6. RESULTADOS COMPUTACIONALES

En este apartado vamos a realizar un análisis del funcionamiento de la metaheurística GRASP en el problema POC, considerando dos objetivos distintos: En el primero de ellos aplicaremos la metaheurística a un problema de maximizar la suma de los pesos de los trabajos procesados. En el segundo de los casos, el problema sobre el que se trabajará tendrá como función objetivo (FO) maximizar el número de trabajos procesados. Para estudiar la calidad de los resultados obtenidos, los compararemos con los conseguidos por otras técnicas metaheurísticas aplicadas sobre los mismos problemas, sacando, posteriormente las conclusiones pertinentes.

6.1. Generación de la batería de problemas

Para la realización del estudio llevado a cabo en este proyecto es fundamental disponer de una buena batería de problemas que abarquen un amplio conjunto de situaciones diversas. Esto nos permitirá comprobar el comportamiento del algoritmo de una forma más general. Además, al no ceñirnos únicamente a casos particulares, las conclusiones que se obtengan tendrán un mayor significado y estarán más consolidadas.

Para el desarrollo de esta batería nos vamos a basar en las heurísticas propuestas en *Koon et al y Gabrel*, cuyas características describimos a continuación.

El conjunto de problemas a utilizar, se va a generar de forma aleatoria modificando el valor de un parámetro denominado ratio de utilización "*r*", diseñado por *Koon et al.* para este problema. Este parámetro es un indicador de la carga de trabajo esperada por unidad de capacidad de los recursos de los que se dispone y su formulación matemáticas es como sigue:

$$r = (n / T) \times (D / 15)$$

Donde:

- *D* representa la duración del trabajo cuyo tiempo de procesado alcance el valor más elevado.

- T indica la longitud del horizonte de planificación
- n denota el número de trabajos.

Los valores que va a tomar el ratio de utilización serán:

- Ratio de utilización bajo: r = 1,5.
- Ratio de utilización medio: r = 3.
- Ratio de utilización alto: r = 6.

El horizonte de planificación que vamos a considerar es de T = 1000 unidades de tiempo, mientras que los valores que tomará n serán n = 25, n = 50, n = 100 y n = 200.

Asimismo, el número de máquinas que se van a utilizar en los experimentos variará entre: m = 4, m = 8 y m = 16.

El número de máquinas que pertenece a cada una de las clases de máquinas se determinará de manera uniforme a partir de los valores de m, mientras que los trabajos serán asignados de manera aleatoria a una de las clases de trabajo.

Para cada uno de los trabajos J_i el tiempo de procesamiento d_i será generado aleatoriamente de una distribución $U(0, D)$ mientras que el instante de comienzo será obtenido de una distribución $U(0, T\text{-}d_i)$, teniendo entonces que el instante de finalización del trabajo f_i será la suma del instante de comienzo más la duración del mismo, es decir, $d_i + s_i$.

Por último, destacar que se generarán diez casos para cada una de las $(r\ ,n\ ,m)$ combinaciones, resultando un total de 330 problemas diferentes.

6.2. Objetivo: Maximizar la suma de los pesos de los trabajos

En este primer caso, aplicaremos la metaheurística GRASP a la resolución de problemas en los que se trata de maximizar la suma de los trabajos procesados. Antes de resolver los problemas, vamos a realizar un completo análisis de sensibilidad que nos permita adecuar correctamente la metaheurística a este caso particular. Una vez conformada, obtendremos los resultados y los compararemos con los logrados por el algoritmo exacto y por el algoritmo TABÚ.

6.2.1. Análisis de sensibilidad

El objetivo que queremos conseguir con el análisis de sensibilidad es conformar la metaheurística GRASP de tal modo, que los resultados que se obtengan en su aplicación sobre los dos casos mencionados con anterioridad estén lo más próximo posible de los valores óptimos.

Vamos a comenzar el análisis estudiando el caso en el que se busca maximizar la suma de los pesos de los trabajos procesados. Los parámetros que utilizaremos para ello son los siguientes:

- Función índice.
- Tamaño de la RCL.
- Número de iteraciones.

Función índice. En este primer caso, si lo que queremos es maximizar la suma de los pesos de los trabajos procesados, nos interesa que la metaheurística incluya en la solución a aquellas tareas cuyo *peso / duración* sea mayor. Por ello, se han elegido para el estudio los índices expuestos en la tabla 6.1:

Función índice
Peso
Peso / Duración
Peso^2
Peso^2 / Duración

Tabla 6.1: Valores de la función índice.

Tamaño de la RCL. El tamaño de la RCL se va a definir de dos formas:

En el primero de los casos, para determinar el número de candidatos a entrar en la solución en cada una de las iteraciones, tomaremos el candidato de mayor índice y multiplicaremos dicho valor por un factor α comprendido entre 0 y 1 de manera que serán candidatos todos aquellos trabajos cuyos valores del índice sean mayores o iguales que el resultado del producto anterior. Así pues, en este primer caso tenemos que el tamaño de la RCL dependerá del valor de α tomado. Para la realización del análisis tomaremos los descritos en la tabla 6.2:

Valores de a
$\alpha = 0{,}4$

$\alpha = 0,6$
$\alpha = 0,8$

Tabla 6.2: Valores del parámetro α..

En el segundo de los casos el tamaño de la lista estará fijado de antemano en un valor concreto de manera que el número de candidatos en cada iteración será constante. Los tamaños elegidos son los detallados en la tabla 6.3:

Tamaños RCL
T.RCL = 4
T.RCL = 5
T.RCL = 6

Tabla 6.3: Valores de los tamaños de la RCL.

Número de iteraciones. Debido a que en el funcionamiento de la metaheurística GRASP la aleatoriedad juega un papel muy importante, partimos de la idea de que el algoritmo no alcanzará el óptimo en todos los casos por lo que haremos que se detenga una vez alcanzado un número de iteraciones determinado. Es importante tener en cuenta el hecho de que, dado el funcionamiento de la metaheurística GRASP, a mayor número de iteraciones mayor es la probabilidad de alcanzar una buena solución, pero, a su vez, mayor es el tiempo invertido por el algoritmo. Los valores con los que vamos a trabajar son los expuestos en la tabla 6.4:

Número de iteraciones
N.I. = 500
N.I. = 1000
N.I. = 2000

Tabla 6.4: Valores del número de iteraciones.

Una vez definidos los valores de los parámetros con los que vamos a trabajar, nos queda precisar la estrategia a seguir:

Lo primero que vamos a hacer es determinar con cuál de los índices obtenemos mejores resultados. Para ello fijaremos el número de iteraciones en 1000, un valor no muy elevado pero suficientemente representativo, y probaremos la metaheurística con todos los índices propuestos.

Dado que en esta primera fase, el estudio es más cualitativo que cuantitativo, vamos a aplicar una serie de medidas que disminuyan el tiempo de ejecución de la metaheurística, sin que por ello se vea afectado el nivel de calidad de los resultados obtenidos.

En primer lugar, no será necesario procesar todo el grupo de problemas, sino que será suficiente con un conjunto de ellos lo bastante representativo. Se ha considerado que una colección de problemas apta en cuanto a número de tipos de máquinas, número de tipos de trabajos, carga de trabajo, número de trabajos y número de máquinas sería la expuesta en la tabla 6.5:

Número de problemas	60
Número de problemas por clase	10
Índice de carga (r)	1.5 / 3 / 6
Número de trabajos (n)	50
Número de máquinas (m)	4 / 8
Número tipos trabajo (A)	3
Número tipos máquinas (C)	2 / 4

Tabla 6.5: Características del conjunto de problemas.

Recordando el funcionamiento de la metaheurística GRASP, tenemos que tras la construcción de la solución, ésta, era sometida a un proceso de búsqueda local basado en la vecindad. En principio, podemos considerar que esta medida afecta a todas las soluciones por igual por lo que podemos suprimir su ejecución en esta primera fase.

Una vez definida la estrategia para esta primera fase del estudio, pasamos a describir los resultados obtenidos. En las tablas 6.6 y 6.7 se exponen, para cada uno de los casos considerados, los valores medios conseguidos por la función objetivo para cada conjunto de diez problemas.

Caso 1:

- Asignación: Se asignan los trabajos a las máquinas compatibles más desocupadas en ese instante (A.L.).
- T. RCL: Variable. Depende del valor de α.

Fichero					F. índice	Valores medios de FO		
r	n	m	A	C		α =0.4	α =0.6	α =0.8
1.5	50	4	3	2	Peso	2184,3	2185	2181,9
					Peso / Duración	2158,8	2148,8	2134,7
					Peso^2	2183,9	2182,5	2176,6
					Peso^2 / Duración	2172,2	2169	2154,8
1.5	50	8	3	4	Peso	2142,2	2161	2151,2
					Peso / Duración	2112,4	2112,5	2108,3
					Peso^2	2158	2156,8	2155,7
					Peso^2 / Duración	2135,3	2127,9	2113,7
3	50	4	3	2	Peso	1627,2	1638,2	1643,2
					Peso / Duración	1619,9	1611,5	1590,4
					Peso^2	1643	1641,9	1622,4
					Peso^2 / Duración	1639,6	1632	1618,7
3	50	8	3	4	Peso	1681,9	1707,2	1716,4
					Peso / Duración	1662,8	1657,2	1634,9
					Peso^2	1708,7	1726,6	1714
					Peso^2 / Duración	1701,5	1704,7	1677,8
6	50	4	3	2	Peso	1036,1	1056	1057
					Peso / Duración	1044,2	1035	1004,1
					Peso^2	1058,8	1058,2	1038,4
					Peso^2 / Duración	1058,5	1054,6	1037,6
6	50	8	3	4	Peso	1222,6	1259	1252,6
					Peso / Duración	1225,5	1218,7	1173,1
					Peso^2	1252,8	1255,7	1253,5
					Peso^2 / Duración	1252,6	1241,1	1219,1

Tabla 6.6: Valores medios de la FO asociados a α; A.L.

Caso 2:

- Asignación: Se asignan los trabajos a las máquinas compatibles de forma aleatoria (A. A.).

- T. RCL: Variable. Depende del valor de α.

Fichero					F. índice	Valores medios de FO		
r	n	m	A	C		a =0.4	a =0.6	a =0.8
1.5	50	4	3	2	Peso	2184,6	2186,4	2185,8
					Peso / Duración	2163,6	2155,9	2148,1
					Peso^2	2184,8	2185,6	2178,1
					Peso^2 / Duración	2172,9	2172,2	2159,3
1.5	50	8	3	4	Peso	2140,5	2155	2152,7
					Peso / Duración	2120,6	2118,4	2119,4
					Peso^2	2155,3	2155,5	2150,5
					Peso^2 / Duración	2129,6	2135,5	2138,1
3	50	4	3	2	Peso	1630,4	1640,7	1641,8
					Peso / Duración	1616,2	1613,3	1599,9
					Peso^2	1645,5	1641,2	1632,3
					Peso^2 / Duración	1641,5	1637,6	1632,7
3	50	8	3	4	Peso	1672,2	1701,5	1727,7
					Peso / Duración	1673,3	1678,5	1679,8
					Peso^2	1712,7	1725,6	1717,3
					Peso^2 / Duración	1705,9	1708,4	1709,3
6	50	4	3	2	Peso	1035,2	1055,3	1058,2
					Peso / Duración	1044,1	1033,6	1027,6
					Peso^2	1059,6	1058,6	1039,6
					Peso^2 / Duración	1062,6	1056,6	1044,7
6	50	8	3	4	Peso	1215,5	1253,8	1256,2
					Peso / Duración	1227,1	1235,3	1233,3
					Peso^2	1260,8	1265,5	1248,1
					Peso^2 / Duración	1259	1256,6	1261,9

Tabla 6.6: Valores medios de la FO asociados a a ; A. A.

Valores promedio asociados al caso 1:

F. índice	Valores promedio de FO		
	a =0.4	a =0.6	a =0.8
Peso	1649,1	1667,7	1667,1
Peso / Duración	1637,3	1630,6	1607,6
Peso^2	1667,5	1670,3	1660,1
Peso^2 / Duración	1660,0	1654,9	1637,0

Tabla 6.7: Valores promedio de la FO asociados a α; A. L.

Valores promedio asociados al caso 2:

F. índice	Valores promedio de FO		
	α **=0.4**	α **=0.6**	α **=0.8**
Peso	1646,4	1665,5	1670,4
Peso / Duración	1640,8	1639,2	1634,7
Peso^2	1669,8	1672,0	1661,0
Peso^2 / Duración	1661,9	1661,2	1657,7

Tabla 6.8: Valores promedio de la FO asociados a α; A. A.

Observando los resultados obtenidos en los valores promedio relativos a estos dos casos, podemos observar que se producen ciertas incongruencias con lo que debería ser el desarrollo normal de la metaheurística.

Tomemos, por ejemplo, los resultados de la función índice “Peso / Duración” en los dos casos considerados. Si los representamos gráficamente:

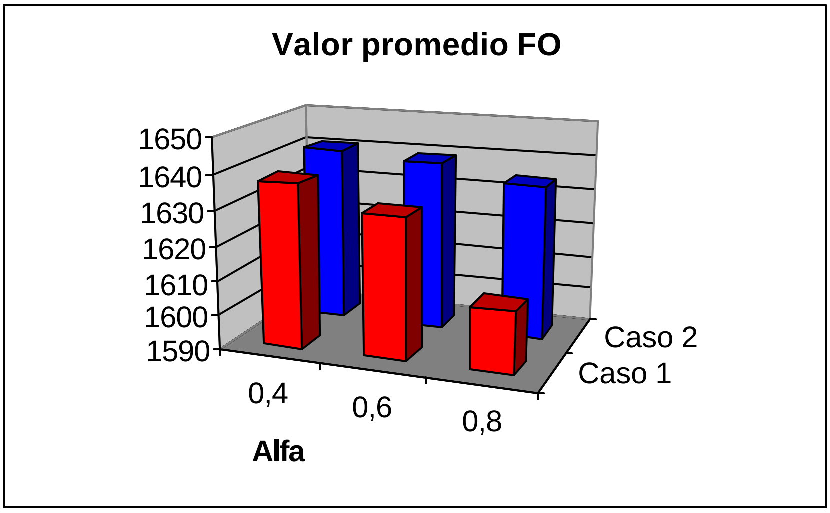

Figura 6.1: Valores promedio del índice "Peso / Duración"; Caso 1 y 2.

Vemos que en ambos casos, los valores promedios de la FO decrecen al aumentar el valor del parámetro α, lo cual va en contra del propio funcionamiento de la metaheurística. Además, esto no sólo ocurre con el "Peso / Duración", sino que se extiende a la mayoría de las funciones utilizadas.

Para estudiar las razones que pueden haber dado lugar a este hecho, vamos a recordar algunos aspectos del algoritmo utilizado:

Cuando hablábamos de las características de la metaheurística GRASP, hacíamos especial hincapié en la necesidad de que tuviera un carácter aleatorio. Esto lo lográbamos haciendo pasar un número de trabajos suficiente de la lista de candidatos (LC) a la lista restringida de candidatos (RCL).

Para tratar de conseguir este objetivo, en los dos casos considerados se ha definido un parámetro α ($0 \leq \alpha \leq 1$), de forma que, en cada iteración, la RCL está formada por aquellos candidatos que cumplen que:

$$\text{Peso (candidato)} \geq \alpha \cdot V$$

Siendo V el mayor peso de los candidatos en esa iteración.
Sin embargo, se ha comprobado que, para los valores de α que nos aseguran la entrada únicamente de los mejores candidatos en la RCL ($\alpha = 0.8$ ó $\alpha = 0.9$), el número de candidatos seleccionado en cada iteración es excesivamente pequeño (1 ó 2), por lo que el algoritmo se encierra siempre en las mismas soluciones, consiguiendo valores demasiado pequeños.

Si reducimos el valor de α, vemos que los valores promedios crecen. Esto se debe a que aumenta el número de candidatos que constituyen la RCL en cada iteración.

A simple vista, podríamos considerar como válidos los resultados conseguidos con valores pequeños de α; Sin embargo, hemos de tener en cuenta que, al reducir α, el número de candidatos no deseables que entra en la RCL aumenta y, dado que el elemento que formará parte de la solución se selecciona aleatoriamente, el nivel de calidad de la solución obtenida puede verse reducido.

Con todo ello, podemos afirmar que esta primera vía de investigación no ha resultado ser apta para satisfacer los objetivos propuestos en este apartado.

Pasemos, pues, a estudiar los siguientes casos:

Caso 3:

- Asignación: Se asignan los trabajos a las máquinas compatibles más desocupadas en ese instante (A. L.).
- T. RCL: Fijado de antemano.

Fichero					F. índice	Valores medios de la FO				
r	n	m	A	C		T.RCL=3	T.RCL=4	T.RCL=5	T.RCL=6	T.RCL=7
1.5	50	4	3	2	**Peso**	2181,9	2181	2181,3	2183,6	2183,7
					Peso / Duración	2180,7	2182,1	2182,1	2184,2	2183
					Peso^2	2179,1	2180,4	2180,6	2181,6	2183,4
					Peso^2 / Duración	2178,6	2181,1	2181,2	2182,6	2184,4
1.5	50	8	3	4	**Peso**	2151,1	2149,7	2149,7	2149,7	2156,8
					Peso / Duración	2151,3	2152,7	2152,8	2157,3	2150,8
					Peso^2	2149,6	2154,1	2154,2	2162,3	2153,4
					Peso^2 / Duración	2152,9	2149,5	2147,4	2151,5	2153,9
3	50	4	3	2	**Peso**	1633,8	1633,3	1634,6	1640,8	1637
					Peso / Duración	1633,6	1635,4	1639,4	1642,5	1635,6
					Peso^2	1633,9	1637,8	1638,5	1637,3	1640
					Peso^2 / Duración	1633,2	1638,3	1640	1640,1	1638,1
3	50	8	3	4	**Peso**	1709,2	1708,8	1710,2	1714	1705
					Peso / Duración	1709,3	1713,6	1705,5	1703	1702,1
					Peso^2	1704,2	1716,3	1719,1	1706,1	1712,5
					Peso^2 / Duración	1708	1710,8	1707	1710,8	1704,8
6	50	4	3	2	**Peso**	1048,8	1046,1	1053,8	1055,3	1049,2
					Peso / Duración	1047,5	1051,2	1054,8	1051,8	1056,6
					Peso^2	1047,8	1048,5	1059,2	1049,4	1056,2
					Peso^3 / Duración	1047,9	1053,1	1057,9	1049,4	1053,2
6	50	8	3	4	**Peso**	1240,7	1244,6	1244,3	1247,5	1251,5
					Peso / Duración	1237,8	1243,4	1248,8	1251,2	1244,9
					Peso^2	1240	1250,8	1247,5	1246,8	1243,5
					Peso^2 / Duración	1244,3	1247,4	1250,1	1256,9	1242,9

Tabla 6.9: Valores medios de la FO asociados a los tamaños RCL; A. L.

Caso 4:

- Asignación: Se asignan los trabajos a las máquinas compatibles de forma aleatoria (A. A.).
- T. RCL: Fijado de antemano.

Fichero					F. índice	Valores medios de la FO				
r	n	m	A	C		T.RCL=3	T.RCL=4	T.RCL=5	T.RCL=6	T.RCL=7
1.5	50	4	3	2	**Peso**	2181,3	2181,6	2184	2185,3	2184,4
					Peso / Duración	2180,5	2180,5	2181,7	2185,2	2183,6
					Peso^2	2182,9	2180,8	2181,9	2184,5	2185,3
					Peso^2 / Duración	2182	2177,9	2183,5	2187	2182,5
1.5	50	8	3	4	**Peso**	2152,9	2153,4	2152,4	2148,9	2152,8
					Peso / Duración	2160	2154,8	2150,7	2145,7	2153
					Peso^2	2156,3	2153	2150,8	2155,6	2149,2
					Peso^2 / Duración	2158	2160	2156,8	2153,9	2148,7
3	50	4	3	2	**Peso**	1639,1	1638,5	1642,1	1638,8	1639,8
					Peso / Duración	1638,6	1640,8	1640	1637,7	1636,2
					Peso^2	1637,1	1636,9	1637,3	1638,6	1638,2
					Peso^2 / Duración	1640,4	1639,1	1639,9	1643,6	1638,2
3	50	8	3	4	**Peso**	1718,7	1719,8	1709,8	1710,9	1713,4
					Peso / Duración	1704,4	1714,4	1707,4	1706,8	1694,1
					Peso^2	1716,6	1711,2	1706,9	1703,4	1704,4
					Peso^2 / Duración	1726,3	1709,6	1712	1704,8	1709,2
6	50	4	3	2	**Peso**	1048,6	1051,3	1054,1	1054,1	1053,7
					Peso / Duración	1045,8	1056,8	1055,8	1060,2	1052
					Peso^2	1046,3	1053,6	1052,5	1055,2	1057,5
					Peso^3 / Duración	1047,1	1054,4	1054,3	1050,4	1057
6	50	8	3	4	**Peso**	1245,3	1254,8	1248,8	1253,8	1246,8
					Peso / Duración	1250,9	1245,1	1248,4	1245,2	1247,3
					Peso^2	1249,7	1241,8	1245,9	1249,8	1242,2
					Peso^2 / Duración	1245,4	1247	1248,4	1254,4	1253,3

Tabla 6.10: Valores medios de la FO asociados a los tamaños RCL; A. A.

Valores promedio asociados al caso 3:

F. índice	Valores promedio				
	T.RCL=3	T.RCL=4	T.RCL=5	T.RCL=6	T.RCL=7
Peso	1660,9	1660,6	1662,3	1665,2	1663,9
Peso / Duración	1660,0	1663,1	1663,9	1665	1662,2
Peso^2	1659,1	1664,7	**1666,5**	1663,9	1664,8
Peso^2 / Duración	1660,8	1663,4	1663,9	1665,2	1662,9

Tabla 6.11: Valores promedio de la FO asociados a los tamaños RCL; A. L.

Valores promedio asociados al caso 4:

F. índice	Valores promedio				
	T.RCL=3	T.RCL=4	T.RCL=5	T.RCL=6	T.RCL=7
Peso	1664,3	**1666,6**	1665,2	1665,3	1665,2
Peso / Duración	1663,4	1665,4	1664,0	1663,5	1661,0
Peso^2	1664,8	1662,9	1662,6	1664,5	1662,8
Peso^2 / Duración	**1666,5**	1664,7	1665,8	1665,7	1664,8

Tabla 6.12: Valores promedio de la FO asociados a los tamaños RCL; A. A.

Si, al igual que antes, representamos gráficamente los resultados obtenidos por la función índice "Peso / Duración" en los dos casos, resulta:

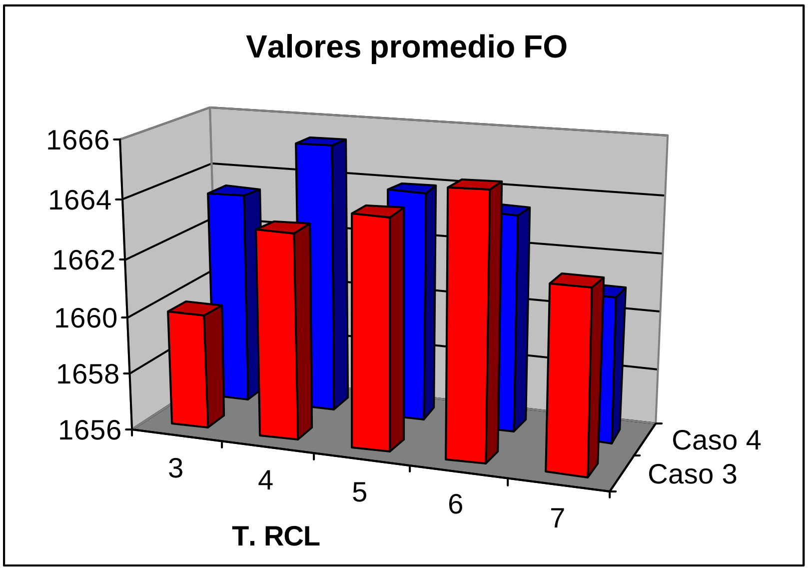

Figura 6.2: Valores promedio del índice “Peso / Duración”; Caso 3 y 4.

Este comportamiento puede explicarse de la manera siguiente:

Cuando el tamaño de la RCL es pequeño, nos encontramos con una situación análoga a la descrita anteriormente, es decir, en cada una de las iteraciones, el algoritmo dispone de pocos trabajos para seleccionar a aquel que va a formar parte de la solución, por lo que pierde aleatoriedad y disminuye el nivel de calidad de las soluciones alcanzadas. Ahora bien, a medida que aumenta el número de candidatos que constituyen la RCL, la metaheurística va ganando en aleatoriedad y va obteniendo cada vez mejores soluciones. Esta conducta se detiene en un punto a partir del cual, en cada iteración, el tamaño de la RCL comienza a ser excesivo de forma que se permite que candidatos de escaso interés desde el punto de vista de la función objetivo puedan formar parte de la solución. Esto implica que el nivel de calidad de las soluciones conseguidas vuelva a disminuir.

Tenemos, pues, que en ambos casos los resultados logrados siguen un patrón de conducta que sí es acorde con el funcionamiento de la metaheurística. Por tanto, y dado que este comportamiento se extiende también al resto de las funciones índices utilizadas, podemos afirmar que esta vía de investigación sí ha resultado ser adecuada para los intereses de la función objetivo considerada en este apartado.

Una vez analizada la validez de la metaheurística, nos queda averiguar con cuál de las funciones índices propuestas se consiguen los mejores resultados.

Observando los valores de las tablas, encontramos que el mínimo error se obtiene tomando el "Peso" como función índice y realizando una asignación aleatoria (Ver valor marcado en negrita en la tabla 6.12). Ahora bien, conseguimos resultados muy similares asignando sobre la máquina más libre con la función índice "Peso^2" asignando aleatoriamente con el índice "Peso^2 / Duración" (Ver valores marcados en negrita en las tablas 6.11 y 6.12). En la gráfica 6.3, representada a continuación, se detalla una comparación de los valores promedios obtenidos con estos tres índices.

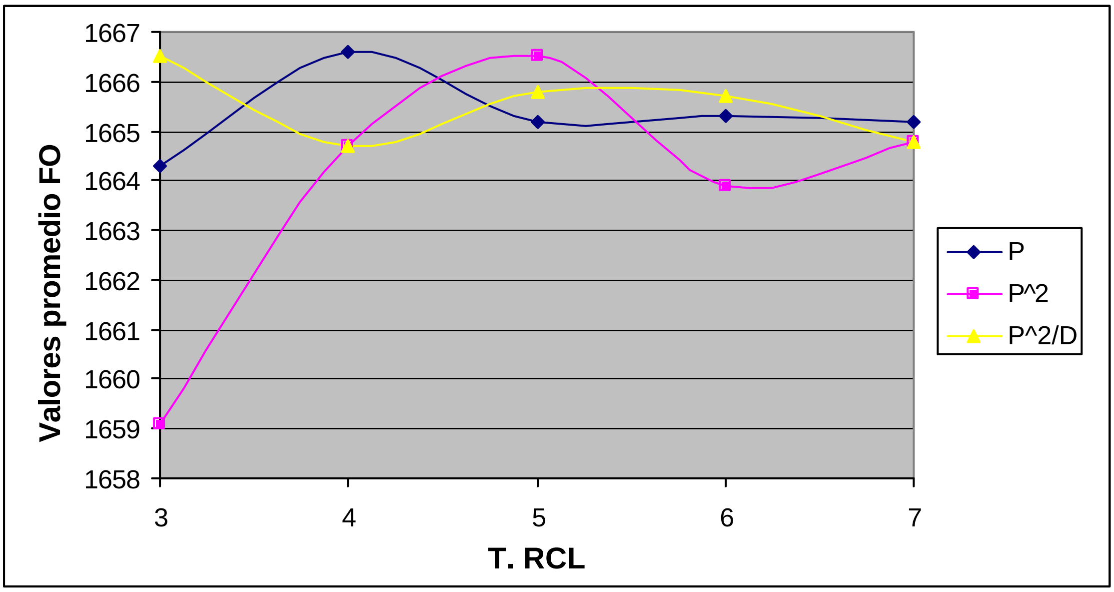

Figura 6.3: Comparación de los valores promedio del "Peso", "Peso^2" y el "Peso^2 / Duración".

Con todo lo realizado hasta el momento, hemos conseguido reducir el número de funciones índices candidatas a sólo tres. Nuestro objetivo sigue siendo identificar cuál de ellas es la que nos ofrece los mejores resultados, por lo que, a continuación, aplicaremos la metaheurística GRASP con cada uno de los índices mencionados sobre todo el conjunto de problemas. Los valores medios resultantes nos permitirán obtener entonces una solución concluyente. En el algoritmo incluiremos, además, el módulo de búsqueda local, con idea de que los resultados nos sirvan para comparaciones posteriores.

Los resultados obtenidos son los expuestos en la tabla 6.13:

Fichero					F. Índice: P	F. Índice: P^2	F. Índice: P^2 / D
r	n	m	A	C			
1.5	25	4	3	2	1007,3	1007,3	1007,3
1.5	50	4	3	2	2187,9	2186	2186,8
1.5	50	8	3	4	2168,4	2164,4	2167,2
1.5	100	4	3	2	4120,6	4106,1	4119,4
1.5	100	8	3	4	4510,6	4531,2	4499,1
1.5	100	16	3	4	4892,4	4907	4892,2
1.5	100	16	4	4	4755,8	4750,7	4744,9
1.5	200	4	3	2	8294,3	8263,9	8280
1.5	200	8	3	4	9212,4	9245,9	9198,1
1.5	200	16	3	4	9395	9450,7	9379,6
1.5	200	16	4	4	9619,1	9608,4	9608,5
3	25	4	3	2	852,8	852,8	852,8
3	50	4	3	2	1645,3	1643	1642,7
3	50	8	3	4	1736,6	1728,8	1736

3	100	4	3	2	3350,7	3332	3343
3	100	8	3	4	3655,8	3648,8	3646,5
3	100	16	3	4	3787,9	3788	3779,6
3	100	16	4	4	3664,3	3658,3	3646,7
3	200	4	3	2	6679,4	6640,9	6672
3	200	8	3	4	7130,7	7148	7130,3
3	200	16	3	4	7420,9	7442,9	7412
3	200	16	4	4	7613,2	7579,1	7620,2
6	25	4	3	2	574,1	574,1	577,2
6	50	4	3	2	1067	1063,6	1063,3
6	50	8	3	4	1264,7	1259,2	1264,7
6	100	4	3	2	2454	2440,4	2453,5
6	100	8	3	4	2506,4	2502,8	2499
6	100	16	3	4	2668,8	2658,7	2648,4
6	100	16	4	4	2583,2	2576,5	2583
6	200	4	3	2	4733,2	4734,8	4738,2
6	200	8	3	4	5126,2	5115,1	5113,4
6	200	16	3	4	5011,3	5002,3	5005,2
6	200	16	4	4	5282,3	5250,5	5279,5

Tabla 6.13: Valores medios de la FO para el "Peso", "Peso^2" y "Peso^2 / Duración".

	F. Índice: P	**F. Índice: P^2**	**F. Índice: P^2 / D**
Valores promedio FO	**4271,9**	4268,6	4266,4

Tabla 6.14: Valores promedio de la FO para el "Peso", el "Peso^2" y el "Peso^2 / Duración".

Los resultados expuestos en la tabla 6.14, que representan los valores promedio de la FO logrados con cada uno de los índices, nos permiten afirmar que la opción más idónea cuando se utiliza la metaheurística GRASP en la resolución de este tipo de problemas es la mostrada a continuación:

	OPCIÓN
Función índice	Peso
T. RCL	4, fijado de antemano
Mét. de asignación	Aleatorio

Tabla 6.15: Valores finales de los parámetros.

Una vez fijados los parámetros y el método de asignación con los que se consiguen los mejores resultados, nos queda, para concluir este análisis de sensibilidad, estudiar la dependencia del comportamiento de la metaheurística con el número de iteraciones. Para ello, y fijando los parámetros en los valores anteriormente definidos, aplicaremos la metaheurística sobre todo el conjunto de problemas, variando el número de iteraciones entre 500, 1000 y 2000 iteraciones.

Los resultados obtenidos se presentan en la tabla 6.16:

Fichero					N.I. = 500	N.I. =1000	N.I. =2000
r	n	m	A	C			
1.5	25	4	3	2	1007,3	1007,3	1007,3
1.5	50	4	3	2	2187,7	2187,9	2188,9
1.5	50	8	3	4	2158,6	2168,4	2170,6
1.5	100	4	3	2	4108,9	4120,6	4123,4
1.5	100	8	3	4	4490	4510,6	4519,4
1.5	100	16	3	4	4879,8	4892,4	4903,5
1.5	100	16	4	4	4736,3	4755,8	4760,3
1.5	200	4	3	2	8269,3	8294,3	8298,7
1.5	200	8	3	4	9169,2	9212,4	9228,1
1.5	200	16	3	4	9371,1	9395	9404,8
1.5	200	16	4	4	9593,2	9619,1	9627,1
3	25	4	3	2	852,8	852,8	852,8
3	50	4	3	2	1641,3	1645,3	1645,9
3	50	8	3	4	1726,8	1736,6	1742,2
3	100	4	3	2	3333,6	3350,7	3355
3	100	8	3	4	3624,2	3655,8	3669,5
3	100	16	3	4	3763,7	3787,9	3795,6
3	100	16	4	4	3642,6	3664,3	3670,7
3	200	4	3	2	6635,1	6679,4	6694,1
3	200	8	3	4	7084,5	7130,7	7139,9
3	200	16	3	4	7380,4	7420,9	7441,2
3	200	16	4	4	7560,5	7613,2	7621
6	25	4	3	2	574,1	574,1	577,2
6	50	4	3	2	1058,1	1067	1067,7

6	50	8	3	4	1257,2	1264,7	1267,5
6	100	4	3	2	2439,1	2454	2455,6
6	100	8	3	4	2486,4	2506,4	2512,6
6	100	16	3	4	2636,2	2668,8	2676,6
6	100	16	4	4	2571,6	2583,2	2588,8
6	200	4	3	2	4704,2	4733,2	4738,1
6	200	8	3	4	5085,1	5126,2	5133,1
6	200	16	3	4	4977,5	5011,3	5019,3
6	200	16	4	4	5247	5282,3	5291,2

Tabla 6.16: Valores medios de la FO según el número de iteraciones.

	N.I. = 500	**N.I. =1000**	**N.I. =2000**
Valores promedio FO	4250,1	4271,9	4278,4

Tabla 6.17: Valores promedio de la FO según el número de iteraciones.

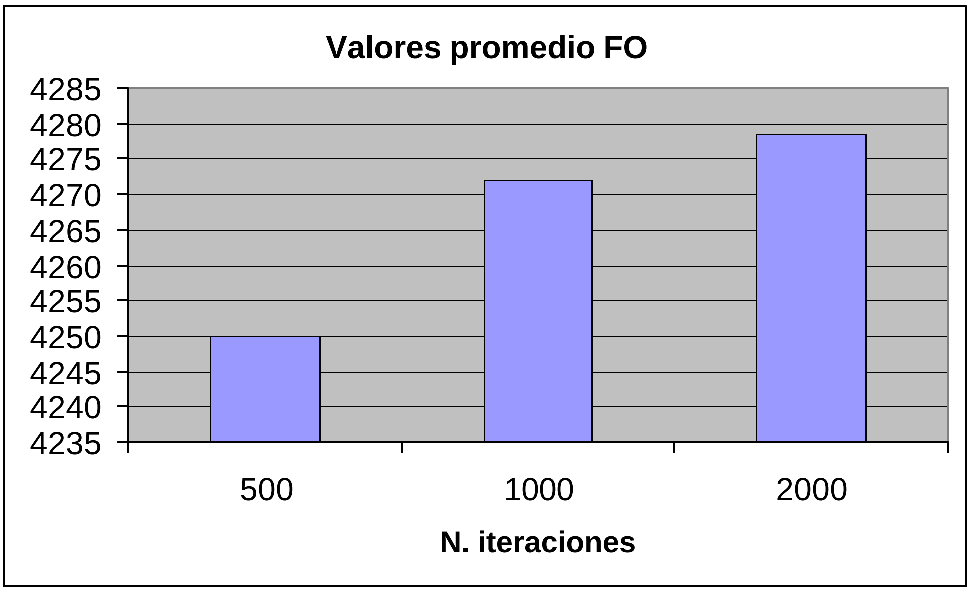

Figura 6.4:Valor promedio de la FO en función del número de iteraciones.

En los resultados de la tabla 6.17, representados gráficamente en la figura 6.4, se advierte que, al aumentar el número de iteraciones con las que trabaja la metaheurística, se produce un incremento en el valor promedio alcanzado por la FO. Esto resulta obvio si tenemos en cuenta que, al aumentar el número de iteraciones, estamos ofreciendo a la metaheurística un mayor número de oportunidades para construir soluciones, lo que implica que la probabilidad de obtener un valor más cercano al óptimo aumenta.

Fijémonos ahora en los tiempos medios de computación asociados a los tres casos anteriores. Éstos aparecen representados en la tabla 6.18:

	N.I. = 500	N.I. =1000	N.I. =2000
Tiempos medios	0:00:47	0:01:33	0:03:14

Tabla 6.18: Tiempos medios de computación asociados al número de iteraciones.

Si representamos gráficamente los incrementos medios producidos en la FO en función de los tiempos medios de computación que han sido necesarios para generarlos, tenemos que:

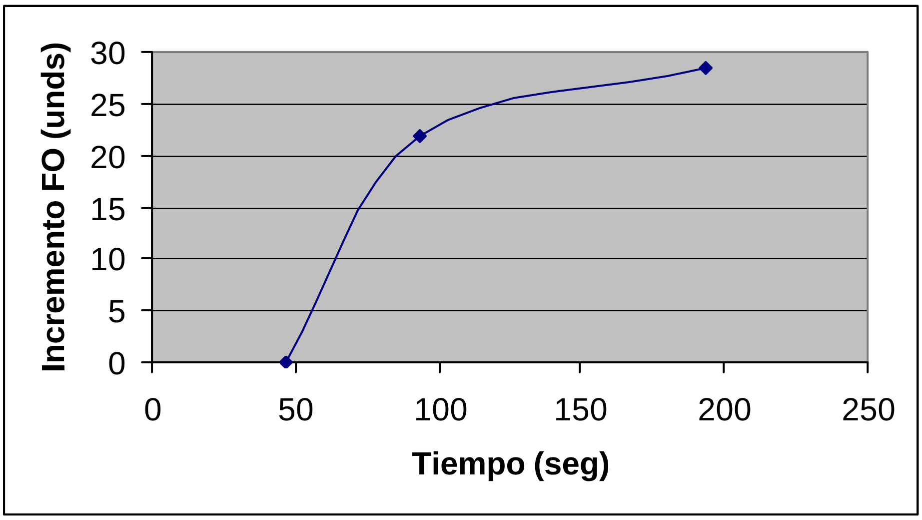

Figura 6.5: Incrementos medios en la FO en función de los tiempos de computación.

Fijándonos en la representación gráfica 6.5 notamos que, aunque hemos obtenido mejoras tanto al pasar de 500 a 1000 iteraciones como al pasar de 1000 a 2000 iteraciones, en ambos casos ni los valores conseguidos ni los tiempos de computación necesarios son similares.

Cuando al comienzo de este proyecto citábamos las características que debe tener una buena metaheurística, hacíamos especial hincapié en que una de las funciones más importantes que debe cumplir cualquier metodología de este tipo es ofrecer información de calidad, en un tiempo razonable, acerca del problema sobre el que trabaja. En la situación en la que nos encontramos y a tenor de los resultados obtenidos, resulta justificado asumir el cambio de 500 a 1000 iteraciones, pues la metaheurística consigue un incremento del valor medio de la FO de 21,6 unidades requiriendo, para ello, un tiempo medio de computación extra de tan solo 46 segundos. Esta decisión se ve, además, reforzada si tenemos en cuenta que el incremento conseguido se refleja sobre todo en los problemas de 100 y 200 trabajos ya que, si nos fijamos en la tabla de resultados 6.16, las diferencias existentes entre los valores medios de la FO en los problemas de 25 y 50 trabajos están, en general, muy por debajo del incremento medio resultante.

Al estudiar la posibilidad de cambiar de 1000 a 2000 iteraciones, encontramos que esta opción no resulta tan interesante como la anterior, puesto que, con el cambio, aumentaríamos el tiempo medio de computación en 1:41 segundos para obtener un incremento del valor medio de la FO de tan sólo 6,5 unidades. Además, hay que tener en cuenta que, si seguimos aumentando el número de iteraciones, este comportamiento se irá agudizando, puesto que, al estar cada vez más cerca del valor óptimo, a la metaheurística le costará más trabajo mejorar los valores obtenidos.

Por tanto, concluimos que, tomar un número de iteraciones igual a 1000, resulta ser la opción más idónea para resolver este tipo de problemas mediante la metaheurística GRASP aquí planteada.

Una vez definidos los valores de los parámetros a utilizar, damos por concluido el análisis de sensibilidad, recordando, en la siguiente tabla, los resultados obtenidos:

	OPCIÓN
Función índice	Peso
T. RCL	4, fijado de antemano
Mét. de asignación	Aleatorio
N. iteraciones	1000

Tabla 6.15: Valores finales de los parámetros a utilizar.

6.2.2. Comparación de resultados

En este apartado, vamos a realizar una comparación de los resultados obtenidos por la metaheurística GRASP, trabajando ésta con el "Peso" como función índice, una asignación aleatoria y un tamaño de la RCL igual a cuatro (caso más favorable de los estudiados), con los valores obtenidos por el algoritmo exacto y con los generados por el algoritmo tabú sobre los mismos problemas, tomando, tanto para la metaheurística GRASP como para el algoritmo tabú, un número de iteraciones igual a 1000.

La comparación con los valores exactos nos va a permitir tener una idea del nivel de calidad de las soluciones obtenidas cuando se aplica la metaheurística aquí generada a problemas del tipo POC. Hemos de tener en cuenta, eso sí, que la aplicación del algoritmo exacto visto en el punto cuatro, a problemas de este tipo, resulta más compleja y costosa a medida que aumentan los tipos de trabajos, las clases de máquinas y el número de máquinas utilizadas. Por ello, sólo dispondremos de los valores exactos en los casos en los que m=4, A=3 y C=2. Dichos valores quedan expuestos en la tabla 6.18:

Fichero					Valores medios exactos
r	n	m	A	C	
1.5	25	4	3	2	1007,3
1.5	50	4	3	2	2191,4
1.5	100	4	3	2	4136,4
1.5	200	4	3	2	8378,8
3	25	4	3	2	852,8
3	50	4	3	2	1651,9
3	100	4	3	2	3384,7
3	200	4	3	2	6847,7
6	25	4	3	2	579,2
6	50	4	3	2	1072,6
6	100	4	3	2	2527,3
6	200	4	3	2	5014,8

Tabla 6.18: Valores medios obtenidos con el algoritmo exacto.

Procedemos ahora a comparar los valores de la tabla 6.18 con los obtenidos sobre los mismos problemas por la metaheurística GRASP. Los errores relativos resultantes se definen en la siguiente tabla:

Fichero					Valores exactos	Valores GRASP	ERROR (%)
r	n	m	A	C			
1.5	25	4	3	2	1007,3	1007,3	0,00
1.5	50	4	3	2	2191,4	2187,9	0,16
1.5	100	4	3	2	4136,4	4120,6	0,38
1.5	200	4	3	2	8378,8	8294,3	1,02
3	25	4	3	2	852,8	852,8	0,00
3	50	4	3	2	1651,9	1645,3	0,40
3	100	4	3	2	3384,7	3350,7	1,01
3	200	4	3	2	6847,7	6679,4	2,52
6	25	4	3	2	579,2	574,1	0,89
6	50	4	3	2	1072,6	1067	0,52
6	100	4	3	2	2527,3	2454	2,99
6	200	4	3	2	5014,8	4733,2	5,95

Tabla 6.19: Errores medios del GRASP respecto a los valores exactos.

	GRASP
Error promedio (%)	**1,32**

Tabla 6.20: Error promedio del GRASP respecto a los valores exactos.

Para estudiar más detalladamente el comportamiento de la metaheurística GRASP, vamos a representar gráficamente su evolución en función del índice de solapamiento ρ y del número de trabajos n. Esto nos permitirá detectar qué parámetros tienen una mayor influencia sobre su funcionamiento.

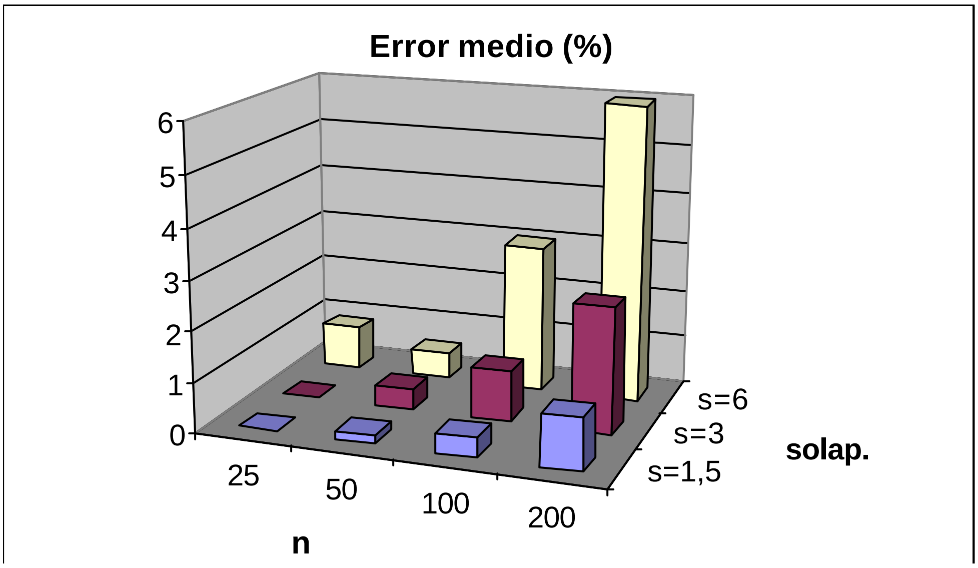

Figura 6.6: Error medio en función de n y r.

Hemos de tener en cuenta que un tamaño de la RCL constante no resulta ser igual de útil para todos los casos. Si el tamaño de la RCL permanece invariable a medida que aumentamos el número de trabajos a considerar, vamos a obtener como resultado una segura pérdida de aleatoriedad, lo que, como ya se explico anteriormente, traerá consigo un descenso en el nivel de calidad de los resultados obtenidos. Una posible solución a este defecto, consiste en generar una RCL cuyo tamaño dependa del número de trabajos que se tengan que procesar en cada uno de los casos. Esta solución, que en principio parece sencilla, no lo es tanto, pues si el tamaño de la RCL resulta ser demasiado grande, tareas de poco interés desde el punto de vista de la FO entrarán a formar parte de la RCL, con lo que aumenta el riesgo de que la solución empeore.

Por otro lado, al aumentar ρ, el número de soluciones factibles en cada caso también aumenta. Considerando que el número de iteraciones se puede entender como el número de oportunidades que se le ofrece a la metaheurística para conseguir una buena solución, al mantener este valor fijo, estamos repercutiendo negativamente en el funcionamiento del algoritmo. Ya se ha visto anteriormente que un aumento en el número de iteraciones produce, efectivamente, una mejora en la solución. Sin embargo, esta mejora trae consigo también un aumento en el tiempo de computación necesario para la resolución del problema, y esto, como ya hemos explicado, va en contra del principio de creación de la propia metaheurística.

En lo que se refiere a la influencia de los parámetros en el comportamiento de la metaheurística, se ha comprobado experimentalmente que, a medida que crecen los valores del índice de solapamiento ρ y del número de trabajos n, el error en el que se incurre al aumentar el valor de ρ manteniendo n constante es mayor que el debido a aumentar n manteniendo ρ constante, por lo que, podemos afirmar, que la metaheurística se adapta peor a cambios en el índice de solapamiento que a cambios en el número de trabajos.

De los 120 problemas considerados en esta primera comparación, la metaheurística GRASP generada en este proyecto obtiene la solución óptima en 49 de ellos, lo que supone un **40,83 %** de los casos.

La distribución de los óptimos en función de los parámetros del problema es la siguiente:

n	Óptimos
25	28
50	19
100	2
200	0

Tabla 6.21: Distribución de los valores óptimos en función de n.

r	Óptimos
1,5	20
3	15
6	14

Tabla 6.22: Distribución de los valores óptimos en función de r.

Obviamente, al aumentar el índice de solapamiento y el número de trabajos, no sólo aumenta el error cometido sino que también disminuye el número de óptimos encontrados.

Pasamos ahora a comparar los resultados obtenidos por la metaheurística GRASP implementada, con los conseguidos por el algoritmo tabú, habiéndose aplicado ambos métodos sobre todo el conjunto de problemas.

En la tabla 6.22 se presenta una comparación de los valores medios de la FO obtenidos con:

- Metaheurística GRASP, con el “Peso” como función índice, asignación aleatoria y tamaño de la RCL igual a 4.
- Algoritmo tabú con lista tabú de tamaño variable.
- Para ambos, N.I.=1000.

Fichero					Valor FO (Tabú)	Valor FO (GRASP)
r	n	M	A	C		
1.5	25	4	3	2	1000,8	1007,3
1.5	50	4	3	2	2185,3	2187,9
1.5	50	8	3	4	2140,8	2168,4
1.5	100	4	3	2	4097	4120,6
1.5	100	8	3	4	4506,5	4510,6
1.5	100	16	3	4	4921,7	4892,4
1.5	100	16	4	4	4749	4755,8
1.5	200	4	3	2	8262,7	8294,3
1.5	200	8	3	4	9265,3	9212,4
1.5	200	16	3	4	9516,3	9395
1.5	200	16	4	4	9661,4	9619,1
3	25	4	3	2	852,3	852,8
3	50	4	3	2	1635,7	1645,3
3	50	8	3	4	1716,5	1736,6
3	100	4	3	2	3320,6	3350,7
3	100	8	3	4	3654,8	3655,8
3	100	16	3	4	3814,4	3787,9
3	100	16	4	4	3654,3	3664,3
3	200	4	3	2	6703,8	6679,4
3	200	8	3	4	7198,3	7130,7
3	200	16	3	4	7509,7	7420,9
3	200	16	4	4	7652,2	7613,2
6	25	4	3	2	579,2	574,1
6	50	4	3	2	1066	1067
6	50	8	3	4	1262,8	1264,7
6	100	4	3	2	2462	2454
6	100	8	3	4	2549,5	2506,4
6	100	16	3	4	2710,6	2668,8
6	100	16	4	4	2620,7	2583,2
6	200	4	3	2	4856,3	4733,2
6	200	8	3	4	5209,1	5126,2
6	200	16	3	4	5162,3	5011,3
6	200	16	4	4	5416,2	5282,3

Tabla 6.23: Comparación de los valores medios de la FO entre tabú y GRASP.

	Tabú	GRASP
Valor promedio FO	**4300,4**	4271,9

Tabla 6.24: Comparación de los valores promedio de la FO entre tabú y GRASP.

Tal y como podemos observar en la tabla 6.24, el algoritmo tabú obtiene un valor promedio de la FO superior en 28,5 unidades al conseguido por la metaheurística GRASP.

Aunque, según los resultados de la tabla 6.23, la metaheurística GRASP resulta ser muy competitiva para problemas en los que $\rho \leq 3$ y $n \leq 100$, al igual que ocurría en la comparación con el algoritmo exacto, a medida que aumentamos los valores de ρ, dicha metaheurística va perdiendo definición, acumulando, cada vez, errores más elevados. Si, a medida que la complejidad de los problemas crece, la diferencia entre los valores medios de la FO obtenidos por ambas metaheurísticas también aumenta, no cabe la menor duda de que el algoritmo tabú resulta ser más idóneo que la metaheurística GRASP en lo que se refiere a la resolución de este tipo de problemas. En la figura representada a continuación, en la que se describen las diferencias entre los valores medios de las FO en función de los índices de solapamiento, se puede observar de forma clara lo descrito anteriormente:

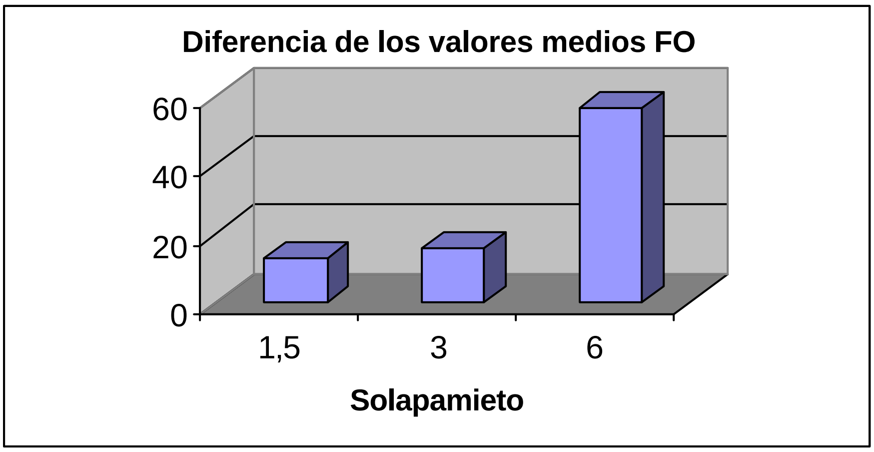

Figura 6.7: Diferencia de los valores medios de la FO en función de r.

Si realizamos ahora una comparación caso por caso encontramos que, en los 330 problemas estudiados:

- La metaheurística GRASP mejora los resultados obtenidos por el algoritmo tabú en 118 ocasiones, lo que supone un **35,76%** del total de casos considerados.
- El algoritmo tabú mejora los resultados obtenidos por la metaheurística GRASP en 179 ocasiones, lo que supone un **54,24%** del total de casos considerados.

- En 33 ocasiones, ambos algoritmos consiguen los mismos resultados, lo que supone un **10%** del total de casos considerados.

La distribución de estos resultados en función de los parámetros del problema se expone en las siguientes tablas:

n	Tabú	GRASP	Igualdad
25	2	4	24
50	13	38	9
100	65	55	0
200	99	21	0

Tabla 6.25: Distribución de los mejores valores en función de n.

r	Tabú	GRASP	Igualdad
1,5	53	44	13
3	55	44	11
6	71	30	9

Tabla 6.26: Distribución de los mejores valores en función de r.

6.2.3. Análisis de los resultados

En este apartado se ha desarrollado un estudio cuya finalidad ha sido fijar los valores de los parámetros de la metaheurística GRASP, de forma que se obtuvieran los mejores resultados al aplicar dicha metaheurística a problemas del tipo POC con objetivo la maximización de la suma de los pesos de los trabajos procesados.

Los resultados de este análisis han sido los siguientes:

1. Experimentos previos respecto a la función índice de GRASP, pusieron de manifiesto que el uso de la componente adaptativa del método no proporcionaba mejoras respecto a los índices analizados en el análisis de sensibilidad.
2. La metaheurística GRASP, trabajando con una RCL variable cuyo tamaño esté determinado por un factor α, no ha resultado ser apta para la resolución de este tipo de problemas.

3. La duración de los trabajos no ha resultado ser un factor determinante en la resolución del problema pues los mejores resultados se han conseguido tomando como función índice el “Peso”.
4. Fijando el número de iteraciones en 1000, la metaheurística ha logrado la relación más positiva entre resultados y tiempos de computación.

La comparación de los resultados obtenidos por la metaheurística GRASP con los conseguidos por el algoritmo exacto y el algoritmo tabú, han mostrado lo siguiente:

1. Trabajando con los 120 problemas en los que se cuenta con la solución óptima, la metaheurística GRASP obtiene un error promedio del 1,32%, alcanzando el valor óptimo en 49 de los casos. Esto supone un porcentaje de acierto del 40,83 %.
2. Considerando los 320 casos estudiados, el algoritmo tabú obtiene un valor promedio de la FO de 4300,4, frente al 4271,9 obtenido por la metaheurística GRASP. Esto supone un error relativo entre ambos del 0,67%.
3. Se ha demostrado que, en este caso, el comportamiento de la metaheurística GRASP resulta ser especialmente sensible a las variaciones producidas en el índice de solapamiento ρ, de forma que, al aumentar dicho índice, el error acumulado en los resultados también crece.
4. En lo que se refiere a este caso, queda probado que la metaheurística GRASP no supera los resultados logrados por el algoritmo tabú, puesto que, aunque para problemas sencillos resulte más exacta la metodología GRASP, a medida que aumenta la complejidad del problema, el algoritmo tabú va resultando cada vez más efectivo.

6.3. Objetivo: Maximizar el número de trabajos procesados

En este apartado vamos a utilizar la metaheurística GRASP para resolver problemas del tipo POC en los que el objetivo a lograr es maximizar el número de trabajos procesados.

En todos los casos estudiados hasta ahora, el peso de los trabajos ha sido el factor que ha determinado la asignación de cada una de las tareas a las máquinas existentes. Sin embargo, en este nuevo escenario nuestro objetivo es incluir en la solución el mayor número de trabajos posible, y para conseguirlo, en cada una de las iteraciones, debemos incluir en la RCL a aquellas tareas con una duración menor. Así aseguramos, en la asignación de la siguiente iteración, un mayor espacio disponible en las listas de trabajo de cada una de las máquinas. El peso de cada uno de los trabajos es, ahora, irrelevante.

En lo que se refiere al resto de parámetros, utilizaremos los mismos valores con los que obtuvimos los mejores resultados en los casos anteriores, puesto que nos interesa que el funcionamiento de la metaheurística, salvo en el valor de la función índice, sea el mismo.

Los valores de los parámetros de funcionamiento que utilizaremos en este caso serán, pues, los siguientes:

	OPCIÓN
Función índice	Duración
T. RCL	4, fijado de antemano
Mét. de asignación	Aleatorio
N. iteraciones	1000

Tabla 6.27: Valores de los parámetros de la metaheurística GRASP.

A continuación, vamos a realizar una comparación de los resultados obtenidos por la metaheurística GRASP, fijando sus parámetros en los valores anteriores, con los conseguidos por el algoritmo tabú, trabajando ambos sobre todo el conjunto de problemas disponibles.

6.3.1. Comparación de resultados

Fichero					Valor FO (TABU)	Valor FO (GRASP)
r	n	m	A	C		
1.5	50	4	3	2	39,3	39,4
1.5	50	8	3	4	40,6	40,7
1.5	100	4	3	2	75,2	75,4
1.5	100	8	3	4	80,4	80
1.5	100	16	3	4	89,6	87,9
1.5	100	16	4	4	87,5	87,3
1.5	200	4	3	2	148,4	147,5
1.5	200	8	3	4	164,3	161,7
1.5	200	16	3	4	176,3	170,7
1.5	200	16	4	4	179,6	175
3	50	4	3	2	29,1	29,1
3	50	8	3	4	29,1	29,4
3	100	4	3	2	56,5	57
3	100	8	3	4	62,3	61,2
3	100	16	3	4	59,4	60,5
3	100	16	4	4	61,9	62,3
3	200	4	3	2	110,5	110,1
3	200	8	3	4	120,4	117,7
3	200	16	3	4	127,2	124,6
3	200	16	4	4	129,9	128
6	50	4	3	2	18,6	18,7
6	50	8	3	4	21,5	21,9
6	100	4	3	2	41,3	41,5
6	100	8	3	4	41,5	41,8
6	100	16	3	4	42,7	44,1
6	100	16	4	4	42,5	44
6	200	4	3	2	79,9	80,4
6	200	8	3	4	86,3	85
6	200	16	3	4	84,3	86,7
6	200	16	4	4	89,5	90,7

Tabla 6.28: Comparación de los valores medios de la FO entre tabú y GRASP.

	Tabú	**GRASP**
Valor promedio FO	**80,52**	80,01

Tabla 6.29: Comparación de los valores promedio de la FO entre tabú y GRASP.

Si representamos gráficamente los valores obtenidos por cada una de las metodologías en función de ρ, tenemos que:

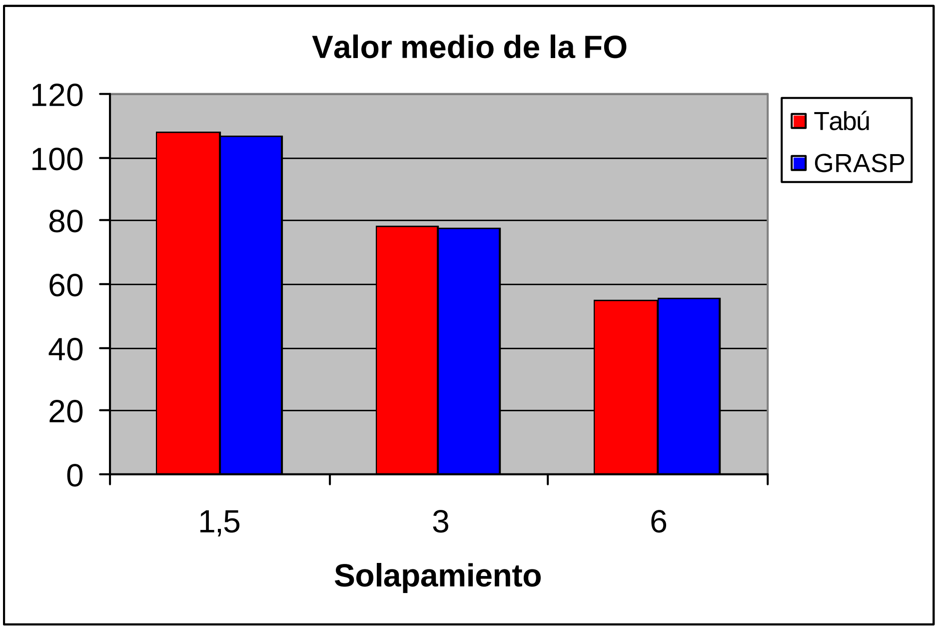

Figura 6.8: Valores medios de la FO para tabú y GRASP en función de r.

En este caso, tal y como vemos en la tabla 6.29, también ocurre que, en los 300 casos estudiados, el algoritmo tabú obtiene un valor promedio de la FO mayor que el conseguido por la metaheurística GRASP. Sin embargo, al estudiar la figura 6.8, nos damos cuenta de que la metaheurística GRASP, a pesar de que para problemas sencillos (ρ y n pequeños) obtiene peores resultados que el algoritmo tabú, a medida que aumenta el valor de ρ va resultando más efectiva, de forma que cuando ρ = 6 ya supera los valores logrados por el algoritmo tabú. Esta tendencia se pone también de manifiesto en las tablas de distribución de resultados expuestas a continuación:

n	Tabú	GRASP	Igualdad
50	4	14	42
100	31	51	38
200	77	26	17

Tabla 6.30: Distribución de los mejores valores en función de n.

r	Tabú	GRASP	Igualdad

1,5	55	17	28
3	43	23	34
6	14	51	35

Tabla 6.31: Distribución de los mejores valores en función de r.

Efectivamente, en la tabla 6.31, observamos como el número de veces que la metaheurística GRASP mejora los resultados logrados por el algoritmo tabú va creciendo a medida que aumenta el valor del índice ρ. Además, hemos de tener en cuenta que si continuáramos aumentando la complejidad de los problemas a resolver, la metaheurística GRASP se mostraría cada vez más útil.

Por todo lo anterior, podemos concluir que la metaheurística GRASP resulta más idónea que el algoritmo tabú cuando se trata de resolver problemas del tipo POC en los que el objetivo es maximizar el número de trabajos procesados.

Realizando, como en el punto anterior, una comparación caso por caso, obtenemos que, en los 300 problemas estudiados:

- La metaheurística GRASP mejora los resultados obtenidos por el algoritmo tabú en 91 ocasiones, lo que supone un **30,33%** del total de casos considerados.
- El algoritmo tabú mejora los resultados obtenidos por la metaheurística GRASP en 112 ocasiones, lo que supone un **37,33%** del total de casos considerados.
- En 97 ocasiones, ambos algoritmos consiguen los mismos resultados, lo que supone un **32,33%** del total de casos considerados.

6.3.2. Análisis de los resultados

En este apartado tratamos el problema POC suponiendo el mismo peso para todos los trabajos, por lo que el objetivo se convierte en el de maximizar el número de trabajos procesados.

En lo que se refiere a la implantación de la metaheurística para este nuevo escenario, cabe destacar lo siguiente:

1. El comportamiento exigido a la metaheurística en este caso es muy similar al del apartado anterior por lo que, exceptuando el valor de la función

índice, se ha mantenido el valor de todos los parámetros de funcionamiento.

2. Dado que la resolución del problema se basa ahora en asignar a las máquinas la mayor cantidad de trabajos se ha decidido tomar la ”Duración” como función índice.

La comparación de los resultados conseguidos por la metaheurística GRASP con los logrados por el algoritmo tabú ha deparado lo siguiente:

1. Considerando los 300 casos estudiados, el algoritmo tabú obtiene un valor promedio de la FO de 80,52 frente al 80,01 obtenido por la metaheurística GRASP. Esto supone un error relativo entre ambos del 0,64%.
2. Se ha comprobado que la eficiencia de la metaheurística GRASP crece a medida que aumenta la complejidad del problema, es decir, a medida que aumenta el valor de ρ, la calidad de los resultados obtenidos con la metaheurística GRASP va siendo más elevada.
3. En lo que se refiere a este caso, queda probado que la metaheurística GRASP resulta ser más apropiada que el algoritmo tabú, puesto que, aunque para problemas sencillos resulte más exacta la metodología tabú, a medida que aumenta la complejidad del problema, la metaheurística GRASP va resultando cada vez más efectiva. De hecho, para $\rho = 6$ (caso más complejo de los estudiados) el error relativo resultante entre los valores medios de la FO es ya favorable a la heurística GRASP y su valor es de 1,22%.