

***METAHEURÍSTICAS PARAMETRIZADAS
PARALELAS APLICADAS A UN PROBLEMA DE
OPTIMIZACIÓN DE COSTES EN LA EXPLOTACIÓN
DE RECURSOS HÍDRICOS***

José Matías Cutillas Lozano

Murcia, 15 de septiembre de 2011

1. Motivación
2. El problema de optimización de costes
3. Metaheurísticas parametrizadas
 - Esquema unificado parametrizado
 - Esquema unificado parametrizado paralelo
 - Modelado y autooptimización
4. Experimentos
5. Conclusiones

La explotación de recursos hídricos

- La explotación de recursos hídricos conlleva un coste en el consumo de energía eléctrica para realizar bombeos de agua.
- Existen una serie de restricciones técnicas que se deben cumplir.
- Nuestro objetivo es aplicar un algoritmo que permita optimizar el coste eléctrico respetando las restricciones.
- Se pretende mejorar un problema inicial real resuelto mediante algoritmos genéticos.
- El espacio de posibles soluciones es muy grande y los métodos exhaustivos no son aplicables aquí.
- En nuestro trabajo se han utilizado técnicas metaheurísticas.

Metaheurísticas parametrizadas

- Se realizan experimentos con varios parámetros y funciones para adaptar una metaheurística a nuestro problema.
- Se experimenta con varias metaheurísticas para obtener una que sea buena para nuestro problema.
- Utilizamos un esquema unificado parametrizado de metaheurísticas:
 - diferentes valores de los parámetros proporcionan diferentes metaheurísticas o combinación/hibridación de éstas.

Metaheurísticas parametrizadas paralelas

- Se requieren muchos experimentos para seleccionar una metaheurística satisfactoria y adaptarla al problema.
- Se resolverán un gran número de problemas de optimización.
- Utilizamos un esquema unificado parametrizado paralelo de metaheurísticas:
 - Se paralelizan a la vez las diferentes metaheurísticas obtenidas del esquema parametrizado, con los parámetros paralelos para la optimización del tiempo de ejecución.

Modelado y autooptimización

- Queremos establecer el número de hilos óptimo (que proporciona menor tiempo de ejecución) para cada método de la metaheurística.
- Se realizan experimentos variando el número de hilos de ejecución y los parámetros de cada función, correlacionando ambas variables.
- Se pretende utilizar estas correlaciones para autoseleccionar el número de hilos óptimo de cada función.

El problema de optimización de costes

Nuestro sistema hídrico está formado por un conjunto de bombas B de extracción de agua funcionando en unos determinados rangos horarios diarios R . Así se constituye el individuo y se estructura en forma de vector de tamaño $B \cdot R$. El conjunto de individuos es la población según nomenclatura de algoritmos evolutivos.

Podemos ver todas las variables que forman nuestro problema en el siguiente ejemplo:

Bombas: 5

Rangos Horarios: 3

Volumen total acumulado diario (m³): 10000 (+10000)

Caudal total mínimo (m³/h): 200

Caudales (m³/h): 258 142 243 247 277

Tarifas (€/Kw h): 0.168 0.112 0.056

Potencias (Kw): 103 154 314 132 234

El problema de optimización de costes

Conductividad máxima permitida ($\mu\text{S}/\text{cm}$): 2500

Conductividades ($\mu\text{S}/\text{cm}$): 1484 1923 1157 1588 2861

Volúmenes máximos concedidos diarios (m^3): 9245 6721 7455 2274 6462

Profundidades nivel dinámico (m): 152 186 170 270 173

Profundidades nivel dinámico máximas (m): 290 262 260 349 271 (+3)

Estatus pozos: 1 1 1 1 1

Individuo ejemplo (tamaño B·R): 0 1 1 0 1 0 1 1 0 1 1 1 1 1 0

El problema consiste en: obtener el individuo que opere de manera óptima de acuerdo a cierto criterio respetando todas las restricciones del sistema.

El problema de optimización de costes

Se ha utilizado la siguiente función objetivo como criterio de optimización:

$$C_e = \sum_{i=1}^R \sum_{j=1}^B T_i \cdot P_j \cdot N_i \cdot x_{ij}$$

donde C_e es el coste de energía eléctrica consumida por la combinación de bombas seleccionada, T_i es la tarifa energética en el rango horario i , N_i es el número de horas de funcionamiento de las bombas durante todo el rango horario i , P_j es la potencia eléctrica consumida por la bomba j y x_{ij} es la variable binaria de valor 1 para bomba encendida y 0 para bomba apagada.

El problema de optimización de costes

La anterior función de bondad está sujeta a las siguientes restricciones:

- *R1. Satisfacción de la demanda:*

$$H \cdot \sum_{i=1}^R \sum_{j=1}^B Q_{ij} \cdot x_{ij} = V_{dT}$$

- *R2. Mantenimiento del caudal mínimo para cada tramo horario:*

$$\sum_{j=1}^B Q_j \cdot x_j \geq Q_{\min.}$$

- *R3. Cumplimiento de los volúmenes máximos de explotación de cada pozo j:*

$$\forall j, \quad \frac{24}{R} \cdot Q_j \sum_{i=1}^R x_{ij} \leq V_{conc.kj}$$

El problema de optimización de costes

- *R4. Mantenimiento de la conductividad media por debajo del límite para cada tramo horario:*

$$\frac{\sum_{j=1}^B Q_j \cdot \sigma_j \cdot x_j}{\sum_{j=1}^B Q_j \cdot x_j} \leq \sigma_{\text{lím.}}$$

- *R5. Cumplimiento de profundidades máximas de niveles dinámicos.*

donde Q_{ij} es el caudal extraído del pozo j en el intervalo horario i , V_{dT} es el volumen diario total demandado, H es el nº de horas de cada intervalo, Q_j es el caudal extraído del pozo j , $Q_{\text{min.}}$ es el caudal mínimo total en tubería para cada tramo horario, σ_j es la conductividad de cada pozo y $\sigma_{\text{lím}}$ es la conductividad límite de la mezcla de aguas.

Objetivos

- Obtener una herramienta para aplicar y adaptar metaheurísticas de forma eficiente a nuestro problema de optimización de costes.
- Utilizamos para ello un esquema unificado parametrizado experimentando con las siguientes metaheurísticas: Algoritmos Genéticos, GRASP, Scatter Search y sus combinaciones/hibridaciones.
- Mejorar los tiempos de ejecución aplicando un esquema unificado parametrizado en memoria compartida basado en OpenMP, con funciones parametrizadas paralelas.
- Modelar el sistema correlacionando el número de hilos óptimo como función de los parámetros de cada método.

Esquema unificado parametrizado de metaheurísticas

Inicializar (S, ParamIni)

mientras (NO CondiciondeFin(S, ParamCondFin))

 SS = Seleccionar (S, ParamSel)

 si ($|SS| > 1$) SS1 = Combinar (SS, ParamCom)

 sino SS1 = SS

 SS2 = Mejorar (SS1, ParamMej)

 S = Incluir (SS2, ParamInc)

finmientras

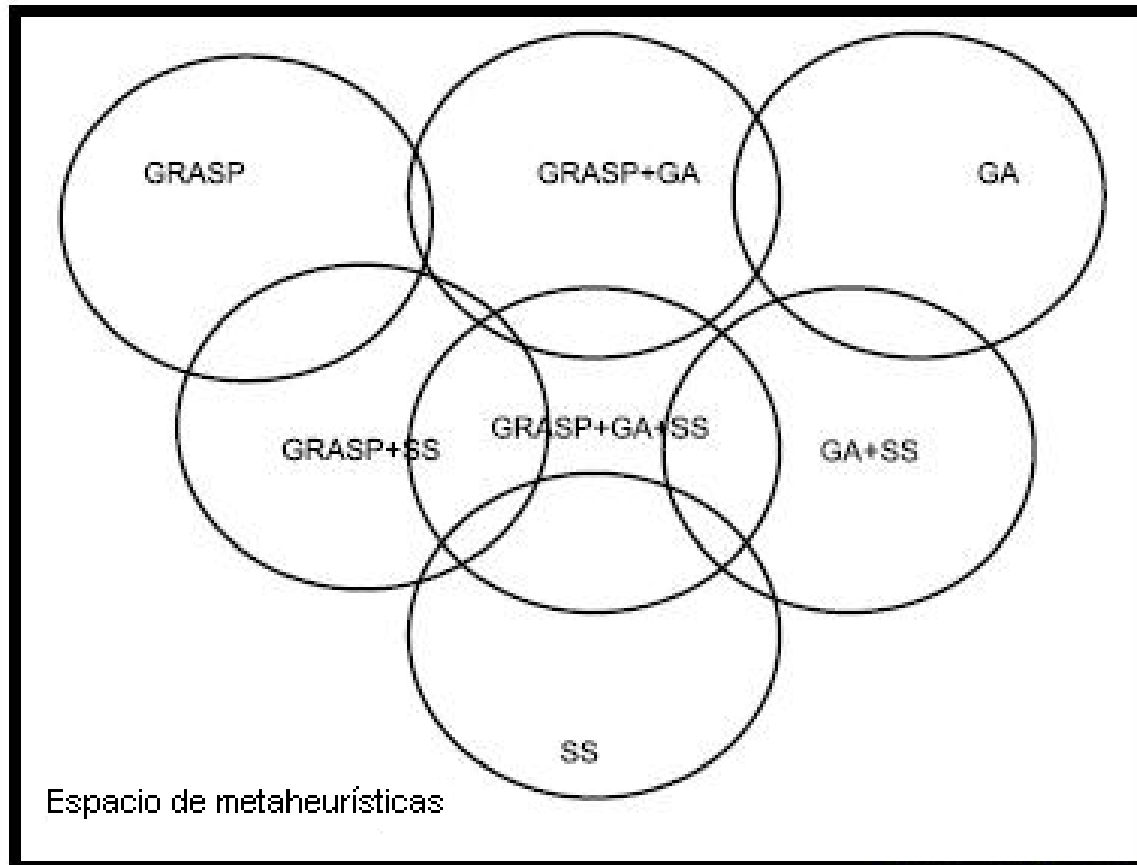
- Facilita trabajar con diferentes metaheurísticas reutilizando funciones
- El uso de parámetros facilita el trabajo con diferentes metaheurísticas/hibridación/combinación seleccionando diferentes valores de parámetros en las funciones

Parámetros de la metaheurística

- ParamInic = (NEIIni, PEIIni, IMEIni, NEFIIni)
- ParamSel = (NEMSel, NEPSel)
- ParamCom = (NMMCom, NMPCCom, NPPCom)
- ParamMej = (PEBMej, IMBMej, PEMMej, IMMMej)
- ParamInc = (NEMInc)
- ParamCondFin = (NMFin, NIRFin)

Metaheurísticas

- Metaheurísticas puras: GRASP (GR), Algoritmos Genéticos (GA), Scatter Search (SS)
- Combinaciones: GRASP+GA, GRASP+SS, GA+SS, GRASP+GA+SS



Esquema parametrizado paralelo

Inicializar (S, ParamIni, ThreadsIni)

mientras (NO CondiciondeFin(S, ParamCondFin))

 SS = Seleccionar (S, ParamSel)

 si ($|SS| > 1$) SS1 = Combinar (SS, ParamCom, ThreadsCom)

 sino SS1 = SS

 SS2 = Mejorar (SS1, ParamMej, ThreadsMej)

 S = Incluir (SS2, ParamInc, ThreadsInc)

finmientras

- ThreadsIni = (ThreadsIni, Threads1Ini, Threads2Ini)
- ThreadsMej = (Threads1Mej, Threads2Mej, Threads1Mut, Threads2Mut)

Esquema parametrizado paralelo

- Paralelización independiente de las funciones, con parámetros de paralelismo (número de threads) de cada función.
- Dos niveles de paralelismo. Se puede ajustar el paralelismo cambiando el número de hilos de cada nivel.
- Identificamos funciones con el mismo esquema paralelo:

Paralelismo de un nivel

```
omp_set_num_threads ( threads-un-nivel )
```

```
#pragma omp parallel for
```

```
bucle en elementos
```

```
    tratar elementos
```

p. ej.: GenerarConjuntoInicial (dentro de Inicializar), Combinar

- Se persigue reducir el tiempo de ejecución.

Esquema parametrizado paralelo

Paralelismo de dos niveles (anidado)

omp_set_num_threads (threads-primer-nivel)

#pragma omp parallel for

bucle en elementos

segundo-nivel (threads-primer-nivel)

segundo-nivel (threads-primer-nivel) :

omp_set_num_threads (threads-segundo-nivel (threads-primer-nivel))

#pragma omp parallel for

bucle en elementos

tratar elementos

p. ej.: Mejorar

- En este caso no se persigue reducir el tiempo de ejecución, sino mejorar la función de bondad mediante búsqueda y mejora locales.

Sistemas computacionales

- *Ben-Arabí* (Centro de Supercomputación de Murcia):
 - *Ben*: HP Integrity Superdome SX2000 con 128 cores con procesador Intel Itanium-2 dual-core Montvale.
 - *Arabí*: cluster de 102 nodos, cada uno con 8 cores con procesador Intel Xeon Quad-Core L5450 (se usa un nodo).
- Computador *Saturno* del Grupo de Computación Científica y Programación Paralela de la Universidad de Murcia, con 24 cores estructurados en cuatro nodos NUMA idénticos con 8183 MB de memoria compartida y tres niveles de memoria caché: L1 (32 KB), L2 (256 KB) y L3 (12 MB).

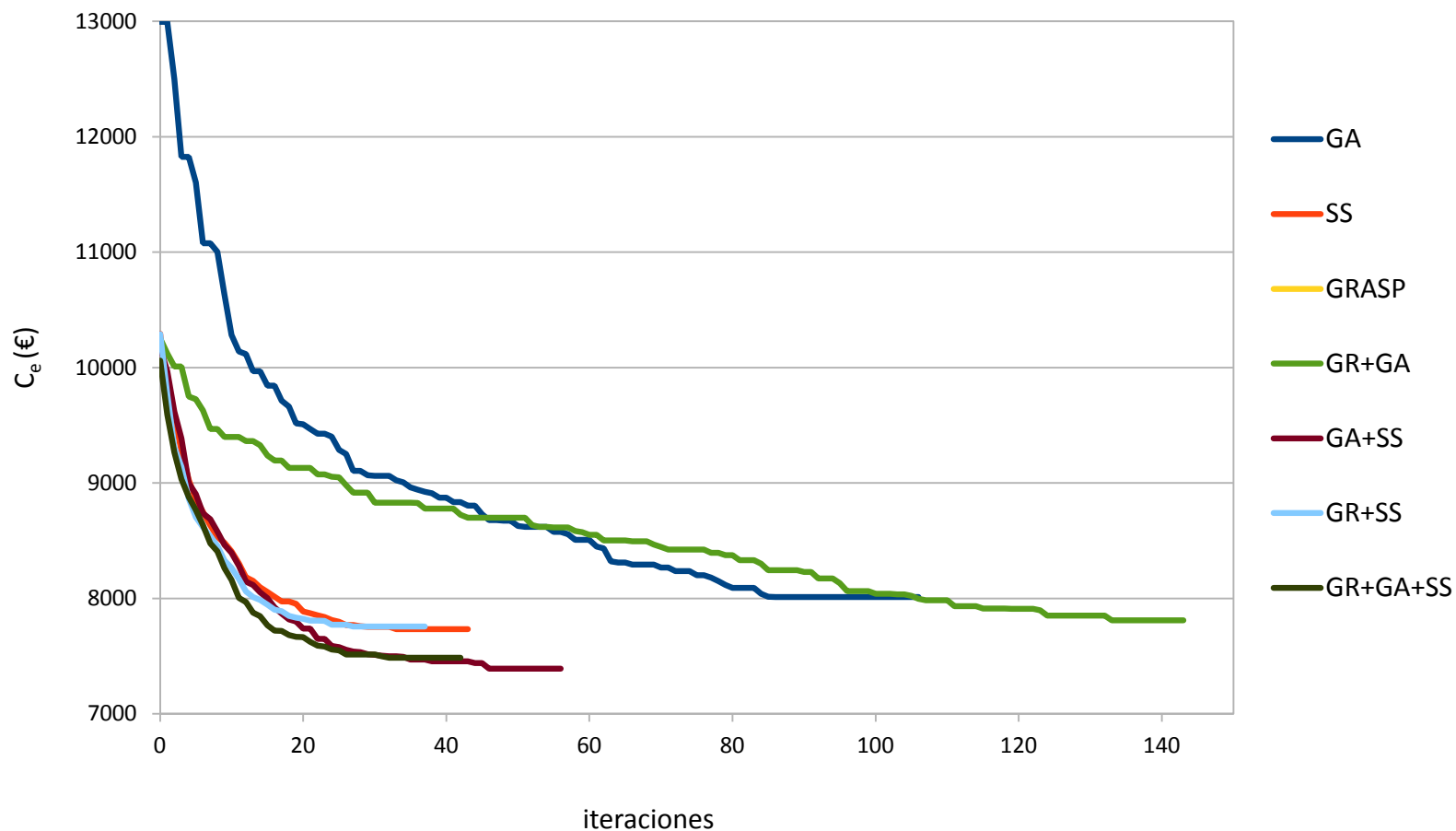
Parámetros

- Parámetros de las metaheurísticas y combinaciones de metaheurísticas utilizados en los experimentos:

		GRASP	GA	SS	GR+GA	GA+SS	GR+SS	GR+GA+SS
Inicializar	NEIIni	200	100	100	200	100	200	200
	NEFIni	1	100	20	100	50	20	50
	PEMIni	100	0	100	100	100	100	100
	IMEIni	10	0	10	10	10	10	10
Seleccionar	NEMSel	0	100	10	100	25	10	25
	NEPSel	0	0	10	0	25	25	25
Combinar	NMMCom	0	50	90	50	90	90	90
	NMPCom	0	0	100	0	100	100	100
	NPPCom	0	0	90	0	90	90	90
Mejorar	PEBMej	0	0	100	0	100	100	100
	IMBMej	0	0	5	0	5	5	5
	PEMMej	0	10	0	10	10	0	10
	IMMMej	0	5	0	5	5	0	5
Incluir	NEMInc	0	100	10	100	25	10	25

Experimentos secuenciales

Valores de coste óptimos obtenidos por las diferentes metaheurísticas en sucesivas iteraciones, para el problema con 50 bombas y 6 franjas horarias



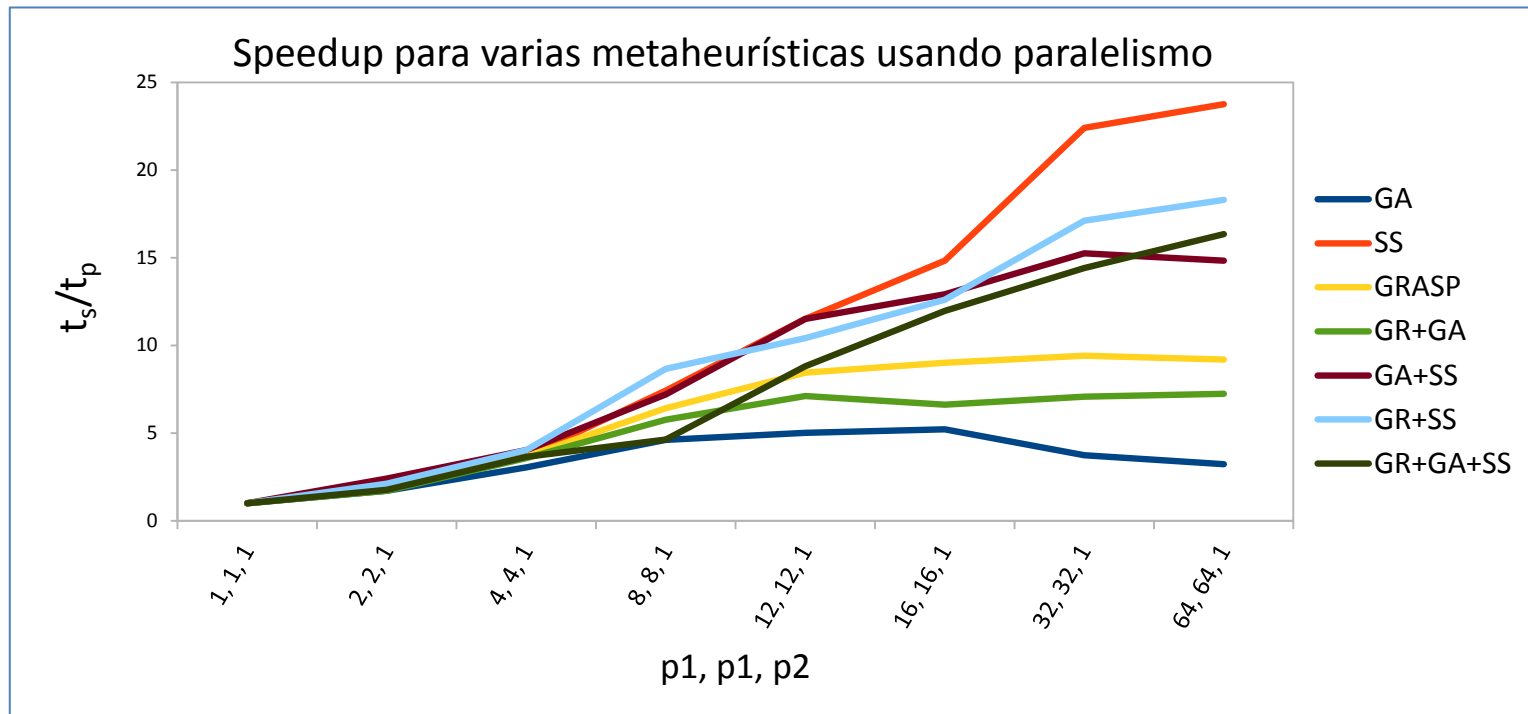
Experimentos con paralelismo global

- Se analiza el paralelismo de la metaheurística en su conjunto.
- Valores de los parámetros para la metaheurística considerada en los experimentos (salvo que se especifiquen otros):

<i>NEIni</i>	<i>NEFin</i>	<i>PEMIni</i>	<i>IMEIni</i>	<i>NEMSel</i>	<i>NEPSel</i>	<i>NMMCom</i>
100	50	50	10	25	25	90
<i>NMPCom</i>	<i>NPPCom</i>	<i>PEBMej</i>	<i>IMBMej</i>	<i>PEMMej</i>	<i>IMMMej</i>	<i>NEMInc</i>
100	90	50	10	50	5	25

Experimentos con paralelismo global

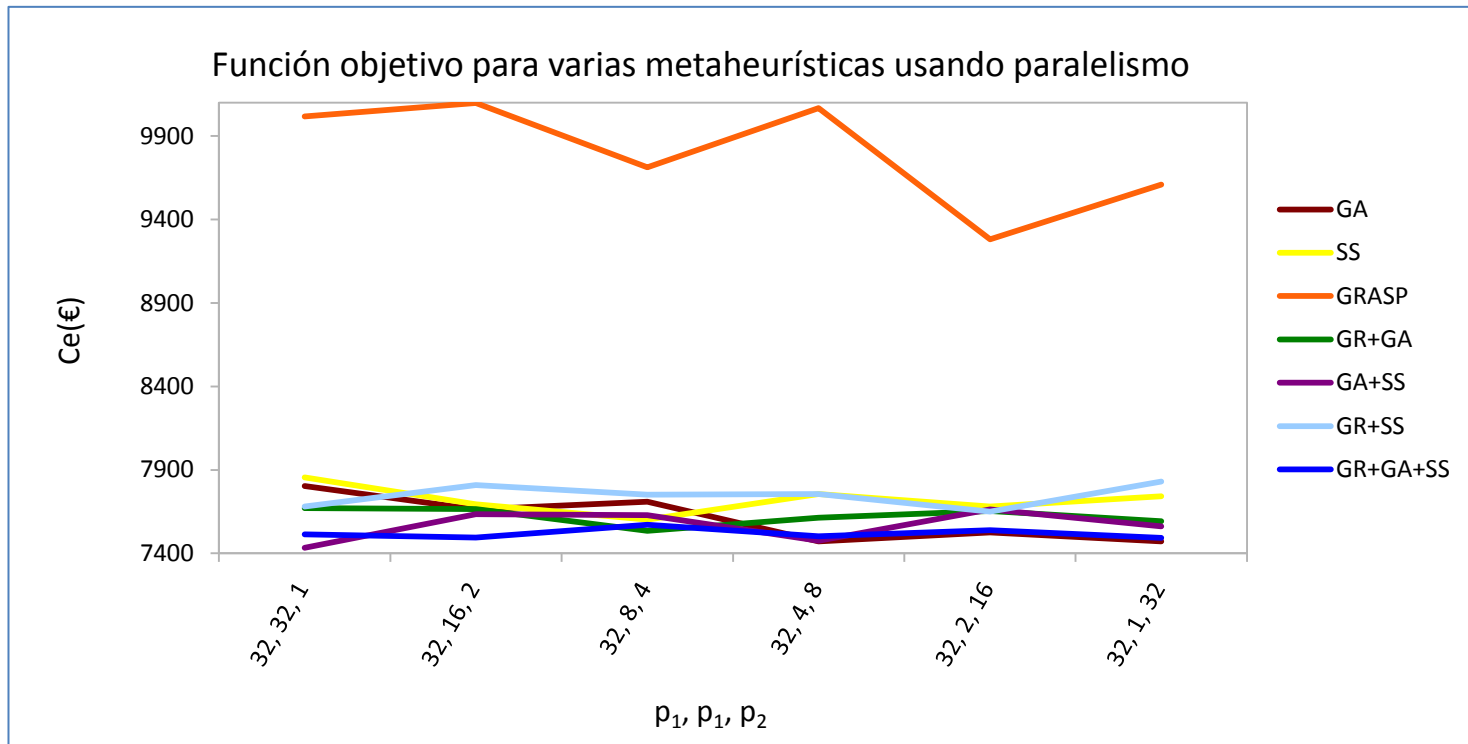
- Speedup para cada metaheurística lanzando distinto número de hilos (*Ben*).
 - Problema: $B=50$, $R=6$.



- Se mejora el tiempo secuencial:
 - GA, GR+GA, GA+SS y GRASP: máximo speedup en 16 a 32 hilos,
 - SS, GR+SS y GR+GA+SS: máximo speedup por encima de 64 hilos.

Experimentos con paralelismo global

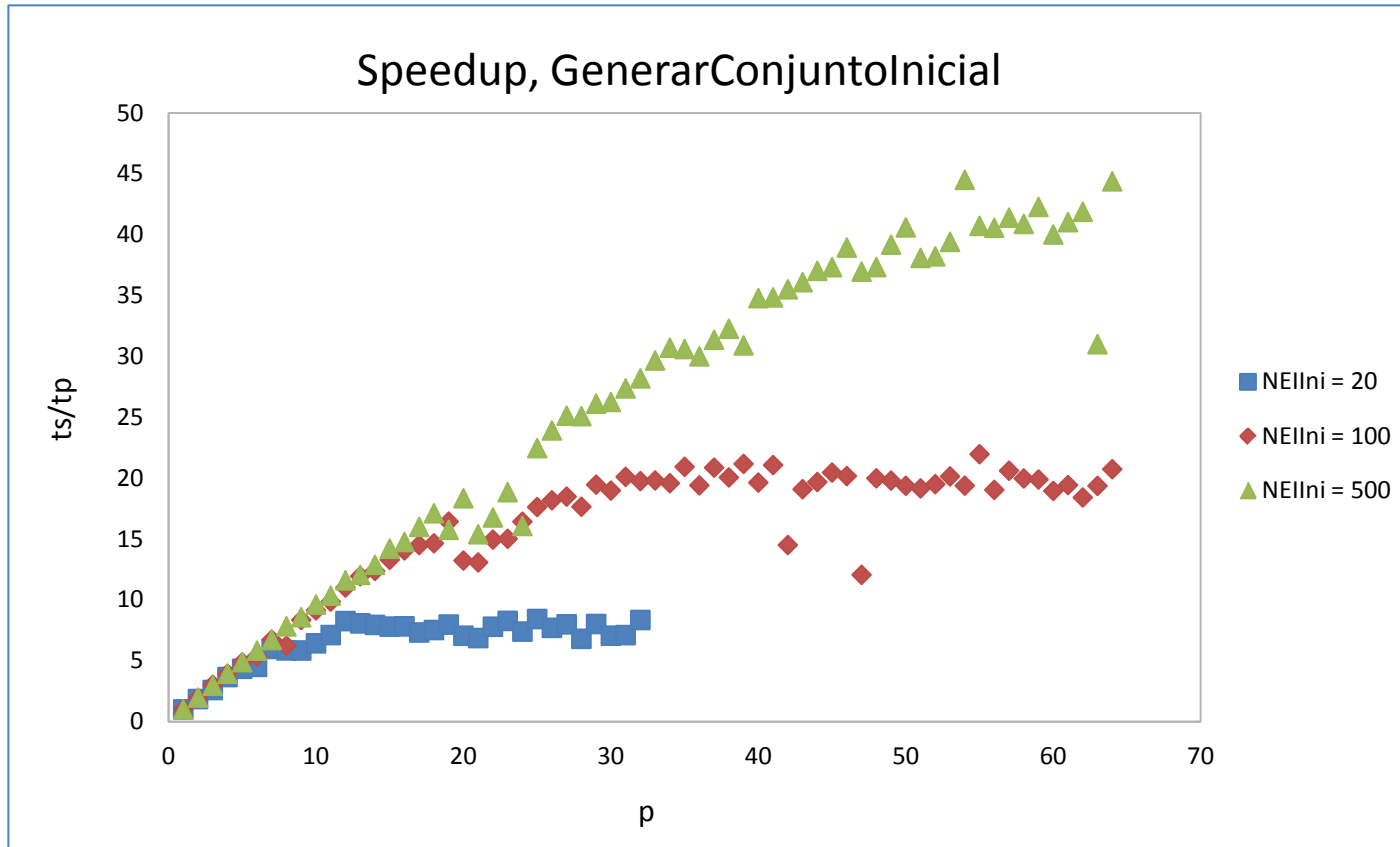
- Función de bondad para cada metaheurística lanzando distinto número de hilos de segundo nivel (Ben).
 - Problema: $B=50, R=6$.



- La mejora en la función objetivo es pequeña en todas las metaheurísticas respecto a los experimentos secuenciales

Experimentos paralelos. Paralelismo de un nivel

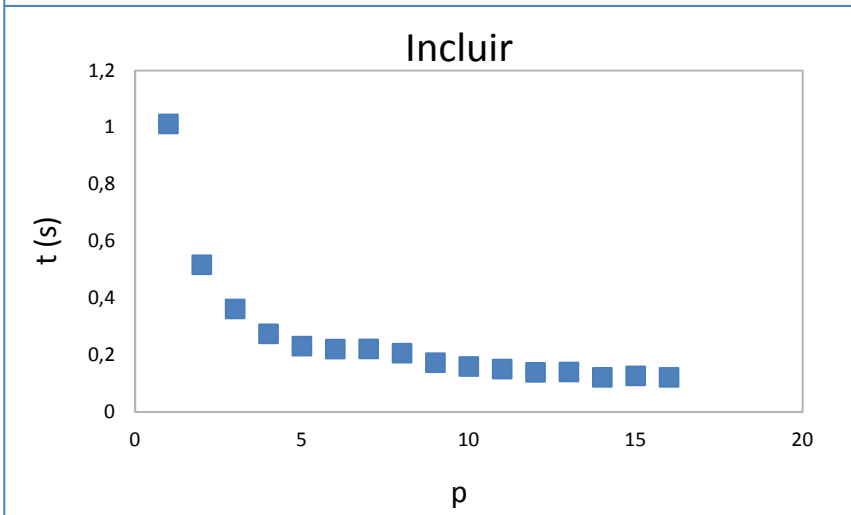
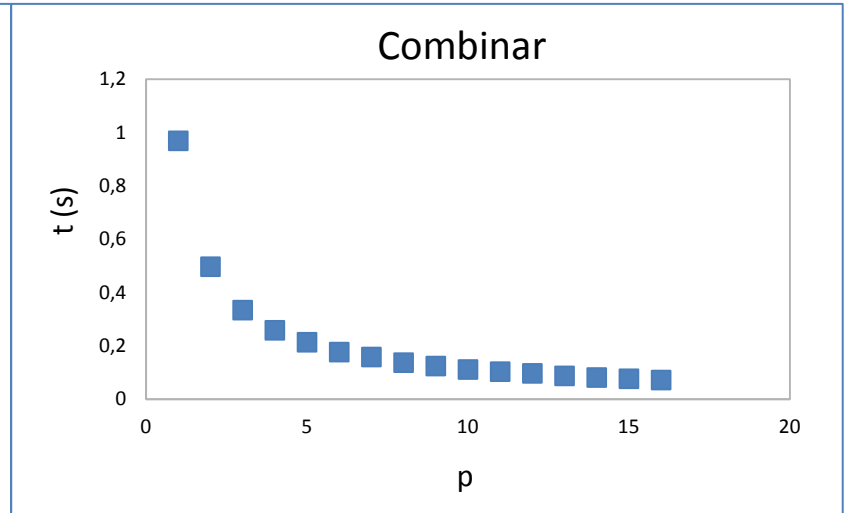
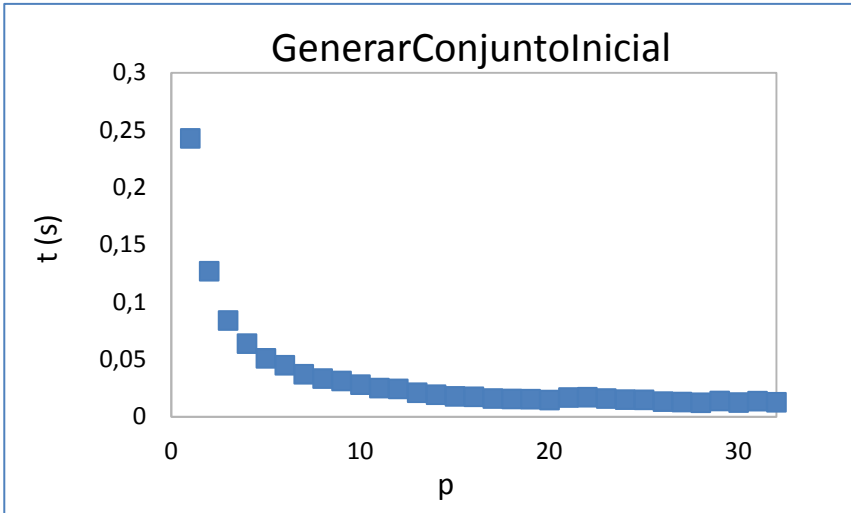
Speedup para varios tamaños de población inicial para la función *GenerarConjuntoInicial*



El número de hilos óptimo depende del tamaño inicial del conjunto *NEIIni*

Experimentos paralelos. Paralelismo de un nivel

Paralelismo de un nivel (*Ben*):



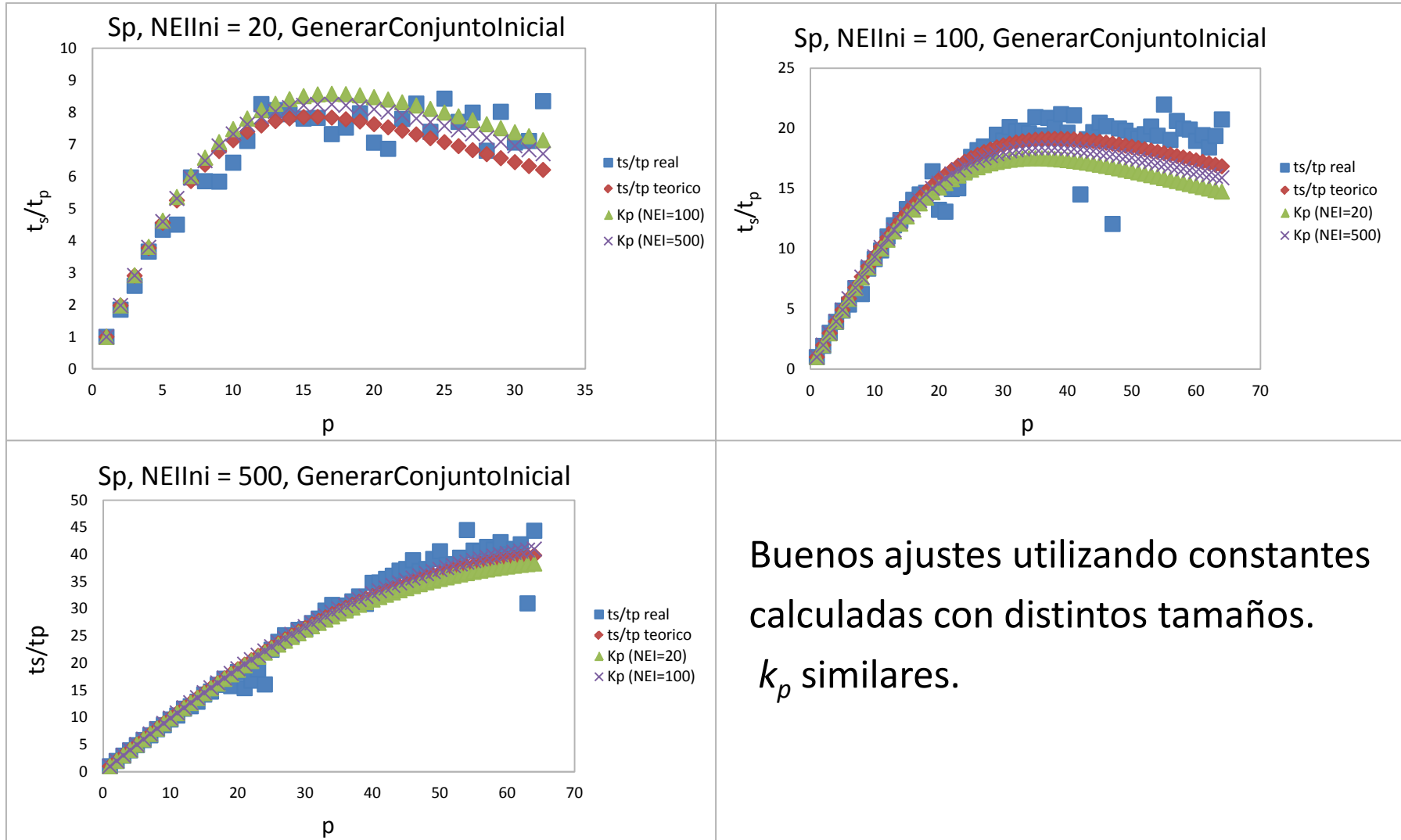
Modelamos GenerarConjuntoInicial según:

$$t_{gen} = \frac{(k_g \cdot NEIIni)}{p} + k_p \cdot p$$

El comportamiento del resto de funciones con paralelismo de un nivel es similar a GenerarConjuntoInicial.

Experimentos paralelos. Paralelismo de un nivel

Speedup para varios tamaños de *NEI*ni en GenerarConjuntoInicial (*Ben*):



Buenos ajustes utilizando constantes calculadas con distintos tamaños. k_p similares.

Paralelismo de un nivel. Modelado.

Obtenemos los valores de las constantes k_g y k_p que determinan la ecuación del modelo:

$$t_{gen} = k_g \cdot NEIIni \quad ; \quad k_g = 2.38 \cdot 10^{-3} \text{ s. (de la pte. de } t_{gen} \text{ vs. } NEIIni)$$

$$k_p = \frac{\sum_{i=1}^n (p_i \cdot t_{gen,i} - k_g \cdot NEIIni)}{\sum_{i=1}^n p_i^2}$$

; del ajuste para varios $NEIIni$:

$NEIIni$	k_p
20	1,94E-04
100	1,63E-04
500	1,76E-04

Podemos hacer una media de las constantes k_p , lo que nos permite calcular el nº de hilos óptimo para cada tamaño inicial:

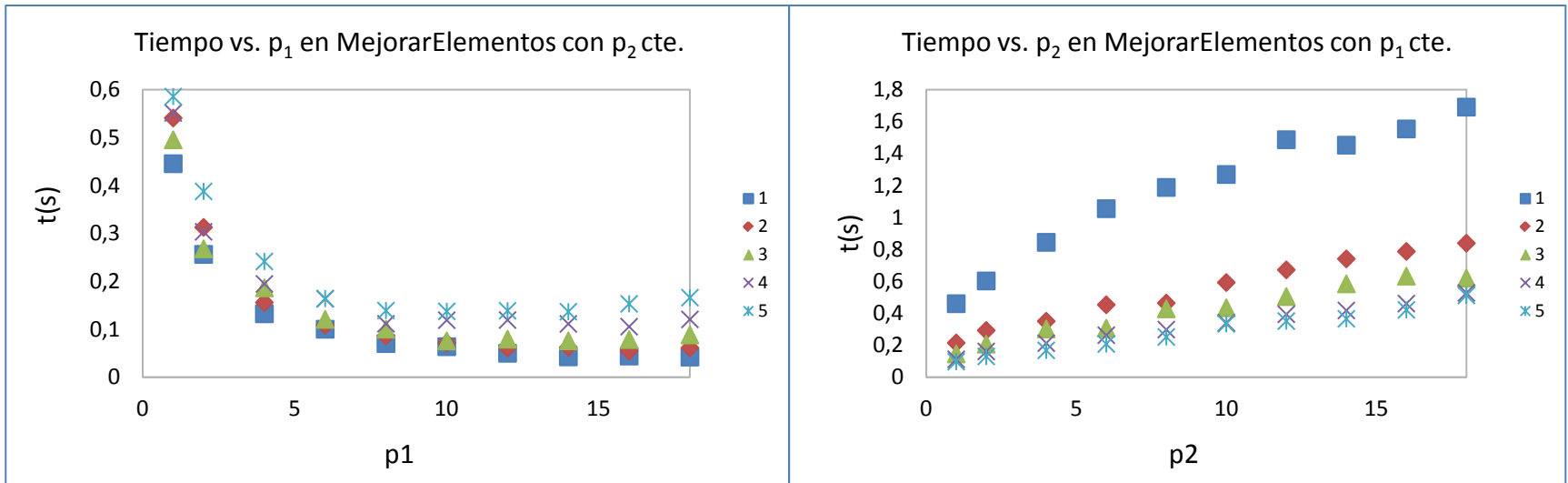
$$p_{opt.} = \sqrt{\frac{k_g}{k_p} \cdot NEIIni}$$

	NEIIni		
	20	100	500
$p_{opt.}$ (modelo)	16	37	82
$p_{opt.}$ (real)	25	55	64
$Sp_{opt.}$ (modelo)	8,6	19	41
$Sp_{opt.}$ (real)	8,4	22	44

Los valores óptimos de speedup alcanzados se ajustan bastante a los reales, a pesar de que el número de hilos modelado se diferencia moderadamente del real.

Experimentos paralelos. Paralelismo de dos niveles

- Paralelismo de dos niveles para *MejorarElementos* (Ben):



- Idéntico comportamiento en la función *Mutar*.
- El paralelismo anidado no mejora el tiempo de ejecución: sólo contribuye a la búsqueda y mejora en el vecindario de individuos
- Modelamos *MejorarElementos* según:

$$t_{mej} = \frac{k_m \cdot \frac{NEI_{ini} \cdot PEM_{ini} \cdot IME_{ini}}{100}}{p_1} + k_{p,1} \cdot p_1 + k_{p,2} \cdot p_2$$

Experimentos paralelos. Modelado

- Calculamos las constantes que definen el modelo para la función *MejorarElementos*:

$$k_m = \frac{\sum_{i=1}^n \left(\frac{NEIIni_i \cdot PEMIni_i \cdot IMEIni_i}{100} \cdot t_{meji} \right)}{\sum_{i=1}^n \left(\frac{NEIIni_i \cdot PEMIni_i \cdot IMEIni_i}{100} \right)^2} \quad \text{con } k_m = 9,10 \cdot 10^{-4} \text{ s. A partir del ajuste}$$

$$k_{p,1} = \frac{\sum_{i=1}^n p_{2,i} \cdot t_{meji} - \sum_{i=1}^n k_m \cdot \frac{NEIIni_i \cdot PEMIni_i \cdot IMEIni_i}{100 \cdot p_{1,i}} \cdot p_{2,i} - k_{p,2} \cdot \sum_{i=1}^n p_{2,i}^2}{\sum_{i=1}^n p_{1,i} \cdot p_{2,i}}$$

$$k_{p,2} = \frac{\sum_{i=1}^n p_{1,i}^2 \cdot \left(\sum_{i=1}^n p_{2,i} \cdot t_{meji} - \sum_{i=1}^n k_m \cdot \frac{NEIIni_i \cdot PEMIni_i \cdot IMEIni_i}{100 \cdot p_{1,i}} \cdot p_{2,i} \right)}{\sum_{i=1}^n p_{1,i}^2 \cdot \sum_{i=1}^n p_{2,i}^2 - \left(\sum_{i=1}^n p_{1,i} \cdot p_{2,i} \right)^2}$$

$$= \frac{\sum_{i=1}^n p_{1,i} \cdot p_{2,i} \cdot \left(\sum_{i=1}^n p_{1,i} \cdot t_{meji} - \sum_{i=1}^n k_m \cdot \frac{NEIIni_i \cdot PEMIni_i \cdot IMEIni_i}{100} \right)}{\sum_{i=1}^n p_{1,i}^2 \cdot \sum_{i=1}^n p_{2,i}^2 - \left(\sum_{i=1}^n p_{1,i} \cdot p_{2,i} \right)^2}$$

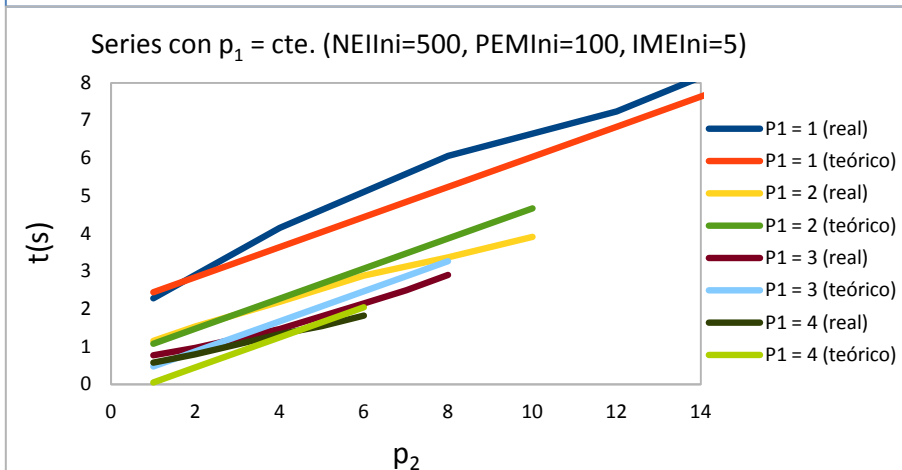
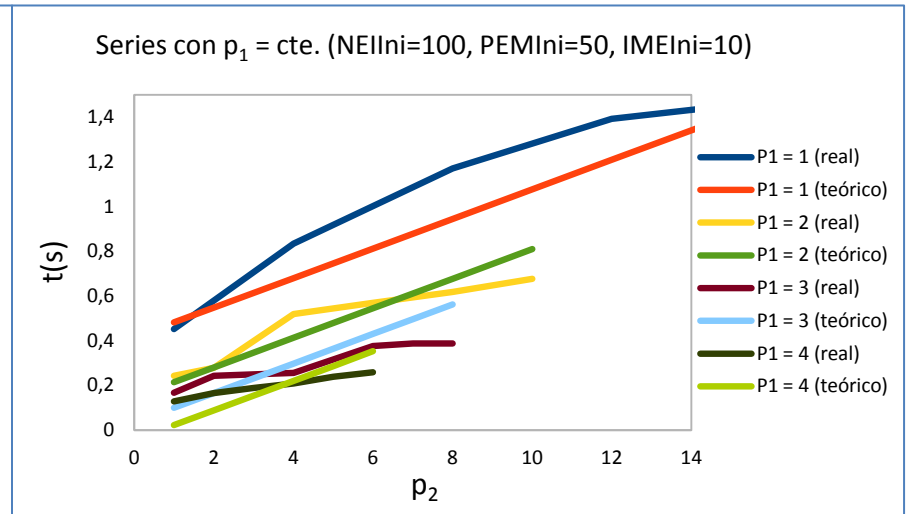
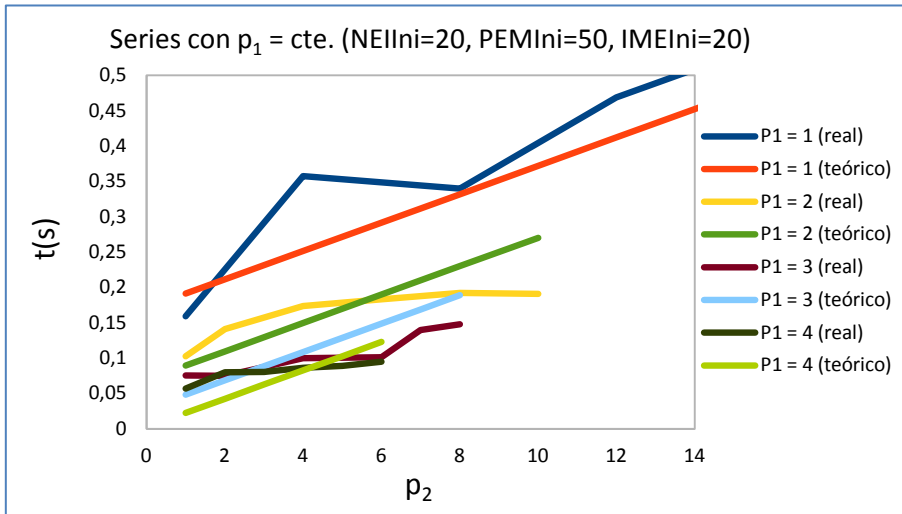
Valores de $k_{p,1}$ y $k_{p,2}$ a partir del ajuste:

Combinación	<i>NEIIni</i>	<i>PEMIni</i>	<i>IMEIni</i>
1	20	50	20
2	100	50	10
3	500	100	5

Combinación	$k_{p,1}$	$k_{p,2}$
1	-1,08E-02	2,01E-02
2	-3,95E-02	6,62E-02
3	-2,28E-01	3,99E-01

Paralelismo de dos niveles. Modelado

- Comparativa de tiempos de ejecución real y teórico a partir de los ajustes para varias combinaciones de parámetros en la función *MejorarElementos* (Ben):



- Los ajustes son mejores con valores altos de los parámetros.

- Se ajustan mejor las series con pocos hilos de primer nivel.

Conclusiones

- Aplicación satisfactoria de un esquema unificado parametrizado de metaheurísticas a nuestro problema de optimización de costes en la explotación de recursos hídricos.
- Los mejores resultados secuenciales, en cuanto a función de bondad, se obtienen con la combinación de metaheurísticas GA+SS y GR+GA +SS, mejorándose los resultados iniciales con algoritmos genéticos.
- Aplicación de un esquema unificado parametrizado en memoria compartida. Se han obtenido versiones paralelas de las metaheurísticas mejorándose el tiempo de ejecución respecto al secuencial.
- Se ha demostrado la dependencia del número de hilos con los parámetros de la metaheurística. Se han modelado algunas funciones representativas del sistema obteniéndose resultados satisfactorios, optimizando el esquema paralelo.

Trabajos futuros

- A partir del conjunto de parámetros establecido se podrán diseñar hiperheurísticas para obtener la mejor combinación de parámetros para el problema (mejor metaheurística).
- Diseño de un mecanismo de selección automática del número óptimo de hilos en cada función del esquema paralelo, utilizando para ello los modelos teóricos del capítulo 4, y obteniendo el esquema metaheurístico parametrizado paralelo con autooptimización.
- Extensión del esquema parametrizado de memoria compartida a paso de mensajes y GPUs.
- Desarrollo de una interfaz de usuario atractiva.
- Utilización del software desarrollado en un entorno real.

Referencias

- [1] Alba, E.: *Parallel Metaheuristics: A New Class of Algorithms*. Wiley Interscience, 2005.
- [2] Almeida, F., Cuenca, J., Giménez, D., Llanes-Castro, A., Martínez-Gallar, J.P.: A framework for the application of metaheuristics to tasks-to-processors assignment problems. *Journal of Supercomputing* (published online, September 2009).
- [3] Almeida, F., Giménez, D., López-Espín, J.J.: A parameterised shared-memory scheme for parameterised metaheuristics. In: *Proc. Int. Conf. CMMSE 2010*. páginas, está en el laboratorio
- [4] Almeida, F., Giménez, D., López-Espín, J.J.: Comparación de metaheurísticas para la obtención de modelos de ecuaciones simultáneas. VII Congreso Español sobre Metaheurísticas, Algoritmos Evolutivos y Bioinspirados, Valencia, 7-10 septiembre 2010.
- [5] Almeida, F., Giménez, D., López-Espín, J.J.: Parameterized schemes of metaheuristics: basic ideas and applications. En revisión.
- [6] Cuenca, J., Giménez, D., López-Espín, J.J., Martínez-Gallar, J.P.: A proposal of metaheuristics to schedule independent tasks in heterogeneous memory-constrained systems. In: *Proc. IEEE Int. Conf. on Cluster Computing*. IEEE Computer Society, pp. 422–427 (2007).
- [7] Cuenca, J., Giménez, D.: *Improving Metaheuristics for Mapping Independent Tasks into Heterogeneous Memory-Constrained Systems*. ICCS (1), LNCS 5101, Springer, 2008, 5101, 236-245.
- [8] Cutillas, L.G.: *Metaheurística aplicada a la optimización de los criterios de producción de aguas subterráneas*. Proyecto Sondea. Proyecto Final de Carrera, Universidad de Alicante, 2008.
- [9] Dréo, J., Pétrowski, A., Siarry, P., Taillard, E.: *Metaheuristics for Hard Optimization*. Springer, 2005.
- [10] Glover, F., Kochenberger, G. A.: *Handbook of Metaheuristics*. Kluwer, 2003.
- [11] López-Espín, J.J., Giménez, D.: Genetic algorithms for simultaneous equation models. In: *DCAI*, pp. 215–224 (2008).
- [12] Raidl, G.R.: A unified view on hybrid metaheuristics. *Hybrid Metaheuristics, Third International Workshop*, LNCS 4030, 2006, 1-12.

Muchas gracias
por su atención