matemática: Una publicación de FCNM – ESPOL
2013, Vol. 11, No. 2

IMPLEMENTACIÓN DE UN ALGORÍTMO GRASP PARA EL PROBLEMA DE COLORACIÓN DE GRAFOS APLICADO A LA CALENDARIZACIÓN DE EXÁMENES EN UNA INSTITUCIÓN EDUCATIVA

Delgado Erwin[1]

Resumen. *Una de las tareas que enfrentan las instituciones educativas cada año, es la planificación de los horarios de clases y exámenes. Su dificultad radica en que diversas restricciones operativas surgen en el momento de la planificación. Dada la naturaleza del problema descrito anteriormente, la calendarización de exámenes pertenece al conjunto de problemas de optimización combinatoria categorizado NP-Duro por lo que resulta complejo resolverlo por métodos exactos. La ventaja es que la calendarización de exámenes es un problema operativo por lo que bastaría con obtener soluciones factibles de gran calidad, no necesariamente la óptima, en tiempos computacionales razonables. Una de las herramientas utilizadas para el efecto, es la construcción de heurísticas basadas en metaheurísticas por la fortaleza en la exploración inteligente en el espacio de soluciones. Con base en lo anterior, en el presente trabajo se desarrollará un algoritmo heurístico basado en la metodología GRASP el mismo que se lo aplicará en la confección de horarios de exámenes sujetos a un conjunto de restricciones de diversas índoles.*

Palabras Claves: GRASP, Coloración de Grafos, Metaheurísticas, Calendarización de exámenes.

Abstract. *One of the tasks facing educational institutions each year is planning class schedules and exams. His difficulty is that various operational constraints arise at the time of planning. Given the nature of the problem described above, the test scheduling belongs to the set of combinatorial optimization problems categorized NP-Hard, making it complex to solve by exact methods. The advantage is that the scheduling of examinations is an operational problem it would be sufficient to obtain high quality feasible solutions, not necessarily optimal, in reasonable computational times. One of the tools used for this purpose is the construction of metaheuristics based heuristics for intelligent exploration strength in the solution space. Based on the above, in this paper we develop a heuristic algorithm based on the GRASP methodology the same as it applied in the preparation of test schedules subject to a set of constraints of various kinds.*

Keywords: GRASP, graph coloring, metaheuristics, timetabling examinations.

Recibido: Agosto 2013
Aceptado: Septiembre 2013

1. INTRODUCCIÓN

Una de las tareas que enfrentan las instituciones educativas cada año, es la planificación de los horarios de clases y exámenes. Su dificultad radica en que diversas restricciones operativas surgen en el momento de la planificación. Específicamente, en el caso de la calendarización de exámenes, generalmente se intenta elaborarlos considerando diversas restricciones como por ejemplo que los mismos deben ser receptados en uno y sólo un periodo de tiempo y aula previamente definida, la misma que debe admitir un número máximo de estudiantes. De igual manera, la calidad de los calendarios está basado en promover aquellos que impidan que estudiantes rindan más de un examen en un mismo día, entre otras restricciones operativas. Considerando lo anterior, y el hecho de que la calendarización de exámenes pertenece al conjunto de problemas combinatorios con elevada complejidad computacional, hace que éste sea complejo de resolver por métodos exactos.

La ventaja, es que la calendarización de exámenes es un problema operativo por lo que bastaría con obtener soluciones factibles de gran calidad, no necesariamente la óptima, en tiempos computacionales razonables. Una de las herramientas utilizadas para el efecto, es la construcción de heurísticas basadas en metaheurísticas por la fortaleza en la exploración inteligente en el espacio de soluciones.
Considere un conjunto de N exámenes $E = \{e_1, e_2, \dots, e_N\}$, los mismos que deben ser planificados en algún periodo $p \in \{p_1, p_2, \dots, p_m\}$ Un calendario factible es aquel que satisface las siguientes restricciones fuertes:

- Todos los exámenes planificados son asignados a algún periodo de tiempo
- Todos los exámenes planificados son asignados a alguna aula.
- Ningún estudiante podrá rendir más de un examen en un mismo periodo de tiempo.
- En cada periodo de tiempo, no puede asignarse un aula para receptar dos o más exámenes
- La capacidad de las aulas no debe ser excedida.
- Alguna otra restricción operativa definida por la organización.

[1] *Delgado Erwin, M.Sc., Profesor de la Escuela Superior Politécnica del Litoral (ESPOL).*
(e_mail: edelgado@espol.edu.ec).

La calidad de un calendario factible está definida por el valor de la función objetivo que, entre otras cosas, penaliza alguna de las siguientes restricciones suaves:

- Materias de un mismo nivel no pueden ser planificadas en el mismo día
- A cada estudiante se le receptará máximo un examen por día.
- En caso de que a un estudiante se le recepte más de un examen por día, estos no deben ser consecutivos.
- Alguna otra restricción operativa definida por la organización.

En este contexto, el problema de calendarización de exámenes consiste en la asignación de aula y horario a cada uno de los exámenes planificados en un periodo académico, satisfaciendo cada una de las restricciones fuertes establecidas por la organización, minimizando las penalizaciones por el no cumplimiento de las restricciones suaves.

2. FORMULACIÓN MATEMÁTICA DEL PROBLEMA DE CALENDARIZACIÓN DE EXÁMENES

Antes de abordar el modelo matemático para el problema de calendarización de exámenes es importante establecer algunas componentes del mismo.

2.1 CONJUNTOS Y PARÁMETROS

1. Exámenes:
   a) E: Conjunto de exámenes.
   b) se_i: Número de estudiantes registrados en el examen $i \in E$
2. Estudiantes:
   a) S: Conjunto de estudiantes
   b) $t_{is}\begin{cases} 1 \; si \; el \; estudiante \; s \; debe \; rendir \\ \quad el \; examen \; i \\ \quad 0 \; sino \end{cases}$
3. Aulas
   a) R conjunto de aulas
   b) s_r: Capacidad del aula $r \in R$
4. Periodos
   a) P: Conjunto de periodos
   b) H^{aft}: Conjunto de pares de exámenes. $\forall(e_1, e_2) \in H^{aft}$ examen e_1 debe receptarse estrictamente después del examen e_2.
   c) H^{coin} : Conjunto de pares de exámenes. $\forall(e_1, e_2) \in H^{coin}$ los exámenes e_1 y e_2 deben ser receptados en el mismo periodo.
   d) H^{excl} : Conjunto de pares de exámenes. $\forall(e_1, e_2) \in H^{excl}$ los exámenes e_1 y e_2 no deben ser receptados en el mismo periodo.
5. Penalizaciones
   a) w^{2R}: penalización por exámenes consecutivos.
   b) w^{2D}: penalización por exámenes receptados en el mismo día.

2.2 VARIABLES

1. $xp_{ip} = \begin{cases} 1, si \; el \; examen \; i \; debe \; programarse \\ \quad en \; el \; período \; p \\ \\ \quad 0, si \; no \end{cases}$
2. $xr_{ir} = \begin{cases} 1, si \; el \; examen \; i \; debe \; programarse \\ \quad en \; el \; aula \; r \\ \\ \quad 0, si \; no \end{cases}$
3. $C2R_s$ Variable que indica el número de ocasiones que el estudiante s tiene dos exámenes consecutivos en un día.
4. $C2D_s$ Variable que indica el número de ocasiones que el estudiante s tiene dos exámenes planificados en un día.

2.3 FORMULACIÓN MATEMÁTICA

A continuación se presenta un modelo de programación no lineal presentado en [2] que permite encontrar una solución ´optima con la mínima penalización por violación de las restricciones suaves.

$$\text{Min} \sum_{s\in S} w^{2R} C2R_s + w^{2D} C2D_s \quad (1)$$
$$\sum_{r\in R} xr_{ir} = 1, \quad \forall i \in E \quad (2)$$
$$\sum_{p\in P} xp_{ip} = 1, \quad \forall i \in E \quad (3)$$
$$se_i xp_{ip} xr_{ir} \leq sr_r, \forall p \in P, \forall i \in E, \forall r \in R \quad (4)$$
$$\sum_{i\in E} t_{is} xp_{ip} \leq 1, \quad \forall p \in P, \forall s \in S \quad (5)$$
$$xp_{ip} + xp_{jq} \leq 1, \forall p, q \in P_p \leq q, \forall (i,j) \in H_{aft} \quad (6)$$
$$xp_{ip} = xp_{jq}, \forall p \in P, \forall (i,j) \in H_{coin} \quad (7)$$
$$xp_{ip} + xp_{jq} \leq 1, \forall p \in P, \forall (i,j) \in H_{excl} \quad (8)$$
$$C2R_s = \sum_{\substack{i,j\in E \\ i\neq j}} \sum_{\substack{p,q\in P, \\ q=p+1, \\ ypq=1}} t_{is} t_{js} \, xp_{ip} xp_{jq} \quad (9)$$
$$C2D_s = \sum_{\substack{i,j\in E \\ i\neq j}} \sum_{\substack{p,q\in P, \\ q>p+1, \\ ypq=1}} t_{is} t_{js} \, xp_{ip} xp_{jq} \quad (10)$$
$$xp_{ip}, xr_{ir} \in \{0,1\} \quad (11)$$
$$C2R_s, C2D_s \in Z, C2R_s \geq 0, C2D_s \geq 0 \; \forall s \in S \quad (12)$$

La ecuación (1) establece la función objetivo la cual es una combinación lineal del número de estudiantes que tienen exámenes consecutivos, así como exámenes que se toman en el mismo día. La restricción (2) establece que cada examen sea receptado en alguna aula. La restricción (3) establece que cada examen sea receptado en algún

periodo. La restricción (4) garantiza que ningún examen sea receptado en un aula con capacidad menor al número de alumnos que deben rendir la prueba. La restricción (5) garantiza que cada estudiante en cualquier periodo rinda a lo mucho un examen. La restricción (6) garantiza el cumplimiento de que para cada par ordenado de exámenes que pertenezcan a Haft, uno debe estar planificado estrictamente después del otro. De manera similar se analiza las restricciones (7) y (8). Las ecuaciones (9) y (10) permiten determinar el número de estudiantes que tienen planificado más de un examen en el mismo día, así como los que rinden dos exámenes consecutivos. Por ´ultimo las restricciones (11) y (12) se refieren al carácter de las variables declaradas en el modelo.
Uno de los enfoques utilizado para obtener soluciones factibles de buena calidad es por medio de la utilización de métodos que permite resolver el problema de coloración de grafos. ´Este es el enfoque que se ha considerado en el presente proyecto.

3. REVISIÓN DE LITERATURA

Uno de los enfoques utilizados para resolver el PCG es por métodos exactos. Así, Isabel Méndez et al [9] propone un modelo de programación entera, el mismo que es resuelto por un algoritmo de planos cortantes. De igual manera, Hans Bodlaender [11] desarrolla un modelo exacto incorporando memoria polinomial mientras que C. Lucet et al [8] propone un método exacto basado en una descomposición lineal del grafo. Sin embargo estos métodos tienen limitaciones por la cardinalidad del conjunto de vértices del grafo Precisamente, el PCG es uno de los problemas de optimización combinatoria, categorizado como NP-Hard [4], por tal motivo múltiples heurísticos se han diseñado con el objeto de obtener soluciones factibles de buena calidad en tiempos de ejecución adecuados. Uno de ellos es el realizado por [15] quién implementó un algoritmo glotón, ordenando en primer lugar los vértices de acuerdo al grado de adyacencia y asignando en cada iteración un color cada vértice obteniendo así una solución factible. De igual manera, Cédric Avanthay et al [6], desarrollaron una búsqueda local en un vecindario variable. Uno de los trabajos a destacar, dentro de la categoría de métodos de búsqueda local, es el propuesto en [16], en el cual se implementa una búsqueda local con una función de evaluación de la coloración de un grafo, la misma que está en términos del tamaño de cada partición del conjunto de vértices, así como del número de arcos que violen la condición de coloración.

Dentro de los trabajos que se fundamentan en algoritmos evolutivos, podemos destacar los realizados m por Zhipeng Lu [5], quien desarrolló un algoritmo memético para este problema, resolviendo iterativamente el k-coloring desde un valor de k, disminuyéndolo hasta que no exista alguna solución factible. Este algoritmo combina un gen ético con una búsqueda tabú, incorporando un operador de cruce adaptativo; y el realizado por A. Eiben et al [10] que desarrolla un algoritmo evolutivo para la determinación de buenas soluciones factibles. Incorpora un mecanismo adaptativo que permite cambiar el fitness de la función objetivo en cada iteración. Por otro lado, A. Hertz et al [7], diseña un algoritmo fundamentado en la búsqueda tabú, la misma que la comparó con los resultados obtenidos por algunos trabajos desarrollados con recocido simulado observando un mayor perfomance para instancias de gran cardinalidad.

4. GRASP

El GRASP[2] es una metaheurística en la que cada iteración consiste de dos etapas: construcción y búsqueda local [17]. A continuación se muestra el seudocódigo del algoritmo GRASP, considerando un problema de minimización:

Algorithm 1 GRASP

```
procedure GRASP(MaxIter,Seed)
    for k=1,...,MaxIter do
        Solución1 ← EtapaContrucción(Seed)
        Solución ← BúsquedaLocal(Solución1)
        if Solución<BestSolución then
            BestSolución=Solución
        end if
    end for
    return BestSolución
end procedure
```

El objetivo de la fase de construcción es la determinación de una solución factible la cual es mejorada en la fase de búsqueda local. En este sentido, el fundamento de la fase de construcción de una solución factible es un algoritmo glotón no determinístico, en la que iterativamente se incorporan los elementos a una solución parcial. El elemento a ser incorporado en la solución parcial es escogido aleatoriamente desde una lista RCL, la

[2] *Greedy Randomized Adaptive Search Procedure.*

cual está formada por aquellos elementos que produzcan los más bajos costos incrementales en la función objetivo. Luego que el elemento seleccionado es incorporado en la solución parcial, esta es actualizada y se procede a iterar nuevamente hasta que todos los vértices sean coloreados respectando las restricciones establecidas. Luego de la obtención de la solución inicial, se procede a realizar una búsqueda local, la misma que en cada iteración mejora la solución actual hasta alcanzar un óptimo local. En el presente trabajo se ha utilizado una adaptación del algoritmo GRASP desarrollado por [3], el cual permite encontrar soluciones factibles al problema de coloración de grafos.

5. IMPLEMENTACIÓN DEL ALGORITMO GRASP AL PROBLEMA DE CALENDARIZACIÓN DE EXAMENES

Una de las aplicaciones de los algoritmos diseñados para el Problema de Coloración de Grafos es en la confección de horarios de exámenes en una institución educativa. En este contexto, se ha considerado aplicar una adaptación del algoritmo desarrollado en [3] en la planificación de los horarios de exámenes en un semestre específico de estudios, de una carrera específica de una institución educativa de nuestro medio, por lo que se hace necesario establecer diversas consideraciones operativas propias de la organización que son importantes en el diseño de horarios factibles.

5.1. RESTRICCIONES

Diversas restricciones operativas se presentan en el momento de elaborar cualquier planificación. Específicamente, en la confección de los horarios de exámenes en la que se pretende aplicar una adaptación del algoritmo desarrollado en [3], se han considerado las siguientes restricciones.

1. Restricciones fuertes.
a) Uno de los mayores problemas que tiene el estudiantado de cualquier institución educativa es la que exámenes considerados fuertes se planifiquen el mismo día. Por ello, y como una forma de impulsar su desempeño académico, se ha considerado que los exámenes de las materias fuertes ubicadas en el mismo nivel se lo recepten en días diferentes.
b) De igual manera, se ha considerado un máximo de 30 aulas disponibles en un día, por lo que en cada día no se puede programar materias que en conjunto superen el número máximo de aulas disponibles.

2. Restricciones suaves.
a) Se ha considerado el hecho de que en cada semestre planificado se penalice la posibilidad de evaluar más dos exámenes del mismo nivel.

Con base en lo establecido anteriormente, es necesario conocer las materias y el número de paralelos aperturados por cada materia. En la tabla 1 se muestran las materias por nivel. De igual manera, entre paréntesis se muestra el número de paralelos por cada materia.

TABLA I
Implementación de un algoritmo GRASP para el problema de coloración de grafos aplicado a la calendarización de exámenes en una institución educativa
Materias impartidas por niveles

Nivel	Materias				
1	MatI914(14)	MatI2496(1)	MatI039(1)		
2	MatI985(11)	MatI640(13)	MatI910(1)	MatI749(10)	MatI236(2)
3	MatI185(6)	MatI623(2)	MatI966(10)	MatI444(1)	MatI947(11)
4	MatI127(1)	MatI415(1)	MatI166(8)		
5	MatI402(1)	MatI496(1)	MatI458(1)		
6	MatF177(1)	MatI591(1)	MatF115(1)	MatF144(1)	MatI477(1)
7	MatI514(3)				
8	MatI326(2)	MatI484(2)			
9	MatI548(1)	MatI913(1)	MatI529(1)	MatI618(1)	

De igual manera, en la tabla 2 en cada fila se muestran las materias en la que las evaluaciones de la primera columna deben estar planificadas en días distintos a la restante por su nivel de complejidad.

6. RESULTADOS COMPUTACIONALES

A continuación se exponen los resultados obtenidos de la implementación del algoritmo GRASP en el problema de calendarización de exámenes:

TABLA II

Implementación de un algoritmo GRASP para el problema de coloración de grafos aplicado a la calendarización de exámenes en una institución educativa

Materias que no deben evaluarse en el mismo día

Materia	Materias que deben evaluarse en días diferentes a la de la columna 1					
MatI548	MatI185	MatI947				
MatI127	MatI966	MatI166	MatI947			
MatF115	MatI591	MatI402	MatI477			
MatI913	MatI548	MatI185				
MatI591	MatF115	MatI477				
MatI185	MatI548	MatI913	MatI947			
MatI402	MatF115					
MatI749	MatI910	MatI640	MatI914	MatI985		
MatI477	MatF115	MatI591	MatI947			
MatI966	MatI127	MatI640	MatI947			
MatI166	MatI127	MatI947				
MatI640	MatI910	MatI966	MatI914	MatI985		
MatI914	MatI910	MatI749	MatI640			
MatI985	MatI910	MatI749	MatI640			
MatI947	MatI548	MatI127	MatI185	MatI477	MatI966	MatI166

TABLA III

Implementación de un algoritmo GRASP para el problema de coloración de grafos aplicado a la calendarización de exámenes en una institución educativa

Horario de exámenes de materias del primer nivel

	Lu	Ma	Mi	Ju	Vi
8:30 10:30		MATF039			
11:00 13:00					
13:30 15:30			MATI914		MATI2496
16:00 18:00					

TABLA IV

Implementación de un algoritmo GRASP para el problema de coloración de grafos aplicado a la calendarización de exámenes en una institución educativa

Horario de exámenes de materias del segundo nivel

	Lu	Ma	Mi	Ju	Vi
8:30 10:30		MATI640		MATI910	
11:00 13:00					
13:30 15:30	MATI749		MATI985		MATI236
16:00 18:00					

Como se puede observar en cada una de las tablas III, IV, V, VI, VII, VIII, IX, X, XI, se cumplen cada una de las restricciones fuertes establecidas anteriormente.

TABLA V

Implementación de un algoritmo GRASP para el problema de coloración de grafos aplicado a la calendarización de exámenes en una institución educativa

Horario de exámenes de materias del tercer nivel

	Lu	Ma	Mi	Ju	Vi
8:30 10:30	MATI947	MATI185		MATI966	
11:00 13:00					
13:30 15:30			MATI623		MATI444
16:00 18:00					

TABLA VI

Implementación de un algoritmo GRASP para el problema de coloración de grafos aplicado a la calendarización de exámenes en una institución educativa

Horario de exámenes de materias del cuarto nivel

	Lu	Ma	Mi	Ju	Vi
8:30 10:30				MATI166	
11:00 13:00		MATI415			
13:30 15:30			MATI127		
16:00 18:00					

TABLA VII

Implementación de un algoritmo GRASP para el problema de coloración de grafos aplicado a la calendarización de exámenes en una institución educativa

Horario de exámenes de materias del quinto nivel

	Lu	Ma	Mi	Ju	Vi
8:30 10:30				MATI496	
11:00 13:00		MATI402			
13:30 15:30					MATI458
16:00 18:00					

TABLA VIII

Implementación de un algoritmo GRASP para el problema de coloración de grafos aplicado a la calendarización de exámenes en una institución educativa

Horario de exámenes de materias del sexto nivel

	Lu	Ma	Mi	Ju	Vi
8:30 10:30		MATF177		MATI477	
11:00 13:00		MATI591			
13:30 15:30	MATF115				MATF144
16:00 18:00					

TABLA IX

Implementación de un algoritmo GRASP para el problema de coloración de grafos aplicado a la calendarización de exámenes en una institución educativa

Horario de exámenes de materias del séptimo nivel

	Lu	Ma	Mi	Ju	Vi
8:30 10:30				MATI514	
11:00 13:00					
13:30 15:30					
16:00 18:00					

TABLA X

Implementación de un algoritmo GRASP para el problema de coloración de grafos aplicado a la calendarización de exámenes en una institución educativa

Horario de exámenes de materias del octavo nivel

	Lu	Ma	Mi	Ju	Vi
8:30 10:30				MATI484	
11:00 13:00					
13:30 15:30	MATI326				
16:00 18:00					

TABLA XI

Implementación de un algoritmo GRASP para el problema de coloración de grafos aplicado a la calendarización de exámenes en una institución educativa

Horario de exámenes de materias del noveno nivel

	Lu	Ma	Mi	Ju	Vi
8:30 10:30	MATI618			MATI913	
11:00 13:00					
13:30 15:30	MATI529		MATI548		
16:00 18:00					

7. CONCLUSIONES Y RECOMENDACIONES

En el presente trabajo se ha utilizado una adaptación del algoritmo basado en la metodología GRASP desarrollado en [3] aplicado en el problema de calendarización de exámenes. En este contexto, se consideró aplicarlo en la planificación de un semestre académico de una carrera de pregrado de una institución educativa. Al observar los resultados obtenidos se ha podido evidenciar la utilidad de heurísticas en problemas de optimización combinatoria con elevada complejidad por cuanto nos entrega soluciones factibles de gran calidad a un costo computacional razonable. Sin embargo, sólo se ha considerado como restricciones fuertes el hecho de que materias de complejidad elevada sean evaluadas en días distintos, así como el número de paralelos evaluados en un día específico no supere un límite, por lo que se debe considerar restricciones adicionales como el hecho de existen materias que los estudiantes dejan rezagadas periodo a periodo y que en definitiva afecta la confección de calendarios factibles.

REFERENCIAS BIBLIOGRÁFICAS Y ELECTRÓNICAS

[1]. **DING-ZHUE DU, PANOS M. PARDALOS**. (2005), "*Handbook of combinatorial optimization*". Volumen B, Springer.

[2]. **BARRY MCCOLLUM, PAUL MCMULLAN.** (2007), "*The Second International Timetabling Competition: Examination Timetabling Track*". Techinical Report. Queen's University Belfast.

[3]. **E. DELGADO**, "*Diseño e implementación de un algoritmo GRASP para el problema de coloración de grafos*".

[4]. **DING-ZHUE DU, PANOS M. PARDALOS**. (2005). "*Handbook of combinatorial optimization*", Volumen B, Springer.

[5]. **ZHIPENG LU, JIN-KAO HAO.** (2009). "*A memetic algorithm for graph coloring*", European Journal of Operations Research, **ELSEVIER.**

[6]. **CÉDRIC AVANTHAY, ALAIN HERTZ, NICOLAS ZUFFEREY**. (2003). "*A variable neighborhood search for graph coloring*". European Journal of Operational Research, ELSEVIER.

[7]. **A. HERTZ, D. DE WERRA.** (1987). "*Using Tabu Search Techniques for Graph Coloring*". Computing by Springer-Verlag.

[8]. **C.LUCET, F. MENDES, A. MOUKRIM.** (2004). "*An Exact method for graph coloring*". Computer & Operations Research, ELSEVIER.

[9]. **ISABEL MÉNDEZ, PAULA ZABALA.** (2007). "*A cutting plane algorithm for graph coloring*". Discrete Applied Mathematics. ELSEVIER.

[10]. **A. EIBEN, J. VAN DER HAUW, J. VAN HENNERT. "***Graph Coloring with Adaptive Evolutionary Algorithms*". Journal of Heuristics, volume 4:1.

[11]. **HANS BODLAENDER, DIETER KRATSCH.** (2006). "*An exact algorithm for graph coloring with polynomial memory*". Utrecht University.

[12]. **DAVID JOHNSON, CECILIA ARAGÓN, LYLE MCGEOCH, CATHERINE SCHEVON,** (1990). "*Optimization by simulated annealing: An Experimental Evaluation; Part II, Graph Coloring and Number Partition*", Operations Research Society of America.

[13]. **ERIC SOPENA,** (2001). "*Oriented Graph Coloring*", ELSEVIER.

[14]. **WERRA, D.** (1990). "Heuristics for Graph Coloring", Computational Graph Theory, Comput. Suppl. 7, Springer, Vienna, 191-208.

[15]. **BRÉLAZ, D.,** (1979). "*New methods to color the vertices of a graph*", Communications of the Assoc. of Comput. Machinery 22, 251-256.

[16]. **JOHNSON, D. S., ARAGON, C. R., MCGEOCH, L. A., AND SCHEVON, C.** "*Optimization by simulated annealing: An experimental evaluation*"; part ii, graph coloring and number partitioning. Operations Research, 39(3):378406.

[17]. **RESENDE M., RIBEIRO C.,** (2002). Greedy Randomized Adaptive Search Procedures, AT&T Labs Research Technical Report. Hanbook in Metaheuristic, Fred Glover.

[18]. **GLOVER FRED, KOCHENBERGER.** (2003). "*HandBook of Metaheuristics*", International Series in Operations Research & Management Science.

[19]. **TALBI EL-GHAZALI.** (2009). Metaheristics from design to implementaction, John Wiley & Sons, Inc., Hoboken, New Jerse.