

Organización de asignaturas en una facultad mediante algoritmos genéticos

Fernando Nellmieldin y Mariano Peyregne

Trabajo práctico final de “Inteligencia Computacional”, II-FICH-UNL.

Resumen—El problema de acomodar los horarios, llamado **Timetabling Problem** en inglés, es una difícil tarea que se debe realizar en cualquier institución educativa. Se tiene un conjunto de recursos limitados tales como la cantidad de profesores, alumnos o aulas, y se los debe asignar a cada una de las asignaturas que se cursan en la institución, de manera de minimizar el impacto negativo tanto en los alumnos como en los docentes. Se quieren evitar situaciones tales como que se cursen simultáneamente materias del mismo año o de años adyacentes, ya que con frecuencia hay estudiantes que pueden asistir a asignaturas de distintos años.

En este trabajo abordamos el **Timetabling Problem** utilizando un **Algoritmo Genético** y modelando las restricciones en una simple función de **fitness**. Si bien no consideramos todas los requisitos que participan en este problema, encontramos resultados satisfactorios con respecto al programa semanal obtenido. Primero presentamos el modelado del problema, luego los operadores definidos para el algoritmo genético y el **fitness**. Por último se presentan los resultados obtenidos y algunos ejemplos.

Palabras clave—timetabling problem, restricciones, algoritmo genético, optimización

I. INTRODUCCIÓN

La distribución de los horarios en que se dictan las asignaturas de la facultad se debe actualizar todos los años e implica un trabajo iterativo hasta que se cumplan todas las restricciones, o al menos se minimice la cantidad de conflictos. Entre los requisitos a cumplir, se encuentran los inherentes a los planes de las carreras, tales como que no se asignen materias del mismo año al mismo horario, o incluso de años consecutivos, ya que muchos alumnos asisten a asignaturas de distintos cuatrimestres por problemas de correlatividades. Asimismo, otras limitaciones surgen por la disponibilidad de aulas (y sus correspondientes equipamientos y límite de alumnos) y los docentes asignados a cada clase. Sobre este último, con frecuencia surge que cada profesor impone una restricción del tipo: “sólo de mañana” o “los lunes y martes únicamente”. De la definición de todas estas limitaciones, vemos que encontrar una distribución de clases durante la semana se torna una tarea casi imposible para cualquier ser humano.

En este trabajo, nos enfocamos en solucionar el **Timetabling Problem** (TTP) para un conjunto limitado de restricciones. Además, simplificamos la definición del problema. Debido a la variedad, artículos anteriores como el de Colorni y otros [1], tomaron casos particulares para su estudio. Nosotros especializamos este trabajo en nuestra propia carrera: Ingeniería en Informática. Por lo tanto, modelamos las asignaturas de los 5 años de la misma (y del mismo semestre). También, imitando a Perez de la Cruz y Ramírez Rodríguez [2], se resumieron las clases de cada materia a 3: una de teoría y dos de práctica, que pueden ser de dos o tres horas cada una (a las que los alumnos

deben acudir a ambas). En total existen 20 materias que se dictan en un semestre, de 3 clases cada una, lo que suma un total de 60 clases a ubicar. Un ejemplo para las primeras asignaturas se ve en la Tabla I. Luego se estudiará el caso más general.

TABLA I
ASIGNATURAS MODELADAS COMO CLASES

Asignatura	Clase	ID Clase
Matemática	Matemática (Teoría, 2 horas)	1
Matemática	Matemática (Práctica 1, 3 horas)	2
Matemática	Matemática (Práctica 2, 3 horas)	3
Programación	Programación (Teoría, 2 horas)	4
Programación	Programación (Práctica 1, 2 horas)	5
Programación	Programación (Práctica 2, 2 horas)	6

No se tuvo en consideración la distribución de los profesores ni sus restricciones añadidas, por lo que se asume que cada docente tiene a su cargo una sola clase y puede darla en cualquier franja horaria. En nuestra institución, como en muchas, se dicta más de una carrera. En lugar de modelar la distribución de clases y aulas para todas las carreras, se tomó un camino alternativo: se limitarán las habitaciones ocupadas por el cursado de estudio, de manera que sólo se puede ocupar determinado número de salas en simultáneo.

Por el lado de las restricciones, elegimos las siguientes a satisfacer:

- **Limitación de aulas:** No se pueden utilizar más de 4 aulas al mismo tiempo
- **Solapamiento en años adyacentes:** No se deben superponer las clases de asignaturas de un año, con las clases del año anterior o siguiente, para dar la posibilidad a los alumnos de cursar ambos semestres.
- **Superposición en un mismo año:** No es posible que se dicten dos clases del mismo año en el mismo momento, ya que imposibilita a los estudiantes a hacerse presente en ambas clases.

Con respecto a la distribución de horarios, siguiendo el enfoque de Enzhe y Ki-Seok [3], y Martínez y otros [4], se divide cada día, de lunes a viernes, en 5 módulos o franjas horarias. A cada módulo se le asigna un ID único, que servirá para identificarlo en el algoritmo genético. En la Fig. 1 se aprecian los módulos y el ID asignado a cada uno. El uso de estos identificadores se explicará en la próxima sección.

II. ALGORITMO GENÉTICO

Para la solución del problema, se utilizó un AG [5], [6], en el que se definió un fenotipo como se muestra en la Fig. 2. De acuerdo a los ID definidos para cada clase como en la Tabla I, se le asigna una posición específica en el

Horario	Lunes	Martes	Miércoles	Jueves	Viernes
8 a 10	0	5	10	15	20
10 a 13	1	6	11	16	21
14 a 16	2	7	12	17	22
16 a 18	3	8	13	18	23
18 a 21	4	9	14	19	24

Fig. 1. Módulos definidos para la semana

fenotipo a cada una. Así, la clase con identificador igual a 1, será la primera en el fenotipo. Lo mismo se aplica a ID 2, que estará ubicada segunda, y de igual manera con el resto. En cada posición del fenotipo, se almacena un número que referencia en qué bloque está situada. Siguiendo con el ejemplo de la Fig. 2, se aprecia que la clase 1 se dictará en el módulo 12, que al consultar la Fig. 1 vemos que corresponde a los miércoles a las 14 horas.

A partir de la representación fenotípica, se codifica en un genotipo binario de 300 bits de extensión, ya que se toman 5 bits para cada elemento del fenotipo (que está en el rango 0 – 24).

12	13	16	1	3	...	20
----	----	----	---	---	-----	----

Fig. 2. Representación fenotípica

A. Inicialización

Se parte de una población inicial aleatoria. Para generarla, se toman las clases de a una y se le asigna un bloque aleatorio en el rango 0 – 24. De esta forma, se comenzará con muchos individuos con una evaluación de fitness pobre, ya que no se espera que cumplan ninguna restricción al inicio, pero todos serán válidos en el sentido de que no se asignarán bloques fuera del rango 0 – 24. Esta propiedad se mantendrá durante todo el algoritmo.

B. Selección de individuos para la siguiente generación

La población siempre se mantiene en el mismo tamaño. En cada generación, se deben elegir a los individuos que formarán parte de la generación siguiente. Se utilizó el método de selección por ventanas. A los individuos elegidos se les aplicará, si corresponde, las cruza y mutaciones que se describen en la próxima sección. Además, a través de elitismo, siempre se hereda el mejor individuo (según su fitness) a la próxima generación. Por último, se elige un término de brecha generacional del 10% del tamaño de la población. Por lo tanto, sin importar el fitness, se elige aleatoriamente cierta cantidad de individuos que serán integrantes del próximo conjunto. De esta manera, nos aseguramos que siempre haya cierta diversidad en el conjunto de búsqueda.

C. Operadores genéticos

Se definieron dos operadores de mutación y uno de cruza. Estos operadores se aplican en representación genotípica, pero cuidan de realizarse en posiciones que se corresponden con el fenotipo, de manera de no generar individuos inválidos. Cada uno de estos operadores tiene una probabilidad de aplicarse definida al inicio, por lo que puede suceder que se realicen ambas mutaciones y no la cruza, viceversa o cualquier otra combinación.

C.1. Cruza: Se utiliza una cruza de dos puntos. Se cuenta con dos padres y se generan aleatoriamente dos posiciones en el genotipo de ambos. Luego se intercambia el contenido genético entre éstos. Las posiciones elegidas se corresponden con los límites de cada elemento del fenotipo, de manera de no generar individuos con bloques de horarios inválidos.

C.2. Mutación por movimiento: Se genera una posición en el fenotipo al azar, que identificará a un bloque donde está asignada cierta asignatura. A continuación se buscará un módulo que se encuentre libre y que sea del mismo tamaño (2 o 3 horas según corresponda). Si se encuentra un espacio libre (antes de un número limitado de iteraciones), se moverá la clase al nuevo bloque. En la Fig. 3 se ve un ejemplo.

Horario	Lunes	Martes	Miércoles	Jueves	Viernes
8 a 10	Mat	Química			
10 a 13		POO	APS	Mat 2	PDS
14 a 16			IComp		
16 a 18					
18 a 21					

Horario	Lunes	Martes	Miércoles	Jueves	Viernes
8 a 10	Mat	Química		IComp	
10 a 13		POO	APS	Mat 2	PDS
14 a 16					
16 a 18					
18 a 21					

Fig. 3. Mutación por movimiento

C.3. Mutación por permutación: Se toma una posición aleatoria del fenotipo. A partir de ésta, y dependiendo de si es de 2 o 3 horas, se genera una nueva posición aleatoria del mismo horario. Para continuar, se intercambian las posiciones de ambas, de manera de que donde se ubicaba la primera, ahora se ubique la segunda y viceversa. En la Fig. 4 mostramos un ejemplo ilustrativo.

Horario	Lunes	Martes	Miércoles	Jueves	Viernes
8 a 10	Mat	Química			
10 a 13		POO	APS	Mat 2	PDS
14 a 16			IComp		
16 a 18					
18 a 21					

Horario	Lunes	Martes	Miércoles	Jueves	Viernes
8 a 10	Mat	IComp			
10 a 13		POO	APS	Mat 2	PDS
14 a 16			Química		
16 a 18					
18 a 21					

Fig. 4. Mutación por permutación

D. Función de fitness

La función de fitness que evalúa la calidad de la solución representada por cada individuo, debe tener en cuenta qué tanto cumple con las restricciones definidas. Para ello, el fitness está formado por 3 términos, uno por cada requisito a satisfacer. Se probaron dos tipos de funciones de fitness, una aditiva y otra multiplicativa. Primero se expondrán cada uno de los términos y luego la relación entre éstos.

D.1. Sobrepasso de Aulas: Cuenta la cantidad de veces que se supera la capacidad de aulas definida. Si se dispone de 4 aulas, contará para cada bloque horario, la cantidad de clases que se dictan simultáneamente. Si se sobrepasa el límite, se sumará 1 en este contador.

D.2. Solapamiento de años adyacentes: Se suman la cantidad de oportunidades en las que clase de años adyacentes se dictan en el mismo módulo. Por ejemplo, este contador sumará 1 cada vez que el individuo represente que un curso de segundo año se solape con cualquier clase de primer o tercer año.

D.3. Superposición de clases: Se aumentará el contador de esta restricción cuando dos clases del mismo año se asignen al mismo bloque horario, ya que significará que un alumno debería estar al mismo tiempo en dos clases distintas.

Una vez definido los términos de la función de fitness, procedemos a enunciar su forma. Primero presentamos la función de aptitud aditiva en la Ecuación 1:

$$fitness_{suma} = \frac{a_1}{p+1} + \frac{a_2}{q+1} + \frac{a_3}{r+1} \quad (1)$$

donde los a_i representan el peso o importancia que se le da a cada término del fitness y se definen según la Tabla II. Por otra parte, p , q y r representan los contadores, datos que se enuncian en la Tabla III.

TABLA II
VALORES DE LOS PARÁMETROS DEL FITNESS SUMA

Parámetro	Valor
a_1	0,10
a_2	0,35
a_3	0,55

TABLA III
NOMENCLATURA DE LOS VALORES DEL FITNESS

Contador	Nombre
p	Sobrepaso de aulas
q	Solapamiento de años adyacentes
r	Superposición de clases

A continuación, enunciamos la forma del fitness producto en la Ecuación 2:

$$fitness_{producto} = \frac{1}{(p+1)^{b_1}} \frac{1}{(q+1)^{b_2}} \frac{1}{(r+1)^{b_3}} \quad (2)$$

donde p , q y r son como en la Ecuación 1 y los b_i representan la importancia de cada término y cuyos valores se muestran en la Tabla IV.

TABLA IV
VALORES DE LOS PARÁMETROS DEL FITNESS PRODUCTO

Parámetro	Valor
b_1	1
b_2	2
b_3	5

Como vemos, en ambos casos se le da mayor importancia al término de superposición, ya que ésta es la mayor de las restricciones que debemos satisfacer, y sin ella, la tabla de horarios obtenida sería de poca utilidad para cualquier alumno. El segundo término en importancia es el solapamiento entre asignaturas de años adyacentes, ya que es una de las condiciones más deseables de una tabla horaria.

III. RESULTADOS

La elección de una u otra función de fitness es arbitraria. En ambas se obtienen resultados similares. Es de destacar que la función productoria tiene “peor” fitness, debido a que los valores están exponenciados. Sin embargo, al comparar la calidad de la solución en término de los contadores p , q y r , apreciamos lo antedicho: que se logran resultados similares.

Para probar el desempeño del algoritmo, se ejecutó para diversos conjuntos de configuraciones. Se estimaron los parámetros del algoritmo que daban mejores resultados, los cuales se resumen en la Tabla V.

TABLA V
PARÁMETROS DEL ALGORITMO GENÉTICO

Parámetro	Valor
tamaño de población	500
brecha generacional	50
elitismo	1
cantidad de generaciones	300-500
probabilidad cruza	0.8
probabilidad mutación movimiento	0.2-0.25
probabilidad mutación permutación	0.2-0.25

En la Fig. 5 mostramos un resultado donde se alcanza fitness perfecto, ya que se satisfacen todas las restricciones. En la Fig. 6 vemos la tabla horaria que es obtenida en esta ejecución.

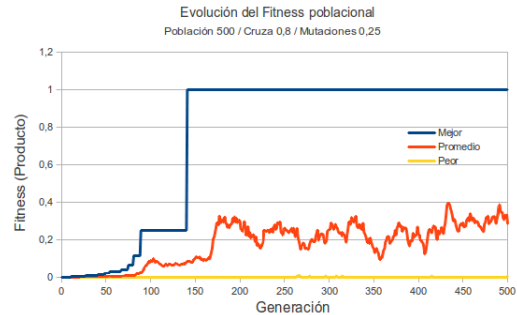


Fig. 5. Resultado con fitness perfecto

	Lunes	Martes	Miercoles	Jueves	Viernes
8-10	MBasica (T)		FPro (T)	FPro (P1)	FPro (P2)
10-13	MBasica (P2)			CTecnica I (P1)	
14-16	Quimica (T)			CTecnica I (P2)	
16-18		MBasica (P1)	CTecnica I (T)		Quimica (P1)
18-21	Quimica (P2)				
8-10		TComp (T)			
10-13		Fisica I (P2)			TComp (P1)
14-16		Calculo II (P2)	ISoft I (P1)		Calculo II (T)
16-18	TComp (P2)			Fisica I (P1)	
18-21		ISoft I (T)	Calculo II (P1)	Fisica I (T)	ISoft I (P2)
8-10	TProg (P2)		EDigital (P1)	Estadística (T)	Estadística (P1)
10-13	TProg (P1)		CGrafica (P1)	EDigital (P2)	
14-16	TProg (T)			CGrafica (T)	
16-18			Estadística (P2)		EDigital (T)
18-21	CGrafica (P2)				
8-10		APS (P1)			
10-13		PDS (P2)			MCont (P2)
14-16			APS (T)		SO (P1)
16-18		PDS (P1)		MCont (P1)	
18-21		MCont (T)	SO (T)	APS (P2)	SO (P2)
8-10			PFinal (P2)	Redes II (T)	PDI (T)
10-13			PFinal (P1)	Economía (P2)	
14-16	Redes II (P1)	Economía (P1)		Redes II (P2)	
16-18	PDI (P2)		PDI (P1)		Economía (T)
18-21	PFinal (T)				

Fig. 6. Tabla horaria obtenida con fitness perfecto

Para contrastar, en las Fig. 7 y Fig. 8 se muestra el resultado para un fitness regular. Los resultados que corresponden a este fitness se muestran en la Tabla VI. Se

puede apreciar en la Fig. 7 que los parámetros utilizados no son los óptimos según la Tabla V. Además, en esta figura se sombrearon los solapamientos entre años adyacentes.

TABLA VI

RESULTADOS OBTENIDOS EN LOS CONTADORES DEL FITNESS REGULAR

Contador	Resultado
Sobrepaso de aulas	0
Solapamiento de años adyacentes	9
Superposición de clases	0

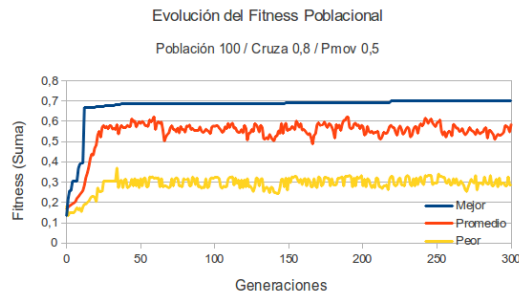


Fig. 7. Resultado con fitness regular

	Lunes	Martes	Miercoles	Jueves	Viernes
8-10	Química (P2)	MBasica (P2)			
10-13	CTec I (P2)			CTec I (P1)	
14-16	FPro (T)	FPro (P2)	Química (T)		
16-18	MBasica (P1)	FPro (P1)	CTec I (T)		Química (P1)
18-21	MBasica (T)				
8-10			Física I (P1)		Calculo II (T)
10-13		TComp (P2)	TComp (P1)		ISoft I (T)
14-16				ISoft I (P2)	TComp (T)
16-18			Calculo II (P2)		Calculo II (P1)
18-21		Física I (P2)	ISoft I (P1)	Física I (T)	
8-10	Estadística (P2)	Estadística (T)			CGrafica (P1)
10-13	EDigital (T)			EDigital (P1)	
14-16		Estadística (P1)	CGrafica (T)		
16-18	TProg (P1)			TProg (P2)	
18-21	EDigital (P2)		TProg (T)		CGrafica (P2)
8-10		APS (P1)			
10-13			SO (T)		
14-16	MCont (T)		MCont (P1)	SO (P2)	APS (T)
16-18		MCont (P2)	APS (P2)	PDS (P2)	
18-21		PDS (P1)			SO (P1)
8-10	PFinal (P2)		Redes II (P1)	Economía (T)	
10-13	Economía (P1)	PFinal (P1)			
14-16		Redes II (T)			
16-18	Redes II (P2)	PDI (P2)			Economía (P2)
18-21	PDI (T)		PDI (P1)	PFinal (T)	

Fig. 8. Tabla horaria obtenida con fitness regular

Es de destacar que todos estos resultados fueron obtenidos para la simplificación de las clases que se enunciaba al principio, por lo que estas soluciones son sólo del caso simplificado. En la Fig. 9 mostramos el resultado obtenido para el caso real, en el que ya no hay necesariamente 3 clases por asignatura, si no que hay más (principalmente en los primeros 2 años) o menos (en los años avanzados). Además se debe aclarar que se dicta más de una comisión por materia, por lo que los alumnos pueden elegir a cuál asistir. Por lo tanto, el contador de solapamientos ya no es tan representativo.

IV. CONCLUSIONES

Si bien en el presente trabajo se realizaron muchas simplificaciones de cara a reducir la complejidad, la inclusión de más requisitos no presenta, en la práctica, una mayor dificultad. A pesar de que las soluciones obtenidas no son utilizables debido a las restricciones no consideradas, concluimos que el uso de un algoritmo genético para encarar el TTP fue la elección correcta.

	Lunes	Martes	Miercoles	Jueves	Viernes
8-10			MBasica (P1)	MBasica (P4)	Química (P3)
10-13		MBasica (T)	FPro (P2)	Química (T)	FPro (P1)
14-16	CTecnica (P1)	FPro (Pratica3)	FPro (T)	Química (P1)	MBasica (P3)
16-18	MBasica (P2)	CTecnica (T2)		Química (P2)	
18-21	CTecnica (T1)		CTecnica (P2)		
8-10	TComp (P1)	Calculo II (P1)	Calculo II (P3)	Física I (P2)	TComp (P2)
10-13	Física I (T)	Física I (L3)	Calculo II (T1)	ISoft (T)	Calculo II (T2)
14-16	ISoft (P)		Física I (P1)	TComp (T1)	Física I (L2)
16-18	TComp (P3)	Calculo II (P5)	Física I (L1)	TComp (T2)	Física I (L4)
18-21		Física I (P4)	Calculo II (P2)	Calculo II (P4)	Física I (P3)
8-10	CGraf (P)	Estadística (T)			
10-13					
14-16		TProg (P)		CGraf (T)	Estadística (P)
16-18					
18-21	EDigital (T)		TProg (T)		EDigital (P)
8-10					
10-13	APS (T)	MContinuo (T)			
14-16	PDS (T)		PDS (P)		
16-18	MContinuo (P)		SO (T)	APS (P)	
18-21		SO (P)			
8-10			PFinal (T)		PDI (T)
10-13			EyC (T)	Redes II (T)	Redes II (P)
14-16		EyC (P)			PDI (P)
16-18					
18-21					

Fig. 9. Tabla horaria obtenida para el caso real

Uno de los problemas que encontramos durante el desarrollo del trabajo, fue que la diversidad de cada generación es pequeña debido a que la función de fitness (la cual se basa en las restricciones planteadas) tiene un limitado conjunto de valores posibles. Esto provoca que fenotipos muy distintos tengan exactamente el mismo fitness, y este fenómeno se repita para toda la población.

A. Trabajo Futuro

Como trabajo futuro, se plantea la inclusión de más restricciones a la función de aptitud, tales como requerimientos propios de los profesores o relacionadas a la disponibilidad y equipamiento de las aulas. En un análisis preliminar, concluimos que no habría mayor dificultad, siguiendo el enfoque actual, en implementar estas extensiones.

Además, se plantea la modificación del modelado de la semana en bloques de longitud y ubicación fijas, para pasar a franjas horarias en término de numeros reales. De esta manera, se daría la posibilidad de que las clases comiencen, por ejemplo, a las 9:45 horas, y terminen a las 11:15. Sin embargo, esto implicaría hacer un total cambio del fenotipo y las correspondientes modificaciones en los operadores, por lo que no se consideró en el trabajo actual y es una de las mayores limitaciones del mismo.

REFERENCIAS

- [1] D. Datta, D. K., y C. Fonseca, "Solving class timetabling problem of IIT kampur using multi-objective evolutionary algorithm," 2006.
- [2] C. Pérez de la Cruz y J. Ramírez Rodríguez, "Un algoritmo genético para un problema de horarios con restricciones especiales," *Revista de Matemática: Teoría y Aplicaciones*, no. 18, pp. 215–229, 2011.
- [3] Y. Enzhe y S. Ki-Seok, "A genetic algorithm for a university weekly courses timetabling problem," *International Transactions in Operational Research*, no. 9, pp. 703–717, 2002.
- [4] F. Martínez Ruiz, E. García Sánchez, J. Muñoz Arteaga, y C. Castañeda Ramírez, "Timetabling académico usando algoritmos genéticos y programación celular," 2006.
- [5] D. E. Goldberg, *Genetic Algorithms in Search, Optimization, and Machine Learning*. Addison-Wesley Professional, 1989.
- [6] O. Roeva, *Real-World Applications of Genetic Algorithms*. InTech, 2012.
- [7] A. Colomi, M. Dorigo, y V. Maniezzo, "A genetic algorithm to solve the timetabling problem," *Computational Optimization and Applications*, 1993.