

Desempeño de metaheurísticas sin memoria en el problema del LABS

Jhon Edgar Amaya

Laboratorio de Computación de Alto Rendimiento (LCAR), Universidad Nacional Experimental del Táchira, San Cristóbal, Venezuela, jedgar@unet.edu.ve

María de los Ángeles Tarazona

Departamento de Matemática, Universidad Nacional Experimental del Táchira, San Cristóbal, Venezuela, maria.tarazona@unet.edu.ve

ABSTRACT

This paper addresses the low-autocorrelation binary sequences (LABS) problem, which is a difficult combinatorial optimization problem. The LABS problem has been studied since the early 60's by physicists and researchers in the area of artificial intelligence, since the problem has many applications in various fields (chemistry, physics, telecommunications, etc.). This research analyzes the performance of different metaheuristics without memory for solving the LABS problem, such as: Hill Climbing (HC), Iterated Local Search (ILS), Genetic Algorithm (GA), a memetic algorithm (MA) and a cooperative version. In particular, we performed a basic analysis of the performance of each algorithm in solving the LABS problem. A comparative method was used for statistical analysis of data. Experimental results show that the cooperative algorithm presented the best results for instances of LABS problem analyzed.

Keywords: Low Autocorrelation Binary Sequences (LABS), Iterated Local Search (ILS), Genetic Algorithm (GA), Memetic Algorithm (MA), cooperative algorithms.

RESUMEN

El presente trabajo aborda la búsqueda de secuencias binarias de baja autocorrelación (LABS), el cual es un problema difícil de optimización combinatoria. El problema del LABS ha sido estudiado desde la década de los 60's por las comunidades de físicos y en el área de inteligencia artificial, pues el problema tiene muchas aplicaciones en diversas áreas (radares, química, física, telecomunicaciones, entre otras). En la investigación se analiza el comportamiento de diferentes metaheurísticas sin memoria para solucionar el problema del LABS, como lo son: Hill Climbing (HC), Búsqueda Local Iterada (ILS), Algoritmo Genético (GA), un Algoritmo Memético (MA) y una versión cooperativa. En particular, se realizó un análisis básico del desempeño de cada algoritmo en la solución del problema del LABS, utilizando un método comparativo estadístico para el tratamiento de los datos. Los resultados experimentales muestran que el algoritmo cooperativo presentó los mejores resultados para las instancias del problema del LABS analizadas.

Palabras claves: Secuencias binarias de baja autocorrelación (LABS), Búsqueda Local Iterada (ILS), Algoritmo Genético (GA), Algoritmo Memético (MA), Algoritmos Cooperativos.

1. INTRODUCCIÓN

La optimización combinatoria tiene una enorme aplicabilidad en diversas ramas del conocimiento, con su uso se busca obtener las mejores soluciones posibles a modelos matemáticos que representan problemas del mundo real. La búsqueda de secuencias binarias de baja autocorrelación (LABS) es uno de los problemas más desafiantes de optimización combinatoria. En las últimas décadas, muchos enfoques de diferentes disciplinas se han utilizado

para abordar este problema, pues tiene muchas aplicaciones en diversas áreas (química, física, telecomunicaciones, entre otras). Por esta razón la solución del problema del LABS ha sido estudiada a través de métodos exactos y a través de diferentes heurísticas, obteniéndose así algunos de los resultados óptimos conocidos hasta la fecha de varias instancias.

Las metaheurísticas son métodos que permiten mejorar la consecución de soluciones de alta calidad en poco tiempo y con recursos computacionales limitados. En la literatura existen un sinnúmero de metaheurísticas originadas desde campos científicos. Las metaheurísticas se pueden clasificar mediante diferentes parámetros. Uno de esos parámetros consiste en el uso o no de algún mecanismo de memoria. Las metaheurísticas sin memoria generalmente están basadas en procesos markovianos. Un ejemplo de estos esquemas se puede ver en los algoritmos de Hill Climbing (HC), de Búsqueda Local Iterada (ILS) y en los Algoritmos Genéticos (GA). En años recientes, se ha mostrado que estas metaheurísticas por si solas no evitan el estancamiento en mínimos locales, debido a esto se han propuesto mecanismos híbridos, es decir la composición de 2 o más metaheurísticas. Esta composición pueden ser integrativa o cooperativa, dando origen a esquemas como el Algoritmo Memético (MA) (Cotta & Moscato, 2011) y esquemas cooperativos como el planteado en (Amaya et al., 2011).

El artículo se estructura como sigue. La sección 2 explica el problema del LABS y se muestran algunos de los trabajos hechos en esta área. En la sección 3 se detallan los diferentes enfoques de algoritmos utilizados, haciendo hincapié en las decisiones de diseño. La sección 4 muestra los resultados obtenidos en las instancias seleccionadas para el problema del LABS, así como los criterios de comparación. Por último las conclusiones se presentan en la sección 5.

2. SECUENCIAS BINARIAS DE BAJA AUTOCORRELACIÓN (LABS)

El problema del LABS es considerado un problema de optimización combinatoria, cuyo objetivo es obtener una secuencia $S = S_1 S_2 \dots S_n$ de longitud n y donde $S_i \in \{-1, 1\}$ con $1 \leq i \leq n$, lo cual es equivalente a $S \in \{-1, 1\}^n$, con el fin de minimizar la función de energía $E(S)$, de la siguiente forma:

$$E(S) = \sum_{k=1}^{n-1} C_k^2(S) \quad (1)$$

Es decir, corresponde a la suma cuadrática de la función de autocorrelación $C_k(S)$, donde,

$$C_k(S) = \sum_{i=1}^{n-k} S_i S_{i+k} \quad (2)$$

Para medir la calidad de una determinada secuencia (Golay, 1977) ha propuesto una medida denominada factor de mérito, definida como

$$F(S) = \frac{n^2}{2E(S)} \quad (3)$$

Lo cual conduce a una definición alternativa como lo plantean (Gallardo et al, 2009), donde el problema ahora se convierte en un problema de maximización combinatoria:

$$F_n = \max_{S \in \{-1,1\}^n} F(S) \quad (4)$$

Para visualizar mejor la descripción del problema del LABS, a continuación se presenta un ejemplo utilizando como referencia la instancia $n = 8$. En primer lugar, supóngase la secuencia $S_1 = -1, 1, 1, -1, -1, 1, 1, -1$ a partir de la cual se calculan las funciones de correlación mediante la ecuación (2), obteniéndose $C_1(S_1) = -1, C_2(S_1) = -6, C_3(S_1) = 1, C_4(S_1) = 4, C_5(S_1) = -1, C_6(S_1) = -2, C_7(S_1) = 1$ con un valor de energía $E(S_1) = 60$ y un factor de merito $F(S_1) = 0,533$. Ahora supóngase la secuencia $S_2 = 1, 1, -1, -1, -1, 1, 1, -1$. En este caso se obtiene $C_1(S_2) = -1, C_2(S_2) = -4, C_3(S_2) = -3, C_4(S_2) = 0, C_5(S_2) = 3, C_6(S_2) = 0, C_7(S_2) = -1$ con un valor de energía $E(S_2) = 36$ y un factor de merito $F(S_2) = 0,889$. De lo anterior se desprenden varios elementos que comentar. Primero, la secuencia S_2 es mejor que S_1 debido a que $E(S_2) < E(S_1)$ (o en su defecto $F(S_2) < F(S_1)$). En segundo lugar, dos secuencias con la misma cantidad de bits del mismo signo (positivo o negativo) dependen de la distribución de los mismos para minimizar la energía.

2.1. TRABAJO PREVIO

Durante las últimas décadas el problema del LABS ha sido abordado en la literatura científica mediante métodos exactos y heurísticos. En (Golay, 1982), se publicaron las primeras soluciones óptimas calculadas mediante la realización de una búsqueda de enumeración exhaustiva. Mertens en (Mertens, 1996) utilizó un método de Branch&Bound y un procedimiento de ruptura de simetría para resolver el problema. En (Dotu & Van Hentenryck, 2006) se presenta el algoritmo de búsqueda estocástica local como Tabu Search (TS), el cual fue capaz de alcanzar óptimos globales más rápido que los métodos de enumeración. (Gallardo et al., 2007) desarrollaron algoritmos meméticos (MA) dotados de un mecanismo de TS inspirado por (Dotu & Van Hentenryck, 2006). Más concretamente, su investigación ofrece un análisis de las diferentes estrategias de búsqueda locales, que son usadas de manera integrativa y bajo un esquema lamarkiano dentro de los MA. Sus resultados mostraron que el algoritmo propuesto fue eficiente en el contexto planteado, realizando la búsqueda de soluciones óptimas para diferentes instancias del problema del LABS de forma más rápida que sus predecesores. En (Halim et al., 2008) los autores adoptan y mejoran significativamente el marco de TS de (Dotu & Van Hentenryck, 2006), por lo que se tienen resultados que mejoran el estado de la técnica de los MA. Según los autores, las mejoras se obtuvieron luego un análisis profundo de la trayectoria de búsqueda, y el establecimiento de un resultado más eficaz del proceso de diversificación. En (Hulianytskyi & Sokol, 2011) ponen a prueba dos marcos diferentes de metaheurísticas para la solución del problema del LABS, a saber: Tabu Search (TSv7) y un algoritmo memético optimizado. Hacen hincapié en operadores eficaces de recalcular de energía, concluyendo que estos enfoques muestran un rendimiento similar que cambia de forma idéntica con el crecimiento de la complejidad del problema.

3. METODOLOGÍA

Las metaheurísticas utilizadas en la presente investigación son metaheurísticas sin mecanismos de memoria, a saber: el algoritmo de Hill Climbing (HC), Búsqueda Local Iterada (ILS), Algoritmo Genético (GA), un Algoritmo Memético (MA) y una versión basada en Modelos Cooperativos. A continuación se describen de forma resumida cada una de ellas:

3.1 HILL CLIMBING (HC)

Se considera un enfoque de búsqueda local bien conocido que está basado en todos los pares de la vecindad, es decir, dos permutaciones son vecinas si sólo difieren en dos posiciones en la secuencia. Un enfoque de Hill

Climbing ascendente (HC) se define sobre la base de la estructura de vecindad de la solución actual que es recorrida, y la mejor solución encontrada se toma como la nueva solución actual, siempre que sea mejor que la actual (de lo contrario, se considera que existe un estancamiento).

3.2. BÚSQUEDA LOCAL ITERADA (ILS)

La Búsqueda Local Iterada (ILS) (Stützle, 1999) es una metaheurística que propone una estrategia exploratoria de mejora de las soluciones obtenidas por una determinada heurística mediante un proceso de iteración. En el esquema se incluye una heurística base que mejora los resultados mediante la repetición de dicha heurística. La estrategia ILS actúa de la siguiente forma: dada una solución obtenida por la heurística base, se aplica un cambio o mutación que da lugar a una solución intermedia. La aplicación de la heurística base a esta solución intermedia aporta una nueva solución que, si supera cierto criterio de aceptación, pasa a ser la nueva solución del ciclo de iteración. Aunque la heurística base suele ser una búsqueda local, puede aplicarse cualquier otra estrategia, determinística o no.

3.3 ALGORITMO GENÉTICO (GA)

Los algoritmos genéticos (GA) son técnicas probabilísticas de optimización global basadas en mecanismos observados en la selección natural y en la teoría de selección natural de Darwin. Los GA encuentran un notable balance entre la exploración y el aprovechamiento del espacio de búsqueda. (Schmitt, 2001). Los GA son menos susceptibles de quedar atrapados en óptimos locales en comparación con los métodos de búsqueda local. Los GA se consideran una metaheurística basada en poblaciones, ya que cada individuo correspondería a una potencial solución. Los elementos fundamentales al diseñar y aplicar un algoritmo genético son: la función objetivo, la representación genética y operadores genéticos.

3.4 ALGORITMO MEMÉTICO (MA)

Los algoritmos meméticos (MA) (Moscato & Cotta, 2003) son procedimientos basados en población que se combinan con esquemas de búsqueda local junto con los operadores genéticos como el cruce y la mutación y que han mostrado ser técnicas efectivas y rápidas para diversos problemas. Los MA poseen las ventajas de los métodos basados en poblaciones, así como la posibilidad de mejorar los resultados por medio de búsqueda local realizando un proceso de intensificación en las zonas del espacio de búsqueda que resultan prometedoras. Los algoritmos meméticos están inspirados en los algoritmos evolutivos y son considerados esquemas híbridos. Los elementos fundamentales al diseñar y aplicar un algoritmo memético son: la función objetivo, la representación genética, los operadores genéticos, el método de búsqueda local y su aplicación.

3.5 ALGORITMOS COOPERATIVOS

El modelo consiste de un pool de agentes (Amaya, 2010) que son cargados con una metaheurística particular y con soluciones iniciales aleatorias para explorar subespacios de búsqueda del problema de forma concurrente. El proceso de búsqueda es realizado por cada agente durante una cantidad de tiempo particular (o número de evaluaciones si fuese el caso), luego de lo cual se procede a elegir la mejor solución encontrada por cada agente y se distribuye en el modelo cooperativo mediante una topología de comunicación específica. Las topologías de comunicaciones planteadas son:

- **Ring.** Donde existe una lista circular de agentes en el que cada nodo sólo envía la información a su sucesor.
- **Broadcast.** Una topología en la que la mejor solución en cada punto de sincronización se transmite a todos los agentes.
- **Random.** Una topología donde cualquiera de dos agentes puede eventualmente comunicarse en cualquier punto de sincronización.

En general, se utiliza la siguiente notación para referirse a un mecanismo cooperativo. Sea ϕ un algoritmo cooperativo que está compuesto por n agentes (a_1, a_2, \dots, a_n) , una topología de comunicación $\Lambda \in \{Broadcast, Random, Ring\}$ y un número de ciclos $cyc\ max$, entonces, $\phi = cyc\ max\ \Lambda(a_1, a_2, \dots, a_n)$

4. RESULTADOS

Se utilizaron las instancias de 51 a 60 bits del problema del LABS. Con el objetivo de tener una medida comparativa para los diferentes experimentos se utilizó como medida el número de evaluaciones. Se estimó la ejecución equivalente a 5.4×10^5 s para 9×10^8 evaluaciones C:\Users\Jhon Amaya\Dropbox\Maria\Article\Article_Final.html LyXconv\Article_Final2.html - fn1x0. Se diseñaron los algoritmos en Java 1.7.0 y se ejecutaron 30 veces para cada instancia en una máquina Intel(R) Core(TM)2 Duo CPU E7400 @ 2.80GHz con SO Debian Linux.

Los parámetros utilizados para cada instancia se describen a continuación. En el algoritmo de Hill Climbing se eligió una vecindad de $4n$ individuos --- n es el número de bits de la secuencia--- y cada vez que la búsqueda local se considera estancada esta se reactiva a partir de una solución seleccionada al azar. El ILS utiliza una versión basada en la propuesta de (Blum et al., 2005), es decir, HC como método de búsqueda local con exploración parcial de la vecindad, y una perturbación basada en el intercambio de bloques, llamado RBI. El parámetro NIC que es un contador del número de iteraciones sucesivas sin mejora de la solución actual se colocó en 2000. En cuanto al GA, empleamos un modelo elitista estacionario en el cual se reemplaza al peor individuo de la población (población con 20 individuos, probabilidad cruce = 1.0 y la probabilidad de mutación = $1/n$), la selección consiste en un torneo binario, se utiliza un cruce denominado APX (Larrañaga et al., 2003) y la mutación se realiza mediante la aplicación de RBI. Se diseñaron 2 versiones particulares de algoritmos meméticos, modificando el enfoque de mejora local, para ello se utilizaron los métodos de búsqueda local definidos en la Sección 3.1 y 3.2, es decir, HC e ILS, llamando a las diversas versiones de los MA como MAHC y MAILS, respectivamente. Los parámetros utilizados por los diferentes MA son idénticos a los usados en el algoritmo genético. Además, en todos los MA, la búsqueda local se aplica a cualquier individuo con una probabilidad de $p_{LS} = 0,05$ y como máximo $LSevals = 200$ evaluaciones. Se definieron algoritmos cooperativos con diversas topologías cooperativas, con un número de agentes igual 3, y para las topologías, se consideraron dos propuestas diferentes: una en la cual cada agente ejecuta una metaheurística diferente en el conjunto {MAILS, ILS, MAHC}, y otra en la que todos los agentes implementan MAILS. Una vez más, se imponen los mismos límites de tiempo y se tiene en cuenta el $cycmax$ igual a 3. En total 6 algoritmos a saber : 3Br(MAILS,ILS,MAHC)_20, 3Br(MAILS,ILS,MAHC)_20, 3Br(MAILS,ILS,MAHC)_20, 3Br(MAILS,MAILS,MAILS)_20, 3Br(MAILS,MAILS,MAILS)_20 y 3Br(MAILS,MAILS,MAILS)_20. En la Tabla 1 se muestran los resultados computacionales correspondientes.

Para determinar el impacto de los diferentes parámetros se realizó un análisis de rangos como se muestra en la Figura 1. Del análisis de rangos se desprende que los mejores algoritmos son la familia de algoritmos cooperativos $3\Lambda(MAILS,MAILS,MAILS)_20$. Además, se observa que los algoritmos híbridos son mejores que sus algoritmos constitutivos.

A fin de determinar la incidencia de los parámetros de los algoritmos utilizados, se llevo a cabo una nueva experimentación bajo las mismas consideraciones anteriores en los algoritmos basados en poblaciones, es decir GA, MAILS, MAHC, $3\Lambda(MAILS,MAILS,MAILS)$ y $3\Lambda(MAILS,ILS,MAHC)$. En particular se extendieron los algoritmos basados en poblaciones con 30 y 50 individuos. En las Tablas 2 y 3 se muestran los resultados computacionales correspondientes. Del análisis de rangos (ver Figura 2) se desprende que los mejores algoritmos en su mayoría presentan mejores resultados para poblaciones superiores a 20 pero no necesariamente un aumento del número de individuos tiene una relación proporcional con el rendimiento del algoritmo basado en poblaciones.

Tabla 1. Resultados computacionales de las metaheurísticas HC, ILS y GA, MAHC y MAILS, así como los cooperativos 3A(MAILS,ILS,MAHC) y 3A(MAILS,MAILS,MAILS).

Método	Parámetro	51	52	53	54	55	56	57	58	59	60
HC	Media	276,3	290,7	302,3	320,3	339,8	346,4	364,8	387,9	404,6	416,4
	Desv. est.	15,9	22,4	16,7	15,7	17,5	27,3	24,2	25,3	28,9	20,7
ILS	Media	229,8	238,0	247,3	268,6	273,1	282,5	293,7	311,9	321,4	330,4
	Desv. est.	18,7	13,0	16,9	19,3	21,7	16,2	28,0	24,4	24,6	28,5
GA_20	Media	264,3	274,1	283,1	299,5	311,5	323,1	331,2	357,0	358,2	378,1
	Desv. est.	16,4	26,1	25,1	22,5	29,0	28,5	28,0	26,2	26,2	31,4
MAHC_20	Media	262,2	266,1	282,9	297,1	305,3	321,1	326,7	347,9	361,9	371,1
	Desv. est.	19,2	23,3	25,5	22,3	26,2	28,9	20,2	30,7	35,3	32,6
MAILS_20	Media	166,5	177,2	179,2	197,9	193,1	206,8	219,3	227,9	228,7	240,1
	Desv. est.	8,6	6,1	10,0	11,5	14,4	9,3	12,0	7,2	17,7	13,9
3Br(MAILS,ILS,MAHC)_20	Media	172,3	181,2	186,7	200,3	204,2	209,2	222,0	231,7	241,0	249,2
	Desv. est.	9,3	9,1	13,3	10,5	14,2	8,9	10,5	13,3	16,8	11,6
3Ra(MAILS,ILS,MAHC)_20	Media	173,3	187,3	188,1	207,3	204,9	213,2	232,3	234,6	237,5	253,7
	Desv. est.	12,4	21,5	16,2	19,2	19,0	21,9	21,7	13,0	19,3	19,0
3Ri(MAILS,ILS,MAHC)_20	Media	170,9	182,3	187,5	198,2	197,1	211,7	222,0	229,5	242,5	248,3
	Desv. est.	7,1	10,2	13,1	9,5	18,5	11,9	12,9	13,4	16,3	11,4
3Br(MAILS,MAILS,MAILS)_20	Media	163,1	174,9	174,7	191,8	192,7	198,7	214,1	220,2	223,5	239,5
	Desv. est.	6,5	5,2	7,4	10,6	15,0	6,1	13,0	9,2	16,0	11,4
3Ra(MAILS,MAILS,MAILS)_20	Media	162,7	176,8	175,7	192,3	190,7	202,0	212,7	220,7	226,3	238,4
	Desv. est.	7,4	6,0	7,6	9,4	14,4	7,0	13,0	10,2	13,3	11,0
3Ri(MAILS,MAILS,MAILS)_20	Media	163,4	176,3	177,5	189,4	185,4	202,3	216,4	220,2	226,3	239,2
	Desv. est.	7,2	7,0	8,9	11,8	15,3	7,8	9,9	9,5	13,8	8,9

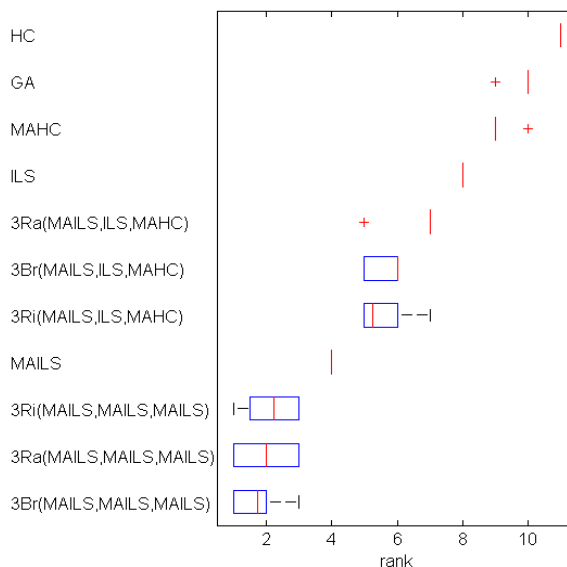


Figura 1: Distribución de rangos de cada algoritmo en todas las instancias. Cada caja incluye el segundo y tercer cuartil de la distribución, la mediana es marcada con una línea vertical.

Tabla 2. Resultados computacionales de las metaheurísticas con 30 individuos.

Método	Parámetro	51	52	53	54	55	56	57	58	59	60
GA30uc	Media	259,1	265,2	279,7	281,7	299,1	314,8	322,1	330,6	351,0	365,6
	Desv. est.	20,7	22,8	23,9	27,8	24,2	31,4	23,9	26,1	26,6	21,5
MA30HCFdot2	Media	251,0	266,3	281,1	291,0	303,4	312,9	335,2	333,3	348,9	370,7
	Desv. est.	21,5	16,8	23,4	22,2	23,8	23,3	19,3	22,3	26,3	22,9
MA30ILSFdot2	Media	167,8	176,5	178,0	194,2	190,5	204,1	214,8	224,2	231,5	237,2
	Desv. est.	7,3	6,3	7,5	12,2	16,7	9,1	13,0	9,1	14,4	11,8
Br(MAILS,ILS,MAHC)3i30	Media	170,9	182,8	186,1	199,3	203,4	209,3	226,7	227,9	238,1	246,4
	Desv. est.	9,1	8,8	11,9	8,5	15,1	10,9	10,8	12,9	16,5	11,0
Ra(MAILS,ILS,MAHC)3i30	Media	173,7	183,9	196,1	204,3	202,3	218,4	230,7	232,5	243,7	250,0
	Desv. est.	13,8	9,7	28,3	15,5	23,8	23,3	22,0	20,1	19,6	12,1
Ri(MAILS,ILS,MAHC)3i30	Media	172,5	181,3	188,9	197,7	200,3	212,1	222,8	227,9	235,3	248,1
	Desv. est.	8,8	8,7	12,1	12,3	13,1	13,1	12,7	11,7	12,5	11,3
Br(MAILS,MAILS,MAILS)3i30	Media	163,0	176,5	172,4	189,9	187,8	200,4	214,3	216,5	220,5	237,3
	Desv. est.	6,3	5,9	3,7	12,2	14,0	8,7	10,0	9,6	15,3	9,0
Ra(MAILS,MAILS,MAILS)3i30	Media	165,7	175,2	177,1	187,3	191,5	202,5	216,5	221,8	231,3	239,9
	Desv. est.	6,2	5,5	8,8	11,7	12,7	7,1	10,2	9,0	12,2	10,4
Ri(MAILS,MAILS,MAILS)3i30	Media	162,5	176,8	175,3	190,2	187,1	201,2	212,1	219,7	231,0	239,5
	Desv. est.	7,6	6,0	7,9	11,6	14,7	6,7	9,7	12,3	13,9	10,2

Tabla 3. Resultados computacionales de las metaheurísticas con 50 individuos.

Método	Parámetro	51	52	53	54	55	56	57	58	59	60
GA50uc	Media	233,7	252,9	266,1	283,3	293,5	298	309,5	319,7	343,1	351,5
	Desv. est.	23,5	17,1	15,7	24,8	20,4	19,4	24,3	25,5	20,6	28,9
MA50HCFdot2	Media	244,7	256,1	268,3	282,7	286,1	307,5	320,1	340,5	351,9	356,1
	Desv. est.	21,4	21	18,1	23,7	19,6	23,5	23,6	20,9	20,6	29,2
MA50ILSFdot2	Media	165,7	174,9	177,5	195	193,7	201,1	216,9	224,2	233,9	241,5
	Desv. est.	6,5	6,6	6,2	10,7	15,1	7,4	8,3	11,8	13	9,2
Br(MAILS,ILS,MAHC)3i50	Media	166,7	181,2	183,1	201,9	198,9	207,3	224,8	229,5	241,5	247,9
	Desv. est.	10,2	8,6	12,1	10,4	14	10,1	9,8	10,3	14,8	11,4
Ra(MAILS,ILS,MAHC)3i50	Media	175,3	193,3	194,3	203,8	211,7	213,7	231,7	241,8	238,5	249,7
	Desv. est.	14,4	22,7	21,3	17,5	24,4	22,6	29,2	30,3	20,5	20,5
Ri(MAILS,ILS,MAHC)3i50	Media	168,2	182,5	187,1	200,6	196,6	211,7	224	232,7	240,2	248,9
	Desv. est.	8,2	9	12,2	5,3	15,4	10,2	13,2	9,8	14	13,9
Br(MAILS,MAILS,MAILS)3i50	Media	164,5	173,6	175,2	191,8	187,7	200,8	214,8	218,6	220,6	238,8
	Desv. est.	6,7	5,8	7,6	11,2	12	5,8	10,7	10,3	14,5	9,8
Ra(MAILS,MAILS,MAILS)3i50	Media	164,9	176	178	190,2	192,6	201,2	211,9	218,3	224,1	236,7
	Desv. est.	7,2	7,3	7,7	12	12,5	6,1	10,7	10,4	16,8	8,9
Ri(MAILS,MAILS,MAILS)3i50	Media	164,1	172,9	174,9	193,4	187,3	200,7	214,4	217,8	227	239,7
	Desv. est.	6,5	5,5	7,1	11,2	12,7	7,1	11,8	10,8	14,5	8,6

Tabla 4. Resultados computacionales de los algoritmos cooperativos con 4 ciclos.

Método	Parámetro	51	52	53	54	55	56	57	58	59	60
Br(MAILS,MAILS,MAILS)4	Media	165,1	176,4	175,6	192,1	187,3	202,1	213,6	220,5	231,0	237,5
	Desv. est.	7,0	5,3	4,8	10,9	14,6	7,4	11,5	11,9	15,3	7,8
Ra(MAILS,MAILS,MAILS)4	Media	164,9	176,3	179,5	196,9	186,3	202,4	215,9	219,1	226,2	238,3
	Desv. est.	6,9	6,6	9,7	9,4	14,3	8,0	10,3	9,6	16,8	9,1
Ri(MAILS,MAILS,MAILS)4	Media	163,5	176,8	176,3	191,3	189,3	201,5	211,7	223,1	227,7	240,1
	Desv. est.	7,4	6,8	7,9	11,2	14,7	9,1	12,6	8,4	14,3	8,4
Br(MAILS,MAILS,MAILS)4i30	Media	166,2	175,2	176,5	189,4	186,6	201,2	213,6	218,9	227,8	238,0
	Desv. est.	6,5	6,7	7,6	11,6	13,0	7,2	12,0	9,6	13,9	8,5
Ra(MAILS,MAILS,MAILS)4i30	Media	164,5	174,3	174,8	195,0	190,6	202,7	214,4	217,3	222,6	237,3
	Desv. est.	6,6	5,3	7,2	10,7	12,4	7,1	9,4	8,1	14,7	9,2
Ri(MAILS,MAILS,MAILS)4i30	Media	161,1	177,7	176,5	193,4	191,5	203,3	213,1	219,1	228,5	236,7
	Desv. est.	6,8	7,4	7,0	10,7	13,9	7,8	9,2	11,1	14,3	11,1
Br(MAILS,MAILS,MAILS)4i50	Media	162,5	175,7	174,8	191,0	190,2	200,3	212,9	219,7	229,9	243,6
	Desv. est.	7,6	5,4	6,2	10,9	9,8	6,4	12,0	9,2	13,6	8,0
Ra(MAILS,MAILS,MAILS)4i50	Media	164,2	175,6	175,9	194,2	191,1	200,3	209,3	215,7	226,2	239,2
	Desv. est.	7,5	4,9	8,6	9,1	14,4	6,6	11,2	10,4	15,1	9,6
Ri(MAILS,MAILS,MAILS)4i50	Media	163,0	175,3	176,7	190,7	184,9	198,3	207,9	223,7	224,9	240,0
	Desv. est.	6,6	6,3	7,8	11,0	13,9	5,4	10,2	7,7	15,0	8,4

Tabla 5. Resultados computacionales de los algoritmos cooperativos con 5 ciclos.

Método	Parámetro	51	52	53	54	55	56	57	58	59	60
Br(MAILS,MAILS,MAILS)5	Media	164,7	174,1	175,7	192,3	188,6	200,5	216,5	218,3	227,0	238,5
	Desv. est.	6,7	7,6	8,1	11,3	15,9	6,6	9,9	10,6	15,0	10,3
Ra(MAILS,MAILS,MAILS)5	Media	164,3	173,9	173,6	193,9	188,7	201,3	213,5	222,6	227,1	236,8
	Desv. est.	7,4	7,3	6,5	9,7	12,7	8,3	11,0	9,7	15,8	9,1
Ri(MAILS,MAILS,MAILS)5	Media	164,2	175,9	178,1	191,3	188,7	202,0	213,2	222,3	232,9	236,3
	Desv. est.	5,7	7,1	9,2	11,4	15,1	7,9	11,9	9,2	12,8	10,3
Br(MAILS,MAILS,MAILS)5i30	Media	163,0	176,5	172,4	189,9	187,8	200,4	214,3	216,5	220,5	237,3
	Desv. est.	6,3	5,9	3,7	12,2	14,0	8,7	10,0	9,6	15,3	9,0
Ra(MAILS,MAILS,MAILS)5i30	Media	161,7	174,1	176,9	191,8	190,1	202,7	216,4	221,3	226,5	240,4
	Desv. est.	8,4	6,4	8,6	10,8	14,7	8,2	10,3	9,3	15,7	10,3
Ri(MAILS,MAILS,MAILS)5i30	Media	164,5	176,1	176,0	192,1	188,5	201,1	214,1	220,5	226,1	239,1
	Desv. est.	6,4	7,2	8,1	11,5	15,4	7,0	11,1	9,1	15,3	8,7
Br(MAILS,MAILS,MAILS)5i50	Media	161,8	175,6	178,4	188,6	186,6	200,8	208,9	219,1	230,1	237,1
	Desv. est.	7,3	5,7	9,2	11,3	13,7	7,5	9,8	12,2	15,8	10,3
Ra(MAILS,MAILS,MAILS)5i50	Media	162,9	175,6	176,1	190,5	189,3	201,6	218,0	219,4	221,8	237,3
	Desv. est.	7,8	5,4	8,0	12,1	13,5	7,8	10,1	11,2	14,3	9,2
Ri(MAILS,MAILS,MAILS)5i50	Media	162,7	173,5	177,5	193,9	191,8	202,1	211,3	215,4	221,0	238,8
	Desv. est.	5,9	6,5	8,6	10,6	12,8	8,0	9,0	9,4	15,3	11,1

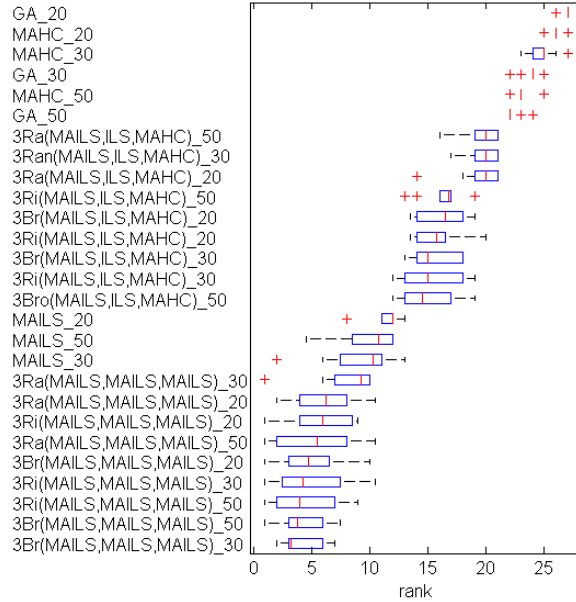


Figura 2: Distribución de rangos de los algoritmos basados en poblaciones en todas las instancias.

Adicionalmente, se llevaron a cabo experimentos con relación a los parámetros de los algoritmos cooperativos, como por ejemplo, el número de ciclos. En particular, se probaron con 3, 4 y 5 ciclos. Las Tablas 4 y 5 muestran los resultados correspondientes. En la Figura 3 se muestra una relación de los diferentes parámetros de los algoritmos cooperativos para todos los experimentos mencionados. En la Figura 3.a se observa la relación de las topologías en particular Random vs {Ring, Broadcast}, se observa que Broadcast logra superar a Random en 8 instancias y ring en 6 instancias. En la Figura 3.b se observan los números de ciclos relacionando 3 ciclos con 4 y 5 ciclos. Se muestra que con 5 ciclos parece obtener mejores resultados. Finalmente, en la Figura 3.c se observa la relación de poblaciones con 20 individuos vs {30,50} individuos. Aquí los resultados son un poco más dispersos pero se puede observar que los experimentos con 30 individuos son ligeramente superiores.

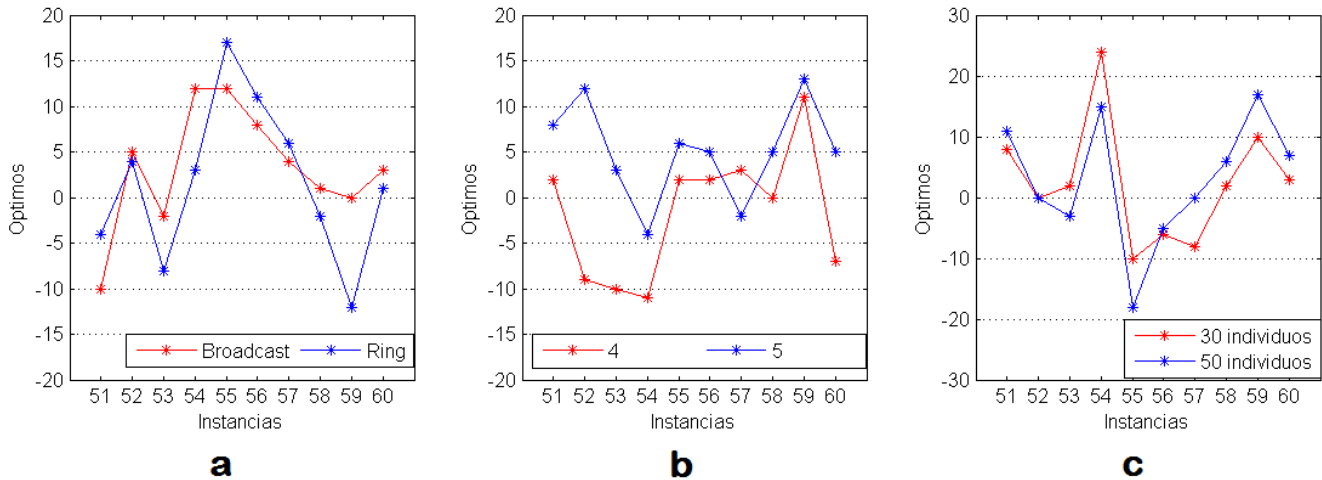


Figura 3: Relación de los diferentes parámetros de los algoritmos cooperativos.

5. CONCLUSIONES

Se mostraron dos técnicas híbridas novedosas sobre el problema del LABS. Una integrativa como un algoritmo memético basado en ILS y otra basada en mecanismos cooperativos. Las propuestas integrativas y cooperativas muestran un comportamiento en general mejor que las metaheurísticas clásicas como GA o ILS para este problema, pero no tan buenas como aquellas basadas en mecanismos constructivos o en mecanismos basados en memoria como por ejemplo MA con TS, sin embargo, se observa que el mecanismo cooperativo tiene margen de mejora al implementar agentes con metaheurísticas con algún mecanismo con memoria, ya que los experimentos sugieren que a medida que mejoran las metaheurísticas constitutivas mejor es el comportamiento del algoritmo cooperativo. Como trabajo futuro se proponen el estudio de nuevas propuestas cooperativas e híbridas, así como la prueba en nuevas instancias del problema del LABS.

AGRADECIMIENTO

Nuestro más sincero agradecimiento al Decanato de Investigación de la Universidad Nacional Experimental del Táchira por su apoyo y soporte durante el desarrollo de esta investigación.

REFERENCIAS

- Amaya, J. E., Cotta, C. and Fernández, A. (2010). “Hybrid Cooperation Models for the Tool Switching Problem Nature Inspired Cooperative Strategies for Optimization” (NICSO 2010), Springer Berlin, 284, 39-52.
- Blum, C., Blesa, M., and Roli, A. (2005). “Combining ILS with an Effective Constructive Heuristic for the Application to Error Correcting Code Design”. In 6th Metaheuristics International Conference (MIC'2005), Vienna, Austria, pp. 114--119.
- Dotu, I. and Van Hentenryck, P. (2006). “A note on low autocorrelation binary sequences,” in Principles and Practice of Constraint Programming - CP 2006, ser. Lecture Notes in Computer Science, F. Benhamou, Ed. Springer Berlin / Heidelberg, vol. 4204, pp. 685–689.
- Gallardo, J. E., Cotta, C. and Fernandez, A. (2007). “A memetic algorithm for the low autocorrelation binary sequence problem,” in Proceedings of the 9th annual conference on Genetic and evolutionary computation. GECCO '07. New York, NY, USA: ACM, pp. 1226–1233.
- Golay, M. (1977). “Sieves for low autocorrelation binary sequences,” Information Theory, IEEE Transactions on, vol. 23, no. 1, pp. 43 – 51.
- Golay, M. (1982). “The merit factor of long low autocorrelation binary sequences (corresp.),” Information Theory, IEEE Transactions on, vol. 28, no. 3, pp. 543 – 5492.
- Halim, S. Yap, R. and Halim, F. (2008). “Engineering stochastic local search for the low autocorrelation binary sequence problem,” in Principles and Practice of Constraint Programming. Lecture Notes in Computer Science, P. Stuckey, Ed. Springer Berlin, vol. 5202, pp. 640–645.
- Hulianytskyi, L. and Sokol, V.(2011). “Effective energy recomputation for low autocorrelation binary sequence problem,” International Journal Information Theories and Applications, vol. 18, no. 4, pp. 344–357.
- Mertens, S. (1996). “Exhaustive search for low-autocorrelation binary sequences,” Journal of Physics A: Mathematical and General, vol. 29, pp. L473–L481.
- Moscato, P., and Cotta, C. (2003). Una Introducción a los Algoritmos Meméticos. Inteligencia Artificial, Revista Iberoamericana de Inteligencia Artificial 7, pp. 131-148.
- Schmitt, L. M. (2001). Theory of genetic algorithms. Theoretical Computer Science 259, 1-2, 1---61.
- Stützle, T. (1999). Local search algorithms for combinatorial problems- analysis, algorithms and news applications. DISKI Dissertationen zur Künstlichen Intelligenz.

Authorization and Disclaimer

Authors authorize LACCEI to publish the paper in the conference proceedings. Neither LACCEI nor the editors are responsible either for the content or for the implications of what is expressed in the paper.