	50	0.435437410124341	8520.2171017953
6	100	1000	2	50	0.435437410124341	8533.9483264557
7	100	1000	2	50	0.435437410124341	7808.7013138427
8	100	1000	2	50	0.435980959820314	8044.6474168975
9	100	1000	2	50	0.435437410124341	7612.8302054101
10	100	1000	2	50	0.435437410124341	8002.6716936764
1	100	1000	5	30	8.569475581789990	14005.9657425486
2	100	1000	5	30	8.992077237941800	12825.1842844639
3	100	1000	5	30	8.992077237941800	12830.1532734405
4	100	1000	5	30	8.811729408654520	12270.6144895456
5	100	1000	5	30	8.992077237941800	12191.1196974598
6	100	1000	5	30	8.992077237941800	12097.2421782760
7	100	1000	5	30	8.992077237941800	10703.9588523048
8	100	1000	5	30	8.931081606042660	11381.6975025329
9	100	1000	5	30	8.992077237941800	10296.0592107748
10	100	1000	5	30	8.992077237941800	10984.3672256916

1	100	1000	5	50	8.882690307399260	9651.4178696404
2	100	1000	5	50	8.931081606042660	9457.6338679458
3	100	1000	5	50	8.749836628117160	9522.4019133728
4	100	1000	5	50	8.475356873158590	9442.3410079527
5	100	1000	5	50	8.672547349709330	9449.7247022466
6	100	1000	5	50	8.538401573560320	9395.5174004930
7	100	1000	5	50	8.766635969035950	9598.2750580071
8	100	1000	5	50	8.843852252065900	9437.2332616831
9	100	1000	5	50	8.538780327659130	9322.9588309575
10	100	1000	5	50	8.843852252065900	9745.6911344763
1	100	1500	2	30	0.435437410124341	17165.2176518982
2	100	1500	2	30	0.434892876803426	13918.2091441877
3	100	1500	2	30	0.441848626391628	13449.1544836669
4	100	1500	2	30	0.441848626391628	13578.8703842674
5	100	1500	2	30	0.441848626391628	14004.8897577413
6	100	1500	2	30	0.441848626391628	13906.2645223054
7	100	1500	2	30	0.441848626391628	13305.0513401984
8	100	1500	2	30	0.441848626391628	13797.0383117917
9	100	1500	2	30	0.434892876803426	16038.1857607102
10	100	1500	2	30	0.441848626391628	13160.8131341578
1	100	1500	2	50	0.435437410124341	12297.3494893896
2	100	1500	2	50	0.441848626391628	13280.8390136245
3	100	1500	2	50	0.435980959820314	13647.4382447941
4	100	1500	2	50	0.435437410124341	15663.1379346018
5	100	1500	2	50	0.441848626391628	13079.7710750981
6	100	1500	2	50	0.435437410124341	15738.7023647855
7	100	1500	2	50	0.435980959820314	12902.4060021971
8	100	1500	2	50	0.435437410124341	13942.6846169888
9	100	1500	2	50	0.441848626391628	13980.0632864624
10	100	1500	2	50	0.435459718156003	12924.7923156375
1	100	1500	5	30	8.992077237941800	24725.9477779949
2	100	1500	5	30	8.992077237941800	24377.3787844204
3	100	1500	5	30	8.992077237941800	23053.4691786389
4	100	1500	5	30	8.931081606042660	24683.3928046544
5	100	1500	5	30	8.749836628117160	24101.6643485715
6	100	1500	5	30	8.992077237941800	21071.8422042212
7	100	1500	5	30	8.992077237941800	22386.7034387177
8	100	1500	5	30	8.992077237941800	20252.1823484302
9	100	1500	5	30	8.992077237941800	21428.6992776410
10	100	1500	5	30	8.992077237941800	20634.8885097467

1	100	1500	5	50	8.992077237941800	17831.8187714807
2	100	1500	5	50	8.766635969035950	18083.3065123970
3	100	1500	5	50	8.931081606042660	18437.1445883835
4	100	1500	5	50	8.992077237941800	18453.4161388690
5	100	1500	5	50	8.882690307399260	18362.1762398251
6	100	1500	5	50	8.992077237941800	18750.3953351275
7	100	1500	5	50	8.860311416013110	18405.1507692409
8	100	1500	5	50	8.717401894104110	18836.0475847199
9	100	1500	5	50	8.860311416013110	18347.1645891388
10	100	1500	5	50	8.676077961632920	19589.8063473662
1	100	2000	2	30	0.434892876803426	26317.4844862778
2	100	2000	2	30	0.435437410124341	24782.0246084827
3	100	2000	2	30	0.434892876803426	24120.9052230216
4	100	2000	2	30	0.441848626391628	23400.2720956621
5	100	2000	2	30	0.441848626391628	23365.3869522166
6	100	2000	2	30	0.435437410124341	25307.1996150922
7	100	2000	2	30	0.434892876803426	22660.9436809600
8	100	2000	2	30	0.445503296065505	22816.7640157182
9	100	2000	2	30	0.441848626391628	21169.8771053751
10	100	2000	2	30	0.441848626391628	27015.8462470668
1	100	2000	2	50	0.435437410124341	19778.0256889833
2	100	2000	2	50	0.435980959820314	17294.0644720572
3	100	2000	2	50	0.441848626391628	17604.2600952089
4	100	2000	2	50	0.441848626391628	20723.3290419835
5	100	2000	2	50	0.435980959820314	18616.7030938771
6	100	2000	2	50	0.435980959820314	17804.5866264243
7	100	2000	2	50	0.435437410124341	19487.9547996066
8	100	2000	2	50	0.435437410124341	20927.5461142211
9	100	2000	2	50	0.435980959820314	18629.0487982305
10	100	2000	2	50	0.435437410124341	19827.7985047054
1	100	2000	5	30	8.992077237941800	40412.9396512512
2	100	2000	5	30	8.992077237941800	37362.6848797245
3	100	2000	5	30	8.992077237941800	36280.4691310181
4	100	2000	5	30	8.882690307399260	35684.2331483783
5	100	2000	5	30	8.992077237941800	36154.8535494772
6	100	2000	5	30	8.992077237941800	36885.4156232501
7	100	2000	5	30	8.992077237941800	36518.7995605745
8	100	2000	5	30	8.992077237941800	35574.4100351902
9	100	2000	5	30	8.992077237941800	34179.7411889119
10	100	2000	5	30	8.882690307399260	33281.2101278045

1	100	2000	5	50	8.992077237941800	28356.9034626923
2	100	2000	5	50	8.992077237941800	29004.7193900591
3	100	2000	5	50	8.931081606042660	29112.3876599609
4	100	2000	5	50	8.931081606042660	29971.6705228605
5	100	2000	5	50	8.608987089768810	30228.7520710963
6	100	2000	5	50	8.882690307399260	29711.5859876481
7	100	2000	5	50	8.860311416013110	32012.8071339312
8	100	2000	5	50	8.992077237941800	30641.6096174404
9	100	2000	5	50	8.412137048617040	30951.3782950752
10	100	2000	5	50	8.779513291742390	29698.1909825823
1	150	1000	2	30	0.441848626391628	11206.7729979950
2	150	1000	2	30	0.441848626391628	10745.5938401615
3	150	1000	2	30	0.435437410124341	11765.1820183685
4	150	1000	2	30	0.441848626391628	9697.1084293824
5	150	1000	2	30	0.435437410124341	11024.8929762536
6	150	1000	2	30	0.433482357917095	11106.4210963632
7	150	1000	2	30	0.434892876803426	12697.7229846119
8	150	1000	2	30	0.441848626391628	10730.3424431465
9	150	1000	2	30	0.435437410124341	12298.6976519639
10	150	1000	2	30	0.435437410124341	12768.2560260897
1	150	1000	2	50	0.433482357917095	11039.3766923874
2	150	1000	2	50	0.435437410124341	10469.5661740920
3	150	1000	2	50	0.433482357917095	11248.6333638216
4	150	1000	2	50	0.441848626391628	10818.1745780218
5	150	1000	2	50	0.435437410124341	10953.5659803244
6	150	1000	2	50	0.435980959820314	10445.5303730683
7	150	1000	2	50	0.433482357917095	11938.7608234794
8	150	1000	2	50	0.435980959820314	10205.8776439833
9	150	1000	2	50	0.435437410124341	10804.3908062209
10	150	1000	2	50	0.435980959820314	10881.3781144444
1	150	1000	5	30	8.992077237941800	14292.3784477903
2	150	1000	5	30	8.811729408654520	13297.7739717592
3	150	1000	5	30	8.931081606042660	13532.6531215465
4	150	1000	5	30	8.860311416013110	13565.7853252257
5	150	1000	5	30	8.882690307399260	13937.3186507865
6	150	1000	5	30	8.811729408654520	17684.7056621186
7	150	1000	5	30	8.882690307399260	14539.3151137071
8	150	1000	5	30	8.992077237941800	14130.1781512274
9	150	1000	5	30	8.931081606042660	13702.1172392389
10	150	1000	5	30	8.992077237941800	13813.8702282681

1	150	1000	5	50	8.992077237941800	12941.0609924513
2	150	1000	5	50	8.992077237941800	12442.2129242976
3	150	1000	5	50	8.931081606042660	12771.4659174320
4	150	1000	5	50	8.608987089768810	13986.9896564238
5	150	1000	5	50	8.992077237941800	13499.1001302184
6	150	1000	5	50	8.992077237941800	13436.0066143713
7	150	1000	5	50	8.992077237941800	14679.5219688100
8	150	1000	5	50	8.460823644920300	15201.3438110142
9	150	1000	5	50	8.931081606042660	13365.4715202709
10	150	1000	5	50	8.444219147458720	12582.5581261690
1	150	1500	2	30	0.441848626391628	18795.0345415344
2	150	1500	2	30	0.435437410124341	20796.8141653954
3	150	1500	2	30	0.441848626391628	18644.1680063451
4	150	1500	2	30	0.435437410124341	20182.5914607612
5	150	1500	2	30	0.441848626391628	17355.1940385269
6	150	1500	2	30	0.441848626391628	17379.7044059135
7	150	1500	2	30	0.427242530077470	21787.1557546508
8	150	1500	2	30	0.441848626391628	17846.3808978007
9	150	1500	2	30	0.441848626391628	18665.6942708838
10	150	1500	2	30	0.445503296065505	19744.9119589081
1	150	1500	2	50	0.441848626391628	17898.0084633738
2	150	1500	2	50	0.435437410124341	16789.5051865670
3	150	1500	2	50	0.435437410124341	16617.9556042346
4	150	1500	2	50	0.435437410124341	18358.7532862489
5	150	1500	2	50	0.441848626391628	15762.5862717311
6	150	1500	2	50	0.435980959820314	16024.9192498239
7	150	1500	2	50	0.441848626391628	15982.5613282603
8	150	1500	2	50	0.435437410124341	17341.1770887899
9	150	1500	2	50	0.435437410124341	17135.3983812197
10	150	1500	2	50	0.435980959820314	16509.6157161929
1	150	1500	5	30	8.992077237941800	25868.7126114779
2	150	1500	5	30	8.931081606042660	28647.7366961314
3	150	1500	5	30	8.992077237941800	26622.4040481003
4	150	1500	5	30	8.992077237941800	25542.3222059290
5	150	1500	5	30	8.992077237941800	26110.9056663881
6	150	1500	5	30	8.882690307399260	26986.5877529036
7	150	1500	5	30	8.992077237941800	25725.8028628339
8	150	1500	5	30	8.811729408654520	24787.1183969181
9	150	1500	5	30	8.882690307399260	26628.8940304806
10	150	1500	5	30	8.992077237941800	29551.3213142697

1	150	1500	5	50	8.882690307399260	31722.1561733859
2	150	1500	5	50	8.600712231935930	29750.6617655511
3	150	1500	5	50	8.992077237941800	30951.0786121643
4	150	1500	5	50	8.688072077115240	35373.0584242098
5	150	1500	5	50	8.608987089768810	33262.6798713252
6	150	1500	5	50	8.811729408654520	28551.1903569429
7	150	1500	5	50	8.882690307399260	27289.9871423716
8	150	1500	5	50	8.749836628117160	25081.0445222069
9	150	1500	5	50	8.779513291742390	24885.8676107636
10	150	1500	5	50	8.931081606042660	24477.9018747834
1	150	2000	2	30	0.450243297574335	31856.9272731154
2	150	2000	2	30	0.441848626391628	39613.3708662232
3	150	2000	2	30	0.445503296065505	28252.5423784478
4	150	2000	2	30	0.445503296065505	27969.4368588633
5	150	2000	2	30	0.445503296065505	31185.9540672308
6	150	2000	2	30	0.434892876803426	32685.0390080414
7	150	2000	2	30	0.433482357917095	33464.6493217314
8	150	2000	2	30	0.435437410124341	31609.2709576881
9	150	2000	2	30	0.435437410124341	33965.6563384167
10	150	2000	2	30	0.450916233029988	28375.0252472589
1	150	2000	2	50	0.450916233029988	27681.7962747362
2	150	2000	2	50	0.435980959820314	24149.4789622597
3	150	2000	2	50	0.440216436269326	25040.9551592260
4	150	2000	2	50	0.433482357917095	26582.0638546283
5	150	2000	2	50	0.435437410124341	25884.1939022328
6	150	2000	2	50	0.441848626391628	27897.9226637459
7	150	2000	2	50	0.433482357917095	26507.9231235334
8	150	2000	2	50	0.434892876803426	26424.1108449088
9	150	2000	2	50	0.441848626391628	23728.7680815532
10	150	2000	2	50	0.441848626391628	23871.6068094526
1	150	2000	5	30	8.992077237941800	38316.9249413360
2	150	2000	5	30	8.992077237941800	43966.6050606181
3	150	2000	5	30	8.992077237941800	39808.3059787151
4	150	2000	5	30	8.992077237941800	40608.2282253681
5	150	2000	5	30	8.992077237941800	40507.8073556144
6	150	2000	5	30	8.992077237941800	41248.0812083214
7	150	2000	5	30	8.860311416013110	40517.5261134657
8	150	2000	5	30	8.992077237941800	40682.7922072591
9	150	2000	5	30	8.992077237941800	42554.5541949860
10	150	2000	5	30	8.992077237941800	41128.2798857757

1	150	2000	5	50	8.992077237941800	37587.5443571760
2	150	2000	5	50	8.717401894104110	38951.5626205916
3	150	2000	5	50	8.992077237941800	37327.1589074793
4	150	2000	5	50	8.992077237941800	35995.7748814816
5	150	2000	5	50	8.992077237941800	36673.1732068699
6	150	2000	5	50	8.992077237941800	40764.2365803635
7	150	2000	5	50	8.992077237941800	39639.1842384852
8	150	2000	5	50	8.992077237941800	37981.0411559057
9	150	2000	5	50	8.931081606042660	39174.1350658563
10	150	2000	5	50	8.992077237941800	39824.7762230758

Resume.txt

#Cases	#Generations	#Chromosomes	#Max_Groups_Created	%Population Truncated	Max_Fitness_Found	Ocurr_Max_Fitness_Found	Avg_Max_Fitness	Avg_Time(ms)
10	50	1000	2	30	0.441848626391628	20.00%	0.437506173554110	5362.0949559671
10	50	1000	2	50	0.435980959820314	20.00%	0.432087505460700	4646.4457606773
10	50	1000	5	30	8.423498475441320	10.00%	6.789639494282930	6717.6180299092
10	50	1000	5	50	5.839091821104490	10.00%	4.780123490354020	5413.6346692896
10	50	1500	2	30	0.441848626391628	30.00%	0.437969889709688	11121.1210357698
10	50	1500	2	50	0.441848626391628	30.00%	0.437525014934997	9750.7759324244
10	50	1500	5	30	7.962555122232610	10.00%	6.708740918708420	14936.2616732652
10	50	1500	5	50	6.040349070650540	10.00%	5.163438744769950	11546.7806665933
10	50	2000	2	30	0.441133575261640	10.00%	0.437104079009714	20500.8098828035
10	50	2000	2	50	0.441848626391628	20.00%	0.436936974893693	15500.6815528337
10	50	2000	5	30	8.161791345226350	10.00%	6.807600689041520	24035.7553733556
10	50	2000	5	50	5.879991061399490	10.00%	5.316766171666000	18548.5616862377
10	100	1000	2	30	0.441848626391628	70.00%	0.439925261511442	9115.7831083933
10	100	1000	2	50	0.441848626391628	10.00%	0.436187241690264	8094.6114138958
10	100	1000	5	30	8.992077237941800	70.00%	8.925682726207980	11958.6362457039
10	100	1000	5	50	8.931081606042660	10.00%	8.724303513881420	9502.3195046775
10	100	1500	2	30	0.441848626391628	70.00%	0.439816354847259	14232.3694490925
10	100	1500	2	50	0.441848626391628	30.00%	0.437471715746888	13745.7184343579
10	100	1500	5	30	8.992077237941800	80.00%	8.961753613769420	22671.6168673037
10	100	1500	5	50	8.992077237941800	30.00%	8.867074228406650	18509.6426876549
10	100	2000	2	30	0.445503296065505	10.00%	0.438845125229098	24095.6704029873
10	100	2000	2	50	0.441848626391628	20.00%	0.436937073256187	19069.3317235298
10	100	2000	5	30	8.992077237941800	80.00%	8.970199851833290	36233.4756895581
10	100	2000	5	50	8.992077237941800	30.00%	8.838203407945130	29969.0005123346
10	150	1000	2	30	0.441848626391628	40.00%	0.437751938078439	11404.0990464336

10	150	1000	2	50	0.441848626391628	10.00%	0.435655080997687	10880.5254549843
10	150	1000	5	30	8.992077237941800	30.00%	8.908754577403140	14249.6095911668
10	150	1000	5	50	8.992077237941800	50.00%	8.833657928394220	13490.5731661458
10	150	1500	2	30	0.445503296065505	10.00%	0.439471240474142	19119.7649500720
10	150	1500	2	50	0.441848626391628	30.00%	0.437469484943721	16842.0480576442
10	150	1500	5	30	8.992077237941800	60.00%	8.946065505714650	26647.1805585433
10	150	1500	5	50	8.992077237941800	10.00%	8.792739018611700	29134.5626353705
10	150	2000	2	30	0.450916233029988	10.00%	0.441876810016167	31897.7872317017
10	150	2000	2	50	0.450916233029988	10.00%	0.438995451105647	25776.8819676277
10	150	2000	5	30	8.992077237941800	90.00%	8.978900655748930	40933.9105171460
10	150	2000	5	50	8.992077237941800	80.00%	8.958510140368120	38391.8587237285

ANEXO E.

Pruebas estadistics del UMDA-CHIP

Pruebas no paramétricas

Estadísticos descriptivos

	N	Media	Desviación típica	Mínimo	Máximo
Tiempo	360	17890.2089	9554.51718	4115.96	43966.61
MilFitness	360	4170.8810	3892.81833	402.81	8992.08

Prueba de Kolmogorov-Smirnov para una muestra

		Tiempo	MilFitness
N		360	360
Parámetros normales[a,b]	Media	17890.2089	4170.8810
	Desviación típica	9554.51718	3892.81833
Diferencias más extremas	Absoluta	.104	.330
	Positiva	.104	.330
	Negativa	-.075	-.198
Z de Kolmogorov-Smirnov		1.969	6.268
Sig. asintót. (bilateral)		.001	.000

Pruebas no paramétricas

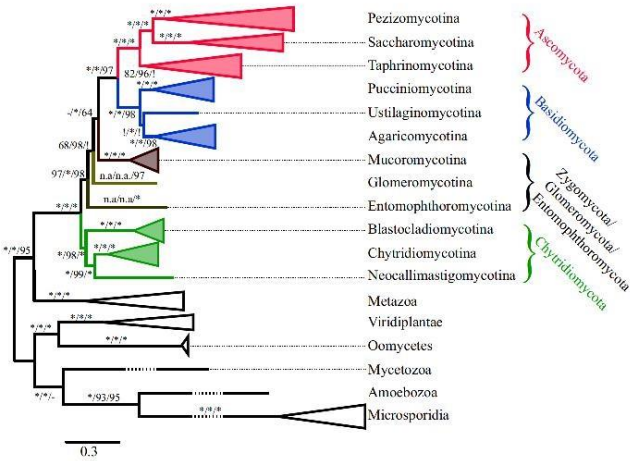
Prueba de Kruskal-Wallis

Estadísticos de contraste[a,b]

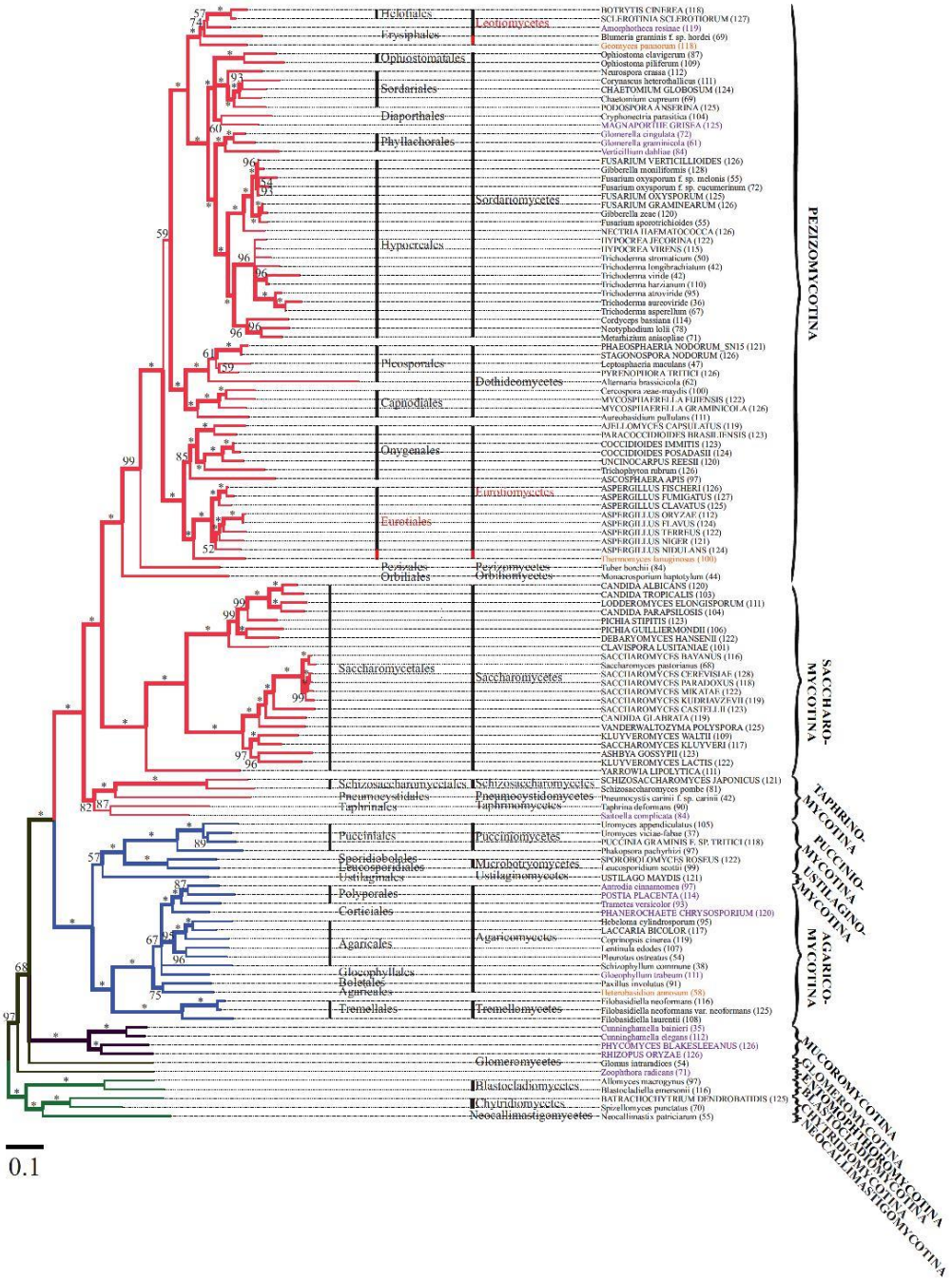
	Tiempo	MilFitness
Chi-cuadrado	353.909	314.014
gl	35	35
Sig. asintót.	.000	.000

ANEXO F.

Árbol filogenético que Ebersberger y sus colegas (Ebersberger et al., 2009) incluyen en su publicación. Puede consultar el artículo y árbol en la siguiente dirección: https://www.researchgate.net/profile/Martin_Eckart/publication/36789883_A_stable_backbone_for_the_fungi/links/09e41505b447636461000000.pdf



Nature Precedings : hdl:10101/npre.2009.2901.1 : Posted 24 Feb 2009

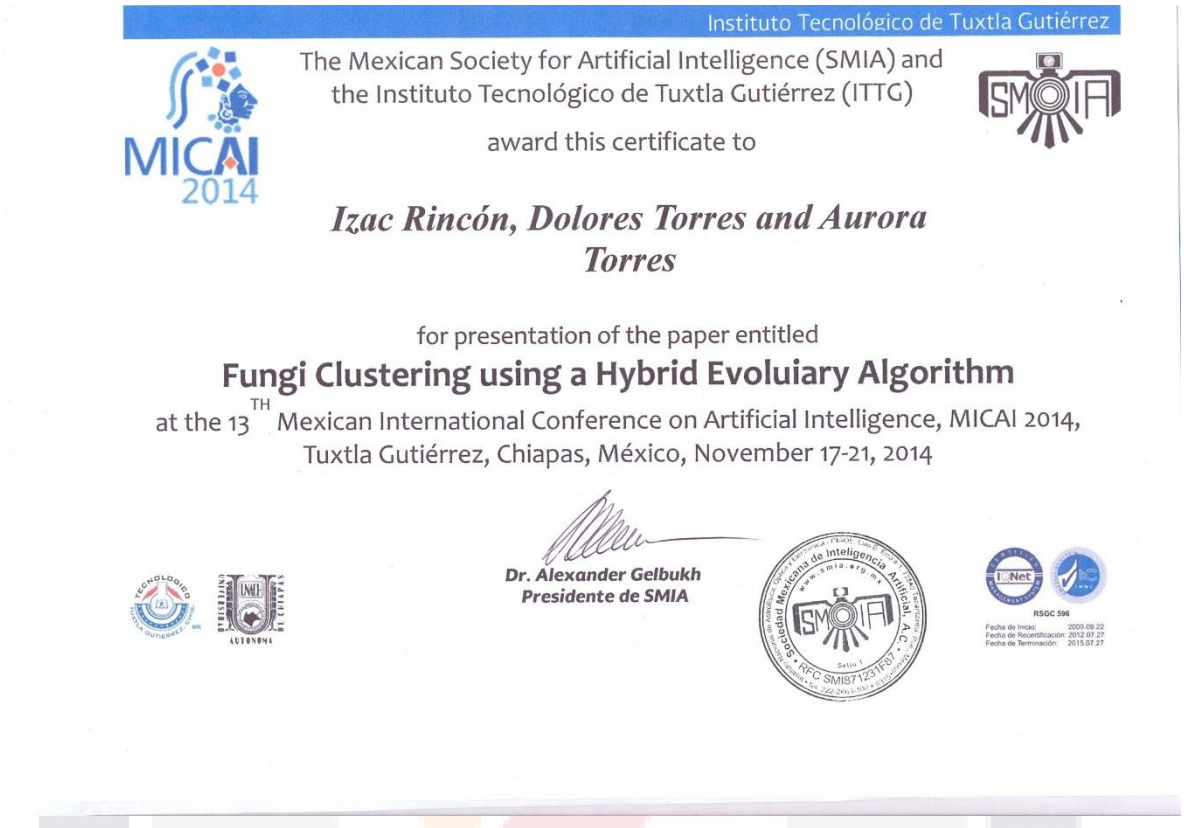


ANEXO G.

Diseño e implementación de un Algoritmo Genético Híbrido. Se dio cumplimiento con el diseño, implementación y ajuste de los parámetros de un algoritmo denominado AGH-CHIP, posteriormente los resultados fueron presentados en:

"Quinto Congreso Internacional: La investigación en el posgrado" en la Universidad Autónoma de Aguascalientes bajo el título "Aplicación de un Algoritmo Evolutivo Hibrido para la clasificación de hongos" (Rincón Miranda et al., 2014).

"Seventh International Workshop on Hybrid Intelligent Systems" dentro de MICAI 2014 bajo el título "Fungi clustering using a Hybrid Evolutionary Algorithm" (Torres Soto et al., 2014).



Instituto Tecnológico de Tuxtla Gutiérrez

The Mexican Society for Artificial Intelligence (SMIA) and the Instituto Tecnológico de Tuxtla Gutiérrez (ITTG)

award this certificate to

Izac Rincón, Dolores Torres and Aurora Torres

for presentation of the paper entitled

Fungi Clustering using a Hybrid Evoluiary Algorithm

at the 13TH Mexican International Conference on Artificial Intelligence, MICAI 2014, Tuxtla Gutiérrez, Chiapas, México, November 17-21, 2014

Dr. Alexander Gelbukh
Presidente de SMIA

"17vo Seminario de Investigación" de la Universidad Autónoma de Aguascalientes con el título "Algoritmo Genético Híbrido para la Clusterización de Hongos mediante su Información Proteómica".

UNIVERSIDAD AUTONOMA DE AGUASCALIENTES

17vo Seminario de Investigación

La Universidad Autónoma de Aguascalientes
a través de la Dirección General de Investigación y Posgrado

Otorgan la presente

CONSTANCIA

A: María Dolores Torres Soto,
Jesús Izac Rincón Miranda, Aurora Torres Soto

Por su participación como ponentes con el trabajo
"Algoritmo genético híbrido para la clusterización de hongos mediante su información proteómica" en la Mesa de **Ingenierías y Tecnologías**

"Se lumen Proferre"
Aguascalientes, Ags., Mayo 2016

M. en Admón. Mario Andrade Cervantes
Rector

Dra. Guadalupe Ruiz Cuéllar
Directora General de Investigación y Posgrado

Diseño e implementación de un Algoritmo de Estimación de Distribución (EDA). Se dio cumplimiento con el diseño, implementación y ajuste de los parámetros de un algoritmo denominado UMDA-CHIP, posteriormente los resultados fueron presentados en:

"Sexto Congreso Internacional: La investigación en el posgrado" en la Universidad Autónoma de Aguascalientes bajo el título "Clusterización de Hongos por Medio de un EDA" (Rincón Miranda et al., 2015).

CISCI 2016 con el título "Clusterización de Hongos en base a su Información Proteómica por Medio de un EDA-UMDA" (estado «Aceptado para participación»).

"*We wish to pursue the truth no matter where it leads. But to find the truth, we need imagination and skepticism both. We will not be afraid to speculate, but we will be careful to distinguish speculation from fact*" Carl Edward Sagan (1934 – 1996).

"*Science works on the frontier between knowledge and ignorance. We're not afraid to admit what we don't know. There's no shame in that. The only shame is to pretend that we have all the answers.*" Neil deGrasse Tyson, Cosmos: A Spacetime Odyssey (2014)