

# Metaheurísticas híbridas combinando algoritmos evolutivos y enfriamiento simulado: taxonomía y estudio comparativo y de sinergia

Francisco J. Rodríguez, Carlos García Martínez y Manuel Lozano

*Resumen*— El diseño de metaheurísticas con ideas procedentes de los campos de los algoritmos evolutivos y del enfriamiento simulado es una línea de investigación fructífera actualmente. En este trabajo, en primer lugar, clasificamos distintas instancias de este tipo de metaheurísticas de acuerdo a dos conocidas taxonomías para algoritmos híbridos. A continuación, realizamos un estudio experimental para comparar el comportamiento de las distintas instancias presentadas anteriormente. Finalmente, mostramos los resultados del estudio realizado para comprobar la existencia de sinergia en las aproximaciones híbridas estudiadas. Los diversos experimentos realizados proporcionan conclusiones útiles acerca de aquellos esquemas para combinar los algoritmos evolutivos y el enfriamiento simulado que pueden mejorar el rendimiento de esta clase de hibridaciones.

*Palabras clave*— metaheurísticas híbridas, enfriamiento simulado, algoritmos evolutivos

## I. INTRODUCCIÓN

Durante los últimos años han aparecido un gran número de algoritmos de búsqueda que no siguen de forma estricta los conceptos propuestos por una única metaheurística clásica (MH) [1], [2], sino que tratan de obtener lo mejor de un conjunto de MH (e incluso otros tipos de métodos de optimización), actuando de forma coordinada y complementaria para producir una sinergia beneficiosa de forma combinada. Estas aproximaciones se denominan *MH híbridas* (MHH) [3], [4].

El *enfriamiento simulado* (ES) [5], [6] destaca por ser el primer algoritmo que extiende los métodos de búsqueda local, con una estrategia que le permite escapar de los óptimos locales. Para ello, el ES permite movimientos hacia soluciones de peor calidad que la solución actual. La probabilidad de hacer este tipo de movimientos decrece durante el proceso de búsqueda. A pesar de ser un método propuesto en 1983, el ES es aún objeto de nuevos estudios, se utiliza para resolver muchos problemas de optimización y además se selecciona como componente de otros algoritmos de búsqueda [7], [8]. Precisamente, su papel destacado dentro del campo de las MH, genera gran interés en la realización de nuevos estudios para obtener modelos de ES más efectivos.

F.J. Rodríguez, M. Lozano: Universidad de Granada. Departamento de Ciencias de la Computación e Inteligencia Artificial. E-mail: fjrodriguez, lozano@decsai.ugr.es.

C. García Martínez: Universidad de Córdoba. Departamento de Informática y Análisis Numérico. E-mail: cgarcia@uco.es.

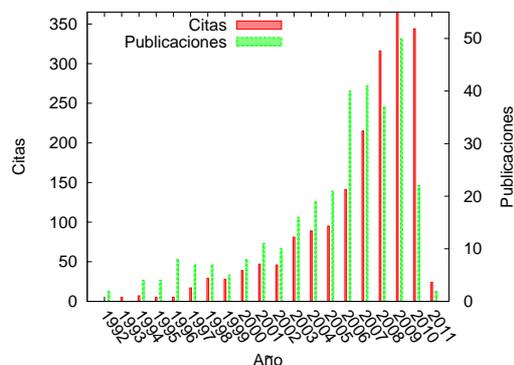


Fig. 1. Número de publicaciones y citas (Web of Science)

Los *algoritmos evolutivos* (AE) [9], [10], se basan en el concepto de una población de individuos, sobre los que se aplican una serie de operadores probabilísticos tales como mutación, selección y (a veces) cruce, para evolucionar hacia individuos con mejor valor de aptitud. La flexibilidad ofrecida por el paradigma evolutivo, permite obtener modelos especializados con el objetivo de actuar de la misma manera que otros métodos de búsqueda, pero de una forma más satisfactoria. Esta práctica es una alternativa reciente para diseñar nuevas MHH. En particular, existen varias propuestas de AE actuando como métodos de búsqueda local y consiguiendo resultados más satisfactorios [11], [12].

Las hibridaciones con AE son cada vez más populares debido a su habilidad para afrontar diversos problemas reales [13], [14]. En este trabajo, nos centramos particularmente en aquellas que consideran ES junto con AE para desarrollar nuevas MHH (MHH-AE/ES), debido a su destacado papel dentro del campo de las hibridaciones con AE. La visibilidad de este tema en el ISI Web of Science nos muestra la relevancia de las MHH-AE/ES actualmente. En la Figura 1, se puede observar el importante número de publicaciones y citas por año, mostrando además una tendencia creciente. Por lo tanto, podemos concluir que aunque los primeros trabajos en esta área datan de 1992, actualmente las MHH-AE/ES son un área de gran interés que cuente con una importante comunidad investigadora asociada a su estudio. De hecho, hoy en día, las MHH-AE/ES son la herramienta elegida para solucionar problemas reales pertenecientes a diversas áreas [15], [16].

El objetivo de este artículo es doble. Por una parte, pretendemos identificar las distintas aproximaciones para construir MHH-AE/ES y clasificarlas de acuerdo a una taxonomía para MHH-AE/ES basada en las taxonomías para MHH propuestas por Talbi [4] y Raidl [3]. Nuestro segundo objetivo es comparar experimentalmente el comportamiento de las distintas clases de MHH-AE/ES según la taxonomía y estudiar las relaciones de sinergia existentes en este tipo de hibridaciones.

El resto de este artículo está organizado de la siguiente forma. En la Sección II, presentamos diversas instancias existentes de MHH-AE/ES y las clasificamos de acuerdo a la taxonomía propuesta. En la Sección III, realizamos un estudio experimental para comparar el comportamiento de las MHH-AE/ES identificadas en la sección anterior. A continuación, en la Sección IV, comparamos el rendimiento de las MHH-AE/ES de acuerdo a la categoría a la que pertenecen. En la Sección V, realizamos un estudio de sinergia de las MHH-AE/ES comparadas. Finalmente, en la Sección VI, presentamos las conclusiones y el trabajo futuro.

## II. MHH BASADAS EN AE Y ES: VISIÓN GENERAL Y TAXONOMÍA

En esta sección, hemos clasificado distintas instancias de MHH-AE/ES encontradas en la literatura en dos grandes grupos en base a dos conocidas taxonomías [3], [4] para MHH:

1. *MHH colaborativas*. Se basan en el intercambio de información entre diferentes MH independientes, ejecutándose de forma secuencial o paralela.

2. *MHH integrativas*. En este caso, un algoritmo es considerado un componente subordinado dentro de una MH maestra que controla el proceso de búsqueda.

Dentro de las MHH colaborativas, definimos dos subcategorías:

1. *Colaborativas en equipo*. Diversas MH trabajan en paralelo e intercambian soluciones, parámetros, etc. Dentro de las MHH-AE/ES, encontramos dos aproximaciones:

- *Múltiples ES*. Considera un conjunto de ES que se ejecutan en paralelo y cooperan para explorar el espacio de búsqueda. Como ejemplos encontramos en la literatura los algoritmos ESA (evolutionary simulated annealing) [17] y SSSA (sample sort simulated annealing) [18]. En la aproximación *múltiples ES*, aunque no se considera la inclusión de AE, subyace el concepto de población de ES. Por este motivo, aunque difícilmente se puede hablar de hibridación en esta categoría, se incluye en el estudio.

- *Múltiples AE y ES*. De la misma forma que en la categoría anterior, se puede considerar un conjunto de ES y AE que se ejecutan en paralelo e intercambian periódicamente información. En este trabajo, se presenta una instancia que considera este tipo de hibridación combinando dos AE (en concreto CHC [19]) y dos procesos de ES, distribuidos en forma

de anillo (DCHCSA, distributed CHC and simulated annealing). En DCHCSA cada 75 evaluaciones de la función de adaptación en cada algoritmo, la solución actual de cada proceso de ES se envía al siguiente algoritmo CHC, reemplazando a su peor solución. De la misma forma, cada algoritmo CHC envía su mejor solución encontrada hasta el momento al siguiente proceso de ES.

2. *Colaborativas por relevos*. En este caso, las distintas MH se ejecutan secuencialmente. Para el caso concreto de las MHHs-AE/ES, podemos distinguir otras dos aproximaciones:

- *AE y después ES*. Al finalizar el AE, los individuos de la población son optimizados mediante un proceso de ES. En la literatura encontramos el algoritmo HHSAGA (highly hybrid simulated annealing and genetic algorithm) [20], que sigue este esquema.

- *ES y después AE*. La instanciación más directa de este esquema es un AE cuya población inicial es inicializada mediante un proceso de ES. En este trabajo, se presenta una implementación de este esquema denominada GA-PSA (genetic algorithm with population initialised by simulated annealing). El algoritmo genético utiliza una población con 60 individuos, mutación de un bit con probabilidad 0,006 y cruce en dos puntos. El procedimiento para inicializar la población del algoritmo genético consiste en añadir las distintas soluciones que va aceptando el proceso de ES. En el momento en que se rechaza una solución, se vuelve a partir de una solución aleatoria y se continúan añadiendo soluciones. Este proceso se repite hasta que se completa la población.

De igual modo que en las MHH colaborativas, dentro de las MHH integrativas, distinguimos entre integrativas en equipo e integrativas por relevos:

1. *Integrativas en equipo*. En esta clase de MHH, una MH (subordinada) se convierte en un componente de otra MH basada en población (maestra). En el caso del tipo de hibridaciones que nos ocupa, encontramos dos tipos:

- *Algoritmos meméticos*. En este caso un algoritmo genético a cargo del proceso de búsqueda utiliza un proceso de ES, dentro de la propia ejecución del algoritmo genético, como procedimiento de búsqueda local. El algoritmo AGA (annealing genetic algorithm) [21] sigue este esquema.

- *Componentes de AE basados en ES*. Esta aproximación consiste en definir distintos operadores del AE siguiendo los principios del ES. En la literatura podemos encontrar distintos componentes de los AE que han sido reemplazados o extendidos mediante ES: mecanismo de selección (guided evolutionary algorithm, GESA [22]), operadores de cruce y/o mutación (adaptive real-parameter simulated annealing-genetic algorithm, ARSAGA [23]) y esquemas de reemplazo (parallel recombinative simulated annealing, PRSA [24]).

2. *Integrativas por relevos*. Esta clase de MHH representa algoritmos en los que una determinada MH está embebida dentro de otra MH basada en trayec-

torias. En la literatura, encontramos dos ejemplos de MHH-AE/ES que siguen este patrón de diseño. En primer lugar, SALGeS (simulated annealing based on local genetic search) [25] presenta un AE diseñado específicamente para desempeñar la función de operador de vecindario dentro del ES. En segundo lugar, GAMSa (genetic algorithm based simulated annealing) [26] extiende la idea de SALGeS considerando la ejecución de múltiples procesos de ES que comparten un único algoritmo genético estacionario para desempeñar la función de operador de vecindario.

### III. ESTUDIO COMPARATIVO DE LAS MHH-AE/ES

En esta sección, realizamos un estudio experimental para comparar el rendimiento de las distintas MHH-AE/ES descritas en la sección anterior. Para la realización del mismo, hemos definido un conjunto de 27 problemas de optimización combinatoria modelados mediante variables binarias y sin restricciones. En la Tabla I, se especifica para cada problema su nombre, dimensión y valor de aptitud del óptimo global (o el mejor encontrado en la literatura). Todos ellos se han formulado como problemas de maximización. Para poder comparar los resultados de los diferentes algoritmos, ejecutaremos todos ellos hasta alcanzar un determinado número de evaluaciones de la función de aptitud ( $10^5$ ). La medida de eficiencia de cada algoritmo, para un determinado problema, se calcula como la media del mejor valor de aptitud encontrado a lo largo de cada una de las 50 ejecuciones independientes realizadas. Los parámetros utilizados por cada uno de los algoritmos son los recomendados en las publicaciones originales.

En la Tabla V, se muestran los resultados medios de cada algoritmo en cada uno de los problemas. Además, en dicha tabla, se muestran los resultados del *t*-test [27], que compara GAMSa (la MHH-AE/ES con mejor comportamiento) frente a las restantes aproximaciones híbridas, indicando para cada problema si los resultados de GAMSa son estadísticamente mejores (+), peores (-) o iguales (~) frente a los de sus competidores. De acuerdo a los resultados de esta tabla, podemos destacar lo siguiente:

- SALGeS y GAMSa alcanzan la mejor solución encontrada por cualquiera de las MHH-AE/SA en 23 de las 27 instancias.
- GAMSa obtiene mejores resultados que PRSA, ARSAGA, GESA, HHSAGA, SSSA y CSA para casi todos los problemas. En los restantes, el test realizado no muestra diferencias estadísticamente significativas.
- Solo GA-PSA, AGA, DCHCSA y SALGeS son capaces de superar a GAMSa en algunos problemas. GP-SA lo hace en un problema, AGA en dos, DCHCSA en cuatro y SALGeS en cinco.

Adicionalmente, se han utilizado test no paramétricos [28] para comparar los resultados de las

TABLA I  
PROBLEMAS CONSIDERADOS

Prob.	Nombre	$N$	$f^*$
1	Royal Road (400,8)	400	1
2	Trap	36	220
3	Deceptive	39	390
4	Bipolar Deceptive	396	1
5	Overlapping Deceptive	399	1
6	M-Sat(100,1200,3)	100	1
7	M-Sat(100,2400,3)	100	1
8	NKLand(48,4)	48	1
9	NKLand(48,12)	48	1
10	HIFF(2,5,true)	32	192
11	HIFF(3,4,false)	81	211
12	PPeaks(10,100)	100	1
13	PPeaks(100,100)	100	1
14	PPeaks(50,150)	150	1
15	PPeaks(50,200)	200	1
16	BQP(bqp50-1)	50	2098
17	BQP(bqp500-1)	500	116586
18	BQP(be120.3.3-1)	120	Des.
19	BQP(be200.8.5)	200	Des.
20	Maxcut(pm1s_80.6)	80	73
21	Maxcut(w09_100.2)	100	2738
22	Maxcut(g05_100.5)	100	1436
23	Maxcut(pw05_100.6)	100	8217
24	Maxcut(ising2.5-250.5555)	250	7919449
25	Mul. Knapsack p.(weish03)	30	4115
26	Mul. Knapsack p.(pet5)	28	12400
27	Mul. Knapsack p.(pb4)	29	95168

distintas MHH-AE/ES. En primer lugar, aplicamos el test de Iman-Davenport [29], detectando diferencias significativas de rendimiento entre los algoritmos comparados (el valor del estadístico, 117,166, es superior al valor crítico, 1,822). Después, aplicamos el test de Holm [30] para comparar el rendimiento del algoritmo con mejor ranking (GAMSa) con respecto al resto. En la Tabla III, se muestra, en la segunda columna, el ranking medio de cada algoritmo sobre el conjunto de problemas y además, en la tercera columna, los resultados del test de Holm. En esta columna, mostramos si el test de Holm ha encontrado diferencias significativas con respecto a GAMSa (+) o no (~). Según se puede observar en la tabla, existen diferencias significativas entre el rendimiento de GAMSa y CSA, ARSAGA, GESA, PRSA, SSSA, HHSAGA, GA-PSA y AGA.

Además, en la Tabla III, se muestran los resultados de aplicar el test de Wilcoxon con  $p$ -value = 0,05 y 0,1, para comparar los resultados de GAMSa y los de los algoritmos para los cuales el test de Holm no detectó diferencias significativas (DCHCSA y SALGeS). En dicha tabla, se muestran los valores de  $R^+$  (asociado a GAMSa) y  $R^-$  (asociado al competidor) del test. Si  $R^-$  es menor que  $R^+$  y que el valor crítico del test, GAMSa es estadísticamente mejor que el correspondiente competidor. Si  $R^+$  es inferior a  $R^-$  y al valor crítico, GAMSa es estadísti-

TABLA II  
COMPARATIVA, MEDIANTE EL TEST DE HOLM, DEL  
RENDIMIENTO DE LAS DISTINTAS MHH-AE/ES

MH-EA/SA	Ranking	Holm
GAMSA	1,81	Ganador
SALGeS	2,33	~
DCHCSA	2,74	~
AGA	4,48	+
GA-PSA	4,98	+
HHSAGA	6,11	+
SSSA	6,43	+
PRSA	7,81	+
GESA	8,59	+
ARSAGA	10,04	+
CSA	10,67	+

TABLA III  
GAMSA vs. SALGeS Y DCHCSA (TEST DE WILCOXON  
CON  $p$ -VALUE=0,05 Y 0,1 Y VALOR CRÍTICO = 107 Y 119,  
RESPECTIVAMENTE)

Algoritmos	$R^+$	$R^-$	Dif? 0,05/0,1
GAMSA vs DCHCSA	262	116	~/+
GAMSA vs SALGeS	259	113	~/+

camente peor que su competidor. Si ni  $R^+$  ni  $R^-$  son menores que el valor crítico, no hay diferencias significativas. La última columna indica si GAMSA es estadísticamente superior (+), inferior (-) o no hay diferencias significativas (~). De acuerdo a estos resultados, el test de Wilcoxon determina que GAMSA es estadísticamente superior con respecto a GAMSA Y DCHCSA.

#### IV. RENDIMIENTO DE LAS MHH-AE/ES POR CATEGORÍAS

En esta sección, estudiamos el comportamiento de las MHH-AE/ES por categorías para comprobar el rendimiento de cada grupo de algoritmos. En la Figura 2, vemos el ranking medio obtenido por cada algoritmo y la categoría a la que pertenece. Las distintas MHH-AE/ES han sido agrupadas por categorías, siendo la altura de cada columna proporcional a su ranking. Por lo tanto, a menor altura de una columna, mejor es el algoritmo asociado a la misma. Teniendo en cuenta esta figura, podemos obtener las siguientes conclusiones:

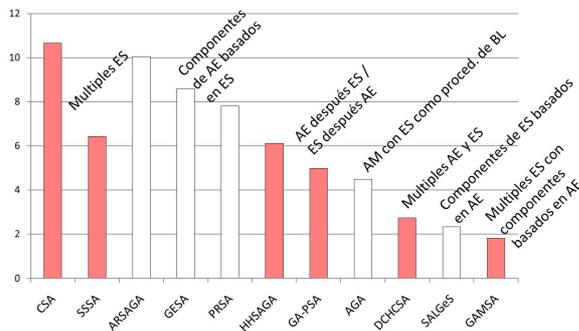


Fig. 2. Ranking medio de las MHH-AE/ES por categorías

- La categoría *múltiples ES* incluye al algoritmo CSA, que presenta el peor rendimiento de entre los algoritmos estudiados. El otro algoritmo perteneciente a esta categoría, SSSA, aparece en el grupo de los 5 algoritmos con peor ranking medio. Las tres instancias de algoritmos de la categoría *componentes de AE basados en ES* están también dentro de estos 5 algoritmos. Una característica compartida por estas dos categorías es que la hibridación emplea formulaciones incompletas de un AE o un ES, es decir, se incluyen principios de una de las MH dentro de la otra (población de soluciones en un ES, *múltiples ES*, y criterio de aceptación tipo ES en un AE, *componentes de AE basados en ES*).

- Los mejores resultados se obtienen en aquellas categorías que conciben la hibridación con formulaciones completas de ES y AE, cooperando de alguna forma.

- La aproximación integrativa está presente en 3 de las 4 MHH-AE/ES con mejor ranking. Estos son AGA (*AM con ES como procedimiento de búsqueda local*), SALGeS y GAMSA (perteneciendo ambos a la categoría *componentes de ES basados en AE*). En estas MHH, se especializa una determinada MH para que desempeñe un determinado papel dentro de la otra MH. En concreto, en AGA, el ES se encarga de refinar las soluciones obtenidas por el AE, mientras que en SALGeS y GAMSA, el AE actúa como operador de vecindario del ES.

- En [31], se estudió anteriormente la técnica de hibridación empleada por SALGeS y GAMSA, que consiste en reemplazar algunos componentes de una MH por AE adaptados para desempeñar la misma tarea que el componente al que reemplazan pero haciéndolo de una manera más efectiva y con un costo computacional relativamente bajo. En este trabajo, los autores presentan un algoritmo de búsqueda local iterativa con un método para generar perturbaciones, el cual es un micro AE que explora las soluciones en el vecindario de una solución particular. Los resultados competitivos obtenidos por este algoritmo frente a otros algoritmos del estado del arte para optimización binaria y el buen comportamiento de GAMSA Y SALGeS en el estudio previo evidencian la efectividad de este paradigma de hibridación.

- La aproximación colaborativa en equipo, a la que pertenece DCHCSA, completa el grupo de los 4 mejores algoritmos. Resulta interesante que sus resultados son mejores que los del esquema colaborativo por relevos (HHSAGA Y GA-PSA)

#### V. ESTUDIO DE LA SINERGIAS

Uno de los aspectos más importantes a la hora de estudiar el comportamiento de una MHH es analizar la sinergia producida por la combinación de las MH que la componen. De hecho, la principal motivación detrás de la hibridación es la explotación del carácter complementario de las diferentes estrategias de optimización involucradas en la MHH, es decir, se espera que los algoritmos híbridos se bene-

ficien de la sinergia [32]. Para comprobar el alcance del efecto de la sinergia cuando se combinan dos o más componentes, o si en realidad el efecto de la sinergia aparece o no, la práctica más usual consiste en comparar el rendimiento del algoritmo híbrido frente al de sus componentes [33], [34], [35]. La sinergia solo aparece cuando la combinación de diversos algoritmos presenta un mejor comportamiento que el de cualquiera de sus componentes por separado. En esta sección, vamos a estudiar la sinergia producida de la combinación de ES y AE en las MHH-AE/ES presentadas anteriormente con respecto a los siguientes algoritmos:

- *ES Estandar* (ES) [6], [5]. El criterio de aceptación elegido es el de metrópolis y el esquema de enfriamiento es el geométrico con  $\alpha = 0,99$ . La temperatura inicial se fija de la siguiente manera: en primer lugar, se generan dos soluciones aleatorias y se establece una probabilidad de aceptación,  $p_d = 0,4$ , del movimiento producido al pasar de la mejor a la peor de las dos soluciones. Entonces, se calcula el correspondiente valor  $T_0$  de acuerdo al criterio de aceptación aplicado. El enfriamiento se produce cada 100 iteraciones.

- *AG Generacional Canónico* (AGGC). Considera una población de 60 individuos, selección mediante torneo binario, mutación de un bit con probabilidad 0,006 y cruce en dos puntos con probabilidad 1.

- *AG Estacionario Canónico* (AGEC). Los parámetros y operadores son los mismos que en AGGC. En cada iteración, se generan dos hijos que reemplazan a sus padres.

- *Cross-generational elitist selection, Heterogeneous recombination, and Cataclysmic mutation* (CHC) [19]. Es un AE que combina una estrategia de selección con una presión selectiva alta, además de diversos componentes para inducir diversidad.

La Tabla IV (en la columna *Resultados*) muestra los resultados de aplicar el test de Wilcoxon con un  $p$ -value=0,05, especificando los valores de  $R^+$  (asociado a la HM-ES/AE) y  $R^-$ . De acuerdo a estos resultados, podemos concluir lo siguiente:

- Las únicas MHH-AE/ES que presentan mejor rendimiento que la sola aplicación de ES, cualquier AG y CHC son SALGeS y GAMSAs. De esta forma, podemos confirmar que esta aproximación para la generación de algoritmos híbridos (*componentes de SA basados en AE*) proporciona un rendimiento superior, explotando y conjugando las ventajas de las diferentes MH que la componen y, por lo tanto, mostrando un efecto de sinergia positiva.

- Considerando el resto de hibridaciones, solamente DCHCSA presenta un resultado positivo en términos de sinergia. Las restantes aproximaciones híbridas no producen mejores resultados, para el entorno de trabajo considerado, que la aplicación por separado de las MH que lo componen. Estos resultados resaltan que la simple combinación de varias MH no asegura el éxito de la hibridación. Es necesario estudiar la forma en la que las MH que componen la

TABLA IV  
MHH-AE/ES vs. ES, ALGORITMOS GENÉTICOS CANÓNICOS Y CHC (COMPARATIVA SOBRE RESULTADOS Y TIEMPOS) (TEST DE WILCOXON CON  $p$ -VALUE=0,05 Y VALOR CRÍTICO = 107)

MHH	MH Simples	Resultados			Tiempos		
		$R^+$	$R^-$	Dif	$R^+$	$R^-$	Dif
CSA	ES	0	378	-	64	314	-
	AGGC	0	378	-	321	57	+
	AGEC	100	278	-	358	20	+
	CHC	0	378	-	309	69	+
ARSAGA	ES	0	378	-	24	354	-
	AGGC	0	278	-	262	116	~
	AGEC	16	362	-	60	318	-
	CHC	0	378	-	277	101	+
GESA	ES	0	378	-	38	340	-
	AGGC	0	378	-	298	80	+
	AGEC	4	374	-	378	0	+
	CHC	0	378	-	301	77	+
PRSA	ES	0	378	-	27	351	-
	AGGC	10	368	-	288	90	+
	AGEC	0	378	-	279	99	+
	CHC	1	377	-	277	101	+
SSSA	ES	0,5	377,5	-	48	330	-
	AGGC	33,5	344,5	-	329	49	+
	AGEC	378	0	+	378	0	+
	CHC	1	377	-	367	11	+
HHSAGA	ES	8	370	-	116	262	~
	AGGC	88	290	-	378	0	+
	AGEC	377	1	+	378	0	+
	CHC	46	332	-	378	0	+
GA-PSA	ES	7,5	370,5	-	68	310	-
	AGGC	0	378	-	359,5	18,5	+
	AGEC	158,5	219,5	~	210	168	~
	CHC	52,5	325,5	-	355	23	+
AGA	ES	30,5	347,5	-	81	297	-
	AGGC	237,5	140,5	~	339	39	+
	AGEC	378	0	+	323	55	+
	CHC	129,5	248,5	~	345	33	+
DCHCSA	ES	206,5	171,5	~	32	346	-
	AGGC	330,5	47,5	+	276	102	+
	AGEC	378	0	+	306	72	+
	CHC	328,5	49,5	+	325	53	+
SALGeS	ES	287,5	90,5	+	27	352	-
	AGGC	374,5	3,5	+	174	204	~
	AGEC	378	0	+	192	186	~
	CHC	343,5	34,5	+	160	218	~
GAMSA	ES	277,5	101,5	+	27	351	-
	AGGC	352,5	25,5	+	188	190	~
	AGEC	378	0	+	214	164	~
	CHC	333,5	44,5	+	199	179	~

hibridación se van a combinar para poder conseguir una sinergia positiva entre ellas.

Finalmente, es interesante analizar el esfuerzo computacional requerido por las MHH-AE/ES con

respecto a las MH que las componen. En la Tabla IV (columna *Tiempos*), podemos ver los resultados de aplicar el test de Wilcoxon, usando como medida de rendimiento el tiempo medio empleado para realizar  $10^5$  evaluaciones de la función de adaptación, en cada problema, sobre 50 ejecuciones independientes. La última columna indica si la correspondiente MHH-AE/ES requiere estadísticamente más tiempo (−), menos (+), o no hay diferencias significativas (∼) con respecto a las MH simples. Dicho test muestra que todas las MHH-AE/ES, excepto HHSAGA, requieren estadísticamente más tiempo que ES para resolver los problemas considerados. Por el contrario, es significativo el hecho de que ninguna MHH-AE/ES requiere un mayor esfuerzo computacional que las MH estándar más costosas (los AE). En particular, podemos resaltar que tanto SALGeS como GAMSa, que son las únicas hibridaciones que presentan resultados positivos en términos de sinergia y mejoran los resultados de sus componentes estándar, no exceden el tiempo requerido por sus componentes estándar más costosos, AGGC, AGEc Y CHC.

## VI. CONCLUSIONES Y TRABAJO FUTURO

En este artículo, hemos presentado una visión general acerca de los distintos esquemas utilizados para construir MHH-AE/ES, clasificándolos de acuerdo a las taxonomías propuestas por Talbi y Raidl para MHH. Además, hemos realizado un estudio experimental comparando el comportamiento de distintas instancias de MHH-AE/ES. Finalmente, el estudio de sinergia realizado ha identificado dos MHH-AE/ES que realmente presentan relaciones de sinergia positivas, siendo sus resultados mejores que los de todos sus componentes estándar de forma independiente. Los resultados de los distintos experimentos realizados han permitido identificar aquellos esquemas utilizados para desarrollar MHH-AE/ES que permiten mejorar el comportamiento de este tipo de algoritmos.

La línea de investigación presentada en este artículo es lo suficientemente interesante como para ser objeto de más estudios. Actualmente queremos extender nuestra investigación, estudiando el comportamiento de las hibridaciones entre ES y AE en otros campos como el de la optimización continua. Además, pretendemos ampliar el estudio, considerando otro tipo de MH basadas en trayectorias (algoritmos voraces iterativos, búsqueda tabú, etc.) como componente para las hibridaciones con los AE.

TABLA V  
RESULTADOS DE LAS MHH-AE/ES SOBRE CADA UNO DE LOS PROBLEMAS

Pr.	PRSA	GA-PSA	ARSAGA	GESA	AGA	HHGASA	SSSA	CSA	DCHCSA	SALGeS	GAMSA
1	0,366 ~	0,579 -	0,085 +	0,104 +	0,690 -	0,548 +	0,123 +	6,52e-02 +	0,338 ~	0,997 -	0,394
2	209 +	217 +	188 +	201 +	216 +	211 +	214 +	179 +	220 ~	220 ~	220
3	375 +	378 ~	363 +	376 +	370 +	374 +	373 +	365 +	382 -	379 -	378
4	0,860 +	0,883 +	0,826 +	0,875 +	0,893 +	0,893 +	0,875 +	0,813 +	0,88 +	0,895 +	<b>0,899</b>
5	0,768 +	0,864 +	0,671 +	0,777 +	0,894 +	0,896 +	0,845 +	0,635 +	0,896 +	0,9 +	0,906
6	0,933 +	0,955 +	0,916 +	0,932 +	0,953 +	0,953 +	0,952 +	0,912 +	0,957 +	0,958 +	<b>0,959</b>
7	0,921 +	0,936 +	0,906 +	0,917 +	0,936 +	0,935 +	0,932 +	0,903 +	0,938 +	<b>0,939 ~</b>	<b>0,939</b>
8	0,727 +	0,744 +	0,673 +	0,72 +	0,759 +	0,737 +	0,741 +	0,674 +	<b>0,766 ~</b>	<b>0,766 ~</b>	<b>0,766</b>
9	0,713 +	0,721 +	0,675+	0,71 +	0,714 +	0,709 +	0,718 +	0,692 +	<b>0,743 ~</b>	0,737 ~	0,741
10	151 +	126 +	113 +	120 +	179 ~	128 +	160 +	116 +	<b>187 ~</b>	176 +	185
11	159 +	164 +	124 +	144 +	171 -	152 +	156 +	120 +	<b>178 -</b>	176 -	169
12	0,841 +	0,987 +	0,725 +	0,842 +	0,988 +	0,993 +	0,939 +	0,680 +	0,998 ~	0,989 ~	<b>1</b>
13	0,828 +	0,976 +	0,73 +	0,857 +	0,976 +	0,979 +	0,935 +	0,709 +	0,993 +	0,995 +	<b>0,998</b>
14	0,805 +	0,979 +	0,687 +	0,79 +	0,967 +	0,98 +	0,937 +	0,656 +	0,996 +	0,994 +	<b>0,999</b>
15	0,747 +	0,987 +	0,659 +	0,759 +	0,980 +	0,979 +	0,923 +	0,627 +	0,997 +	0,992 ~	<b>0,998</b>
16	4641 +	5164 +	3771 +	4325 +	<b>5176 ~</b>	5146 +	5152 ~	3378 +	<b>5176 ~</b>	<b>5176 ~</b>	<b>5176</b>
17	59834 +	95978 +	26523 +	46188 +	117034 +	111884 +	80116 +	16728 +	120215 -	<b>121266 -</b>	119264
18	8140 +	11588 ~	5022 +	7537 +	11597 ~	11408 +	10792 +	3958 +	11597 ~	<b>11599 ~</b>	<b>11599</b>
19	34193 +	49759 +	19042 +	27767 +	50311 +	50298 +	44338 +	14305 +	50410 +	50456 +	<b>50465</b>
20	51 +	68 +	34 +	52 +	70 +	67 +	68 +	31 +	70 +	72 ~	<b>73</b>
21	2122 +	2623 +	1564 +	2076 +	2675 +	2564 +	2529 +	1510 +	2667 +	2700 +	<b>2716</b>
22	1366 +	1420 +	1324 +	1366 +	1422 +	1414 +	1412 +	1319 +	1431 +	1433 +	<b>1435</b>
23	7819 +	8150 +	7514 +	7826 +	8125 +	8123 +	8074 +	7463 +	8187 +	8202 ~	<b>8204</b>
24	5153437 +	7285193 +	3230924 +	4975365 +	7369494 +	7275664 +	6546047 +	2812404 +	7545615 -	<b>7563823 -</b>	7422326
25	3915 +	3922 +	3732 +	3620 +	3965 +	3793 +	3975 +	3720 +	4053 +	4052 +	<b>4081</b>
26	12189 +	12094 +	12084 +	11934 +	11994 +	11913 +	12244 +	12090 +	<b>12400 ~</b>	12398 +	<b>12400</b>
27	116566 +	<b>118204 ~</b>	111258 +	112481 +	<b>118204 ~</b>	117156 ~	<b>118204 ~</b>	109247 +	<b>118204 ~</b>	<b>118204 ~</b>	<b>118204</b>

## AGRADECIMIENTOS

Este trabajo se ha financiado con los proyectos TIN2008-05854 y P08-TIC-4173

## REFERENCIAS

- [1] F Glover and G Kochenberger, Eds., *Handbook of Metaheuristics*, Kluwer Academic Publishers, 2003.
- [2] P Siarry and Z Michalewicz, Eds., *Advances in Metaheuristics for Hard Optimization*, Natural Computing. Springer, 2008.
- [3] G Raidl, "A unified view on hybrid metaheuristics," in *Hybrid Metaheuristics*, F Almeida, MJB Blesa Aguilera, C Blum, JM Moreno Vega, M Pérez Pérez, A Roli, and M Sampels, Eds. 2006, vol. LNCS 4030, pp. 1–12, Springer.
- [4] E Talbi, "A taxonomy of hybrid metaheuristics," *J. Heuristics*, vol. 8, no. 5, pp. 541–564, 2002.
- [5] E Aarts and J Korst, *Simulated Annealing and Boltzmann Machines*, John Wiley & Sons, 1989.
- [6] S Kirkpatrick, C D Gelatt Jr, and M P Vecchi, "Optimization by simulated annealing," *Sci.*, vol. 220, no. 4598, pp. 671–680, 1983.
- [7] D Henderson, S Jacobson, and A Jacobson, "The theory and practice of simulated annealing," in *Handbook of Metaheuristics*, pp. 287–319. Kluwer Academic Publishers Group, 2003.
- [8] P Salamon, P Sibani, and R Frost, *Facts, Conjectures and Improvements for Simulated Annealing*, Monographs on Mathematical Modeling and Computation. SIAM, 2002.
- [9] T Bäck, D Fogel, and Z Michalewicz, *Handbook of Evolutionary Computation*, Institute of Physics Publishers, 1997.
- [10] A Eiben and J Smith, *Introduction to Evolutionary Computing*, Springer-Verlag, 2003.
- [11] C García-Martínez and M Lozano, "Local search based on genetic algorithms," in *Advances in Metaheuristics for Hard Optimization*, P Siarry and Z Michalewicz, Eds., Natural Computing, pp. 199–221. Springer, 2008.
- [12] M Lozano, F Herrera, N Krasnogor, and D Molina, "Real-coded memetic algorithms with crossover hill-climbing," *Evol. Comput.*, vol. 12, no. 3, pp. 273–302, 2004.
- [13] R Mallipeddi and P N Suganthan, "Ensemble of constraint handling techniques," *IEEE Trans. Evol. Comput.*, vol. 14, no. 4, pp. 561–579, 2010.
- [14] J A Vrugt, B A Robinson, and J M Hyman, "Self-adaptive multimethod search for global optimization in real-parameter spaces," *IEEE Trans. Evol. Comput.*, vol. 13, no. 2, pp. 243–259, 2009.
- [15] I Obregon and A Pawlovsky, "A hybrid SA-EA method for finding the maximum number of switching gates in a combinational circuit," *IEICE Electronics Express*, vol. 5, no. 18, pp. 756–761, 2008.
- [16] A Tantar, N Melab, and E Talbi, "A grid-based genetic algorithm combined with an adaptive simulated annealing for protein structure prediction," *Soft Comput.*, vol. 12, no. 12, pp. 1185–1198, 2008.
- [17] M Aydin and V Yigit, *Parallel Simulated Annealing*, pp. 267–288, Parallel Metaheuristics: A New Class of Algorithms. Wiley, 2005.
- [18] D Thompson and G Bilbro, "Sample-sort simulated annealing," *IEEE Trans. Syst., Man, Cybern. B*, vol. 35, no. 3, pp. 625–632, 2005.
- [19] L Eshelman and J Schaffer, "Preventing premature convergence in genetic algorithms by preventing incest," in *Int. Conf. on Genetic Algorithms*, R Belew and LB Booker, Eds. 1991, pp. 115–122, Morgan Kaufmann.
- [20] D Chen, C Lee, C Park, and P Mendes, "Parallelizing simulated annealing algorithms based on high-performance computer," *J. Global Optim.*, vol. 39, no. 2, pp. 261–289, 2007.
- [21] F Lin, C Kao, and C Hsu, "Applying the genetic approach to simulated annealing in solving some NP-hard problems," *IEEE Trans. Syst., Man, Cybern.*, vol. 23, no. 6, pp. 1752–1767, 1993.
- [22] P Yip and Y Pao, "Combinatorial optimization with use of guided evolutionary simulated annealing," *IEEE Trans. Neural Netw.*, vol. 6, no. 2, pp. 290–295, 1995.
- [23] S Hwang and R He, "A hybrid real-parameter genetic algorithm for function optimization," *Adv. Eng. Inform.*, vol. 20, no. 1, pp. 7–21, 2006.
- [24] S Mahfoud and D Goldberg, "Parallel recombinative simulated annealing: A genetic algorithm," *Parallel Comput.*, vol. 21, no. 1, pp. 1–28, 1995.
- [25] C García-Martínez and M Lozano, "Simulated annealing based on local genetic search," in *Proc. of the IEEE Int. Conf. Evolutionary Computation*, 2009, pp. 2569–2576.
- [26] F Rodríguez-Díaz, C García-Martínez, and M Lozano, "A GA-based multiple simulated annealing," in *IEEE Congress on Evolutionary Computation*, 2010, pp. 195–201.
- [27] J Zar, *Biostatistical Analysis*, Prentice Hall, 1999.
- [28] S García, D Molina, M Lozano, and F Herrera, "A study on the use of non-parametric tests for analyzing the evolutionary algorithms' behaviour: A case study on the CEC'2005 special session on real parameter optimization," *J. Heuristics*, vol. 15, pp. 617–644, 2008.
- [29] R Iman and J Davenport, "Approximations of the critical region of the Friedman statistic," in *Communications in Statistics*, 1980, pp. 571–595.
- [30] S Holm, "A simple sequentially rejective multiple test procedure," *Scand. J. Stat.*, vol. 6, pp. 65–70, 1979.
- [31] M Lozano and C García-Martínez, "Hybrid metaheuristics with evolutionary algorithms specializing in intensification and diversification: Overview and progress report," *Comput. Oper. Res.*, vol. 37, pp. 481–497, 2010.
- [32] C Blum, J Puchinger, G Raidl, and A Roli, "Hybrid metaheuristics in combinatorial optimization: A survey," *App. Soft Comput.*, vol. 11, pp. 4135–4151, 2011.
- [33] C Antonio, "A study on synergy of multiple crossover operators in a hierarchical genetic algorithm applied to structural optimisation," *Structural and Multidisciplinary Optimization*, vol. 38, no. 2, pp. 117–135, 2009.
- [34] F Peng, K Tang, G Chen, and X Yao, "Population-based algorithm portfolios for numerical optimization," *IEEE Trans. Evol. Comput.*, vol. 14, no. 5, pp. 782–800, 2010.
- [35] F Herrera, M Lozano, and A Sánchez, "Hybrid crossover operators for real-coded genetic algorithms: An experimental study," *Soft Comput.*, vol. 9, no. 4, pp. 280–298, 2005.