

OPERADORES GENETICOS SOBRE PERMUTACIONES APLICADOS A LA RESOLUCIÓN DEL SUDOKU

Carlos Cervigón Rückauer
Departamento de Ingeniería del Software e Inteligencia artificial
Universidad Complutense de Madrid

Resumen

Los Algoritmos Evolutivos están inspirados en la naturaleza y se basan en un conjunto de modelos basados en la evolución de los seres vivos. Se caracterizan por imitar procesos adaptativos de los sistemas naturales y se basan en la supervivencia del mejor individuo, siendo un individuo una representación de una solución potencial del problema que se implementa como una estructura de datos. La población de individuos se somete a un proceso de selección, que favorece a los mejores según su aptitud, y después a ciertas transformaciones. Cada ciclo de transformación y selección constituye una generación. Se espera que después de cierto número de generaciones el mejor individuo de la población esté cerca de la solución buscada. En este artículo se analizan y comparan diferentes transformaciones (operadores de cruce) sobre los individuos representados por permutaciones, que representan posibles soluciones al juego Sudoku.

Palabras clave: evolución, genético, Sudoku, aptitud, población, mutación, cruce.

1. Introducción

El Sudoku es un pasatiempo que se popularizó en Japón en 1986 y se dio a conocer en el ámbito internacional en 2005. Consiste en un tablero con algunas casillas ocupadas por números y casillas libres. Se trata de rellenar todas las casillas libres respetando una serie de restricciones.

La resolución del Sudoku se ha llevado a cabo utilizando diferentes estrategias, entre las que podríamos destacar los clásicos algoritmos de vuelta atrás y el de ramificación y poda. Ahora tendremos que comprobar si un algoritmo evolutivo es una buena estrategia para resolver dicho juego.

Para resolver el problema de sudoku con un algoritmo genético, como en la mayoría de los AGs, debemos definir tres elementos fundamentales: La representación de los individuos, la función de aptitud y los operadores de transformación.

Hasta el momento, hay buenos estudios hechos sobre tres elementos citados anteriormente para la resolución del sudoku con AG (Mantere, T; Koljonen, J, 2006) (Moraglio, A; Togelius, J; S. Lucas).

En este artículo se presenta una nueva forma de abordar el problema utilizando un algoritmo genético elitista. Nuestra propuesta incluye una nueva representación de los individuos, la inclusión de varios operadores genéticos de cruce y mutación basados en problemas de permutaciones y una nueva función de aptitud, más simple que las propuestas por otros autores, pero no menos eficiente como demuestran los resultados.

Los resultados experimentales obtenidos nos permiten sacar buenas conclusiones con respecto al tratamiento del problema y marcan una línea prometedora de trabajo en la utilización de estos operadores genéticos descritos.

En el apartado 2 se hace un breve repaso de los aspectos más importantes de los algoritmos genéticos y se hace especial hincapié el algoritmo genético simple con elitismo. A continuación, en el apartado 3, se hace un breve repaso de los aspectos más importantes del juego del sudoku a nivel de representación y reglas. El apartado 4 presenta las técnicas, la representación y los operadores utilizados. En el apartado 5 se muestran los resultados experimentales, incluyendo las gráficas de diferentes ejecuciones con diferentes valores de parámetros. Por último, en el apartado 6 se indican las conclusiones obtenidas y el trabajo a realizar en un futuro.

2. Algoritmos genéticos

El algoritmo genético clásico utiliza una población de individuos, que representan posibles soluciones candidatas a un problema. Esta población se somete a un proceso de selección que normalmente favorece a los mejores según su aptitud y después los somete a una serie de transformaciones. Cada generación consiste en un ciclo de transformación y selección. Se espera que después de número de generaciones el mejor individuo de la población esté cerca de la solución buscada.

Los algoritmos genéticos se basan en que los mejores individuos (posibles soluciones a un problema) tienen una mayor probabilidad de reproducirse y sobrevivir que otros individuos menos adaptados al entorno o peores (Davis, 1991), (Goldberg, 1989), (Michalewicz, 1996).

Los individuos tendrán una representación adecuada al contexto. Una representación muy utilizada es identificar una solución como una cadena de valores enteros y un caso particular es la representación con permutaciones.

Cada individuo (posible solución) tiene asignado un valor de una función de aptitud, que mide la calidad de la solución y que es la herramienta que permite interpretar el concepto de individuos mejor adaptados. Aquellos individuos cuyo valor de la función de aptitud sea mejor tendrán más posibilidades de ser seleccionados para construir la siguiente población y por lo tanto, para pasar sus propiedades a los individuos de la siguiente generación.

Los algoritmos genéticos combinan la búsqueda aleatoria, realizada por las transformaciones de la población, con una búsqueda dirigida, llevada a cabo por el operador de selección. Los principales componentes son:

- Población de individuos, que son una representación de posibles soluciones.
- Procedimiento de selección basado en la aptitud de los individuos para resolver el problema.
- Procedimiento de transformación para construir nuevos individuos a partir de los anteriores.

El algoritmo genético simple (Goldberg, 1989), aplicado al la resolución del Sudoku, incorpora los siguientes aspectos:

- Criterio de codificación: Es específico de cada problema y en el caso del Sudoku utilizaremos representación con permutaciones de valores enteros.

- Criterio de inicialización: La población inicial está formada por secuencias de enteros que representan los valores de cada celda vacía del tablero.
- Funciones de evaluación y aptitud: La función de aptitud coincide con la de evaluación. La función de evaluación viene dada a través de una función objetivo explicada posteriormente.
- Operadores genéticos: operadores de cruce y mutación explicados posteriormente.
- Criterio de selección y de reemplazo.
- Criterio de terminación y parámetros de funcionamiento.

Si el algoritmo utiliza elitismo, se garantiza que los mejores individuos no desaparecen con la evolución. Para ello se muestrea una élite de r miembros de entre los mejores de la población. Durante el reemplazo los individuos de la élite se preservan, descartando si es necesario a alguno de sus hijos (siempre que sea peor). Está comprobado que si el algoritmo utiliza elitismo, es posible alcanzar el óptimo.

A continuación se muestra el pseudocódigo de un algoritmo genético simple con elitismo:

```

inicializar población
evaluar población
mientras no se cumpla la condición de parada do
    separa Elite
    selección
    cruce
    mutación
    incluye elite
    evaluar población
fin mientras

```

3. Sudoku

Sudoku es un rompecabezas matemático que se popularizó en Japón en 1986 y se dio a conocer en el ámbito internacional en 2005. Es un pasatiempo que consiste en un tablero de 9 filas x 9 columnas que componen 81 casillas. El tablero está dividido en subcuadrículas de 3x3 casillas. El objetivo del juego es rellenar todas las casillas libres de dicho tablero con los números del 1 al 9 considerando que inicialmente existen algunos números ya colocados en algunas casillas. En el tablero solución se debe cumplir la restricción de que no se repite ningún número en una misma fila, columna o subcuadrícula. La solución es única.

En la Figura 1 se muestra un ejemplo de posible configuración inicial de tablero junto con su solución.

	6		1		4		5	
		8	3		5	6		
2								1
8			4		7			6
		6				3		
7			9		1			4
5								2
		7	2		6	9		
	4		5		8		7	

9	6	6	1	7	4	2	5	8
1	7	8	3	2	5	6	4	9
2	5	4	6	8	9	7	3	1
8	2	1	4	3	7	5	9	6
4	9	6	8	5	2	3	1	7
7	3	5	9	6	1	8	2	4
5	8	9	7	1	3	4	6	2
3	1	7	2	4	6	9	8	5
6	4	2	5	9	8	1	7	3

Figura 1. Configuración inicial de un tablero y su solución.

4. Algoritmo genético para resolver el Sudoku

El Sudoku puede considerarse un problema de optimización con restricciones, a nivel de fila, columna y subcuadrícula. Partiendo de esta base, podemos presuponer una buena idea utilizar un algoritmo evolutivo para resolverlo, dados los buenos resultados obtenidos en este tipo de problemas.

A continuación se analiza la representación utilizada para las posibles soluciones, la función de aptitud y los operadores genéticos apropiados al problema.

Representación del individuo

El Sudoku es un claro ejemplo de representación basado en otros problemas de optimización combinatoria (Lawler, 1985). Cada individuo (tablero) se codifica en un array de 81 posiciones; este array representa el cromosoma que está formado por 9 genes (filas).

De esta forma, supongamos que tenemos el siguiente tablero:

	6				5			5
		8						
2					1			1
8			4		7			6
		6				3		
7			9		1			4
5								2
		7	2		6	9		
	4		5		8		7	

En nuestra representación, cada individuo o cromosoma se representa por un array compuesto por 9 genes de 9 elementos cada uno. Cada gen se corresponde con una fila del tablero.

Las posiciones fijas del tablero se respetan y las posiciones vacías se representan como 0:

0	6	0	0	0	5	0	0	5	0	0	8	0	0	0	0	0	0	...
---	---	---	---	---	---	---	---	---	---	---	---	---	---	---	---	---	---	-----

Cada valor del individuo o cromosoma tendrá un valor del 1 al 9. El cromosoma se inicializa con valores aleatorios entre 1 y 9, respetando las posiciones fijas y sin generar elementos repetidos dentro del gen (fila).

En (Mantere; Koljonen, 2006) se utiliza una representación muy similar donde el individuo es una secuencia de los valores que se encuentran en las subcuadrículas, respetando la restricción de no tener valores repetidos dentro de la subcuadrícula.

Función de aptitud

La función de aptitud se basa en que los elementos de cada fila y columna deben ser los valores del conjunto $\text{Digitos} = \{1,2,3,4,5,6,7,8,9\}$. La función calcula el número de elementos que faltan en cada submatriz y columna. A nivel de fila no existe penalización pues los elementos son permutaciones.

$$\text{Aptitud} = \text{total de faltas de cada subcuadrícula} + \text{total de faltas de las columnas}$$

Se trata de un problema de minimización y el objetivo a conseguir aptitud 0.

En otra función de aptitud propuesta por (Mantere; Koljonen, 2006), se añaden tres tipos de restricciones basadas en que en el Sudoku resuelto, las sumas de los elementos de las filas y las columnas debe ser 45 y que el producto debe ser $9!$. Además, no pueden faltar valores del 1 al 9.

Operadores genéticos aplicados al problema

Selección

Los métodos de selección utilizados para las pruebas son:

- ☐ Selección por torneo. Se escogen tres individuos aleatoriamente, y se selecciona aquel tiene mayor aptitud de entre los tres.
- ☐ Selección por ranking. Se basa en el ranking según la ordenación de los individuos por aptitud decreciente

Cruce

Los operadores de cruce actúan sobre parejas de individuos y suelen producir un par de individuos que combinan características de los progenitores. El operador crea un nuevo par de individuos combinando partes de los dos cromosomas padre.

Diferentes métodos de cruce han sido estudiados en una gran variedad de problemas (Aguirre, K, 1999). En el caso del Sudoku se han obtenido buenos resultados con el operador de cruce geométrico (Moraglio; Togelius).

El operador de cruce mas utilizado es el cruce uniforme. Si probamos este tipo de cruce en el Sudoku, tenemos que añadir la restricción de que los puntos de corte tienen que respetar las filas para que se mantengan las restricciones de elementos no repetidos dentro de la fila.

Se divide el individuo en tres bloques. El primer bloque y el tercer bloque del hijo1 corresponden al padre y el segundo a la madre. En el hijo 2, el primer bloque y el tercer bloque pertenecen a la madre y el segundo bloque al padre.

Para hacer el estudio se han probado otros tipos de cruces:

- ❑ *Cruce uniforme un punto.* Se escoge un punto de cruce divisorio entre filas y se intercambian los cuadros de los padres hasta ese punto.
- ❑ *Cruce uniforme con dos puntos :* Se escogen dos puntos de cruce divisorios entre filas y se intercambian los cuadros de los padres hasta el primer punto, y a partir del segundo punto.
- ❑ *Cruce alterno.* Intercala los cuadros de cada uno de los padres en cada uno de los hijos.
- ❑ *Cruce por emparejamiento parcial y Cruce por orden.*
- ❑ *Cruce “combinado”*

El método que se propone aquí es el que llamamos *cruce “combinado”*. Consiste en recorrer circularmente los valores no prefijados de los padres, asignando de forma alternada a los cromosomas de los hijos cada nuevo número. A medida que se realiza esta operación se comprueba que el nuevo número no se encuentre en el hijo y que no es un valor prefijado, en cuyo caso se le asigna el nuevo número del padre opuesto. Este cruce siempre mantiene la restricción de no tener elementos repetidos a nivel de fila.

Padres

2	4	1	6	7	5	8	3	9
2	4	6	9	1	5	7	3	8

Hijos:

2	4	1	9	7	5	6	3	8
2	4	6	1	9	5	7	3	8

Mutación

La mutación es el operador básico de alteración. Se aplica a individuos solos, realizando una pequeña modificación en alguno de sus genes o en el conjunto, permitiendo explorar nuevas zonas del espacio de búsqueda

En (Ochoa; Harvey; Buxton, 2000), se realiza un estudio en el que demuestra la importancia de las proporciones de mutación y de selección en la eficiencia del algoritmo.

La probabilidad con la que estos cambios se producen suele ser baja, para evitar que el AG realice una búsqueda aleatoria. Sin embargo en ocasiones puede ser necesario aumentar esta probabilidad para recuperar la diversidad de la población. En algunas ocasiones se puede utilizar un operador de regeneración para evitar problemas de convergencia prematura.

Para las pruebas del Sudoku se han implementado las siguientes estrategias de mutación:

- ❑ *Regeneración:* Se inicializa aleatoriamente ese gen respetando las posiciones fijas y la restricción de elementos no repetidos.
- ❑ *Mutación por intercambio con la siguiente casilla.* Se intercambia la casilla con la siguiente válida dentro del cuadro.
- ❑ *Mutación por intercambio con casilla aleatoria.* Se intercambia la casilla con una escogida al azar.
- ❑ *Mutación por intercambio con casilla diagonal.* Se intercambia la casilla con una casilla diagonal a la primera.
- ❑ *Rotación:* Para cada cromosoma que mutamos, seleccionamos aleatoriamente la fila (gen) que va a mutar y dentro de cada fila a mutar se seleccionan aleatoriamente tres elementos y se rotan hacia la derecha.

5. Resultados experimentales

Los parámetros utilizados para probar el algoritmo son 200 individuos, una probabilidad total de cruce de 0.8 y una probabilidad de mutación de 0.2 y un porcentaje de elitismo del 5 por ciento.

Los mejores resultados se han obtenido con el mecanismo de cruce combinado y una probabilidad de mutación por rotación de 0.2. Valores altos de probabilidad de mutación pueden evitar el estancamiento y la convergencia prematura.

En las pruebas hechas sobre un Sudoku de 9x9 de dificultad media se obtuvieron soluciones en el 50% de los casos; en el 40% de los casos se obtuvo una solución con una o dos casillas mal colocadas.

A continuación se muestran las gráficas de resultados, Figuras 2 a 7, variando los siguientes parámetros: número de generaciones, tamaño de la población inicial y probabilidad de cruce. En estas gráficas se puede observar el rango más apropiado de cada uno de los parámetros.

Por ejemplo, en la Figura 1 podemos ver que la mejor aptitud se obtiene a partir de un valor de 250 generaciones. Entre 100 y 250 generaciones la función de aptitud es siempre 2.

En la Figura 4 podemos observar cómo se mejoran los resultados a partir del valor 0.8 de probabilidad de cruce.

Gráficas de análisis de resolución del Sudoku 6 x 6

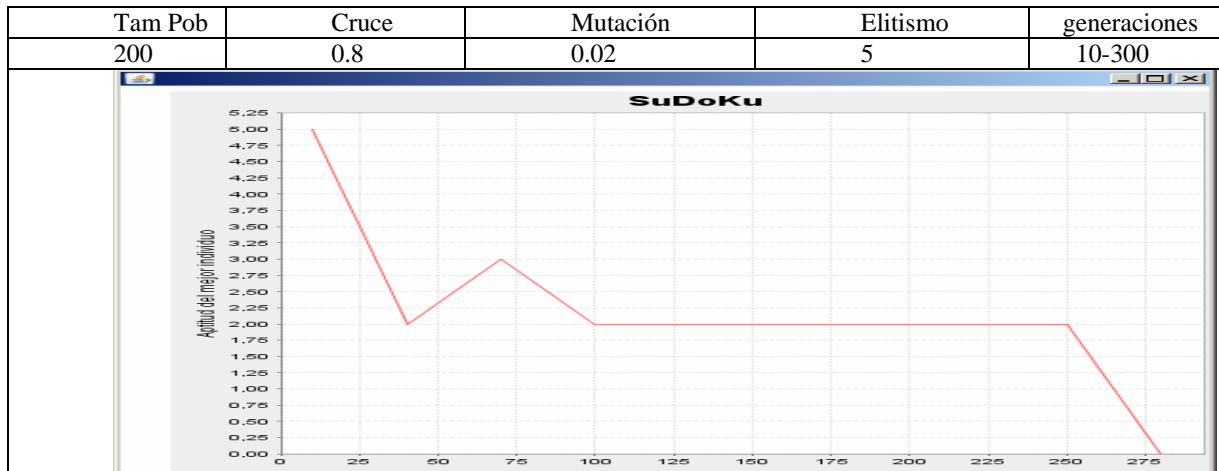


Figura 2. Variación del número de generaciones.

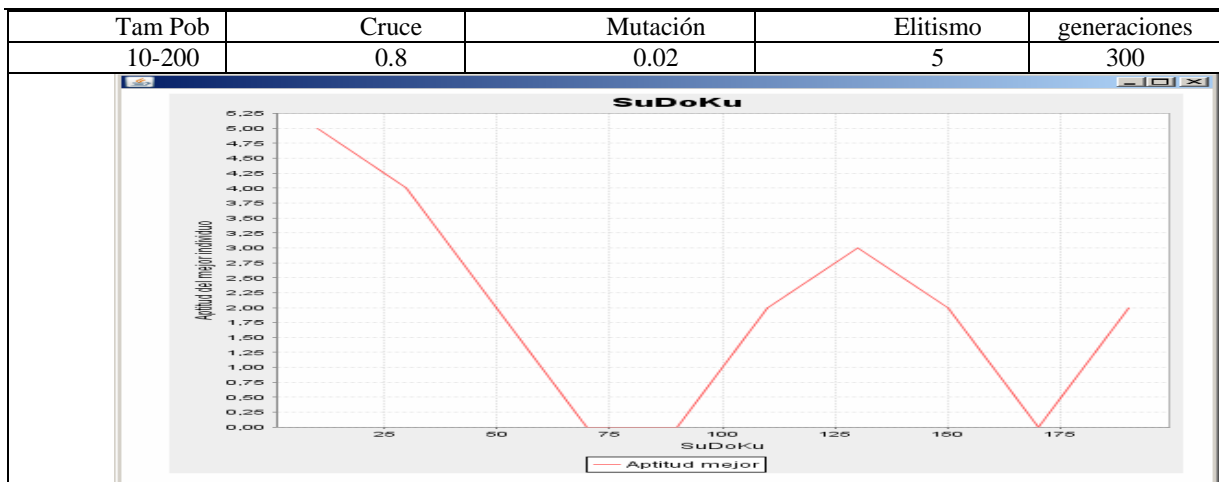


Figura 3. Variación del tamaño de la población.

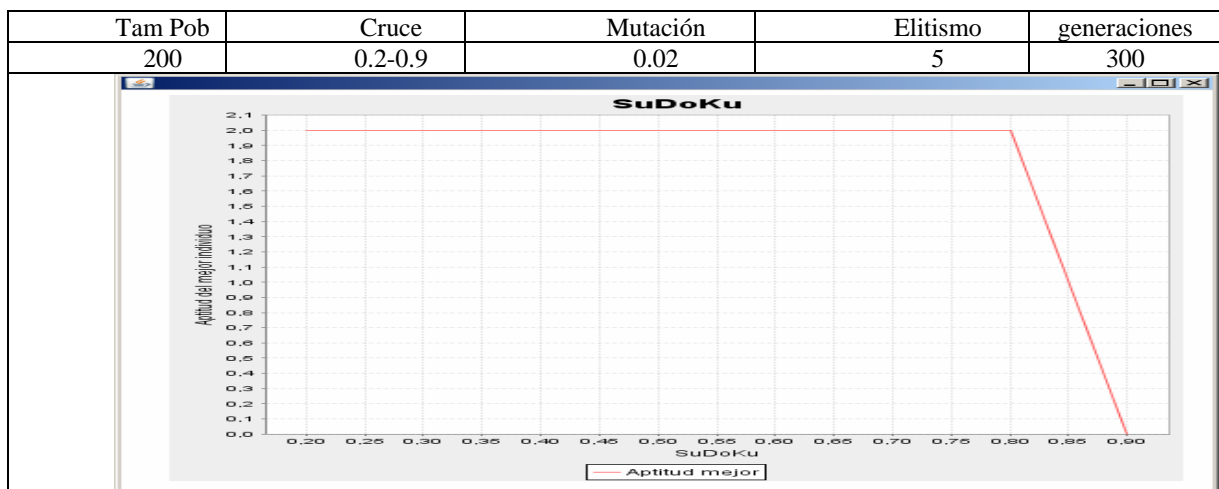


Figura 4. Variación de la probabilidad de cruce.

Gráficas de análisis de resolución del Sudoku 9 x 9

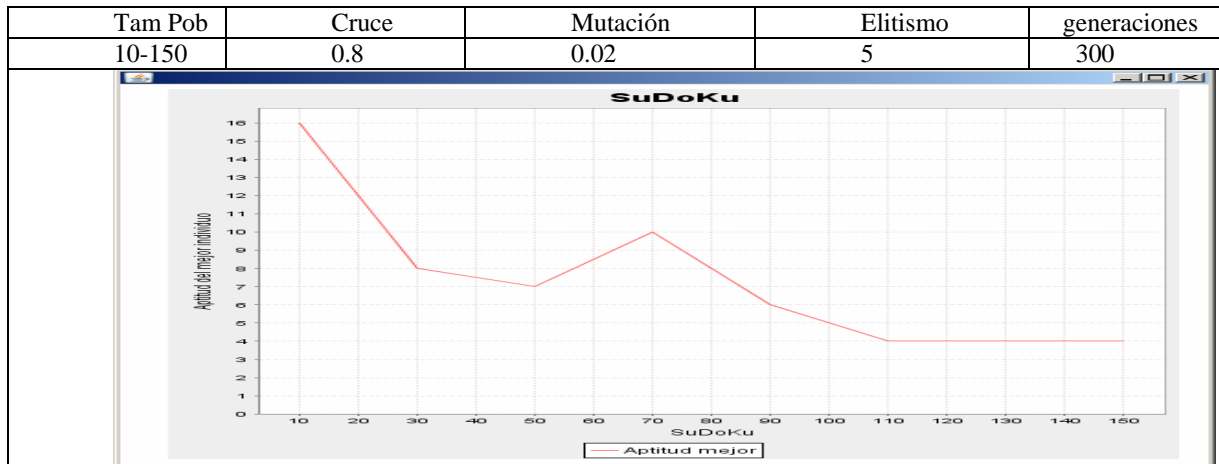


Figura 5. Variación del tamaño de la población.

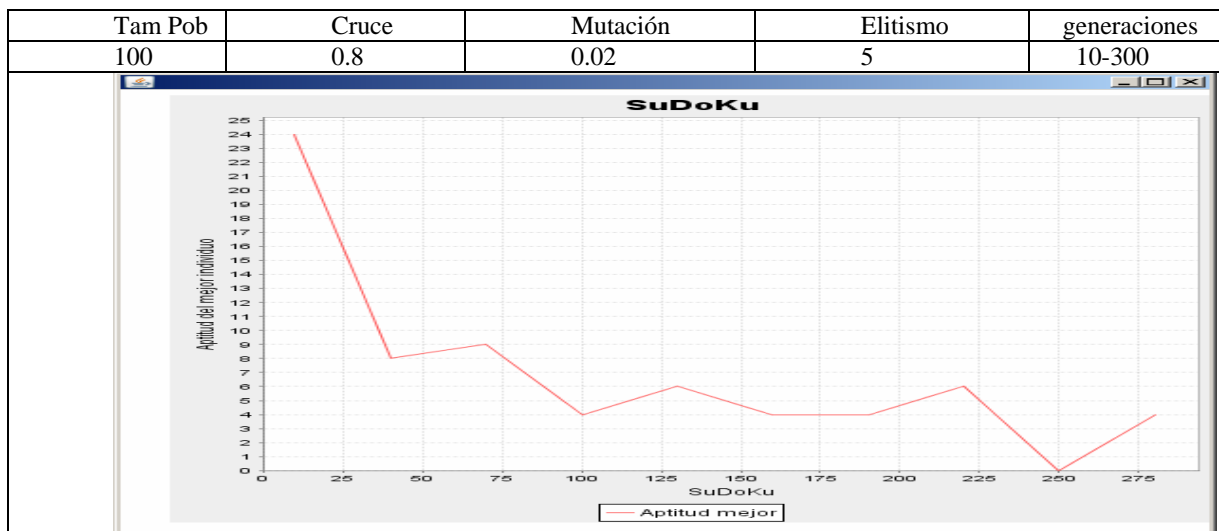


Figura 6. Variación del número de generaciones.

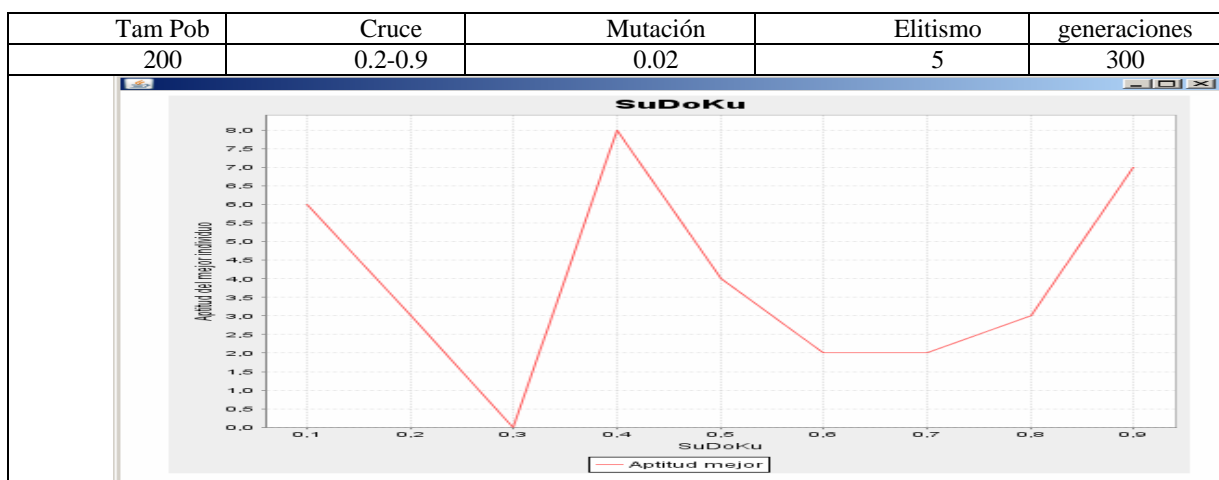


Figura 7. Variación de la probabilidad de cruce.

6. Conclusiones y trabajo futuro

El sudoku tiene solución única, lo que significa que no se consideran soluciones válidas los tableros con situaciones cercanas a la solución. En este sentido, el Sudoku no parece un problema adecuado para resolver utilizando un algoritmo genético; pero si es un problema muy útil para experimentar con diferentes operadores genéticos sobre permutaciones, dado su alto grado de dificultad.

Como consecuencia de esta experimentación se ha concluido con una ligera mejoría en la solución aportada al aplicar el operador de cruce combinado propuesto, frente a operadores de cruce clásicos sobre permutaciones..

Como trabajo futuro se seguirá afinando la función de aptitud de los individuos y se profundizará en nuevos operadores de cruce y mutación. Un buen punto de partida son la ideas tomadas de (Moraglio;Poli, 2004,2005), donde se estudia el cruce geométrico que ha producido buenos resultados en la resolución del Sudoku.

Bibliografía

- Aguirre, H; K. Tonaka, and T. Sugimura. *Cooperative crossover and mutation operators in genetic algorithms*. In Proceedings of the 1999 Genetic and Evolutionary Computation Conference, page 772. Morgan Kaufmann, 1999.
- Davis, L. Handbook of Genetic Algorithm. Van Nostrand Reinhold, 1991
- Goldberg, D.E. Genetic algorithms in search, optimization and machine learning. Addison Wesley, 1989.
- Lawler, E.L., Lentra, J.K., Rinnooy, A.H.G., and Shmoys, D.B. (eds.). The Traveling Salesman problem . A Guided Tour of Combinatorial Optimization. John Wiley & Sons, New York, 1985.
- Mantere, T; Koljonen,J. Solving and Rating Sudoku Puzzles with Genetic Algorithms.
- Michalewicz, Z. Genetic Algorithms + data structures = Evolution Programs. Springer-Verlag, 1996.
- Moraglio, A and Poli, R. Topological interpretation of crossover. Proceedings of GECCO 2004.
- Moraglio, A and Poli,R. Topological crossover for the permutation representation. Proceedings of GECCO 2005.
- Moraglio, A; Togelius, J; S. Lucas: *Product Geometric Crossover for the Sudoku Puzzle*
- Ochoa, G;I. Harvey, and H. Buxton. *Optimal mutation rates and selection pressure in genetic algorithms*. In Proceedings of the 2000 Genetic And Evolutionary Computation Conference, pages 315-322. Morgan Kaufmann, 2000.