Algoritmos Genéticos para la Optimización de Asignación de Espacios Áulicos

Marcelo J. Karanik
José S. Pérez
{marcelo;jsantiago}@frre.utn.edu.ar
Grupo de Investigación Sobre Inteligencia Artificial (GISIA)
U. T. N., Facultad Regional Resistencia – French 414 – Resistencia (3500) Chaco – Argentina

Resumen*. Es sabido que la asignación de recursos finitos a necesidades que cambian dinámicamente es un problema difícil de tratar. Si a esto se le suma la necesidad de optimizar varios objetivos a la vez, las alternativas que puedan ser implementadas son escasas. Este trabajo describe un modelo genético para la optimización de la asignación de espacios áulicos en la Facultad Regional Resistencia de la Universidad Tecnológica Nacional. En la primera parte se describe el problema que se trata de resolver. En la segunda se muestra el modelo utilizado y en la tercera los resultados obtenidos.*

Palabras clave: *algoritmos genéticos, estructura del cromosoma, función de aptitud, optimización multi objetivos.*

1. INTRODUCCIÓN

La asignación de recursos en cualquier organización deja de ser un problema trivial cuando los recursos son limitados y las necesidades van cambiando. Por lo general, la solución de este tipo de problemas trae aparejada una serie de condiciones de eficiencia, tiempo y oportunidad que deben ser tenidas en cuenta, más allá de que la asignación esté correcta. En este trabajo se propone una solución a la asignación dinámica de las aulas de la Facultad Regional Resistencia de la U.T.N. utilizando algoritmos genéticos. Para realizar la asignación, se tienen en cuenta los distintos cursos que se dictan, la cantidad de alumnos por cursos, la capacidad de las aulas y algunas necesidades específicas para el dictado de clases. Este problema puede ser abstraído de la siguiente manera: se supone que existen **n** recursos que son necesarios asignar a [1,.., i,.., n] demandantes del recurso y que las características de los recursos son parecidas en sustancia pero no en características cuantificables. A medida que **n** aumenta, la explosión combinatoria se vuelve inmanejable si se quiere revisar todos los casos. Bajo estas condiciones, se puede pensar en esas combinaciones como un espacio de estados donde cada estado es una configuración posible de asignación. De esta manera, y como revisar un espacio de combinaciones muy grande es dificultoso, conviene dirigir o conducir la búsqueda de la solución que permita resolver el problema de forma eficiente[4].

2. DESCRIPCIÓN DEL PROBLEMA

Dominio de aplicación

La asignación de aulas a distintos cursos de las carreras dictadas en la Facultad Regional Resistencia, de la U.T.N. es un problema que genera inconvenientes tanto a alumnos como a docentes y personal administrativo. Algunas de las características de este problema son:

- *Las aulas tienen distintas capacidades.*
- *Las aulas están acondicionadas de forma distinta.*
- *En ciertos intervalos de tiempo durante la jornada no se utilizan todas las aulas, y en otros que no hay aulas libres.*
- *Los cursos no tienen la misma cantidad de alumnos.*
- *A medida que transcurre el año el número de alumnos por curso tiende a disminuir.*
- *La inasistencia de los docentes genera liberación de aulas durante ciertos módulos.*
- *Eventos especiales pueden requerir el uso de aulas.*
- *Es preferible mínimo movimiento de alumnos en los cambios de hora.*

Los objetivos propuestos son la maximización del uso de los recursos, sin desaprovechar el espacio físico logrando la asignación completa de las aulas mejor acondicionadas, con el mínimo movimiento de alumnos en los cambios de módulo y reasignando las aulas en caso de inasistencias de docentes o alguna circunstancia no prevista.

La cantidad de combinaciones posibles de cursos a asignar está dada por:

$$\sum_{i=1}^{n}\left[\frac{n!}{(n-i)!}\right]^2 \cdot \frac{1}{i!} \qquad (1)$$

n : Cantidad total de cursos i : Cantidad de cursos a asignar.

Como máximo se pueden asignar 19 cursos en forma simultánea, lo que equivale a $1{,}21 \cdot 10^{17}$ combinaciones aproximadamente si se cubrieran todas las aulas. Dicha cantidad aumenta si se producen cambios en la asignación y se desocupan aulas. El número estimado de combinaciones, utilizando (1) y teniendo en cuenta todas las variaciones en la ocupación, pasa a ser $6.99 \cdot 10^{19}$, aproximadamente. Actualmente esta tarea se realiza manualmente, provocando que en ciertos horarios algunas aulas estén colmadas de alumnos y otras con mayor capacidad estén desocupadas o con cantidad de alumnos inferior a dicha capacidad[4].

Algoritmos Genéticos (AG)

Desde el punto de vista de la resolución de problemas, un individuo representa una solución posible a un problema dado[2]. El individuo es evaluado en su capacidad por una función de aptitud, el valor devuelto por dicha función representa cuán buena es la solución.

Los AG son algoritmos probabilísticos caracterizados por el hecho de que un número **p** de posibles soluciones registran el espacio de soluciones en un problema de optimización[1][5][6]. En el comienzo se crean poblaciones iniciales de soluciones que tienen distinto grado de eficiencia que, una vez medidos, sirven para determinar si algún individuo es solución, o si no lo es, qué transformación sufrirá para pasar a la siguiente generación[1][5][6]. Entre estas transformaciones tenemos la **selección**, para elegir los individuos más aptos, el **cruce**, para combinarlos, y la **mutación** para alterar el valor de alguna característica de algún individuo. Todas estas transformaciones tienen como objetivo contribuir a la diversificación de la población[1][3][5][6]. Un algoritmo típico de AG puede escribirse como sigue[6]:

```
procedure GA; {
    t = 0;
    initialize population P(t);
    evaluate P(t);
    until (done) {
            t = t + 1;
            parent_selection P(t);
    recombine P(t)
    mutate P(t);
    evaluate P(t);
    survive P(t);
    }           }
```

Fig. 1. Algoritmo Genético.

3. MODELO DE ASIGNACIÓN

Al trabajar con un modelo genético se deben definir dos aspectos que están relacionados con la implementación: el **cromosoma,** que representa a los individuos de una población, y la **función de aptitud**, que mide qué tan bueno es un individuo como solución del problema.

La mayoría de los desarrollos de aplicaciones que utilizan AG, manejan representación binaria para los individuos. Esto significa que los cromosomas son cadenas de bits cuyos valores representan situaciones particulares en la resolución. Sin embargo, no existe limitación alguna para modelar y trabajar con cromosomas cuyos parámetros sean no binarios. En este caso, para el modelo que nos ocupa, se deben considerar ciertos aspectos de la representación y la aplicación de los operadores no binarios.
La función de aptitud debe representar todos los aspectos relevantes del problema a resolver, ya que el valor que devuelve esta función determina el grado de aptitud del individuo con respecto a la solución óptima y con respecto a los demás individuos de la población.

Definición del Cromosoma

Para establecer el formato del cromosoma se necesitan varias estructuras de datos que contengan la información relevante de los espacios que se asignen (capacidad, acondicionamiento, distancia entre las aulas). Para ello se definen las siguientes estructuras:
capacidad: Vector de **n** elementos que contiene la capacidad de cada aula posible de asignar (variable entera).
capProyecAula: Vector de **n** elementos que indica si el aula tiene las condiciones para utilizar medios audiovisuales o no (variable booleana).
acondicionamiento: Vector de **n** elementos que indica si el aula tiene aire acondicionado o no (variable booleana).
distancia: Matriz de **n** x **n** elementos que contiene la distancia entre aulas (variable entera). Las distancias varían de acuerdo al curso en que se dictó clases antes de la hora actual.
cursoAsignar: Vector de **m** elementos que contiene los datos de los cursos que se dictan en el día. Los datos contenidos en cada elemento son:

- *idCarrera (variable entera)*
- *idCatedra (variable entera)*
- *nombreCurso (variable cadena)*
- *cantAlumnos (variable entera)*
- *horaInicio (variable hora)*
- *horaFinalizacion (variable hora)*
- *necProyecCurso (variable booleana)*

Con estos elementos definidos se establece el formato del cromosoma que se utiliza para manejar la asignación:
cromosoma: Vector de **n** elementos (en principio una posición por cada espacio asignable) que contiene el código del curso asignado al aula (variable entera).
Una vez que se obtienen los valores de todos los vectores se crea cada individuo (cromosoma) de la población 0. La cantidad de individuos de la población es variable y depende de varios factores, como la capacidad de almacenamiento y la velocidad de procesamiento. La generación 0 se crea de forma aleatoria asignando las aulas disponibles a los cursos, según las necesidades.

Función de aptitud

La alternativa de función de aptitud que se propone contempla cuatro aspectos bien diferenciados: *I. La diferencia entre la capacidad del aula y la cantidad de alumnos del curso; II. La capacidad de proyección multimedial; III. La distancia entre las aulas; IV. El acondicionamiento del aula.*
Teniendo en cuenta los aspectos mencionados se define la siguiente función de aptitud:

$$f(\text{cromosoma}_k) = \sum_{i=1}^{n} \begin{bmatrix} g(\text{factorDiferencia}_{ki}) + h(\text{proyectar}_{ki}) + \\ + d(\text{distancia}_{ki}) + w(\text{acondicionamiento}_{ki}) \end{bmatrix} \quad (2)$$

p : Cantidad de individuos de la población; n : Cantidad de espacios asignables.

$1 \leq k \leq p$

La función (2) contempla todos los aspectos de la solución del problema[4], es decir, tiene en cuenta qué tan bien se adaptan las aulas a los requerimientos de los cursos. La función de aptitud es la suma de otras cuatro funciones (g, h, d, w), cada una con un objetivo específico. La primera de esas cuatro funciones tiene como objetivo representar qué tan bien se adapta la capacidad de un aula a un curso dado. La forma de la función g es:

$$g(\text{factorDiferencia}_{ki}) = \alpha \cdot 100^{(1-\text{factorDiferencia}_{ki})^{1,2}} \quad (3)$$

Donde:

$$\text{factorDiferencia}_{ki} = \frac{\text{abs}(\text{capacidad}_i - \text{cursoAsignar}_j.\text{cantAlumnos})}{\text{capacidad}_i} \quad (4)$$

m : Cantidad de cursos a asignar

$1 \leq j \leq m$

α : factor de ponderación o importancia dentro de la función de aptitud.

Si $\text{factorDiferencia}_{ki} > 1$ entonces $\text{factorDiferencia}_{ki}$ se iguala a 1.

Se puede observar en (3) que el $\text{factorDiferencia}_{ki}$ definido en (4), que varía entre 0 y 1, determina el valor que entregará a la función de aptitud (2). En caso de no haber diferencia, es decir que la asignación sea perfecta en cantidad de alumnos, retorna el valor 100 multiplicado por el factor de ponderación α. Nótese que a medida que la proporción en la diferencia aumenta, el valor devuelto por g decrece en forma exponencial, haciéndose prácticamente nulo para diferencias de más del 10%. La segunda función apunta a que aulas con capacidad para utilizar medios audiovisuales, se asignen con mayor preferencia que aquellas sin dicha capacidad. La función se define como sigue:

$$h(\text{proyectar}_{ki}) = \beta \cdot 100 \cdot \begin{bmatrix} \text{coincideRequerimientoTotal} + \\ + 0.8 \cdot \text{coincideRequerimientoParcial} \end{bmatrix} \quad (5)$$

Donde:

$$\text{coincideRequerimientoTotal} = \neg\begin{pmatrix} \text{capProyecAula}_i \ \text{XOR} \\ \text{cursoAsignar}_j.\text{necProyecCurso} \end{pmatrix} \quad (6)$$

$$\text{coincideRequerimientoParcial} = \begin{pmatrix} \text{capProyecAula}_i \ \text{AND} \\ \neg\text{cursoAsignar}_j.\text{necProyecCurso} \end{pmatrix} \quad (7)$$

m : Cantidad de cursos a asignar

$1 \leq j \leq m$

β : factor de ponderación o importancia dentro de la función de aptitud.

En este caso se trata de una función (5) que devuelve un valor después de aplicar dos ecuaciones booleanas (6) y (7), cuyos valores son complementarios. La ecuación (6) devuelve el valor 1 cuando la capacidad de proyectar del aula k coincide con la necesidad de proyección del curso j. Además de eso se considera que se asigne un aula con capacidad de proyectar a un curso que no necesite de dicha capacidad, eso es lo que se obtiene con la ecuación (7). El factor 0.8 que multiplica en (5) a la ecuación (7) surge debido a que la situación de la asignación de un aula con capacidad de proyección a un curso que no lo necesite es aceptable pero no deseable. En caso de que los requerimientos de proyección coincidan, la función (5) devuelve el valor 100 multiplicado por el factor de ponderación β.
La tercera función que determina la aptitud utiliza la misma forma de calcular el factor de la distancia que el que se utiliza para el factor de capacidad:

$$d(\text{distancia}_{ki}) = \delta \cdot 100^{\left(1-\frac{\text{distanciaAnterior}_{ki}}{100}\right)} \quad (8)$$

Donde:

$$\text{distanciaAnterior}_{ki} = \text{distancia}_{rs} \quad (9)$$

m : Cantidad de cursos a asignar

$1 \leq r \leq n \qquad 1 \leq s \leq n$

δ : factor de ponderación o importancia dentro de la función de aptitud.

La idea es minimizar el movimiento de alumnos en la facultad, es decir, aprovechar la asignación anterior de un aula si es que el curso que se dicta pertenece al mismo nivel. El factor distancia_{rs} se obtiene de la matriz distancia que especifica qué tan lejos se encuentra el aula **r** del aula **s**. En caso que no haya cambio, y la asignación pueda hacerse sin movilizar a los alumnos, el factor es 0 y la función (8) devuelve el valor 100 multiplicado por el factor de ponderación δ. A medida que la distancia a recorrer es mayor la función (8) decrece exponencialmente.

La cuarta función componente de la función de aptitud es la más simple:

$$w(\text{acondicionamiento}_{ki}) = \gamma \cdot 100 \cdot \text{acondicionamiento}_{ki} \quad (10)$$

γ : factor de ponderación o importancia dentro de la función de aptitud.

La función (10) solamente revisa si el aula a asignar tiene aire acondicionado y retorna el valor 100 en caso afirmativo, ese valor se multiplica por el factor de ponderación γ.

4. MODELOS PARA LA IMPLEMENTACIÓN

Las alternativas propuestas para detectar la ocurrencia de un evento son: *Por Horario de Inicio(PHI)* o *Por Franja Horaria(PFH)*. En el modelo PHI se genera un evento por cada horario distinto de iniciación de curso. Al ocurrir el mismo, se crea una población inicial donde cada uno de los **p** individuos representa a todas las aulas asignadas a los cursos que tengan el mismo horario de inicio del que genera el evento, en forma aleatoria.

En cambio, en el modelo PFH, los eventos se generan por horarios de comienzo y de finalización de cursos. Cuando no haya cursos asignados en un instante dado, el siguiente evento está dado por el próximo horario de inicio. Al ocurrir éste, se origina una población inicial donde cada uno de los individuos representa aulas asignadas a todos los cursos cuyos horarios de inicio sean menores al primer horario de finalización de los cursos con aulas asignadas en la actual franja horaria.

Una vez que las aulas se asignan, el próximo evento se genera cuando finaliza el primero de los cursos.

Longitud del cromosoma

Se consideran dos alternativas: **cromosomas de longitud estática** y **cromosomas de longitud variable**.

Si se utiliza longitud estática, en todas las ejecuciones del algoritmo genético realizadas en cada evento, los **p** individuos de la población presentan la misma cantidad de **n** genes. Para esto se necesita que cada gen almacene la disponibilidad del aula, indicando el estado de la misma (disponible u ocupada). Esta característica es tenida en cuenta al momento de aplicar los operadores genéticos, de manera tal que no se pueda realizar una apropiación de los recursos ya asignados.

Si se utiliza longitud variable, la disponibilidad de las aulas se considera previa a la ejecución del algoritmo genético. La cantidad de genes de los **p** cromosomas que conforman la población varía en cada una de las ejecuciones del algoritmo, dependiendo del número de aulas disponibles.

Operadores

El propósito de cada uno de los tres operadores básicos está bien definido: se eligen los individuos más aptos (selección), se recombinan algunos de ellos para producir nuevos individuos (cruce o cruza), y se alteran características de algunos para garantizar diversidad (mutación)[2].

Operador de selección

Para la selección se considera la aplicación de dos métodos proporcionales: selección Elitista y selección por Ruleta. Los métodos proporcionales eligen individuos teniendo en cuenta el peso de su aptitud respecto del resto de la población. Otros métodos utilizan una ordenación de individuos basada en su aptitud[2], éstos últimos no son usados en este trabajo.

Para que se preserven los mejores **m** individuos de la generación actual, se los incluye directamente en la siguiente generación. Este número **m** de individuos se determina previamente a la generación de la población inicial. Se busca mejorar la búsqueda local a expensas de la perspectiva global. También provoca que la aptitud del mejor individuo de la población mejore o a lo sumo se mantenga constante, pero nunca decrezca[2].

Además se utiliza una ruleta particionada en ranuras de igual tamaño y se las numeran, a cada individuo se le asigna una cantidad de ranuras proporcional a su aptitud. Se genera un valor aleatorio y dependiendo de este, se obtiene el individuo que pasará a la siguiente generación. Este proceso se repite hasta obtener la cantidad de individuos deseados.

Esta selección intenta contribuir a la siguiente generación con una cantidad adecuada de los mejores individuos, pero no lo garantiza [2].

Operador de Cruza

Debido a cuestiones del dominio del problema se debe evitar que diferentes genes posean iguales características en un mismo cromosoma, lo que representa distintas aulas asignadas a un mismo curso.

Para ello se realiza una modificación de la cruza multipunto: el cromosoma se considera un anillo y se eligen **n** puntos de cruza en forma aleatoria, intercambiándose los segmentos [2]. En este trabajo **n = 2**. Para que se genere un Cromosoma Hijo se obtienen de forma aleatoria los padres. Se eligen dos genes al azar del padre, se tienen en cuenta sus características para que se localicen las mismas en los genes de la madre y se realice el intercambio.

Al solucionar el problema anterior se reduce el espacio de búsqueda ya que no participan de la asignación las aulas disponibles. Esta dificultad se resuelve con una combinación de dos cruzas: una en la que intervienen únicamente aulas asignadas, y otra en que participan tanto aulas asignadas como disponibles.

Operador de Mutación

En la mutación simple la probabilidad se mantiene constante durante las sucesivas generaciones, debido a esto es bastante difícil que la probabilidad elegida sea la adecuada y por ende el operador no sea bien explotado. En este trabajo se utiliza probabilidad adaptativa por convergencia, es decir que se utilizan estadísticas de la población para adaptar la velocidad de mutación[1].Se proponen dos alternativas de implementación:

I. Trasladar las características de un gen a otra posición dentro del mismo cromosoma que represente un aula vacía; II. Intercambiar las características de dos genes que representen aulas asignadas a cursos dentro de un mismo cromosoma .

Secuencia de Aplicación de los operadores

La búsqueda se inicia con una población inicial. Para obtener la siguiente generación se realiza el siguiente proceso:

- *Cierto porcentaje de individuos pasan por selección elitista.*
- *Se obtiene una población temporal aplicando selección por ruleta al resto de la población original.*
- *Este nuevo conjunto de cromosomas (selección elitista más selección por ruleta) se cruza para obtener la diferencia entre la cantidad original y los seleccionados por elitismo.*
- *A este conjunto de individuos obtenidos por cruza se le aplica mutación adaptativa por convergencia. Se utilizan porcentajes*

de mutación en función de la homogeneidad de la población determinados por la cantidad de grupos de individuos iguales y el tamaño de la población. Utilizando la "Diferencia dividida de Newton para la interpolación de polinomios" [7] como parámetro de comparación con la homogeneidad de la población tratada puede determinarse el porcentaje de mutación, teniéndose en cuenta que a mayor homogeneidad mayor porcentaje.

Para la cruza se analizan dos variantes en relación a los padres: en la primera alternativa se obtienen de manera aleatoria sobre el total de la población temporal citada anteriormente; en la segunda, se fuerza a que uno de los padres se obtenga aleatoriamente del conjunto de individuos obtenidos en la selección elitista.

5. IMPLEMENTACION

La estructura del cromosoma no contiene los datos de las aulas ni de los cursos, sino simplemente un enlace entre esas entidades. Esta independencia hace que el manejo de las estructuras de datos sea más eficiente y que no se duplique la información contenida en ellas. De la función de aptitud definida se desprenden varias cuestiones que son convenientes analizar. Primero, la función está acotada, el valor mínimo que puede tomar el peor individuo es **0** (que no cumpla con ninguna de las características de la asignación) y el valor que puede tomar el mejor individuo es **n·100** (es decir, el máximo valor posible por gen multiplicado por **n** que es la cantidad de aulas a asignar en un momento determinado). Si bien estos valores máximos y mínimos son teóricos y en la práctica es poco probable que se los alcance, se puede establecer lo que se considera como individuo apto, por ejemplo, aquel que tenga una aptitud igual o superior al 90% del valor máximo. Segundo, los factores α,β,δ y γ, cuya sumatoria debe ser 1, definen la importancia de cada aspecto de la función de aptitud y pueden variarse si apuntan a destacar uno (o varios) de ellos por sobre los demás. Lo lógico es que se defina el factor de espacio como el de mayor importancia y que los demás aspectos determinen que a misma capacidad de asignación mejor calidad de la asignación. Tercero, la función de aptitud puede incorporar otros factores que se consideren importantes en caso de ser necesario, es decir que es lo suficientemente flexible para incorporar nuevas características para evaluar.

Los operadores que se implementan son selección, cruce y mutación[1][3][5][6]. La utilización de los dos primeros depende del tipo de individuos que se utilicen (cromosomas de longitud estática o variable). Al momento de realizar la selección y cruza en el caso de los individuos de longitud estática se deben implementar heurísticas que contemplen el estado de cada gen (aula ocupada o libre), para evaluar únicamente genes asignados y no realizar sobreasignaciones. Los cromosomas de longitud variable, hacen uso de los operadores sin consideraciones especiales, ya que dichos cromosomas están formados por genes que hacen referencia únicamente a aulas disponibles. El tercer operador mencionado, mutación, tiene como objetivo mover un curso a un aula que no se encuentra utilizada durante la ejecución del algoritmo genético, ya que las características del aula y las necesidades de los cursos son inalterables.

Prueba del Modelo

El modelo fue probado programando[1] una aplicación en Microsoft Visual Basic 6.0 y con el gestor de base de datos Microsoft SQL Server 2000. Las figuras ampliadas pueden encontrarse en el Anexo de este documento.

[1] Colaboración en programación: Ricardo Lentati, Dardo Zibecchi y Darío Aznar. Pertenecientes al GISIA.

La figura 2 corresponde a la interfase utilizada para determinar los parámetros con los que se obtienen mejores resultados durante la asignación de aulas, no se trata de la del usuario final. Los parámetros de configuración se encuentran en el ángulo superior izquierdo (figura 3), junto con el tipo de corrida que se desea realizar: Jornada Completa o por Franjas Horarias. Estos parámetros permiten modificar, en base a sus valores, el tipo de operadores a utilizar y en qué porcentaje. Esto facilita el proceso de prueba y depuración de las distintas combinaciones a utilizar para obtenerse resultados con mejor rendimiento. Para determinar si se realiza la corrida de la aplicación por hora de inicio (PHI) o por franjas horarias (PFH) se utilizan los botones situados a la derecha, debajo de la grilla mayor (figura 4).

Debajo de los parámetros se observa la cantidad de cursos que deben ser tratados, la cantidad de franjas tratadas al final del día y los cursos tratados.

Además se observa si en la corrida se obtuvo un ganador, y la cantidad de franjas que obtienen o no ganadores durante la corrida de un día (figura 5).

Además se brinda información particular de cada franja, en donde se indica la cantidad de cursos que se van a tratar; la aptitud requerida, calculada en base al Porcentaje Mínimo de Aptitud ingresado por pantalla y la cantidad de cursos que se deben tratar. Puede observarse el porcentaje y la aptitud obtenida por el mejor individuo para la franja actualmente tratada (figura 6).

La asignación de las aulas obtenidas del individuo con mayor aptitud, se muestra en la grilla inferior (figura 7). En blanco figuran las aulas disponibles, o aquellas que continúan asignadas de la franja anterior, implicando su no participación en la corrida del algoritmo genético. Por último las aulas que se encuentran resaltadas, implican que es la mejor asignación posible, dadas las necesidades y recursos disponibles, pero que implican una cantidad de inscritos que superan en un 40% a la capacidad del aula.

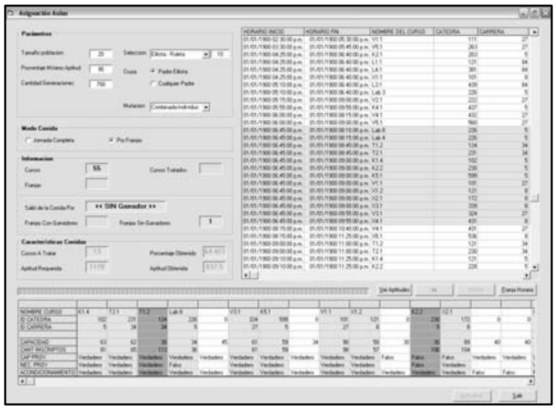

Fig. 2. Interfase de prueba.

Fig. 3. Parámetros de configuración.

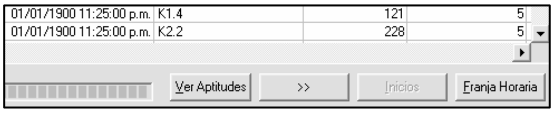

Fig. 4. Parámetros de corrida.

Fig. 5. Obtención de individuos ganadores.

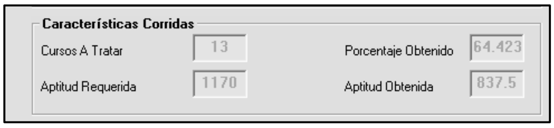

Fig. 6. Características corrida actual.

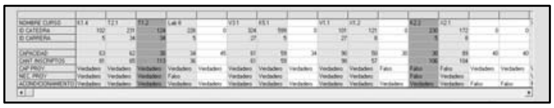

Fig. 7. Asignación resultante.

Resultados Obtenidos

Los parámetros utilizados para una franja horaria de 14 cursos y población de 20 individuos se fijaron de la siguiente manera: porcentaje mínimo de aptitud requerida para que un individuo sea ganador: 90% (implica alcanzar una aptitud de 1260), cantidad de generaciones: 700. Se aplican los operadores: selección elitista; cruza en la que interviene el padre elitista; mutación combinada.

En la figura 8 se observa que la aptitud obtenida luego de 700 generaciones. Se alcanza 1043 de aptitud. A pesar que este individuo no consigue el 90% de aptitud requerida para que se considere ganador, la asignación es aceptable.

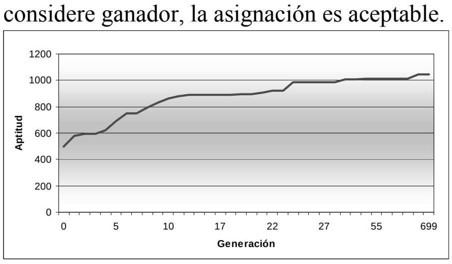

Fig. 8. Aptitud del mejor individuo para 14 asignaciones.

En la figura 9 se observan los mejores individuos obtenidos para un total de 35 corridas con un máximo de 700 generaciones. El desvío estándar es: 78,3071831

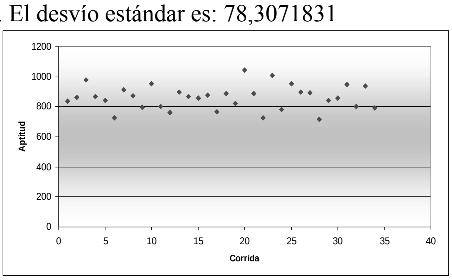

Fig. 9. Dispersión para 14 asignaciones.

Resultados similares se obtienen con simulaciones para distintos valores de asignación (específicamente se probaron 6 y 16 cursos a asignar).

6. CONCLUSIONES

La utilización de algoritmos genéticos para la asignación de aulas ha mostrado un desempeño altamente eficaz en relación a la optimización de la asignación de aulas. Esta eficacia se debe a que el modelo planteado pondera diversos aspectos durante el proceso de resolución del problema.

El cromosoma utilizado contiene solamente la vinculación entre el espacio físico y el curso que se pretende asignar, este tipo de representación flexibiliza el cómputo de la función de aptitud, manteniéndose por separado estructuras relacionadas con la capacidad, el acondicionamiento y la ubicación del aula.

La función de aptitud pondera los cuatro aspectos que se consideran relevantes en la asignación y mediante la determinación inicial de los valores de ponderación, se obtienen distintas soluciones para el mismo problema de asignación. De esta manera es posible, mediante una configuración dada de los parámetros iniciales (α, β, γ, δ), considerar mayor relevancia de algún aspecto sobre otro.

Los valores de la función de aptitud de los individuos ganadores son, en general, superiores al 70% de la asignación teórica prevista (es decir que la combinación de necesidades coincida perfectamente con los recursos disponibles). Este porcentaje es destacable teniendo en cuenta la diversidad de criterios que se sigue para la asignación de las aulas.

Las alternativas descriptas anteriormente para los operadores son el resultado de una serie de pruebas que se realizaron teniendo en cuenta las restricciones establecidas en el propio dominio de problema. Un 30% de selección elitista ha mostrado ser un valor suficiente para que los mejores individuos no desaparezcan en la generación siguiente. La cruza multipunto adaptada a combinaciones de pares de genes de intercambio, genera la diversidad suficiente sin que ello produzca individuos que, por su aptitud, reduzcan considerablemente la aptitud promedio de la población. La mutación adaptativa utilizando la diversidad de los individuos como parámetro de funcionamiento, permite que se exploren otras zonas de dominio de soluciones a medida que la composición de los individuos es más homogénea.

Se ha establecido un método de asignación razonablemente eficiente con complejidad computacional y tiempos de respuesta más que satisfactorios, por ende, se deja planteada la consecución del trabajo con la posible extensión a otros dominios de aplicación.

7. REFERENCIAS

[1] BLICKLE TOBIAS AND THIELE LOTHAR. A comparison of selection schemes used in genetic algorithms. Technical Report 11, Computer Engineering and Communication Networks Lab (TIK), Swiss Federal Institute of Technology (ETH), 1995.

[2] GARCÍA MARTÍNEZ R., SERVENTE M., PASQUINI D. – Sistemas Inteligentes. Nueva Librería SRL. 2003

[3] GOLDBERG D. E. Genetic Algorithms in Search, Optimization and Machine Learning Addison-Wesley Pub. Co. 1989.

[4] KARANIK M. Asignación Dinámica de Aulas Utilizando Algoritmos Genéticos. VII Workshop de Investigadores en Ciencias de la Computación (WICC). 2005.

[5] KOZA J. R. Genetic Programming: A Paradigm for Genetically Breeding Populations of Computer Programs to Solve Problems. Stanford University Computer Science Department technical report. 1990.

[6] SPEARS W. M., DE JONG K.A., BACK T., FOGEL D.B., DE GARIS H. An Overview of Evolutionary Computation. Proc. European Conference on Machine Learning. 1993.

[7] CHAPRA S., CANALE R. Métodos Numéricos para Ingenieros. Mac Graw Hill. 1998

ANEXO: FIGURAS AMPLIADAS.

Fig. 2. Interfase de prueba.

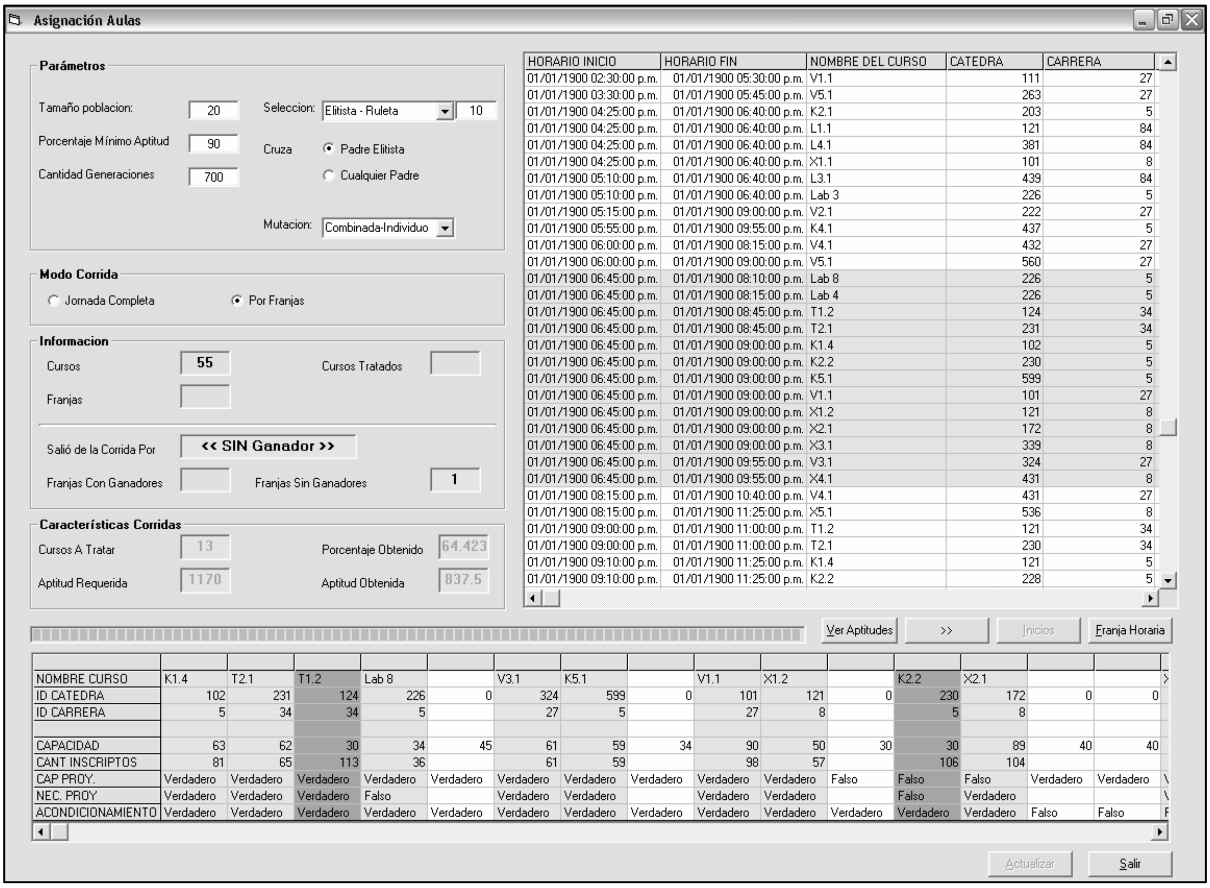

Fig. 3. Parámetros de configuración.

Fig. 4. Parámetros de corrida.

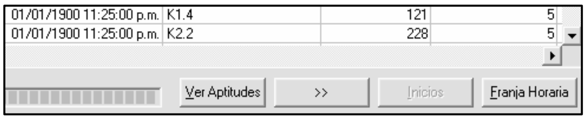

Fig. 5. Obtención de individuos ganadores.

Fig. 6. Características corrida actual.

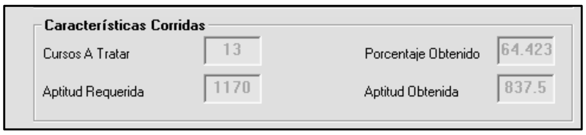

Fig. 7. Asignación resultante.