

EMPAQUETAMIENTO ÓPTIMO BIDIMENSIONAL CON ROTACIÓN DE PIEZAS USANDO UN ALGORITMO HÍBRIDO DE BÚSQUEDA EN VECINDAD VARIABLE Y RECOCIDO SIMULADO.

Optimal two-dimensional packing with rotatable items using a hybrid algorithm of variable neighborhood search and simulated annealing.

RESUMEN

Los problemas de empaquetamiento tienen un amplio espectro de aplicación en ingeniería de producción, ya que muchos problemas requieren la ubicación de rectángulos en una hoja de material rectangular de mayor tamaño. Se presenta un modelo matemático del problema de empaquetamiento bidimensional permitiendo rotación de las piezas, donde el objetivo es minimizar el área desperdiciada, se utiliza una metaheurística híbrida, búsqueda en vecindario variable y recocido simulado. La eficiencia de la metodología propuesta es evaluada usando 50 casos de estudio, comparándose con dos técnicas presentadas en la literatura especializada que no consideran rotación de piezas, obteniendo resultados de excelente calidad.

PALABRAS CLAVES: empaquetamiento bidimensional, optimización combinatorial, recocido simulado, rotación de piezas, vecindario variable,

ABSTRACT

The packing problems have a large spectrum of application in production engineering, as many problems require the location of rectangles in a larger rectangular material. There is a mathematical model of the two-dimensional packing problem allowing rotation, where the objective is to minimize the wasted area, by means of hybrid metaheuristic, variable neighborhood search and simulated annealing. The efficiency of the proposed methodology is evaluated using 50 case studies, comparing with two techniques presented in the specialized literature that don't implement non-oriented items and getting results of excellent quality.

KEYWORDS: combinatorial optimization, non-oriented items, simulated annealing, two-dimensional bin packing, variable neighborhood search.

1. INTRODUCCIÓN

El problema de empaquetamiento multidimensional ha presentado gran interés en los últimos años [1, 2, 3]. En muchos de ellos las piezas a ser ubicadas tienen una orientación única que no permite rotación de las piezas las cuales presentan diferentes formas geométricas. También existen problemas donde se permite rotación de piezas y que son resueltos sin esta consideración [4], presentándose ineficiencias en el proceso de solución. Por lo anterior permitir rotación de piezas en la metodología representa una mejora en el proceso, reflejada en soluciones de mejor calidad y menores esfuerzos computacionales.

En este documento se presenta el problema de empaquetamiento óptimo bidimensional tipo guillotina permitiendo la rotación de piezas y el mismo en la literatura especializada se puede encontrar bajo la denominación *packing of non-oriented items* [3]. Su

Fecha de Recepción: 26 de enero de 2009
Fecha de Aceptación: 28 de abril de 2009

DAVID ÁLVAREZ MARTÍNEZ
Ingeniero de Sistemas y
Computación.

Joven Investigador
COLCIENCIAS-Universidad
Tecnológica de Pereira
davidalv@utp.edu.co

ELIANA TORO OCAMPO

Ingeniera Industrial, *M.Sc.*
Docente Asistente
Facultad de Ingeniería Industrial
Universidad Tecnológica de Pereira
elianam@utp.edu.co

RAMÓN GALLEGO RENDÓN

Ingeniero Electricista. *Ph.D.*
Docente Titular
Programa Ingeniería Eléctrica
Universidad Tecnológica de Pereira
ragr@utp.edu.co

aplicación puede ser tanto para los problemas de *Bin Packing* como de *Strip Packing*.

En este trabajo es permitida la rotación de piezas con el fin de aumentar el espacio de soluciones y así obtener respuestas de mejor calidad a las ya conocidas. El número de piezas colocadas en el tablero está restringido al área específica de trabajo. Para su solución se usa un algoritmo híbrido conformado por un algoritmo de vecindario variable y un algoritmo de recocido simulado.

La metodología propuesta es confrontada con otras de la literatura especializada y para ello se utilizan 50 casos de estudio tomados de [5]. Del análisis realizado se destaca el buen desempeño presentado por la metodología propuesta en la totalidad de los casos que fueron estudiados.

Este artículo tiene la siguiente estructura: en la sección 2 se presenta una definición del problema de

empaquetamiento óptimo bidimensional, en la sección 3 se plantea el modelo matemático, en la sección 4 se presenta el método de solución, en la sección 5 se muestran los casos de prueba y resultados obtenidos, comparando las respuestas con rotación y sin rotación y finalmente en la sección 6 se presentan las conclusiones y perspectivas de trabajos futuros.

2. DESCRIPCIÓN DEL PROBLEMA

El problema empaquetamiento bidimensional con rotación se define como sigue: Las hojas de material son unidades rectangulares y las piezas a ser ubicadas dentro de la misma pueden ser rotadas 90° [1]. El objetivo es ubicar la mayor cantidad de piezas en la hoja de material minimizando la cantidad de material desperdiciado cumpliendo con las siguientes condiciones:

- a) Cada pieza debe estar ubicada completamente en el material paralelo a los ejes del mismo.
- b) Dos piezas no deben estar traslapadas
- c) Las piezas pueden rotar 90° en todas las posibles direcciones.

Este problema tiene muchas aplicaciones interesantes tales como corte de piezas de todo tipo de materiales que no tienen tramas: vidrios, plásticos, espumas, láminas, maderas entre otras.

En la figura 1. Se ilustra el proceso de rotación a 90°.

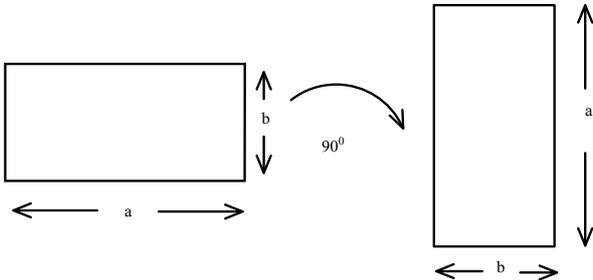


Figura 1. Rotación de una pieza.

3. MODELO MATEMÁTICO

A continuación se describen algunas notaciones necesarias para comprender el modelo matemático del problema.

L : Largo de la hoja de material.

W : Ancho de la hoja de material.

N : Número disponible de rectángulos para posicionar en la hoja de material.

p_i : Largo del rectángulo i .

q_i : Ancho del rectángulo i .

(x_i, y_i) : Variables que indican la localización del rectángulo i teniendo como punto de referencia el vértice inferior izquierdo de la hoja de material.

s_i : Variable binaria que indica si el rectángulo i fue ubicado en la hoja de material. Cuando esto ocurre $s_i = 1$, en caso contrario $s_i = 0$.

(l_{xi}, l_{yi}) : Variables binarias que indican para cual eje de la hoja de material, el lado p_i del rectángulo i está en paralelo.

(w_{xi}, w_{yi}) : Variables binarias que indican para cual eje de la hoja de material, el lado q_i del rectángulo i está en paralelo.

Adicionalmente, existen otras variables que son usadas para indicar el posicionamiento de los rectángulos en relación a otros rectángulos.

a_{ik} : En caso de que sea 1, indica que el rectángulo i está a la izquierda del rectángulo k .

b_{ik} : En caso de que sea 1, indica que el rectángulo i está a la derecha del rectángulo k .

c_{ik} : En caso de que sea 1, indica que el rectángulo i está atrás del rectángulo k .

d_{ik} : En caso de que sea 1, indica que el rectángulo i está al frente del rectángulo k .

M : Número entero muy grande.

La función objetivo consiste en minimizar la cantidad de material desperdiciado tal como se expresa en la ecuación (1).

$$Min \quad L \cdot W - \sum_{i=1}^N p_i \cdot q_i \cdot s_i \quad (1)$$

Sujeto a

1. Evitar superposición de rectángulos en la hoja de material tal como se muestra en las ecuaciones (2), (3), (4) y (5).

$$x_i + p_i \cdot l_{xi} + q_i w_{xi} \leq x_k + (1 - a_{ik}) \cdot M, \forall i, k, i < k \quad (2)$$

$$x_k + p_k \cdot l_{xk} + q_k w_{xk} \leq x_i + (1 - b_{ik}) \cdot M, \forall i, k, i < k \quad (3)$$

$$y_i + p_i \cdot l_{yi} + q_i w_{yi} \leq y_k + (1 - c_{ik}) \cdot M, \forall i, k, i < k \quad (4)$$

$$y_k + p_k \cdot l_{yk} + q_k w_{yk} \leq y_i + (1 - d_{ik}) \cdot M, \forall i, k, i < k \quad (5)$$

2. Garantizar que el par de rectángulos evaluados con las ecuaciones anteriores estén dentro de la hoja de material, tal como se muestra en la ecuación (6).

$$a_{ik} + b_{ik} + c_{ik} + d_{ik} \geq s_i + s_k - 1, \forall i, k, i < k \quad (6)$$

3. Garantizar que un posicionamiento de rectángulos, obedezca a las limitaciones físicas dadas por las dimensiones de la hoja de material, tal como se muestra en las ecuaciones (7) y (8).

$$x_i + p_i \cdot l_{xi} + q_i w_{xi} \leq L + (1 - s_i) \cdot M, \forall i \quad (7)$$

$$y_i + p_i \cdot l_{yi} + q_i w_{yi} \leq W + (1 - s_i) \cdot M, \forall i \quad (8)$$

4. MÉTODO DE SOLUCIÓN

El algoritmo propuesto fue soportado por la metodología presentada en [6]. En la figura 2 se representa el esquema de la metodología implementada.



Figura 2. Esquema de la metodología.

El resumen de la metodología propuesta es el siguiente:

4.1 Preprocesamiento de las piezas

El preprocesamiento consiste en cargar y duplicar las piezas de la siguiente forma, para cada pieza cargada, si el ancho de la pieza es diferente al largo de la pieza entonces, duplique la pieza donde el largo de la pieza duplicada es el ancho de la pieza original y el ancho de la pieza duplicada es el largo de la pieza original, de lo contrario, no la duplique.

4.2 Algoritmo de búsqueda aleatoria binaria

El algoritmo I determina los tipos de cada corte a utilizar sea horizontal o vertical, para este algoritmo se usa un esquema de vecindario variable [7].

Inicialmente el vecindario de búsqueda consiste en modificar de forma aleatoria la orientación de un corte, posteriormente, si esta modificación no genera una mejora en la función objetivo se procede a modificar dos cortes y así sucesivamente. En el momento en que la mejor solución hallada hasta el momento (incumbente) sea actualizada se regresa nuevamente al primer vecindario. Esta metodología de búsqueda local permite diversificar cuando la función objetivo no es mejorada y hacer una búsqueda local más detallada en el momento de actualizarse la incumbente, de esta forma se logra un algoritmo computacionalmente eficiente.

4.3 Algoritmo de búsqueda aleatoria real

El segundo algoritmo determina la distancia a la que se debe realizar cada corte, para ello se utiliza un algoritmo de búsqueda aleatoria que modifica iterativamente los cortes seleccionados por el algoritmo I reduciendo paulatinamente el ancho de búsqueda, mediante un factor de recocido que a medida que el algoritmo evoluciona se hace más determinístico por lo cual este factor se acerca a cero. El algoritmo II está inspirado en la filosofía del recocido simulado [8] en donde a medida que evoluciona el grado de aleatoriedad disminuye.

4.4 Algoritmo constructivo

El algoritmo constructivo ubica las piezas en los diferentes sub-espacios definidos por los tipos de los cortes y las distancias de estos. Cada sub-espacio ubica arreglos de la misma clase de piezas, para esto se determina el desperdicio para cada tipo de arreglo asegurando se cumpla la restricción de demanda, de tal forma que si el arreglo está conformado por piezas duplicadas se modifica la demanda de la pieza original y la duplicada.

5. RESULTADOS

A continuación se presentan los resultados obtenidos de la ejecución de 50 casos de estudio referenciados en [5], en la tabla 1 se presentan las respuestas obtenidas de solucionar los casos con 3 metodologías diferentes: la primera corresponde a los resultados reportados en la referencia [5], la segunda a los resultados reportados en la referencia [6], la tercera corresponde a los resultados obtenidos con el algoritmo propuesto en este artículo. De esta tabla se observa que los mejores resultados fueron obtenidos con el algoritmo propuesto y logra igualar o superar a los dos algoritmos referenciados en los casos estudiados.

Caso de Prueba		Resultados Referencia [5] Sin Rotación		Resultados Referencia [6] Sin Rotación		Resultados Algoritmo Propuesto Con Rotación	
Caso	Área Disponible	Área utilizada	Porcentaje de uso	Área utilizada	Porcentaje de uso	Área utilizada	Porcentaje de uso
1	2004002	1964584	0,9803	1990188	0,9931	1990188	0,9931
2	2941249	2867218	0,9748	2887424	0,9817	2898620	0,9855
3	2709525	2656454	0,98040	2679192	0,9888	2679192	0,9888
4	2479752	2370496	0,9559	2443300	0,9853	2443300	0,9853
5	2319604	2281600	0,9836	2275995	0,9812	2287419	0,9861
6	2976190	2918503	0,9806	2931250	0,9849	2959476	0,9943
7	2265650	2227422	0,9831	2240954	0,9891	2244518	0,9906
8	2272792	2238658	0,9849	2247791	0,9890	2248697	0,9893
9	2764384	2690038	0,9731	2717389	0,9830	2724090	0,9854
10	2624232	2574750	0,9414	2579620	0,9830	2589880	0,9869

11	3235950	3115873	0,9628	3162818	0,9774	3181816	0,9832
12	2921333	2837743	0,9713	2856772	0,9779	2875058	0,9841
13	2553588	2501400	0,9795	2525754	0,9891	2525754	0,9891
14	2436480	2343169	0,9617	2350960	0,9649	2358096	0,9678
15	3072540	3017184	0,9819	3031675	0,9867	3046710	0,9915
16	2174340	2102756	0,9670	2136724	0,9827	2139405	0,9839
17	2401595	2276323	0,9478	2317299	0,9649	2342832	0,9755
18	2827455	2736919	0,9679	2751962	0,9733	2768626	0,9791
19	2181600	2145051	0,9832	2149967	0,9855	2156799	0,9886
20	3157786	2994717	0,9483	3050737	0,9661	3068481	0,9717
21	2953340	2900376	0,9820	2892501	0,9794	2908884	0,9849
22	3204790	3066344	0,9568	3086854	0,9632	3120352	0,9736
23	2763800	2676037	0,9682	2686966	0,9722	2730945	0,9881
24	2408892	2287330	0,9495	2292061	0,9515	2323423	0,9645
25	3163699	3037774	0,9601	3108967	0,9827	3111640	0,9835
26	2805712	2695345	0,9606	2748756	0,9797	2758701	0,9832
27	2526230	2383242	0,9433	2479747	0,9816	2481420	0,9822
28	2532060	2446799	0,9663	2490028	0,9834	2492867	0,9845
29	2230605	2158548	0,9676	2186885	0,9804	2197516	0,9851
30	2514609	2471150	0,9827	2462305	0,9792	2473430	0,9836
31	2727200	2583155	0,9471	2666656	0,9778	2684286	0,9842
32	2699860	2618956	0,9700	2628854	0,9737	2655905	0,9837
33	2262910	2216886	0,9796	2224667	0,9831	2229412	0,9851
34	2330028	2279436	0,9782	2284359	0,9804	2296139	0,9855
35	2882646	2787529	0,9670	2798761	0,9709	2816649	0,9771
36	2658656	2508545	0,9435	2588999	0,9738	2615425	0,9837
37	2817192	2667081	0,9467	2738592	0,9721	2765134	0,9815
38	2618181	2566844	0,9803	2567650	0,9807	2591201	0,9896
39	2097396	2029177	0,9674	2068032	0,9860	2069913	0,9868
40	2597700	2540141	0,9778	2531199	0,9744	2544728	0,9796
41	2718228	2686780	0,9884	2677183	0,9849	2702052	0,9940
42	2379541	2306316	0,9692	2339089	0,9830	2357797	0,9908
43	2519286	2419778	0,9605	2454036	0,9741	2460046	0,9764
44	2575800	2493761	0,9681	2507026	0,9733	2529588	0,9820
45	2497225	2418204	0,9683	2414067	0,9667	2443486	0,9784
46	2817360	2693446	0,9560	2750589	0,9763	2764739	0,9813
47	2484896	2388374	0,9611	2414822	0,9718	2432533	0,9789
48	3007466	2943027	0,9785	2933482	0,9754	2944504	0,9790
49	2868953	2819262	0,9826	2815017	0,9812	2830906	0,9867
50	2234752	2161976	0,9674	2194973	0,9822	2199821	0,9843

Tabla 1. Resultados de la ejecución del caso 1 hasta el caso 50.

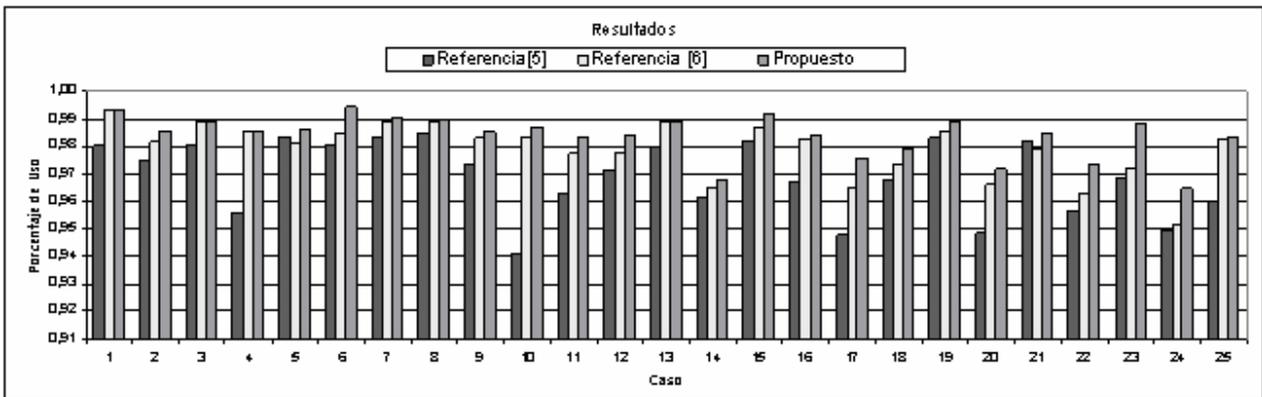


Figura 3. Gráfico comparativo: referencia [5], referencia [6], algoritmo propuesto.

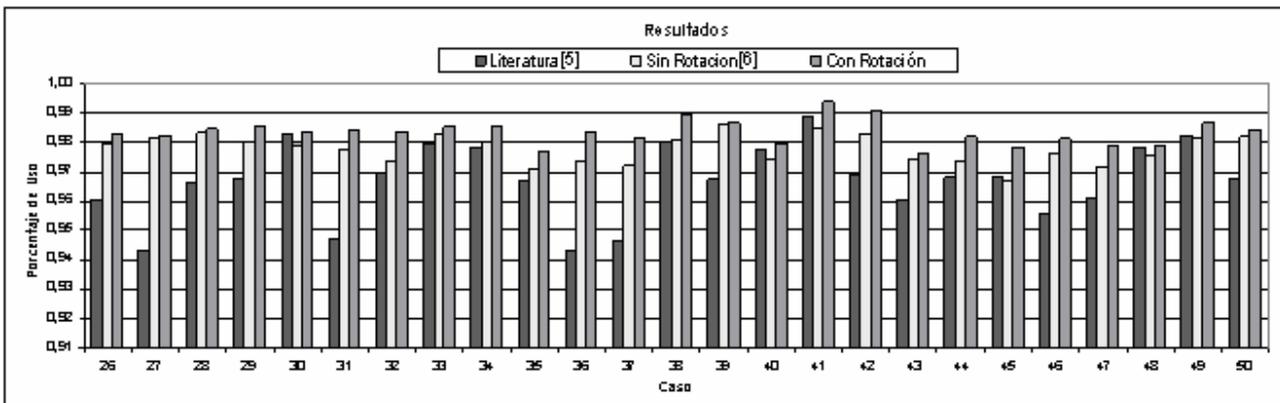


Figura 4. Gráfico comparativo: referencia [5], referencia [6], algoritmo propuesto.

En las figuras 3 y 4 representan gráficamente los resultados presentados en la tabla 1, en estas se visualiza que para todos los casos el algoritmo propuesto supera o por lo menos alcanza las mejores respuestas obtenidas con los algoritmos reportados en [5] y [6].

Resumen de Respuestas	Respecto a [5]	Respecto a [6]
Soluciones no superadas	0	0
Soluciones superadas entre 0 y 1 %	17	44
Soluciones superadas entre 1 y 2 %	20	6
Soluciones superadas entre 2 y 3 %	8	0
Soluciones superadas entre 3 y 4 %	3	0
Soluciones superadas en mas de 4 %	2	0

Tabla 2. Número de casos superados respecto a [5] y [6].

En la tabla 2 y en la figura 5, se presenta un resumen de resultados obtenidos con base en el porcentaje de mejora de de la solución de cada caso de estudio con respecto a las soluciones en [5] y [6]. Con respecto a [6] se observa que en su gran mayoría las respuestas reportadas son

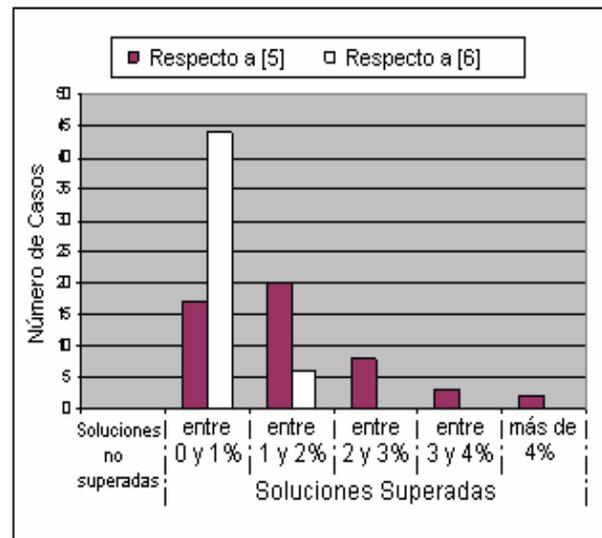


Figura 5. Resumen de Respuestas.

superadas entre 0% y 1%, con respecto a la referencia [5] gran parte de los resultados presentan mejoría mayor al 1%. La inclusión de la rotación de piezas en la metodología ejerce una influencia positiva presentando resultados de mejor calidad a los reportados en [5] y [6].

6. CONCLUSIONES Y RECOMENDACIONES

Se resolvió el problema de empaquetamiento bidimensional con rotación de piezas mediante un algoritmo híbrido constructivo de búsqueda en vecindad variable y recocido simulado, obteniendo resultados de excelente calidad.

Se corrobora que la metodología implementada en [6], es una metodología robusta y con grandes propiedades para solucionar los problemas de empaquetamiento de mercancías.

Se encontraron soluciones superiores nunca antes reportadas en la literatura especializada para los 50 casos de estudio tomados de [5].

Se presenta un modelo matemático que puede ser de gran ayuda para tratar el problema mediante un método exacto, tal como el *Branch and Bound* [9].

Los resultados obtenidos muestran que aplicando rotación de piezas se obtienen soluciones de mejor calidad evaluada en la función objetivo y con tiempos de cómputo inferiores comparados con algoritmos que no consideran rotación.

Existen muchos problemas de ingeniería de producción que podrían considerar rotación de piezas con el fin de obtener mejoras sustanciales en el objetivo a optimizar.

El algoritmo propuesto podría ser extendido a la solución de problemas de *Strip Packing* y empaquetamiento tridimensional.

7. BIBLIOGRAFÍA

- [1] Basal N, Sviridenko M. "New approximability and unapproximability results for 2-dimensional packing". In Proceedings of 15 Annual Symposium on Discrete Algorithms, p 189-196. ACM/SIAM, 2004.
- [2] Caprara A. "Packing 2-dimensional bins in harmony" 43 IEEE symp. on Found of comp. Science. p490-499, 2002.
- [3] Dell Amico M, Martello S, Vigo D. "A lower bound for the non-oriented two dimensional bin packing problema". Discrete Applied Mathematics, 118:13-14, 2002.
- [4] Toro E, Rueda A. "Algoritmo de Búsqueda Tabú aplicado a la solución del problema de corte bidimensional guillotinado". Scientia et Technica No 37. Diciembre 2007.
- [5] Yaodong C. "An exact algorithm for generating homogenous T-shape cutting patterns". Computers & Operations Research. 2007, 34(4): 1107-1120, Disponible en Internet en: <http://www.gxnu.edu.cn/Personal/ydcui/English/Paper.htm>
- [6] Toro E, Garces R, Ruiz H. "Solución del problema de empaquetamiento óptimo bidimensional usando un algoritmo híbrido constructivo de búsqueda en vecindad variable y recocido simulado". Revista Facultad de Ingeniería Universidad de Antioquia No 46 diciembre 2008.
- [7] Hansen P, Nenad J, Moreno J. "Búsqueda en entorno variable . Inteligencia artificial" Revista Iberoamericana de Inteligencia Artificial. Vol 19. 2003.
- [8] Toro E, Gallego R, Escobar A. "Técnicas Metaheurísticas de Optimización". Universidad Tecnológica de Pereira. 2008.
- [9] Romero R, Gallego R, Escobar A. "Programación Lineal Entera". Universidad Tecnológica de Pereira. 2007.