

PONTIFICIA UNIVERSIDAD CATÓLICA DE VALPARAÍSO
FACULTAD DE INGENIERÍA
ESCUELA DE INGENIERÍA INFORMÁTICA

**RESOLUCIÓN DEL PROBLEMA DE BALANCEO DE
MALLAS CURRICULARES MEDIANTE FIREFLY
ALGORITHM**

HERNÁN ANÍBAL JORQUERA MAULÉN
JORGE ANDRÉS AGUILERA MORA

INFORME FINAL DEL PROYECTO
PARA OPTAR AL TÍTULO PROFESIONAL DE
INGENIERO CIVIL EN INFORMÁTICA

Marzo de 2016

PONTIFICIA UNIVERSIDAD CATÓLICA DE VALPARAÍSO
FACULTAD DE INGENIERÍA
ESCUELA DE INGENIERÍA INFORMÁTICA

**RESOLUCIÓN DEL PROBLEMA DE BALANCEO DE
MALLAS CURRICULARES MEDIANTE FIREFLY
ALGORITHM**

HERNÁN ANÍBAL JORQUERA MAULÉN
JORGE ANDRÉS AGUILERA MORA

Profesor Guía: **José Miguel Rubio León**

Profesor Co-referente: **Ricardo Soto De Giorgis**

Carrera: **Ingeniería Civil en Informática**

Marzo de 2016

Dedicatoria

Dedicado a nuestro padres por el apoyo incondicional.

A toda la gente que participó y nos apoyó durante esta gran experiencia de vida.

Índice

Lista De Figuras.....	ii
Lista De Tablas	iii
Resumen	iv
Lista De Abreviaturas.....	v
Abstract.....	v
1 Introducción	1
2 Definición De Objetivos.....	2
2.1 Objetivo general.....	2
2.2 Objetivos específicos	2
3 Estado Del Arte.....	3
4 Marco Teórico	5
4.1 Malla Curricular Universitaria	5
4.2 BACP	7
4.2.1 Entidades y Restricciones del BACP	8
4.2.2 Modelo de programación lineal entera	8
4.3 Optimización	10
4.3.1 Optimización Combinatoria	10
4.3.2 Instancia de un Problema de Optimización	10
4.3.3 Vecindario	11
4.3.4 Óptimo Local	11
4.3.5 Óptimo Global.....	11
4.4 Introducción a las Meta-heurísticas.....	12
4.4.1 Meta-heurística Evolutivas.....	12
4.5 Optimización Basada en las Luciérnagas (FA).....	13
4.5.1 Las Luciérnagas en la Naturaleza	13
4.5.2 Del Comportamiento a la Meta-heurística.	13
4.5.3 Algoritmo FA	14
4.6 Elección del algoritmo FA.....	14
4.7 Esquema general del algoritmo	15
5 Propuesta de Solución	16
5.1 Esquema de Solución Propuesta.....	16
5.2 Estructura de resolución	16
5.3 Lenguaje de programación y entorno de implementación.....	17
6 Prototipo.....	18
6.1 Particularidades de la representación binaria	18

6.2 Resultados Del Prototipo 19
6.2.1 Instancias de Prueba 19
6.2.2 Instancias reales 23
6.2.3 Resultados por instancia 27

7 Conclusiones y Trabajo Futuro 29

8 Referencias..... 30

Lista De Figuras

<i>Figura 4.1 Malla Curricular Universitaria</i>	6
<i>Figura 5.2 Representación matricial</i>	17
<i>Figura 5.3 Vector carga académica</i>	17
<i>Figura 6.1 Frecuencia de soluciones BACP8</i>	20
<i>Figura 6.2 Convergencia BACP 8</i>	20
<i>Figura 6.3 Frecuencia de resultados BACP10</i>	21
<i>Figura 6.4 Convergencia BACP 10</i>	21
<i>Figura 6.5 Frecuencia de soluciones BACP12</i>	22
<i>Figura 6.6 Convergencia BACP 12</i>	23
<i>Figura 6.7 Frecuencia de soluciones para INF</i>	24
<i>Figura 6.8 Convergencia INF</i>	24
<i>Figura 6.9 Frecuencia de soluciones para UPLA</i>	25
<i>Figura 6.10 Convergencia UPLA</i>	25
<i>Figura 6.11 Frecuencia de soluciones para ICI</i>	26
<i>Figura 6.12 Convergencia ICI</i>	27

Lista de Tablas

<i>Tabla 4.1 Tipos de asignaturas</i>	5
<i>Tabla 6.1 Parámetros utilizados en experimentos</i>	19
<i>Tabla 6.3 Resumen resultados</i>	27

Resumen

En el presente informe, se explicará en que consiste el problema de balanceo de mallas curriculares (BACP), describiendo las variables, restricciones y parámetros que presenta el problema en torno a los cursos que componen la malla curricular. Para finalizar se incluyen los resultados obtenidos por el sistema, junto a distintos gráficos que muestran el comportamiento de la aplicación. Además se incluyen los anexos con información acerca del algoritmo en cuestión, los cuales se utilizaron como base de esta investigación.

Abstract

In this report, we explain the balanced academic curriculum problem (BACP), describing the variables, constraints and parameters that has the problem according to the courses that are part of the curriculum. To improve the set of possible solutions, an optimization algorithm based on the behavior of fireflies is applied. Finally the results obtained by the approach will be showed by different graphs illustrating the behavior of the algorithm. In addition we include the information about the algorithm used, which was the basis of this research.

1 Introducción

Las mallas curriculares no son solo importantes para diferenciar una misma carrera que se imparta en 2 o más centros de educación superior. Estas son un instrumento que contiene los conocimientos y objetivos que debe lograr cada estudiante en una carrera específica, estas son diseñadas por docentes, catedráticos y maestros con el fin de satisfacer las necesidades de formación de un profesional, el que deberá ser capaz de solucionar problemas comunes del área en la que se desempeñen.

Diseñar una malla curricular es todo un desafío, no solo porque estas deben formar un profesional integro, esto se debe lograr en un tiempo corto y las materias que se deben abordar muchas veces podrían requerir más tiempo, es por esto que las mallas curriculares deben ser balanceadas ya que dictan la probabilidad de éxito de un estudiante, demasiados ramos en un periodo académico podría llevar al fracaso del futuro profesional. La cantidad de carga por periodo debe ser parecida y a la vez la menor posible dentro del amplio espectro de conocimientos que puedan adquirir. Es por esto que se busca minimizar el costo en diseñar planes curriculares mediante algoritmos con el fin de automatizar y reducir los errores en el proceso.

El problema mencionado se conoce como problema de balanceo de mallas curriculares o *Balanced Academic Curriculum Problem* (BACP), el propósito de esta investigación se centra en resolver el problema BACP mediante la meta-heurística de optimización basada en el comportamiento de las luciérnagas o *Firefly Algorithm* (FA), propuesto por Xin-She Yan [1] el que será explicado a lo largo de todo el presente trabajo.

El algoritmo Firefly, se aplicará a un conjunto de soluciones encontradas en forma previa y representada en forma de matriz binaria. Al poseer un conjunto de soluciones válidas, las cuales representan cada una a una luciérnaga, se procede a optimizar el espacio de soluciones mediante el algoritmo para poder encontrar el óptimo que se desea encontrar.

La meta-heurística básicamente realizará los movimientos correspondientes cuando la intensidad de la solución que se está comparando, logra encontrar una intensidad menor, realizando los movimientos que poseen el algoritmo Firefly en cuanto a las luciérnagas y realizando las reparaciones necesarias y guardando el óptimo encontrado.

Los resultados obtenidos muestran que el algoritmo Firefly posee una convergencia bastante rápida y llegando a los óptimos en casi todas las pruebas realizadas.

2 Definición De Objetivos

2.1 Objetivo general

Resolver el problema de balanceo de mallas curriculares mediante optimización basada en la atracción de las luciérnagas.

2.2 Objetivos específicos

- Diseñar un esquema de solución del problema de balanceo de mallas curriculares basado en optimización con el algoritmo de las luciérnagas.
- Implementar un algoritmo de balanceo de mallas curriculares utilizando optimización con el algoritmo de las luciérnagas.
- Evaluar el rendimiento del algoritmo propuesto mediante la utilización de instancias de prueba e instancias reales.

3 Estado Del Arte

El problema de balanceo de mallas curriculares como fue mencionado busca solucionar la problemática de crear mallas curriculares que sean niveladas y que bajen la carga académica de los estudiantes de las distintas carreras en cualquier tipo de institución de enseñanza. El problema BACP es un problema presente en la librería *CSPLib*, la que es una librería con problemas de restricciones.

En esta sección se busca dar a conocer los autores y técnicas que han desarrollado, modelado este problema hasta el día de hoy, se presentará un resumen cronológico del problema, tratado en distintos lugares, tiempos, personas e instituciones.

El tema fue abordado por primera vez por los autores Castro y Manzano en el año 2001 [2], como fue mencionado anteriormente el fin es lograr una carga académica lo más baja posible y a la vez balanceada. Los autores desarrollaron el problema con técnicas de programación con restricciones. Una de las conclusiones de sus trabajos es que la técnica usada para optimizar es más eficiente y efectiva que la programación lineal entera para resolver problemas de optimización combinatoria.

Otro de los autores que se enfrentaron al desafío de este problema fueron Hnich, Kiziltan y Walsh [3] en el año 2002, en su estudio los autores combinaron los 2 métodos con los cuales se buscó solucionar el problema del balanceo de mallas en estudios pasados, Castro y Manzano señalaron que la programación con restricciones era mejor que la programación lineal entera, Hnich, Kiziltan y Walsh llegaron a la conclusión de que la combinación de la programación lineal entera y la programación con restricciones disminuían el tiempo de ejecución para resolver varias instancias es decir la solución era más eficiente que cada una por si sola.

Lambert et al [4] en el año 2005 utilizaron técnicas híbridas para resolver el problema. Los autores lograron resolver 2 de las instancias del *CSPLib* en 40 segundos aproximados, incluso menos, por otro lado la tercera instancia queda sin resolver pasada una hora. La técnica utilizada por los autores estaba compuesta por Algoritmos Genéticos y Propagación de Restricciones.

Monette et al [5] en el año 2007 estudiaron el problema BACP más allá que los autores anteriores. Lograron resolver las 3 instancias de la *CSPLib* de manera óptima. Además introdujeron nuevas instancias con un generador de instancias, con cientos de cursos y con distintas características. Ellos se enfocaron en el balanceo de carga, extendiendo incluso la idea original de carga máxima sobre los periodos que fueron utilizados en trabajos anteriores, hacia la máxima desviación de la carga promedio y la suma cuadrática y lineal de tales desviaciones.

Castro y Crawford en el año 2007 resolvieron modelos matemáticos utilizando técnicas completas e incompletas, para diseñar mallas curriculares balanceadas. Luego en el año 2013 Rubio et al [6] también desarrollaron el problema BACP, pero ellos utilizaron una meta-heurística relacionada con las colonias de hormigas para solucionar el problema.

Luego otro estudio orientado a las mallas curriculares fue “*The Balanced Academic Curriculum Problem Revisted*”, publicado por Chiarandini, Di Gaspero, Gualandi y Schaerf, en el cual se menciona el hecho que el problema de balanceo de mallas abordado era más simple

que el problema al cual enfrentan las instituciones académicas en la vida cotidiana. Por esto llevaron a cabo una formulación del problema conocido como Problema generalizado de balanceo de mallas curriculares [7] (GBACP). El GBACP toma en consideración incluso los profesores que imparten ramos, los ramos que son compartidos por distintas carreras y además solucionar el problema con más de una malla.

Para finalizar ya en el año 2014 Yusuf Ziya Ünal y Özgür Uysal [8] desarrollaron una nueva solución para el balanceo de mallas, para esto introducen un nuevo modelo llamado *Relevance Based Curriculum Balancing* (RBCB), en él se desarrollan las mallas tomando en cuenta la correlación entre cursos. Aun así no se deja de lado el modelo BACP. En su estudio, los autores usan el *Generalized Quadratic Assignment Problem* (GQAP) para diseñar una malla curricular.

4 Marco Teórico

El marco teórico de este informe corresponde a los siguientes tópicos; El balanceo de mallas curriculares en sí, el cual contiene la información detallada de las mallas curriculares, la meta-heurística que permite la optimización mediante el algoritmo de luciérnagas (FA) [1] que se plantea, la formulación del BACP. Además de otras variables e información de importancia para la comprensión del contexto y optimización.

4.1 Malla Curricular Universitaria

La malla curricular, corresponde a un plan de estudio el cual se conforma de un conjunto de asignaturas o cursos y actividades académicas los que interactúan a lo largo del proceso con un objetivo general y objetivos específicos. Además permite hacer visibles prioridades, secuencias y articulaciones entre los distintos cursos y ciclos correspondientes al ciclo o período de la malla.

Dentro de la malla curricular se pueden distinguir distintos tipos de asignaturas las que forman el plan de estudio, expuestas a continuación en la tabla 4.1.

Tabla 4.1 Tipos de asignaturas

Tipo de asignatura	Definición
Asignaturas básicas	Corresponden a los cursos que proveen al estudiante de conocimientos básicos y cultura.
Asignaturas obligatorias	Corresponden a los cursos que deben ser aprobados por todos los estudiantes que dictan la malla curricular
Asignaturas electivas	Corresponden a los cursos que el alumno puede elegir cumpliendo cierta cantidad obligatorias de créditos mínimos.

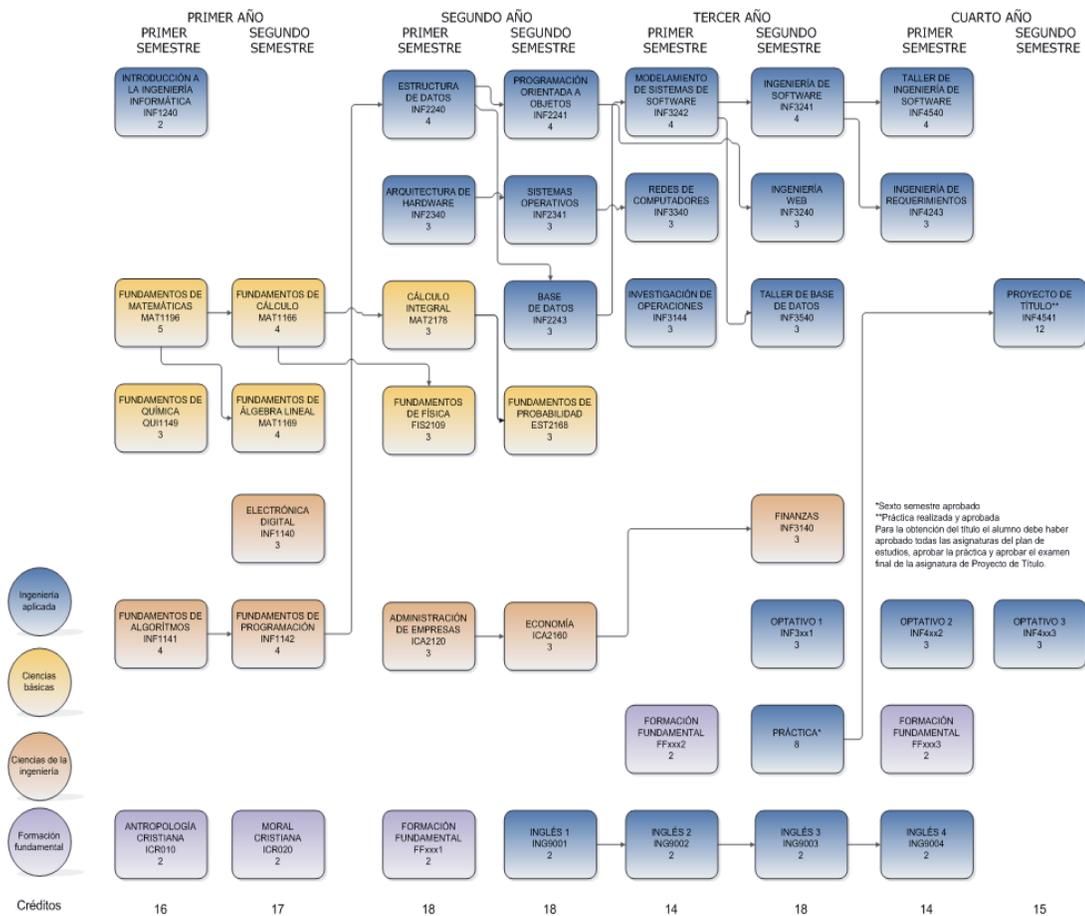


Figura 4.1 Malla Curricular Universitaria

4.1.1 Sistema de Créditos

Dado que los cursos que componen la malla curricular tienen en algunos casos carácter obligatorio y en otros casos de libre elección, en el cual se deben respetar tanto los periodos como los prerrequisitos de cada malla académica, es necesario contabilizar de alguna manera tanto el avance que el alumno posea y su flexibilidad en cuanto al progreso en cierto periodo.

Cada asignatura del plan de estudios tiene una cantidad de créditos asignados, de acuerdo a la cantidad de horas de dedicación que el ramo requiere. Se denomina crédito a la unidad que representa el esfuerzo en horas que un estudiante debería dedicar semanalmente a una asignatura dada.

Los créditos académicos, son un sistema de medida el cual posee las siguientes características:

- Un crédito equivale a 3 “horas” semanales de dedicación (Incluye tanto la cátedra como las horas de estudio y ayudantías).
- Cada “hora” de trabajo es de 45 minutos.
- Por lo tanto, un crédito equivale a 2 horas y 15 minutos (reloj).

Debido a que la malla curricular presenta en su plan de estudios diversos ramos los cuales poseen características especiales (básicos, obligatorios y electivos), es necesario crear una herramienta que asigne y distribuya de manera equilibrada y que cumpla con las restricciones que se derivan de las características de cada asignatura.

El sistema de créditos universitarios permite y facilita la creación de esta herramienta que balancee la malla curricular en forma adecuada, a través de los créditos de cada asignatura es posible cuantificar la carga académica, para así representar el estado de avance del alumno en cualquier periodo dado.

Gracias al valor que se le asigna a cada asignatura, es posible analizar el estado actual y futuro de cada alumno, si el alumno posee en cierto periodo un rendimiento menor al que debe llevar en dicho periodo, el alumno posee un rendimiento bajo con el cual se puede determinar que existen dificultades para ese alumno en cuanto a lo académico. Por el contrario si el alumno posee un rendimiento mayor al promedio hasta ese periodo, se puede determinar que el alumno ha avanzado acorde al plan de estudios o incluso mejor de lo estimado.

Al ser posible cuantificar cada asignatura de forma exacta, obteniendo los límites, reglas y restricciones, genera un contexto en el cual es posible crear una herramienta de optimización y balanceo de mallas curriculares usando una meta-heurística.

4.2 BACP

Balanceo de Mallas Curriculares (BACP), es una herramienta propuesta por Castro C. Y Manzano S. en [2]. Propone la asignación de cursos a los periodos académicos, de manera que la carga producida en cada semestre muestre un balance, es decir, que cada periodo cumpla con tener una cantidad de créditos similar. La malla además debe cumplir con las restricciones que presenta el BACP, como son; Máximos y mínimos de créditos por periodos y cumplir los prerrequisitos que existan.

4.2.1 Entidades y Restricciones del BACP

El BACP consta de las siguientes entidades y restricciones:

- **Cursos:** Los cursos poseen créditos asignados y deben ser dictados dentro del plan de estudios.
- **Períodos:** La malla curricular compuesta por un plan de estudios, se divide en años académicos, cada año académico se divide por un número fijo de intervalos de tiempo (periodos). Cada periodo es lectivo lo cual permite que se dicten cursos. Como ejemplo, una malla curricular que dura 6 años académicos como mínimo, dividida en 2 intervalos al año, tiene 12 periodos en total.
- **Máxima carga:** Por cada periodo existe un máximo de carga permitida.
- **Mínima carga:** Por cada periodo existe un mínimo de carga permitida.
- **Prerrequisitos:** Dentro de la malla curricular, el plan de estudios contiene cursos los cuales deben ser dictados antes que otro curso formando un prerrequisito. Esto produce que se formen pares ordenados de cursos, en el cual se debe cumplir la restricción de cursar el primero antes que el segundo.
- **Distribución balanceada de la carga:** La malla curricular debe quedar balanceada, es decir el número de créditos en cada periodo que se dicta debe poseer un número parecido de créditos, idealmente teniendo el mismo.

Bajo estas entidades y restricciones, el objetivo es encontrar una asignación de cursos para cada periodo que satisfaga todas las restricciones mencionadas.

Se usarán para las pruebas instancias BACP8, BACP10 y BACP12 obtenidas desde CSPLib. Por ejemplo tomando BACP8, está conformado por; 46 cursos con un total de 133 créditos y 8 periodos. Promedio de créditos por período es de $133/8 = 16.625$, es decir que el número máximo de créditos por período es 17.

4.2.2 Modelo de programación lineal entera

El presente problema conocido como BACP e introducido y desarrollado por Castro y Manzano [2], se ha planteado por los autores un modelo de programación lineal entera para el balanceo de mallas curriculares, los parámetros a utilizar son:

- m : Número de cursos
- n : Número de periodos académicos
- α_i : Número de créditos de curso i , donde $i=1, \dots, m$
- β : Carga académica mínima por periodo
- γ : Carga académica máxima por periodo
- δ : Cantidad mínima de cursos por periodo
- ε : Cantidad máxima de cursos por periodo

- Variables de decisión:
 - Vector con los periodos que cada curso tiene asignado

$$x_i = j, \forall i = 1, \dots, m$$

- c : es la máxima carga académica para todos los periodos:

$$c = \text{Max} \{c_1, \dots, c_n\}$$

- La carga académica del periodo j está definida por:

$$c_j = \sum_{i=1}^m \alpha_i \zeta_i, \quad \forall i = 1, \dots, m, \forall j = 1, \dots, n$$

$$\zeta_i = \begin{cases} 1 & \text{si } x_i = j \\ 0 & \text{si } x_i \neq j \end{cases}$$

- Función Objetivo:

$$\text{Min } c$$

- Restricciones:

- Todos los cursos i deben ser asignados a un periodo j :

$$\sum_{j=1}^n x_{ij} = 1, \forall i = 1, \dots, m$$

- El curso b tiene un curso a como prerrequisito:

$$x_{bj} \leq \sum_{r=1}^{j-1} x_{ar}, \forall j = 2, \dots, n$$

- La carga académica máxima está definida por: $c = \text{Max} \{c_1, \dots, c_n\}$. Lo que puede ser representado por el siguiente conjunto de restricciones lineales:

$$c_j \leq c, \forall j = 1, \dots, n$$

- La carga académica del periodo j debe ser mayor o igual que el mínimo requerido:

$$c_j \geq \beta, \forall j = 1, \dots, n$$

- La carga académica del periodo j debe ser menor o igual que el máximo requerido:

$$c_j \leq \gamma, \forall j = 1, \dots, n$$

- El número de cursos del periodo j debe ser mayor o igual que el mínimo requerido:

$$\sum_{i=1}^m \zeta_i \geq \delta, \forall j = 1, \dots, n$$

$$\zeta_i = \begin{cases} 1 & \text{si } x_i = j \\ 0 & \text{si } x_i \neq j \end{cases}$$

- El número de cursos del periodo j debe ser menor o igual que el máximo requerido:

$$\sum_{i=1}^m \zeta_j \leq \varepsilon, \forall j = 1, \dots, n$$

$$\zeta_i = \begin{cases} 1 & \text{si } x_i = j \\ 0 & \text{si } x_i \neq j \end{cases}$$

4.3 Optimización

Un problema de optimización busca minimizar o maximizar el valor de una función. Es importante destacar que la función que se busca maximizar o minimizar, debe ser expresada con respecto de otras variables que están presentes en el problema a resolver.

En estos problemas siempre hay un número de restricciones que deben ser consideradas. Se mencionarán conceptos generales y específicos de problemas relacionados con la optimización.

4.3.1 Optimización Combinatoria

Este tipo de optimización es parte de una de las ramas de las matemáticas. Estos resuelven instancias de problemas explorando espacios de soluciones (usualmente grandes). Eso lo logran reduciendo el tamaño efectivo del espacio y explorando el espacio de búsqueda eficientemente.

4.3.2 Instancia de un Problema de Optimización

La instancia de un problema es una entrada particular del problema, y esta instancia posee una solución que es la salida correspondiente a la instancia del problema que se utilizó como entrada.

Corresponde a un par (F, c) , donde “ F ” es cualquier conjunto dentro del dominio de puntos factibles y “ c ” es la función de costos, tal que [7]:

$$c: F \rightarrow \mathbb{R}$$

El problema es encontrar un $f \in F$, tal que:

$$c(f) \leq c(y), \forall y \in F$$

4.3.3 Vecindario

El vecindario reúne los estados posibles, esto tiene como fin buscar siempre un estado vecino mejor que el actual. Encontrar una solución mediante este método no garantiza obtener un óptimo global, debido a que el espacio explorado no representa la totalidad de las variaciones del problema estudiado.

Dado un punto factible $f \in F$ en una instancia, se define vecindario al conjunto de puntos $N(f)$ que están “cerca” del punto f .

4.3.4 Óptimo Local

Es una solución que es óptima (ya sea máxima o mínima) de un conjunto de vecinos que a su vez son posibles soluciones candidatas.

Dada una instancia (F, c) de un problema de optimización y un vecindario N , una solución f es llamada óptimo local respecto de N si:

$$c(f) \leq c(g), \forall g \in N(f)$$

4.3.5 Óptimo Global

En contraste con el óptimo local, el óptimo global es una solución óptima de todas las posibles soluciones y no solo de un vecindario particular de valores.

Dada una instancia (F, c) de un problema de optimización y $N(f)^*$, el conjunto de todos los vecindarios posibles, f será óptimo global si:

$$c(f) \leq c(y), \forall y \in N(f)^*$$

4.4 Introducción a las Meta-heurísticas

Las meta-heurísticas son estrategias para diseñar procedimientos heurísticos. Por tanto, los tipos de meta-heurística se establecen, en primer lugar, en función del tipo de procedimientos a los que se refiere.

El objetivo principal de las meta-heurísticas es explorar eficientemente el espacio de búsqueda para encontrar soluciones óptimas. Esto se logra a través de estrategias para guiar los procesos de búsqueda, con algoritmos aproximados y no determinísticos. Además se emplean mecanismos para no encontrar óptimos locales [9].

La meta-heurística se compone fundamental de cuatro tipos:

- Las meta-heurísticas de relajación se refieren a procedimientos de resolución de problemas que utilizan relajaciones del modelo original (es decir, modificaciones del modelo que hacen al problema más fácil de resolver), cuya solución facilita la solución del problema original.
- Las meta-heurísticas constructivas se orientan a los procedimientos que tratan de la obtención de una solución a partir del análisis y selección paulatina de las componentes que la forman.
- Las meta-heurísticas de búsqueda guían los procedimientos que usan transformaciones o movimientos para recorrer el espacio de soluciones alternativas y explotar las estructuras de entornos asociadas.
- Las meta-heurísticas evolutivas están enfocadas a los procedimientos basados en conjuntos de soluciones que evolucionan sobre el espacio de soluciones.

Dado que nuestro algoritmo a implementar es el de FA [1], importa entonces explicar las meta-heurística evolutivas.

4.4.1 Meta-heurística Evolutivas

Las meta-heurísticas evolutivas [10] establecen estrategias para conducir la evolución en el espacio de búsqueda de conjuntos de soluciones (usualmente llamados poblaciones) con la intención de acercarse a la solución óptima con sus elementos.

El aspecto fundamental de las heurísticas evolutivas consiste en la interacción entre los miembros de la población frente a la búsqueda que se guían por la información de soluciones individuales.

Las diferentes meta-heurísticas evolutivas se distinguen por la forma en que combinan la información proporcionada por los elementos de la población para hacerla evolucionar mediante la obtención de nuevas soluciones.

4.5 Optimización Basada en las Luciérnagas (FA)

La optimización basada en las luciérnagas es uno de las más nuevas heurísticas inspiradas en los comportamientos naturales para problemas de optimización. Es comúnmente sabido que las luciérnagas poseen un brillo característico inconfundible, e incluso las personas que no han visto una en su vida saben que estas emanan una luz.

El fundamento del algoritmo está orientado al brillo de estas criaturas que en su deseo de aparearse se acercan a cierta luciérnaga la cual libera una luz que es atractiva para las demás. Es este comportamiento el que ha dado luz al algoritmo de optimización el cual considera como función objetivo la intensidad del brillo de las luciérnagas y su necesidad de acercarse a la más brillante u óptima.

4.5.1 Las Luciérnagas en la Naturaleza

Existen diversos tipos de insectos y animales en la naturaleza, que variando de especie en especie, presentan organizaciones y habilidades que hacen que cada elemento de esa colonia cumpla un objetivo para un bien común. Esto se logra a través de diversas formas de comunicación y habilidades que poseen.

Un ejemplo de esto es la luciérnaga, la cual cuando absorben oxígeno, se combina dentro de las células especiales con una sustancia llamada luciferina y reacciona produciendo luz sin apenas generar calor. La luz de la luciérnaga es por lo general intermitente, y brilla de un modo específico en cada especie. Cada forma de brillar es una señal óptica que ayuda a las luciérnagas a encontrar posibles parejas.

4.5.2 Del Comportamiento a la Meta-heurística.

FA surge del comportamiento de las luciérnagas, las que poseen características especiales en cuando a la intensidad de luz asociada al atractivo de una luciérnaga a otra.

El algoritmo de luciérnaga está especialmente indicado para problemas de optimización multimodal. De hecho, FA se ha aplicado en la optimización continua, problema del viajante, clustering, procesamiento de imágenes y selección de características [11].

A continuación se extraen tres propiedades importantes del comportamiento de las luciérnagas [12].

- **Intensidad:** Todas las luciérnagas son unisexuales y se sienten atraídas por otras luciérnagas, independientemente de su sexo.
- El brillo de una luciérnaga se determina por el valor de la función objetivo. Para un problema de maximización, el brillo de cada luciérnaga es proporcional al valor de la función objetivo y viceversa.

- El grado de atracción de una luciérnaga es proporcional a su brillo, y por lo tanto para cualquier par de luciérnagas parpadeantes, la que es menos brillante se moverá hacia la más brillante. Más brillo significa menos distancia entre dos luciérnagas. Sin embargo, si cualquiera de las dos luciérnagas parpadeantes tienen el mismo brillo, éstas se mueven aleatoriamente.

Del análisis de estas propiedades se obtienen las formulas y procesos con los cuales obtener una FA, el cual se mostrará en detalle en el siguiente segmento.

4.5.3 Algoritmo FA

FA es un algoritmo meta-heurístico desarrollado por Dr. Xin-Shi Yang [14], está basado en el comportamiento natural de las luciérnagas que está basado en la bioluminiscencia. Esta luz sirve para comunicarse entre ellas, además puede servir para atraer otras luciérnagas para aparearse y por otro lado su luminosidad puede atraer a sus presas.

Para los problemas de optimización como fue mencionado anteriormente, el brillo es proporcional al valor de la función objetivo. Para algoritmos genéticos el brillo podría ser definido en el mismo sentido que la función fitness [13].

El atractivo de una luciérnaga es proporcional a la luz que emanan y ven a su vez otras luciérnagas, este atractivo puede ser definido como:

$$\beta = \beta_0 e^{-\gamma r^2}$$

Donde β_0 es el atractivo a una distancia $r = 0$. La distancia entre dos luciérnagas r_{ij} es calculada usando el método de distancia cartesiana.

Una luciérnaga i es atraída a una luciérnaga más brillante j y su movimiento está determinado por:

$$x_i = x_i + \beta_0 e^{-\gamma r^2} (x_j - x_i) + \alpha(\text{rand} - 1/2)$$

Donde x_i y x_j es la posición actual de las luciérnagas. El segundo término es debido a la atracción. El tercer término introduce la asignación aleatoria, con α siendo el parámetro de asignación al azar y “rand” es un número aleatorio generado uniformemente distribuido entre 0 y 1.

4.6 Elección del algoritmo FA

La elección del algoritmo general FA está basada en la capacidad que ha sido demostrada por distintos autores y su capacidad para resolver problemas de optimización, la forma en la que se planteara la solución al problema BACP y los estudios que han leído los autores de este ensayo, los ha llevado a la decisión de usar este algoritmo debido a que posee una forma comprobada por Crawford [12] de resolver problemas con representación binaria.

La principal idea de resolución del problema BACP es representar el problema mediante una disposición binaria, debido a que se ha comprobado que plantear la solución de esta manera es efectiva, se puede decir con seguridad que el algoritmo de luciérnagas se acomodara perfectamente a la solución que se planteara más adelante.

4.7 Esquema general del algoritmo

Algoritmo 4.1 – Pseudocódigo del Algoritmo FA

```

Función objetivo  $f(x), x = (x_1, \dots, x_d)^t$ 
Generar una población inicial de  $n$  luciérnagas  $x_i (i = 1, 2, \dots, n)$ 
Intensidad de luz  $I_i$  en  $x_i$  es determinada por  $f(x_i)$ .
Definir el coeficiente de absorción de luz  $\gamma$ 
while( $t < MaxGeneration$ ),
    for  $i = 1:n$  (Todas las  $n$  luciérnagas)
        for  $j = 1:n$  (Todas las  $n$  luciérnagas) (loop interno)
            if ( $I_i < I_j$ )
                Mover luciérnaga  $i$  hacia luciérnaga  $j$ 
            end if
            Variar atractivo con la distancia  $r$  via  $exp[-\gamma r^2]$ 
            Evaluar nueva solución y actualizar intensidad de luz
        end for  $j$ 
    end for  $i$ 
    Clasificar las luciérnagas y encontrar el mejor optimo global  $g_*$ 
end while
Procesar y visualizar

```

La función objetivo es cada una de las posibles soluciones del problema BACP, además se deben inicializar los parámetros de la luciérnaga (γ, β_0 , el tamaño de la población de luciérnaga y el número máximo de generaciones, para el proceso de finalización).

5 Propuesta de Solución

Para resolver el problema BACP, se propone trabajar mediante un algoritmo FA el modelo de programación lineal entera propuesto.

5.1 Esquema de Solución Propuesta

La solución propuesta en este informe es presentada mediante tres simples pasos. Primero se ingresa la instancia BACP, propuesta por *CSPLib*, la cual contiene los datos con los cuales se correrán las pruebas. Segundo se ingresa al algoritmo de FA binario, el cuál analizará la instancia BACP retornando la mejor solución encontrada.

El BACP contiene las restricciones, parámetros y variables decisionales mencionados en la sección 4. Los cursos y periodos académicos forman una malla curricular, las mallas curriculares formarán una matriz binaria de tamaño $N \times M$, al cual se aplicará el algoritmo FA binario.

Además de las variables y parámetros mencionados en el BACP se deben definir las siguientes variables:

- Generar la población total de luciérnagas.
- γ Definir el coeficiente de absorción.
- β_0 Definir coeficiente de atractividad.
- Número máximo de generaciones.

5.2 Estructura de resolución

Es sabido que al momento de buscar la solución a un problema, un aspecto importante de la solución es buscar una representación simple, lógica y útil que sea capaz de resolver la problemática de manera eficiente y eficaz.

Para la solución del problema se ha planteado una representación matricial binaria $C \times P$, es decir, filas con periodos y columnas con cursos, el propósito es que la matriz posea casillas con valores posibles 0 o 1, esta representación indicará si un curso “i” es asignado a un periodo “j” (valor 1), o por el contrario que el curso “i” no este asignado a ese periodo “j” (valor 0). En la Tabla 5.1, se muestra gráficamente representación base de la futura solución al BACP.

		Períodos							
CURSOS	1	0	0	0	0	0	0	0	0
	2	0	0	1	0	0	0	0	0
	3	0	0	0	0	0	0	0	1
	4	0	0	1	0	0	0	0	0
	5	0	0	0	0	1	0	0	0
	6	0	1	0	0	0	0	0	0
	7	0	0	0	0	0	0	0	1

Figura 5.1 Representación matricial

Con la matriz representada, el siguiente paso es la representación de un vector de una dimensión, el cual estará conformado por los cursos dispuestos por las instancias del problema y su respectiva carga académica. El vector estará formado por valores enteros y tendrá un largo C (número de cursos).

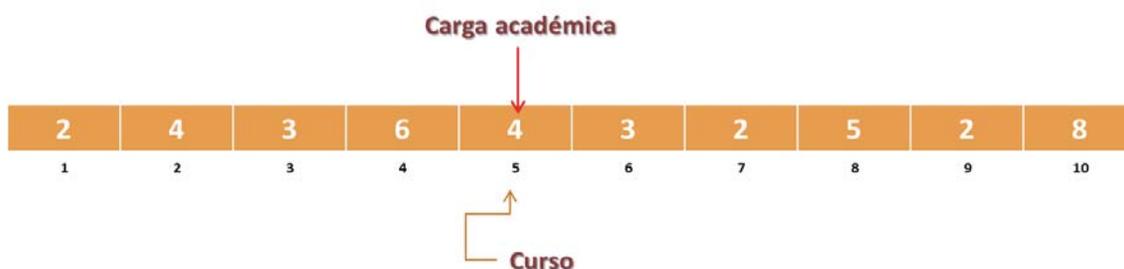


Figura 5.2 Vector carga académica

5.3 Lenguaje de programación y entorno de implementación

El entorno para desarrollar la solución propuesta es en Netbeans IDE 8.0, el lenguaje de programación que usaremos será JAVA, dado las características que posee orientado a objetos, lo cual facilita el planteamiento y resolución de las luciérnagas.

El computador de pruebas tiene sistema operativo Windows 7 Ultimate 64 bits y cuyo procesador es Phenom II X4 3.4 GHZ 8 GB de RAM.

6 Prototipo

El prototipo inicial, constará de una solución del BACP y sus instancias, mediante la Meta-heurística Firefly.

Este prototipo es capaz de generar la solución con mejor calidad, desde el espacio de soluciones.

6.1 Particularidades de la representación binaria

El algoritmo Firefly debe recibir modificaciones para que este lleve a cabo la tarea de optimización de las luciérnagas y sus representaciones binarias. Al aplicar la fórmula de movimiento a una luciérnaga se producen valores que no cumplen con las condiciones de una matriz binaria propuesta en la solución.

La fórmula de movimiento por cada una de las dimensiones de la matriz genera un valor real, al cual se le aplica una función de transferencia.

Una función de transferencia es un modelo matemático que a través de un cociente relaciona la respuesta de un sistema (modelada) con una señal de entrada o excitación (también modelada).

En la teoría de control, a menudo se usan las funciones de transferencia para caracterizar las relaciones de entrada y salida de componentes o de sistemas que se describen mediante ecuaciones diferenciales lineales e invariantes en el tiempo.

Función de transferencia [15]:

$$\text{Tanh}(|X_p|) = \frac{\exp(2|X_p| - 1)}{\exp(2|X_p| + 1)}$$

Luego de aplicar la función de transferencia, los valores deben ser pasados de un espacio de búsqueda continuo a uno discreto, la regla utilizada es la siguiente [15]:

$$x_i^k(t+1) = \begin{cases} 1 & \text{Si } rand < T(x_i^k(t+1)) \\ 0 & \text{en cualquier otro caso} \end{cases}$$

Donde *rand* es un número uniforme al azar entre 0 y 1, $x_i^k(t)$ es un valor de dimensión *k* de una luciérnaga *i* de la iteración.

6.2 Resultados Del Prototipo

Para evaluar los resultados del sistema y su desempeño al momento de generar soluciones, se realizaron 50 pruebas en torno a las instancias de prueba. Los parámetros utilizados en las instancias de prueba fueron los que se presentan en la tabla 6.1:

Tabla 6.1 Parámetros utilizados en experimentos

Parámetros	Valor
Numero de iteraciones	1000
Numero de luciérnagas	30
α	0,5
β	1
γ	1

6.2.1 Instancias de Prueba

Las instancias de prueba son problemas propuestos que sirven para rendir el comportamiento del algoritmo y además como punto de comparación con otros autores que han resuelto el mismo problema. Para el problema BACP existen 3 instancias de prueba las cuales son analizadas y resueltas de forma independiente.

6.2.1.1 BACP 8

El BACP 8 corresponde a la instancia pequeña del problema la cual posee 46 asignaturas que deben ser distribuidas en 8 periodos académicos.

La prueba realizada al sistema en la instancia de prueba BACP 8 entrega los siguientes que se pueden analizar en la Figura 6.1:

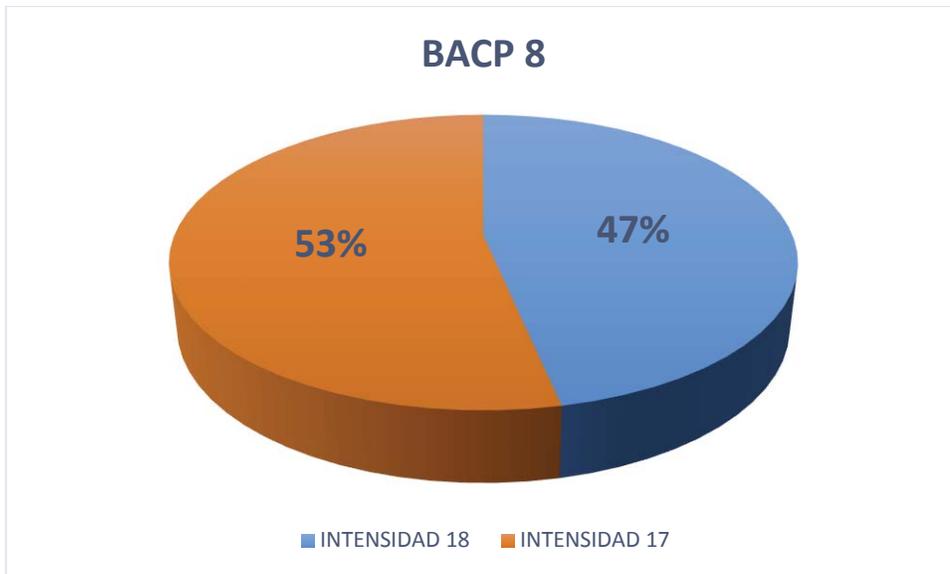


Figura 6.1 Frecuencia de soluciones BACP8

Para la finalización de las pruebas sobre esta instancia de prueba del problema de mallas curriculares se presenta un gráfico de convergencia, el cual muestra como el sistema llega a las soluciones indicadas y como este varía en cada iteración.



Figura 6.2 Convergencia BACP 8

En la figura 6.8 se puede apreciar como el sistema al comienzo del algoritmo cambia de 18 créditos a 17 en las primeras iteraciones y se mantiene estable hasta finalizado el algoritmo.

6.2.1.2 BACP 10

Esta segunda instancia posee 42 asignaturas, las cuales deben ser repartidas en 10 periodos académicos, por lo que la cantidad de créditos mayor (máxima carga académica) es menor que en el caso del BACP 8. Para esta instancia de prueba el sistema con el algoritmo Firefly, realizadas para esta instancia, se logró alcanzar el óptimo conocido. En la Figura 6.3 se muestran los resultados y se evidencia que nuevamente la instancia tiene gran cantidad de respuestas distintas.

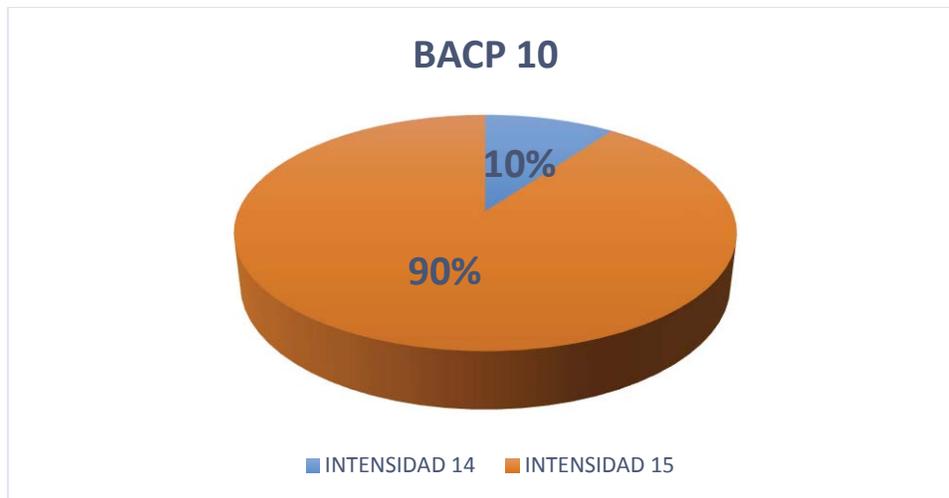


Figura 6.3 Frecuencia de resultados BACP10

A continuación se muestra el gráfico de convergencia para la instancia BACP 10 sobre una prueba que logró el óptimo.

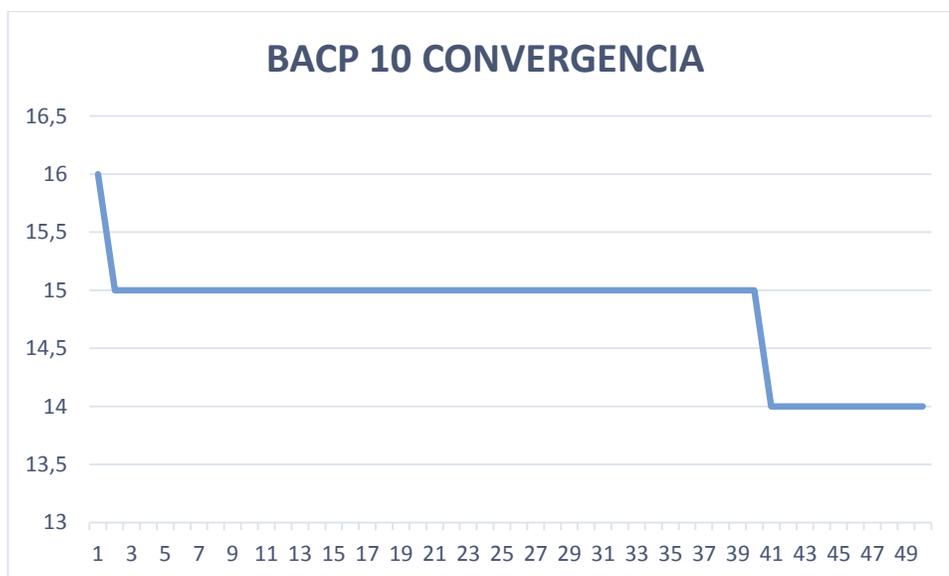


Figura 6.4 Convergencia BACP 10

En la Figura 6.4 se puede apreciar como el sistema varía sus resultados en las primeras iteraciones, empezando desde 16, luego cambiando a 15 y finalmente llegando a 14.

6.2.1.3 BACP 12

La instancia de 12 periodos es la más compleja ya que cuenta con 66 asignaturas, por lo que el tiempo computacional requerido para ejecutarse el algoritmo es mayor que en los casos anteriores. Para esta instancia en la primera ronda de experimentos se logró llegar al óptimo de 18 créditos en varias oportunidades. Los resultados logrados se pueden apreciar en la Figura 6.5.

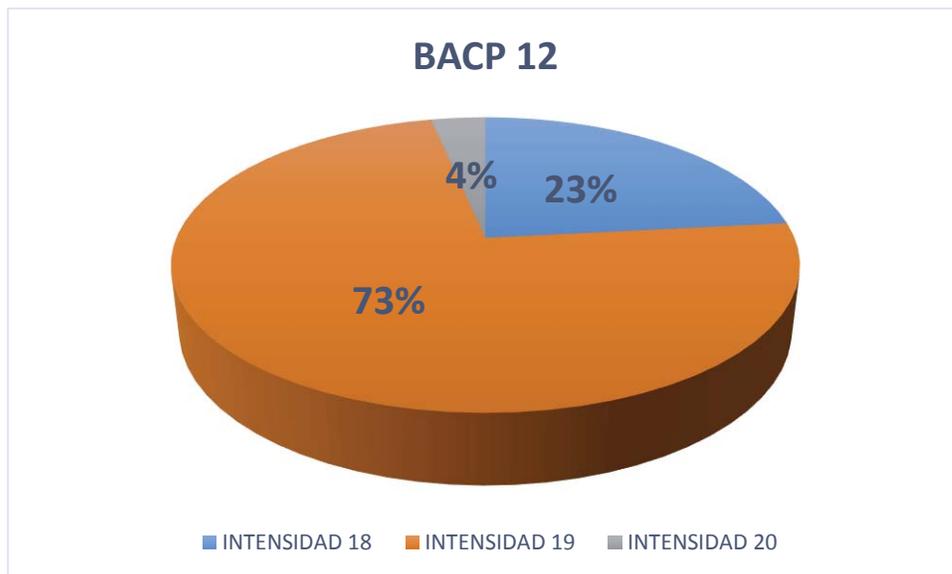


Figura 6.5 Frecuencia de soluciones BACP12

Para finalizar con la instancia BACP 12, se muestra a continuación un gráfico de convergencia para una solución que logró el óptimo en el transcurso de los experimentos.

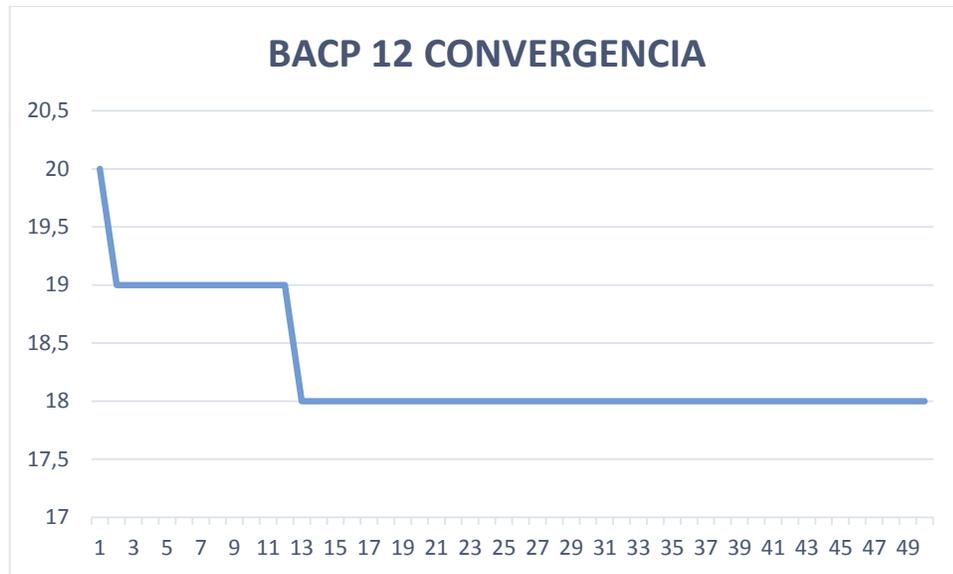


Figura 6.6 Convergencia BACP 12

En la Figura 6.6 se muestra como el sistema para la instancia BACP 12 converge desde los 20 créditos hasta una solución con 19 y luego a otra con 18 créditos y se mantiene constante hasta finalizado el sistema.

6.2.2 Instancias reales

En este apartado se mostrarán los resultados sobre distintos sistemas de mallas curriculares basados en datos reales de distintas universidades conocidas. Las carreras a ser evaluadas corresponden a ingeniería en ejecución informática (8 periodos) e ingeniería civil informática de la Pontificia universidad católica de Valparaíso (12 periodos), e ingeniería informática de la universidad de playa ancha (10 periodos).

6.2.2.1 Ingeniería en Ejecución Informática de la PUCV

En el análisis de las mallas generadas para ingeniería en ejecución informática (INF) se utilizaron 34 asignaturas a ser distribuidas en 8 periodos académicos, de las cuales se encuentran 27 asignaturas obligatorias, 2 ramos de estudios generales (Cultura Religiosa I y II) y 5 asignaturas optativas. Para esta instancia de prueba en la primera ronda de pruebas se logró llegar a un nuevo óptimo de 17 créditos en varias oportunidades. Los resultados logrados se pueden apreciar en la Figura 6.7.

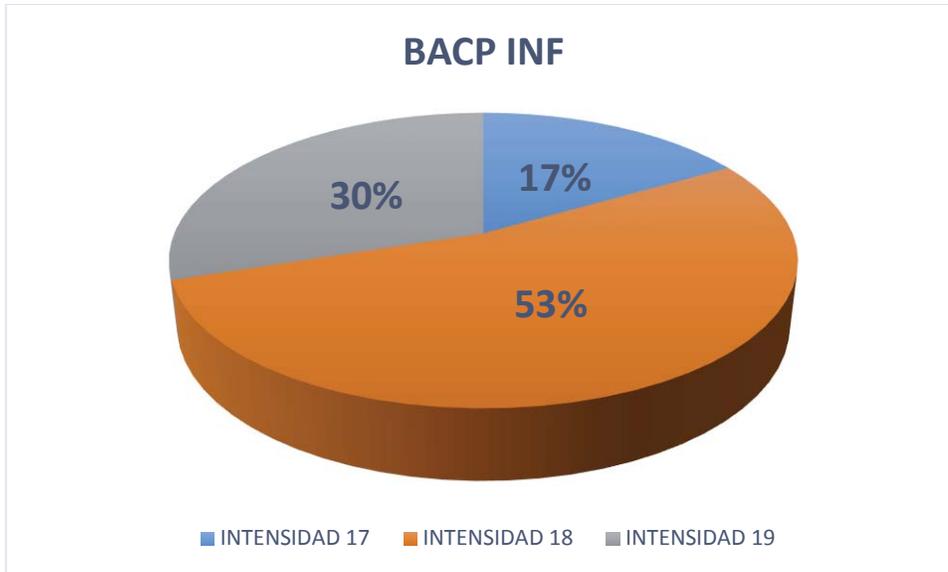


Figura 6.7 Frecuencia de soluciones para INF

Para finalizar las pruebas sobre la instancia INF se muestra un gráfico de convergencias sobre una de las pruebas realizadas en el transcurso de las soluciones.

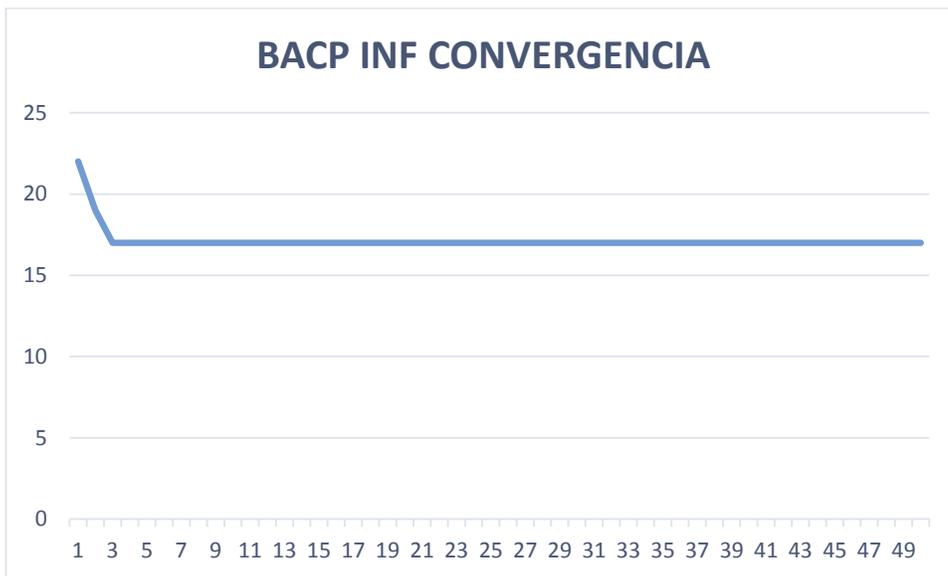


Figura 6.8 Convergencia INF

El sistema converge desde los 22 créditos a los 17 créditos casi instantáneamente en las primeras iteraciones del sistema, como se muestra en la figura 6.8.

6.2.2.2 Ingeniería Informática de la UPLA

Otra de las instancias reales analizadas correspondiente a la carrera de ingeniería informática de la universidad de playa ancha, la cual tiene una duración de 5 años, por lo tanto se deben asignar las 49 asignaturas entre 10 periodos académicos. Para esta instancia de prueba se logró llegar al óptimo de 13. Además se llegó a otros resultados válidos. Los resultados logrados se pueden apreciar en la Figura 6.9.

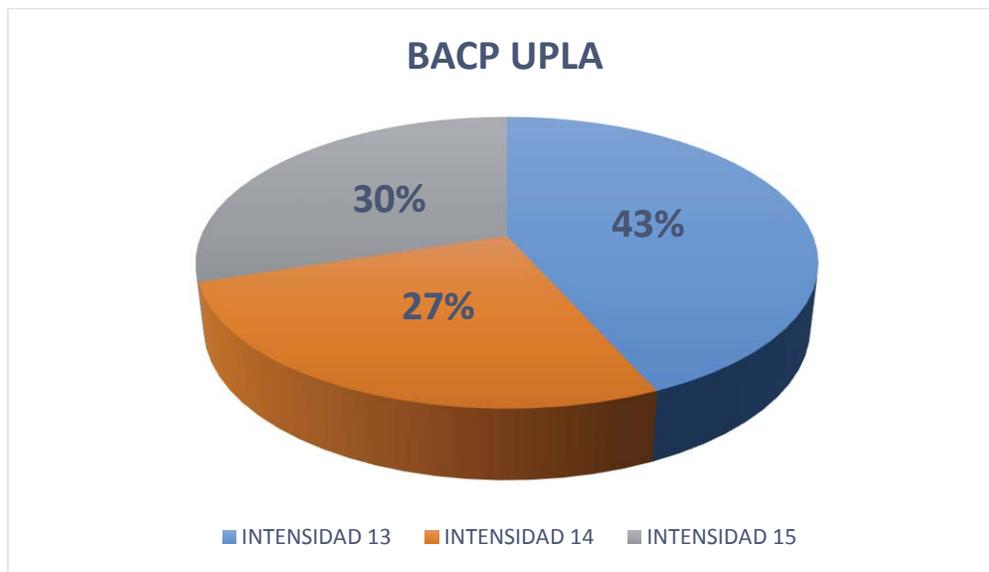


Figura 6.9 Frecuencia de soluciones para UPLA

Para esta instancia real también se realizó un gráfico de convergencia para apreciar el comportamiento del sistema en torno a esta solución.

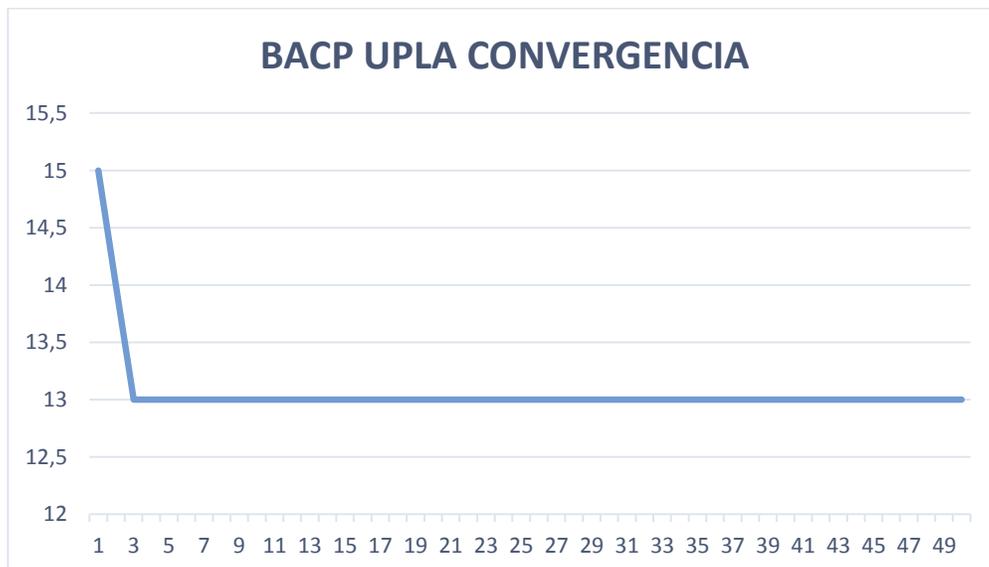


Figura 6.10 Convergencia UPLA

El sistema para la instancia de la Universidad de Playa Ancha y su convergencia en mostrada en la figura 6.10, donde se puede apreciar en concordancia con las otras instancias de prueba anteriores, que el sistema converge en las primeras fases de las iteraciones.

6.2.2.3 Ingeniería Civil Informática de la PUCV

La carrera de Ingeniería Civil Informática de la PUCV (ICI) tiene 53 asignaturas, de las cuales 49 son obligatorias y 4 optativas. En esta instancia no se consideran las asignaturas de estudios generales. Para esta instancia se logró mejorar el óptimo conocido logrando llegar a un resultado de 17 créditos. Además se logró llegar a subóptimos como 18, 19 y 20 créditos. Los resultados logrados se pueden apreciar en la Figura 6.11.

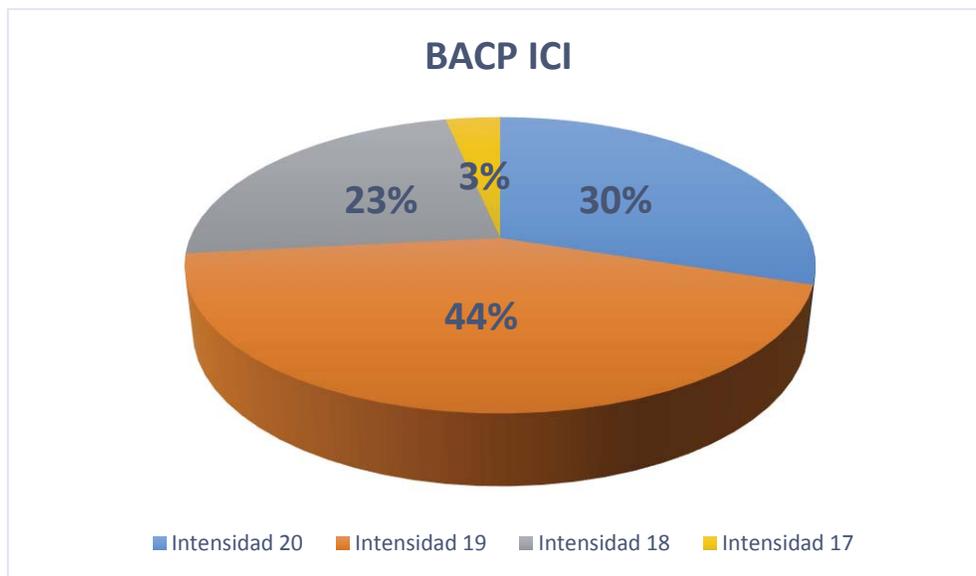


Figura 6.11 Frecuencia de soluciones para ICI

Para finalizar con la ronda de pruebas sobre todas las instancias se realizó nuevamente un gráfico de convergencia sobre la instancia real ICI, el cual se muestra a continuación.

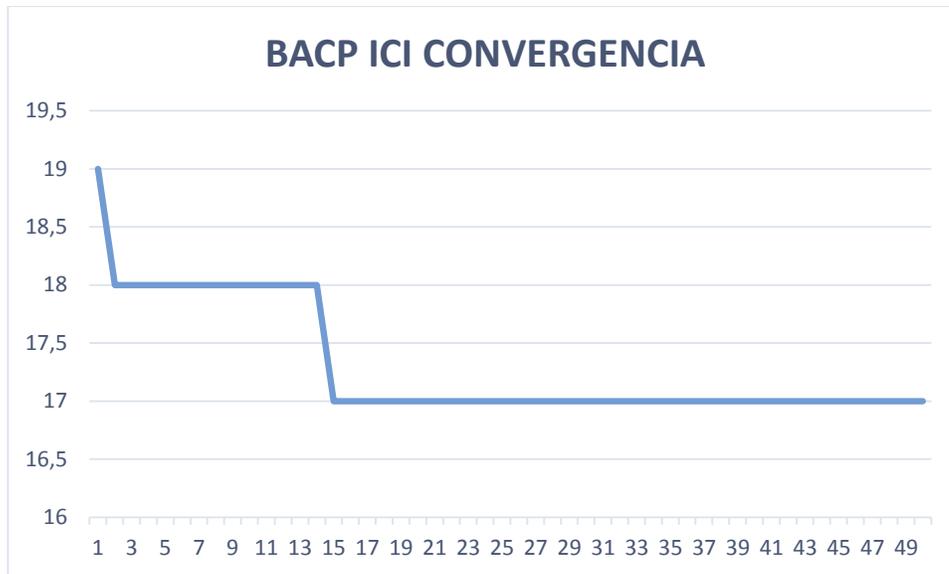


Figura 6.12 Convergencia ICI

El presente caso que se puede analizar en la Figura 6.12, muestra como el sistema converge al óptimo y se mantiene estable en todo el transcurso de las operaciones desde que alcanza el valor. Este caso en particular muestra que el sistema al momento de iniciar encontró una solución factible y luego escala a otra solución logrando llegar a 18 créditos. En el transcurso de su ejecución, nuevamente se produce una actualización en la intensidad logrando una cantidad de 17 créditos.

6.2.3 Resultados por instancia

Los resultados entregados por el sistema en términos generales se consideran alentadores por los desarrolladores del presente trabajo investigativo. Los resultados por cada instancia de prueba y real son mostrados en la Tabla 6.3.

Tabla 6.2 Resumen resultados

	BACP 8	BACP 10	BACP 12	BACP INF	BACP ICI	BACP UPLA
Mejor	17	14	18	17	17	13
Media	17,5	14,9	18,8	18,1	19	13,9
Peor	18	15	20	19	20	15
σ	0,5085	0,3051	0,4842	0,6814	0,8305	0,8604
Óptimo	17	14	18	18	18	13

En la tabla resumen se puede apreciar que en gran parte de las pruebas realizadas se logra llegar al óptimo conocido para dicha prueba, las excepciones encontradas fueron la malla curricular ICI y UPLA, donde no se alcanzó la mejor solución conocida. Es importante destacar

que hay un alto grado de dispersión en los datos entregados por el sistema, esto queda expresado en el valor de la desviación estándar de cada una de las pruebas resaltando en las mallas ICI y BACP 12, ambas con 12 periodos y que se consideran de mayor complejidad.

7 Conclusiones y Trabajo Futuro

El presente trabajo tuvo como finalidad la comprensión del problema BACP, sus singularidades y generalidades, se comenzó por narrar cual era el propósito de este informe y en que se enfocaría, se habló además de cuáles eran las metas pretendidas por los participantes del proyecto.

Luego de analizadas las metas y objetivos del informe y proyecto, fue preciso que se conociera las bases del problema que se abordaría, para esto se realizó un estudio profundo de todas las investigaciones previas y se confeccionó un estado del arte sobre la problemática BACP. Con el estado del arte se logró medir la magnitud del desafío adquirido, se analizó los distintos estudios y se alcanzó un conocimiento profundo sobre el tema en cuestión.

La siguiente etapa consistió en el análisis en profundidad del problema de balanceo de mallas curriculares a través de la sección del marco teórico, se conocieron las aristas del problema y la resolución del tema a través de métodos de programación lineal entera, el cual será reutilizado en el futuro, ya que es una parte consistente y eficaz para llegar a una solución efectiva.

Además se estudió cómo la meta-heurística FA puede llevar a una solución optimizada del problema, teniendo en cuenta que es un algoritmo bastante nuevo, potente y reconocido por diversos autores, además de poseer bastantes aplicaciones, otro aspecto motivante de este algoritmo es su capacidad para resolver problemas con representaciones binarias, además de no haber sido aplicado a la problemática BACP.

Al comprender el funcionamiento del algoritmo resolutivo, se llevó a cabo la confección de un sistema íntegro capaz de resolver el problema planteado, el programa fue capaz de generar soluciones válidas para las instancias del problema y además lograr soluciones óptimas, es importante que las soluciones que el sistema entregue sean mejoradas con el fin de que se acerquen de mejor manera a los óptimos de los problemas.

Los resultados obtenidos, son muy buenos, dado que incluso mejora el óptimo en una de las mallas curriculares. Por lo que el algoritmo de las luciérnagas, ha rendido de gran forma en cuanto a la aplicación de este problema.

Será interesante investigar a futuro cómo se comporta este algoritmo enfrentado a otro tipo de problemas y con otras variantes del algoritmo, las cuales se deben ir eligiendo según el problema que se enfrenta.

El sistema y el trabajo realizado en el transcurso del tiempo en el que se ha confeccionado ha dado los frutos del esfuerzo de los integrantes y además ha arrojado respuestas mejores de las que se esperaban en esta instancia.

8 Referencias

- [1] Yang (2008). Nature-Inspired Metaheuristic Algorithms. Frome: Luniver Press.
- [2] Castro y Manzano, "Variable and Value Ordering When Solving Balanced Academic Curriculum Problems", Proceedings ERCIM WG on constraints, pp. 3-4, Chile, Octubre 2, 2001.
- [3] Hnich, Kiziltan y Walsh, "Modelling a Balanced Academic Curriculum Problem", Proceedings of CP-AI-OR-2002, Vol. 2, pp. 121-131, 2002.
- [4] Lambert, Castro, Monfroy, Riff y Saubion, "Hybridization of Genetic Algorithms and Constraint Propagation for the BACP", Logic Programming, Vol. 3668, pp. 421-423, Berlin, 2005.
- [5] Monette, Schaus, Zampelli, Deville y Dupont, "A cp approach to the balanced academic curriculum problem", Proceedings of the 7th International Workshop on Symmetry and Constraint Satisfaction Problems, 2007.
- [6] Rubio, Palma, Rodriguez, et al., "Solving the Balanced Academic Curriculum Problem Using the ACO Metaheuristic," Mathematical Problems in Engineering, vol. 2013, Article ID 793671, 8 pages, 2013. doi:10.1155/2013/793671.
- [7] Chiarandini, Di Gaspero, Gualandi y Schaerf, "The balanced academic curriculum problem revisited," Journal of Heuristics, vol. 18, no. 1, pp. 119–148, 2012.
- [8] Yusuf Ziya Ünal, Özgür Uysal, et al., "A new mixed integer programming model for curriculum balancing" European Journal of Operational Research, Volume 238, Issue 1, Pages 339–347, 1 October 2014.
- [9] Blum, Roli, Metaheuristics in combinatorial optimization: Overview and conceptual comparison, ACM Computing Surveys (CSUR), v.35 n.3, p.268-308, September 2003 [doi>10.1145/937503.937505].
- [10] Santana, Rodríguez, García López, García Torres, Batista, Moreno Pérez, and Moreno Vega, "Meta-heurística s: Una revisión actualizada." Universidad de La Laguna, España: Departamento de Estadística, Investigación Operativa y Computación, 2004. 205, 214.
- [11] Krishnanand, Ghose, "Glowworm swarm optimization for simultaneous capture of multiple local optima of multimodal functions". Swarm Intelligence 2009.87–124.
- [12] Crawford, Soto, Olivares-Suárez, Paredes. "A Binary FA Algorithm for the Set Covering Problem". Modern Trends and Techniques in Computer Science". 2014. 65-73. ISBN 978-3-319-06739-1.

[13] Sankalap Arora and Satvir Singh. Article: The FA Optimization Algorithm: Convergence Analysis and Parameter Selection. *International Journal of Computer Applications* 69(3):48-52, May 2013.

[14] Yang, “Nature-Inspired Metaheuristic Algorithms”, Luniver Press, 2008.

[15] Crawford, Soto, Olivares-Suárez, Palma, Paredes, Olguín, Norero A Binary Coded Firefly Algorithm that Solves the Set Covering Problem *Romanian Journal of Information Science and Technology* 17(3):252-264 · December 2013.