Algoritmos genéticos multimodales: Un estudio sobre la parametrización del método clearing aplicado al problema “job shop”

M. Elena Pérez[1], Francisco Herrera[2]

Resumen--**Los algoritmos genéticos están especialmente adaptados a la optimización multimodal como es el caso del problema de secuenciación tipo job shop (*scheduling*). Dentro de los muchos trabajos que se han desarrollado en este ámbito, el método que mejores resultados está obteniendo es el *clearing*. Sin embargo, es un método paramétrico y se sabe que su potencia depende de una buena adaptación a las características del problema. Pero no sólo es cierto esto, sino que además es necesario buscar dicha adaptación en función de lo que estemos buscando, una alta eficacia, o (y esto es lo novedoso del estudio) una alta efectividad, entendida esta como la capacidad del algoritmo para encontrar el mayor número de óptimos del problema en una única ejecución.**

Palabras clave—Multimodalidad, clearing, efectividad, job shop.

I. INTRODUCCIÓN.

El objetivo del problema de secuenciación conocido por su término inglés *scheduling* es establecer la secuencia de trabajos que debe realizarse en cada máquina del taller para lograr, como primer objetivo fabricar un conjunto de piezas y, como segundo objetivo, en el menor tiempo posible. Para simplificar el proceso de definición de los problema se utiliza la siguiente nomenclatura n/m/T/E, siendo: *n* el número de trabajo o piezas a realizar, *m* el número de máquinas disponibles en el taller para ello, *T* el tipo de flujo de fabricación (flow shop, job shop, etc..) y, por último, *E* el criterio de eficacia perseguido. (Conway, Maxwell y Miller, 1967; Brucker, 1997)

Este tipo de problemas es *NP-hard* (Garey y Jonson, 1979), por lo que no existe un algoritmo conocido que asegure encontrar la solución óptima en un tiempo polinomial. Por este motivo, ha sido tradicionalmente utilizado como test en la validación de nuevos algoritmos desarrollados.

Además en los últimos años, los estudios se han centrado no ya en encontrar la solución óptima del problema, sino en intentar alcanzar un abanico de soluciones óptimas todas ellas igualmente válidas (multimodalidad) por varios motivos:

- Dar al ejecutor del plan más flexibilidad ante contingencias no siempre cuantificables o incorporables en el modelo del problema (averías de máquinas, roturas de stocks, etc..).
- Cuando el problema es complejo o multimodal, donde mantener la diversidad en la búsqueda es necesario para no quedar atrapado en subóptimos.

Por estas razones, hace ya 20 años surgieron los primeros estudios de multimodalidad con algoritmos genéticos de la mano de Goldberg (Goldberg y Richardson, 1987). Desde entonces han sido numerosos las aproximaciones y algoritmos desarrollados con tal fin. Entre ellos, el que ha demostrado ser de los más potentes (Sareni y Krahenbuhl, 1998; Pérez et al., 2003), es el método *clearing* desarrollado en el año 95.

En este trabajo, pretendemos conocer un poco más sobre el comportamiento de este método con respecto a la variación de sus parámetros: el radio que define la pertenencia a un nicho (σ_{share}), el número de pertenecientes al nicho (*k*), y el sistema de selección, con dos objetivos claros: analizar su influencia en la eficacia del método, y principalmente, sobre su efectividad, entendida esta como la capacidad de alcanzar en una única ejecución del algoritmo el mayor número posible de soluciones óptimas globales del problema. Además, se estudiará si es posible dirigir la búsqueda hacia la exploración del espacio de búsqueda o su explotación, variando la parametrización.

Para ello, en la sección 2 introduciremos brevemente el problema de secuenciación dando una amplia bibliografía para quien desee profundizar, y se describirán los ejemplos sobre los que se realizará la experimentación. En la sección 3 explicaremos detalladamente el método clearing definiendo los parámetros utilizados. La experimentación se describirá en la sección 4 donde además resumiremos los resultados obtenidos. Y

[1] elena@eis.uva.es
Dpto. Organización de Empresas y C.I.M., Universidad de Valladolid, 47011 - Valladolid

[2] herrera@decsai.ugr.es
Dpto. Ciencias de la Computación e I.A., Universidad de Granada, 18071 – Granada

terminaremos con las conclusiones más relevantes al estudio realizado y la ampliación futura de trabajos.

II. TEORÍA DE LA SECUENCIACIÓN (SCHEDULING)

Como ya se ha comentado en la introducción, el problema de secuenciación puede ser caracterizado mediante cuatro dimensiones ($n/m/T/E$), así el problema en el que nos centraremos es ($n/m/J/C_{max}$), ya que hemos elegido la forma más compleja de secuenciación, el job shop (J), y se desea minimizar el tiempo máximo de permanencia en el taller (C_{max}).

Cada trabajo o pieza a realizar está formado por un conjunto de operaciones (O_{ij}), cada una de las cuales deben ser procesadas en una determinada máquina durante un tiempo prefijado de antemano.

El tipo de flujo *job shop* es el más general y complejo, donde cada trabajo tiene su propio flujo o movimiento sobre las máquinas y a su vez, cada máquina tiene su propia secuencia de trabajos.

El criterio de eficacia puede definirse en función de un tiempo: tiempo de terminación (*makespan* o C_{max}), tiempos de retraso (delay times o T_{max}) o tiempo total de flujo en el taller (total flow time o F_{max}), o también puede ser definido como un coste, por ejemplo, cuantificando el coste para la empresa el retraso en la entrega de una pieza al cliente (Brucker, 1997).

Las primeras aproximaciones se orientaron a la definición del modelo (French, 1982) y su resolución mediante métodos matemáticos [Greenberg, 1968; Carlier, y Pinson, 1989]. Sin embargo, debido a las limitaciones del problema las aproximaciones del modelo estaban limitadas al ámbito académico no siendo útiles en la empresa. Por lo que rápidamente surgieron otros enfoques para su resolución, los métodos heurísticos (Panwalkar y Iskander, 1977, Adams et al., 1988), y metaheuristicos como la búsqueda tabu (Glover y Laguna, 1997; Nowicki y Smutnicki, 1996), recocido simulado (Kirkpatrick et al., 1983; Van et al., 1992), algoritmos genéticos (AG en adelante) (Mattfeld, 1995), redes neuronales (Yang, 2000), búsqueda local iterativa (Ramalhinho et al., 2000), y estrategias híbridas de optimización (Wang y Zheng, 2001), entre otras.

Como nosotros intentamos no solo óptimizar el problema (para lo que con la mayoría de las herramientas anteriores nos bastaría), sino además analizar la efectividad, es decir alcanzar el mayor número de óptimos globales del problema, utilizaremos los *AG*.

Para nuestro estudio hemos seleccionados los ejemplos que mejor se adaptan a nuestros objetivos y que, además, son ampliamente utilizados en la bibliografía tratada. Son los denominados mt06, la01, la02, la03, la04 y la05. Todos ellos son de tamaño pequeño. El mt06, la01, la02 y la05 son sencillos lo que nos permitirá alcanzar el óptimo del problema, en la mayoría de los ejemplos fácilmente, y analizar de esta forma el impacto de la variación del parámetro analizado en la eficacia, efectividad y diversidad. Por otro lado, los la03 y la04 son más complejos y nos servirán para validad los resultados sobre eficacia.

La definición de dichos ejemplos, así como un software para crearlos, está disponible en[3]:**.** En la tabla 1 se muestran los valores de los óptimos para cada ejemplo tratado:

TABLA I Óptimos conocidos [4].

Ejemplo	mt06	la01	la02	la03	la04	la05
Óptimo	55	666	655	590	570	593

No se ha encontrado información sobre el número de óptimos de cada problema, salvo del mt06 realizado por los autores en trabajos previos (Pérez, et al., 2003).

III. ALGORITMOS GENÉTICOS MULTIMODALES.

La multimodalidad empezó a ser estudiada con algoritmos genéticos ya a finales de los 80, debido a la innata y directa disposición a este tipo de problemas. Al trabajar con un grupo de soluciones al mismo tiempo, parece lógico que si consiguiéramos mantener la diversidad de la población se podría obtener diferentes óptimos del problema. Siguiendo este planteamiento muchas han sido las aproximaciones a la multimodalidad siendo la primera de ellas los sistemas *sharing*.

El método clásico (*sharing fitness*) (Goldberg y Richardson, 1987) está basado en la penalización de las áreas del espacio de búsqueda con más soluciones en la población. De esta forma, las soluciones pertenecientes a áreas densamente pobladas verán reducida su calidad con el objetivo de permitir la exploración de otras áreas menos pobladas. Para ello se define una nueva calidad que será utilizada durante el proceso de selección, *the shared fitness*, definida según la siguiente ecuación, donde la calidad original del individuo *i* es *fi*, la calidad modificada (*shared fitness)* es *fi** y la función de modificación (*sharing function*) es *Sh(d(i,j))*:

[3] http://people.brunel.ac.uk/~mastjjb/jeb/orlib/jobshopinfo.html
[4] http://koala.ilog.fr/wiki/bin/view/Scheduler/STRandCumulJSSP

$$f_i^* = \frac{f_i}{Sh(i)}$$
$$Sh(i) = \sum_{j=1}^{N} Sh(i, d(i,j))$$
$$con\ i \in \{1, N\} \quad y \quad N = tamaño\ de\ la\ población$$

Donde la función de modificación depende de la distancia entre los individuos *i* y *j* (*d(i,j)*) según la ecuación:

$$Sh(i, d(i,j)) = \begin{cases} 1 - \left(\dfrac{d(i,j)}{\sigma_{share}}\right)^{\alpha} & If \quad d(i,j) < \sigma_{share} \\ 0 & otros \end{cases}$$

Con σ_{share} el radio que define la pertenencia de dos soluciones al mismo nicho.

El objetivo de este tipo de métodos es dividir el espacio de búsqueda en subregiones o nichos. Para ello, en la población, y para cada solución, se busca las soluciones vecinas (definidas como las que están a una distancia menor de un radio predeterminado (σ_{share}), y que determinan el espacio geográfico que comprende su nicho). Si hay muchas soluciones en su nicho, el valor de la función de modificación será elevado y, por ello su calidad será altamente penalizada, frente a otras soluciones en cuyo nicho haya pocas o ninguna solución, por lo que su calidad apenas se verá afectada y el nicho al que pertenece tendrá más posibilidades de ser analizado en el proceso de búsqueda.

Debido a los problemas en mantener la diversidad necesaria se modificó el proceso dando lugar al método denominado *continuously updated sharing* (Oei et al., 1991). La diferencia con el método original es que en lugar de penalizar las áreas presentes en la población de partida de cada generación, se van penalizando aquellas que van siendo elegidas durante la selección. Así, la calidad de una zona en la que ya haya varios padres seleccionados se penaliza limitando la posibilidad de que, nuevamente, sus individuos sean elegidos para la población de padres. Además, se modificó el método de selección dando lugar al torneo binario restringido (*restricted binary tournament*). Este método utiliza un parámetro denominado tamaño de nicho máximo (n^*). Si las dos soluciones que compiten en el torneo pertenecen a nichos con un tamaño de nicho menor que n^*, ganará aquella solución con mayor calidad. En caso contrario, ganará la que tenga mayor riesgo de desaparecer. Puntualizar que se habla siempre de la calidad modificada, y que el tamaño de nicho se refiere al número de soluciones que hay en la población de padres con una distancia menor que el radio definido (σ_{share}) hacia las soluciones involucradas en el torneo.

Basado también en el concepto de nicho, el método *clearing* utiliza el concepto de recursos limitados para eliminar la competencia entre soluciones que son muy diferentes (Pétrowski 1996 y 1997). En la naturaleza los individuos de una misma especie son los que luchan entre si para lograr los recursos limitados existentes en la naturaleza, sin embargo esta lucha no es frecuente entre individuos de especies con muy diferentes características. Así es posible que convivan animales tan dispares como el león y la hormiga (no tienen que luchar por los mismos recursos y por ello sobreviven).

La población inicial (*P(t)* de tamaño *N*) se modifica para formar otra población (*P'(t)* de tamaño *N'*) desde la que se hace la selección. Las soluciones de la población de partida (*P(t)*) se ordenan de mayor a menor calidad, como inicio al proceso se selecciona la primera solución de la ordenación que pasa directamente a la población (*P'(t)*). A esta primera solución se la llama dominadora puesto que no hay mejor solución que ella en la población. Después, se compara cada solución de la población inicial (*P(t)*) con la seleccionada, obteniéndose el conjunto de soluciones que pertenecen al mismo nicho. De este conjunto reservamos las *k* mejores soluciones (las de mayor calidad) mientras que a las restantes se les asigna una calidad nula. De esta forma, limitamos la competencia dentro del nicho. En este punto tenemos una solución en la población (*P'(t)*), mientras que la población de partida (*P(t)*) mantiene las *N* iniciales pero algunas de ellas con calidad cero. El proceso continua volviendo a seleccionar la siguiente solución de la ordenación cuya calidad sea distinta de cero. El proceso termina cuando todas las soluciones que quedan por seleccionar en *P(t)* tienen calidad cero.

Por último, el método implementa la siguiente forma de elitismo. De la población resultante *P'(t)* se selecciona aquellas soluciones cuya calidad sea superior a la media de la calidad original de la población (*P(t)*). El proceso se muestra en la figura 1.

Para quienes deseen ampliar conocimientos sobre otros métodos a continuación referenciaremos las principales aproximaciones al problema multimodal:

- Adaptive niche hierarchical genetic algorithm (Dunwei et al., 2002).
- Niche identification techniques with fitness sharing (Lin y Wu, 2002).

- Deterministic crowding (Mahfoud, 1992).
- Multi-niche crowding genetic algorithm (Cedeño y Vemuri, 1999).
- Restricted competition selection method (Lee et al., 1999).
- Restricted competition selection with pattern search method (Kim et al., 2002).
- Species conserving genetic algorithm. (Li et al., 2002).
- Quick hierarchical fair competition. (Hu y Goodman, 2004).

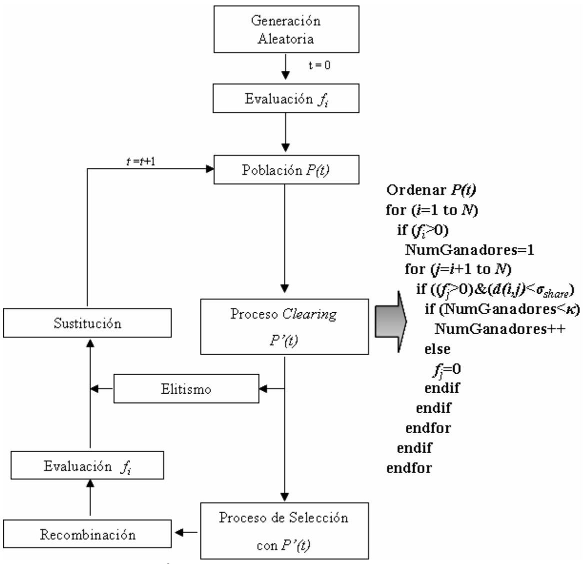

Fig. 1. Algoritmo clearing.

En (Sareni y Krahenbuhl, 1998 y Pérez, et al., 2003) se hace una comparación de algunos métodos (*sharing fitness*, *continously updated sharing* y *deterministic crowding*) y el *clearing* es el que obtuvo un mejor resultado sobre los ejemplos en codificación real y en job shop, respectivamente.

IV. Estudio Experimental

Como ya se ha comentado anteriormente, el estudio se ha realizado sobre seis ejemplos de pequeño tamaño del problema job shop. Sobre cada ejemplo se han realizado las ejecuciones del método clearing variando los valores de los parámetros, el número de ganadores k permitido, el radio que define el nicho y el método de selección. Para el primer parámetro se han considerado los valores 1, 5 y 10, para el segundo hemos analizado los radios 0, 5, 10 20 y 30, y se han ejecutados los AG con dos tipos diferentes de selección: el método *SUS* (*stochastics universal selection*) y *RWS* (*roulette wheel selection*).

Los demás parámetros del AG son constantes para todos los ejemplos:

- Tamaño de población: 50 individuos
- Número de generaciones: 600
- Operador Cruce: OX (*Order crossover*: (Davis, 1989)
- Probabilidad de cruce: 0.8
- Operador Mutación: OBM (*Order based mutation*: Mattfled, 1995)
- Probabilidad de mutación: 0.2

El valor del tamaño de la población y el número de generaciones no presenta ningún problema y es una solución de compromiso entre rapidez y potencia de búsqueda. En problemas de mayor tamaño o complejidad se aumentará preferiblemente en número de generaciones antes que el tamaño de la población. Por otro lado, la codificación implementada ha sido la permutacional con repetición (Mattfeld, 1995) y los operadores utilizados así como sus probabilidades de actuación han surgido de estudios previos (Gento y Pérez, 2002).

Por otro lado, es necesario definir cómo se va a calcular la distancia entre dos soluciones *d(i,j)*. Para el problema de secuenciación puede ser el número de posiciones en las que hay diferentes operaciones. Así, por ejemplo, para un problema 3x3, como el mostrado en la tabla II, la distancia será 5, ya que las posiciones 1, 2, 3, 4 y 6 están ocupadas por diferentes trabajos. Este tipo de distancia se ha comprobado que funciona mejor que las que se definen en la solución codificada (Pérez, 2000).

TABLA II Ejemplo de distancia en secuenciación

Soluciones	Distancia
(1 3 2)(2 3 1)(3 2 1)	5
(3 2 1)(1 3 2)(3 2 1)	

Para terminar la definición de la experimentación, recalcar que todas las ejecuciones se han repetido 30 veces para poder analizar los resultados estadísticamente.

El objetivo del presente trabajo es analizar la influencia de los parámetros sobre, no solo la eficacia del método clearing, objetivo prioritario en cualquier problema de optimización, sino además, la efectividad en su consecución, es decir, analizar si existe un conjunto de parámetros más adecuado para conseguir el mayor número de óptimos globales del problema en una única ejecución del algoritmo. Para todo ello analizaremos los siguientes valores:

- V_m. Valor medio obtenido. Calculado como la media de la mejor solución alcanzada en cada una de las 30 repeticiones realizadas. Medida de eficacia.

- N_e. Número de ejecuciones donde se alcanza el valor del óptimo conocido de las 30 realizadas. Medida de eficacia.
- N_{mo}. Número medio de óptimos alcanzados. Calculado como el valor medio de óptimos encontrados por ejecución exitosa (donde la mejor solución corresponde al óptimo conocido). Medida de efectividad.
- D_m. Distancia media entre óptimos.

En las tablas III y IV se resumen los resultados obtenidos en cuanto a la eficacia del método. Como se puede observar para los problemas mt06, la01, la02 y la05 es bastante fácil alcanzar los óptimos globales del problema y lo hacen, en el caso de mt06, la01 y la05, casi en el 100% de las ejecuciones. Los problemas la02, y sobre todo los la03 y la04, son más complejos hasta el punto de no llegar hasta el óptimo en los dos últimos casos.

TABLA III Valor medio obtenido. V_m

Ejemplo	(1)	(2)	(3)	(4)	(5)	(6)
mt06	55	55.00	55	0	1	*RWS*
la01	666	666.53	666	10	1	
la02	655	671.87	655	20	3	
la03	590	634.03	613	0	1	
la04	570	616.97	595	0	3	
la05	593	593.00	593	*	*	

TABLA IV Número de ejecuciones con óptimos globales

Ejemplo	(7)	(4)	(5)	(6)
mt06	30	0	1	*RWS*
la01	28	10	1	
la02	6	0	1	
la03	---			
la04	---			
la05	30	10	1	

(1) Óptimo conocido
(2) V_m
(3) Mejor solución encontrada de las 30 alcanzadas.
(4), (5) y (6) Valor de los parámetros radio, k y método de selección, respectivamente, con el que se alcanza el mejor valor: V_m (tabla III), N_e (tabla IV) y N_{mo} (tabla V).
(7) N_e

Debido a la extensión de los análisis estadísticos sólo expondremos los resultados obtenidos. Según estos análisis, el método de selección significativamente mejor es el *RWS*. El valor del radio es también significativo para la eficacia del método observándose que cualquier valor salvo el máximo (radio=30) ofrece buenos resultados. Por último, el valor del parámetro k no es significativo, es decir cualquiera de los valores estudiados son igualmente válidos, y por lo tanto, su adaptación al problema no es tan importante como el del resto de parámetros.

En cuanto a la efectividad (N_{mo}), los resultados son espectaculares. Tanto los valores medios como los máximos (columnas 8 y 9 de la tabla V) se obtienen para el valor del radio igual a cero, del parámetro k=1 y la selección *RWS*. Siendo estadísticamente significativo estos tres valores juntos. Es decir, para un radio igual a 0 y cualquier otro valor de k, ya no se obtienen estos resultados.

Recalcar que la población está formada por 50 soluciones, y con estos valores de los parámetros el método ha obtenido, en el peor de los casos, 43 óptimos diferentes. Para que se pueda ver con mayor claridad la diferencia que supone utilizar estos tres valores de forma conjunta, en la tabla VI resumimos el número máximo de óptimos por ejecución para el ejemplo mt06.

TABLA V Número medio y máximo de óptimos globales localizados por ejecución exitosa

Ejemplo	(8)	(9)	(4)	(5)	(6)
mt06	28.67	43	0	1	*RWS*
la01	37.42	46	0	1	
la02	24.32	44	0	1	
la03	---				
la04	---				
la05	40.80	49	0	1	

(8) N_{mo}
(9) Número máximo de óptimos globales distintos localizados en una única ejecución.

TABLA VI Número máximo de óptimos, mt06

	Radio				
	0	**5**	**10**	**20**	**30**
***k*=1**	43	4	3	1	0
***k*=3**	20	7	3	2	0
***k*=5**	15	8	4	2	1
***k*=10**	11	12	7	4	2

Además, mediante estudios previos sobre el problema mt06, los autores habían logrado descubrir 41 diferentes óptimos, obteniéndose ahora seis nuevos óptimos.

En dichos estudios previos, se había detectado que el espacio de soluciones estaba formado por dos grandes macizos (A y B), formados por 19 y 18 óptimos globales respectivamente. Los óptimos muy próximos entre sí (distancia media dentro de los grupos de 5, y con una distancia media entre sí de 12). Además, se tenía catalogadas cuatro óptimos globales de un pico aislado de los anteriores (C) con distancias medias de 19 y 13 respectivamente, y entre si de 2.7 (tabla VII). Los nuevos óptimos son cuatro del primer grupo y dos del segundo.

TABLA VII Distancias medias entre áreas (mt06)

	A	*B*	*C*
A	5		
B	12	5	
C	19	13	2.7

En la tabla VIII se muestran los valores de las distancias medidas obtenidas en la experimentación para el ejemplo mt06.

Como se observa, al aumentar el valor del radio, aumentamos la distancia media entre los óptimos alcanzados en las ejecuciones exitosas. Por lo tanto, se puede afirmar que cuanto mayor es el valor del radio mayor es la diversidad de la búsqueda, ya que se localizan óptimos distantes entre sí. Sin embargo, con el radio igual a cero, se produce una población de óptimos próximos entre sí.

TABLA VIII Distancias medias: Ejemplo mt06

	Radio		
	0	**5**	**10**
***k*=1**	7.51	11.78	13.78
***k*=3**	6.45	10.54	9.53
***k*=5**	4.96	9.49	9.12
***k*=10**	4.19	7.59	8.83

Aunque, en todas las ejecuciones realizadas se obtuvieron óptimos de los dos macizos. Con el radio igual a cero no se obtuvo en ninguna ejecución óptimos del pico aislado, sin embargo con radios mayores en bastantes ejecuciones se alcanzaban picos de las tres zonas.

Esto quiere decir, que con cualquier parametrización se pueden obtener óptimos de las dos principales áreas del espacio. Pero mientras que con el radio menor, las soluciones se agrupan, con radios mayores los óptimos se dispersan entre sí.

Si el espacio de búsqueda del problema del job shop se pudiera dibujar, sería algo parecido a lo mostrado en las figuras 2 y 3.

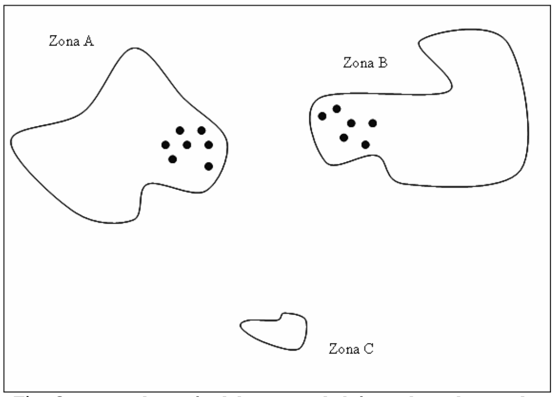

Fig. 2. Explotación del espacio de búsqueda, radio igual a cero

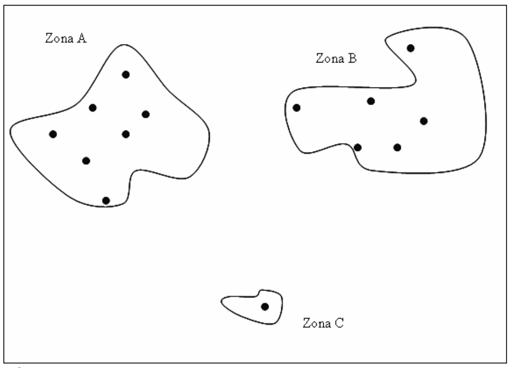

Fig. 3. Diversificación del espacio de búsqueda, radios mayores.

En las siguientes tablas se muestran los valores de las distancias medias para el resto de ejemplos analizados (tablas IX, X y XI). Con ellas, se corrobora los resultados y conclusiones obtenidas para el ejemplo mt06.

Indicar que las celdas sin valor indican que en esa experimentación (30 ejecuciones) no se obtuvo ninguna ejecución exitosa o con un único óptimo (por lo que la distancia no pudo ser calculada).

TABLA IX Distancias medias: Ejemplo la01

	Radio			
	0	**5**	**10**	**20**
***k*=1**	14.01	19.96	24.84	29.34
***k*=3**	9.79	14.84	20.64	20.06
***k*=5**	7.99	14.42	16.84	20.59
***k*=10**	8.86	10.46	15.04	12.27

TABLA X Distancias medias: Ejemplo la02

	Radio			
	0	**5**	**10**	**20**
***k*=1**	9.62	10.85		
***k*=3**	10.05	11.85	21.67	
***k*=5**	5.47	8.29	11.41	17.50
***k*=10**	6.90	9.55	15.67	

TABLA XI Distancias medias: Ejemplo la05

	Radio				
	0	**5**	**10**	**20**	**30**
***k*=1**	24.20	27.72	30.52	33.83	36.67
***k*=3**	19.67	23.89	26.92	31.77	32.83
***k*=5**	18.41	20.86	25.23	30.04	30.87
***k*=10**	16.49	17.92	21.42	27.24	27.52

V. CONCLUSIONES

Los algoritmos genéticos que utilizan el método *clearing* permiten alcanzar múltiples soluciones del problema en una única ejecución. Esto tiene importantes repercusiones en la industria, puesto que en ocasiones no se puede incluir en la definición del problema todas las características del modelo, por ejemplo, cuando son cualitativas o no se conocen a priori, como averías, o falta de materia prima. Con múltiples soluciones, el planificador podrá elegir de entre ellas la que mejor se adapte a las circunstancias reales.

Por otro lado, se ha comprobado que la parametrización debe realizarse en función del objetivo que se busque, y no sólo del problema tratado. Así, si se busca:

- Eficacia: Selección *RWS*, k=1 y σ_{share} = 0, 5, 10 o 20.
- Efectividad: Selección *RWS*, k=1 y σ_{share} = 0.
- Diversidad: σ_{share} altos, a costa de perder efectividad y eficacia.
- Explotación: σ_{share} =0, con alta eficacia y efectividad.

Recalcar que con los valores *RWS*, k=1 y σ_{share} = 0, el 86% como mínimo y el 98% como máximo, de los individuos de la población final eran óptimos globales diferentes del problema. Esto es importante, porque además de ofrecernos una gran cantidad de óptimos para seleccionar el más adecuado, nos asegura **diversidad** en la población, lo que traducido en problemas de mayor complejidad puede indicar un aumento en el rendimiento del algoritmo.

Como líneas futuras, tenemos previsto la aplicación de estos conocimientos a otros tipos de problemas que nada tienen que ver con la secuenciación, para poder comprobar así si existe una parametrización independiente del problema aunque, como ya hemos visto si dependiente del objetivo. Por otro lado, también queremos aplicar esta parametrización en problemas job shop de mayor tamaño y complejidad, para analizar si se validan estas conclusiones.

AGRADECIMIENTOS

Este trabajo ha sido subvencionado por el Ministerio de Educación y Ciencia, en el marco de proyecto TIN2005-08386-C05-01.

REFERENCIAS

[1] Adams, J., Balas, E. y Zawack, D. The shifting bottleneck procedure for job shop scheduling. Management Science, 34. Pp. 391-401. 1988.

[2] Brucker, P. Scheduling algorithms, 2nd Edn. Springer-Verlag. Berlin. Germany. 1997.

[3] Carlier, J. y Pinson, E. An algorithm for solving the job shop problem. Management Science, 35. Pp. 164-176. 1989.

[4] Cedeño, W. y Vemuri, V. R. Analysis of speciation and nichng in the multi-niche crowding GA. Theoretical Computers Science. (229). Elsevier. Pp. 177-197. 1999.

[5] Conway, R. W., Maxwell, W. L. y Miller, L. W. Theory of schelung. Addison-Wesley, Massachusetts. USA. 1967.

[6] Davis, L. Adapting operators probabilities in genetic algorithms. In Proc. of the 3th international conference on genetic algorithms. Schaffer, J. D. (Ed.) Kaufmann. San Mateo. Pp. 375-378. 1989.

[7] Dunwei, G. Fengping, P. y Shifan, X. Adaptive niche hierarchy genetic algorithm. In Proc. of IEEE TENCON. Pp. 39-42. 2002.

[8] French, S. Sequencing and scheduling: An introduction to the mathematics of the job shop. Ellis Horwood. Chichester. USA. 1982.

[9] Garey, M. y Johnson, D. Computers and Intractability: A guide to the Theory of NP-Completeness. Freeman, NY. USA. 1979.

[10] Gento, A.M. y Pérez, M.E. Study on the genetic operators for the job shop problem. In Proc. of the First Spanish Conference on Evolutionary and Bioinspired Algorithms. Mérida. Spain. Pp. 523-530. 2002.

[11] Glover, F. y Laguna, M. Tabu search. Kluwer Academic. Boston. USA. 1997.

[12] Goldberg, D. E. y Richardson, J. Genetic algorithms with sharing for multimodal function optimization. In Proc. of the 2nd international conference on genetic algorithms. Pp. 41-49. 1987.

[13] Greenberg, H. A branch and bound solutions to the general scheduling problem. Operations Research, 16. Pp. 353-361. 1968.

[14] Hu, J. J. y Goodman, E. D. Robust and efficient genetic algorithms with hierarchical niching and a sustainable evolutionary computation model. In GECCO. Deb, K et al., (Eds). Pp. 1220-1232. 2004.

[15] Kim, J., Cho, D., Jung, H. y Lee, C. Niching genetic algorithm adopting restricted competition selection combined with pattern search method. IEEE Transactions on magnetic. (38). Pp. 1001-1004. 2002.

[16] Kirkpatrick, S., Gelatt, C. D. Jr. y Vecchi, M. P. Optimization by simulated annealing. Science. 220. Pp. 671-680. 1983.

[17] Lee, Ch, Cho, D y Jung, H. Niching genetic algorithm with restricted competition selection for multimodal function optimization. IEEE transactions on magnetics. (35). Pp. 1722-1725. 1999.

[18] Li, J., Balazs, M., Parks, G. T. y Clarkson, P. J. A species conserving genetic algorithm for multimodal function optimization . Evolutionary computation. 10(3). Pp. 207-234. 2002.

[19] Lin, C. y Wu, W. Niche identification techniques in multimodal genetic search with sharing scheme. Advances in Engineering Software (33). Elsevier. Pp. 779–791. 2002.

[20] Mahfoud, S. W. Crowding and preservation revisited. Parallel problem solving form nature II. Männer, R. and Manderick, B. (Eds.). Elsevier. Pp. 27-36. 1992.

[21] Mattfeld, D.C. Evolutionary search and the job shop. Investigations on genetic algorithms for production scheduling. Springer. Berlin. 1995.

[22] Nowicki, E. y Smutnicki, C. A fast tabu search algorithm for the job shop problem. Management Science. 42. Pp. 797-813. 1996.

[23] Oei, C. K., Godberg, D. E. y Chang, s. J. Tournament selection, niching and the preservation of diversity. IlliGAL Report No. 91011. University of Illinois. USA. 1991.

[24] Panwalkar, S.S. y Iskander, W. A survey of scheduling rules. Operations Research, 25, Pp. 45-61. 1977.

[25] Pétrowski, A. A clearing procedure as a niching method for genetic algorithms. In Proc. IEEE International conference on evolutionary computation. Japan. Pp. 798-803. 1996.

[26] Pétrowski, A. A new selection operator dedicated to speciatin. In Proc. of the 7th international conference on genetic algorithms. Bäck, T. (Ed.). Kaumann. San Mateo. USA. Pp. 144-151. 1997.

[27] Pérez, E. Análisis de algoritmos genéticos como método de resolución para problemas de programación de operaciones:Caso job shop. Tesis Doctoral. Univ. de Valladolid. 2000.

[28] Pérez, E., Herrera, F. y Hernández, C. Finding multiple solutions in job shop scheduling by niching genetic algorithms. Journal of Intelligent Manufacturing, (14) Pp. 323-341. 2003.

[29] Ramalhinho, H., Marti, O. y Stützle, T. Iterated local search. In Handbook of metaheuristics. Glover, F. and Kochenberger, G. A. (Eds). Kluwer. Pp. 321-354. 2000.
[30] Sareni, B. y Krahenbuhl, L. Fitness sharing and niching methods revised. IEEE Transactions on Evolutionary Computation. 2. Pp. 97-106. 1998.
[31] Van Laarhoven, P. J. M., Aarts, E. H. L. y Lenstra, J. K. Job shop scheduling by simulated annealing. Operations Research. 40. Pp. 112-129. 1992.
[32] Wang, L. y Zheng, D. Z. An effective hybrid optimization strategy for job shop scheduling problems. Computers & Operationas Research. 28. Pp. 585-596. 2001.
[33] Yang, S. y Wang, D. Constraint satisfaction adaptive neural network and heuristics combined approach for generalized job shop scheduling. IEEE Trans. on Neural Networks. 11. Pp. 474-486. 2000.