

Programación de Estudiantes un Caso de Estudio (Universidad Central)

John Camilo Cifuentes Taborda
Universidad Central
Cra 5 # 21 – 38, Bogotá - Colombia.

RESUMEN

Los estudiantes de la jornada nocturna del Departamento de Ingeniería Industrial de la Universidad Central (Colombia), deben programar para el semestre, su horario de clase semanal mediante la selección de cursos que ofrece dicho departamento. La Universidad dispone de unos días al inicio del semestre, en donde se le informa al estudiante la oferta de cursos y sus horarios; con base a esto, el estudiante realiza la selección de sus cursos para crear su horario. Este proceso no es automatizado, por lo anterior, si el estudiante no realiza la inscripción de materias a tiempo, corre el riesgo de no encontrar cupo en las materias que debe cursar o en las que son de su interés. Esta investigación propone un modelo de solución (Student Secheduling Problem - SSP) para este problema del mundo real, con la idea de formularlo como un problema de programación lineal binaria a través de los parámetros necesarios para su ejecución.

PALABRAS CLAVE. Horarios, Programación de estudiantes, Programación binaria.

Área principal: OC, EDU - IO

ABSTRACT

The nocturnal students from the Department of Industrial Engineering at the Central University, have to schedule for the semester their weekly Timetabling, selecting courses offered for this department. The University has a few days at the beginning of the each semester, time in which some personal of the Department informs to the students, about the courses and its Timetabling, based on this, the student makes the selection of courses to create their own schedule. This process is not automated and for this reason, if the student doesn't make the processes of the registration subjects on time, has the risk of not being admitted into the subjects that are interested for him. This research proposes a model solution (Student Secheduling Problem - SSP) for this real world problem, with the idea of formulating it as a binary linear programming problem over the parameters required for its execution.

KEYWORDS. Timetabling, Student Scheduling. Binary programing

Main area: OC, EDU - IO

1. Introducción

La presente investigación pretende proponer una metodología para apoyar el proceso de construcción de horarios en una Universidad, en donde el 75% de los estudiantes hace parte de la jornada nocturna; este proceso ha generado una problemática al interior de la organización, debido a que para el estudiante la construcción de horarios se vuelve compleja a la hora de cumplir con restricciones como el número determinado de créditos a cursar por semestre, el mínimo de horas que debe ver a la semana, las ventanas de tiempo para los cursos y los prerrequisitos existentes entre materias. El problema es reiterativo para estudiantes que no van nivelados en su ciclo académico, debido a pérdida de materias en semestres anteriores, retiros temporales de la universidad por parte del estudiante y cambios de carrera al interior de la institución, entre otros. Es importante mencionar que el problema que se pretende abordar abarca la construcción de horarios de manera particular para cada estudiante, encontrando un conjunto de características y restricciones comunes a los estudiantes que viven semestre a semestre, la problemática de creación de horarios.

la pregunta de investigación que se aborda, consiste en saber si es posible encontrar un modelo matemático que genere horarios para estudiantes de una universidad nocturna, de manera eficiente y con soluciones preferiblemente óptimas, evaluados para instancias que se adapten lo mejor posible a las condiciones del problema real.

2. Marco Teórico

Muchas organizaciones (académicas, de salud, transporte, deporte, etc) deben encarar frecuentemente problemas de generación de horarios, razón por la cual surge la necesidad de utilizar herramientas eficientes para este tipo de situaciones (Zhang D., 2003) (Parker, 1995). En las instituciones educativas las actividades de asignación y construcción de horarios, son consideradas como una de las principales tareas de tipo administrativo (Burke E. K., 2002); así mismo, en las universidades esta actividad puede abarcar varios elementos como la asignación de aulas, la selección de profesores, día y hora de las clases, entre otros.

En el ámbito académico, el problema de programación de horarios (Educational Timetabling Problem), ha sido definido de manera general (Wren A., 1996), como la asignación de recursos sujeta a restricciones de objetos (cursos, exámenes, profesores, etc), que se deben ser colocados en espacios de tiempo, de tal manera que satisfagan un conjunto de objetivos deseables. El Educational Timetabling Problem (ETP), ha sido estudiado por varios investigadores, (De Causmaecker P., 2009) & (Daskalaki S., 2004), y es un problema de gran relevancia a nivel educativo, debido a que el proceso de generación de horarios se ha deseado automatizar y realizar de manera eficiente en todas las organizaciones en donde es vigente.

En la figura 1, se presenta una clasificación del Educational Timetabling Problem – ETP, así mismo, algunas de las restricciones asociadas al problema son (Burke E.K., 1997):

Restricciones duras son aquellas que deben cumplirse:

- Ningún recurso puede ser demandado al mismo tiempo y en el mismo lugar.
- Para cada período de tiempo en el cual se han programado eventos, deben existir recursos suficientes.

Las restricciones blandas no es necesario cumplirlas aunque si es deseable hacerlo, debido al costo de su violación:

- Limitaciones de tiempo entre eventos: Un evento que necesita ser programado antes o después que otro.
- División de eventos en el tiempo: Los estudiantes no pueden ser sometidos a dos o más eventos en un período de tiempo determinado.

- Coherencia: Los profesores desean tener todas sus clases en días seguidos y disfrutar de días libres de clases.
- Asignación de recursos: Los profesores pueden preferir dar su clase en un aula especial, o cierto examen debe ser presentado en un salón en particular.

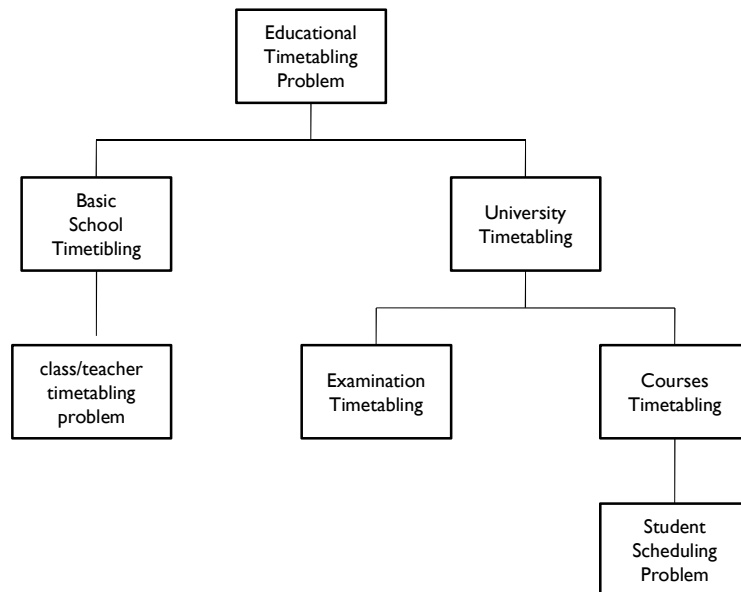


Figura 1. Clasificación General Educational Timetabling Problem (ETP)
Fuente: (Van den Broek J., 2009), (Cheng E., 2002)

Una revisión de la literatura, permite clasificar las técnicas de solución que abordan los problemas ETT y una descripción general del Student Scheduling Problem - SSP:

Técnicas Exactas (Burke E. K., 2002)

En este grupo se destacan los Métodos de Clústers y las Técnicas basadas en restricciones matemáticas, en el primer método los eventos se dividen en grupos que satisfagan las restricciones duras, para luego asignar los grupos a períodos de tiempo que cumplan con las restricciones suaves, sin embargo la metodología puede generar resultados de baja calidad. Por otro lado, en las Técnicas basadas en restricciones el problema es modelado matemáticamente, con la idea de satisfacer un conjunto de restricciones utilizando reglas o métodos matemáticos que permita resolver el problema para todas las restricciones o en su defecto satisfacer algunas de ellas.

Técnicas no Exactas (Burke E. K., 2002), (Cheng E., 2002)

En este grupo aparecen los Métodos Secuenciales y las Metaheurísticas, en el primero se ordenan los eventos usando heurísticas de dominio, para luego asignar los eventos secuencialmente dentro de un período válido de tiempo de tal forma que dentro del período ningún evento esté en conflicto con otro. Por el lado de las Metaheurísticas, estos métodos inician con una o más soluciones de partida, y aplican estrategias de búsqueda para tratar de escapar de óptimos locales. Muchos de estos métodos pueden producir soluciones de alto nivel pero a expensas de un considerable tiempo computacional.

El Student Scheduling Problem - SSP

El SSP es un problema derivado del ETT, en donde se busca generar horarios para un conjunto de estudiantes, con la idea de satisfacer las restricciones individuales de cada alumno, que pueden estar relacionadas con la disponibilidad de tiempo, las materias que pueden cursar y sus cursos de preferencia.

Una revisión de referencias, permite obtener una visión de cómo ha sido abordado el problema por varios autores, en estudios como los de (Head C., 2007), se presenta un enfoque en

el cual tanto los horarios y el lugar que ocupan los alumnos en los cursos, se construyen simultáneamente, la idea de este proceso es crear una programación basada en una heurística en donde se pretende hacer un buen uso del tiempo del estudiante, y utilizar casi que de manera óptima los recursos físicos y humanos de la Universidad. De forma similar, trabajos relacionados (Pongcharoen P., 2008), proponen una herramienta orientada a la programación de horarios de cursos y estudiantes, denominada Stochastic Optimisation Timetabling Tool (SOTT), en esta, se utilizan varias técnicas como Algoritmos Genéticos, Simulado Recocido y Búsqueda Aleatoria para resolver el problema. Otro trabajo importante a referenciar, es el documento presentado por (Van den Broek J., 2009), en este los autores organizan grupos prioritarios de materias que debe cursar el estudiante, con el ánimo de construir su horario y plantean un problema multi-objetivo con la idea de abarcar posibles soluciones al problema.

Por otro lado, trabajos más generales (Daskalaki S., 2004) muestran casos de estudio para la asignación de aulas resueltos como modelos de programación entera. De manera similar algunas investigaciones (Saldaña A., 2007) tienen como objeto resolver la programación de horarios para una universidad, donde se desea programar las asignaturas a ser dictadas teniendo en cuenta elementos del modelo como profesores, días, horarios y aulas. Otros autores (Carrasco M.P., 2004), proponen trabajos con principios de inteligencia artificial en donde se plantean problemas de programación de horarios Clases/Profesor (class/teacher timetabling problem - CTTTP) como modelos de optimización combinatoria, resueltos por medio de redes neuronales. Investigaciones similares proponen heurísticas para resolver el problema de asignación de aulas (Castro, 2004), en donde se utiliza el principio de Slot/Sección que permite formular el problema como un modelo de asignación que considera programación binaria.

3. Objetivos

General

Desarrollar un modelo que brinde una solución factible, de calidad y bajo tiempo computacional, al problema de programación de estudiantes (Student Scheduling Problem), en una universidad en donde la mayoría de los alumnos son nocturnos.

Específicos

- Desarrollar un modelo matemático con las principales características del problema en cuestión, considerando las restricciones reales que tienen los estudiantes al programar sus clases.
 - Realizar un estudio de los principales parámetros que sean necesarios para la implementación del modelo definido.
 - Identificar las materias críticas ó prioritarias que los estudiantes deben ver para su formación académica, y que son prerrequisito para un mayor número de materias posteriores.
 - Determinar las ventajas y desventajas del modelo planteado y de los métodos de solución utilizados con el fin de medir la utilidad del mismo y la aplicación práctica para la universidad.

4. Propuesta de Solución

Se pretende abordar el problema de manera similar a la propuesta por otros autores, desarrollando un modelo matemático MIP que permita dar solución a los objetivos propuestos. En este orden de ideas, para el desarrollo del modelo, se tienen los siguientes elementos:

Conjuntos

Los conjuntos encontrados que recogen la información del problema son:

(1) Estudiantes (S): Grupo de alumnos inscritos para un semestre en jornada nocturna.

S conjunto de estudiantes indexado en s

(2) Cursos (C): Materias ofrecidas en el plan de estudios de Ingeniería Industrial para la jornada nocturna.

C conjunto de cursos indexado en c

(3) Sección (I): Grupo de estudiantes que toman con el mismo profesor, a la misma y hora y en el mismo salón, una sección de un curso. Un curso puede tener más de una sección, y dicha sección es dictada en días y horas específicos.

$I(c)$ conjunto de secciones asociadas a un curso (c) e indexadas en i

(4) Días-Horas (D): Son los días de la semana en los cuales se dictan las secciones de cada curso a una hora específica para los estudiantes nocturnos.

$D(i)$ conjunto de días-hora asociados a una sección (i) e indexados en d

(5) Prioridades (P): Nivel de importancia que tiene el curso dentro del pensum de materias ofrecidas por la universidad, a la hora de ser programada dentro de los horarios de cada estudiante.

$P(c)$ conjunto de prioridades asociadas a un curso e indexados en p

(6) Créditos (R): Es un indicador que se utiliza en las universidades, como unidad de medida que expresa el volumen del aprendizaje de un curso, según lineamientos del Ministerio de Educación Nacional de Colombia.

$R(c)$ conjunto de créditos asociadas a un curso e indexados en r

Parámetros Generales del Modelo

Matriz de Prerrequisitos de Cursos (PC_{ic}): El número de filas y de columnas que contiene la matriz, está dado por la cantidad de cursos que contiene el pensum de estudios, la materia "i" de la fila tiene como prerrequisito directo la materia "j" de la columna.

$$PC_{ij} = \begin{cases} 1 & \text{Si el curso (i) tiene como pre-rrequisito al curso (j)} \\ 0 & \text{De lo contrario} \end{cases}$$

Ejemplo:

Matriz de Prerrequisitos de Cursos (PC_{cc})

Cursos	A	B	C	D	E	F	G	H	I
A	0	0	0	0	0	0	0	0	0
B	0	0	0	0	0	0	0	0	0
C	0	0	0	0	0	0	0	0	0
D	1	1	0	0	0	0	0	0	0
E	0	1	1	0	0	0	0	0	0
F	0	1	0	0	0	0	0	0	0
G	0	1	0	1	0	1	0	0	0
H	1	0	1	0	1	0	0	0	0
I	0	0	1	0	1	0	0	0	0

Matriz de Cumplimiento de Prerrequisitos (CP_{cs}): El número de filas se da por la cantidad de cursos del pensum de estudios, mientras que el número de columnas depende del número de estudiantes para lo que se va a realizar la programación de horarios.

$$CP_{cs} = \begin{cases} 1 & \text{Si el estudiante (s) puede cursar la materia (c)} \\ 0 & \text{De lo contrario} \end{cases}$$

Ejemplo:

Matriz de Cumplimiento de Prerrequisitos CP_{cs}

Cursos	Estudiantes		
	1	2	3
A	0	0	0
B	0	0	0
C	0	0	1
D	1	0	1
E	1	0	1
F	1	0	1
G	0	1	0
H	0	1	0
I	0	1	0

Matriz de Horarios por Sección (HC_{cid}): Está matriz posee tres dimensiones, el índice “c” indica el curso a referenciar, el índice “i” indica el número de secciones existentes para cada curso, y el índice “d” indica conjuntamente el día y la hora para los cuales hay actividad académica en la jornada nocturna durante una semana.

$$HC_{cid} = \begin{cases} 1 & \text{Si para la materia “c” existe una sección “i” que tenga programada en el día y la hora “d”, una hora de clase.} \\ 0 & \text{De lo contrario} \end{cases}$$

Ejemplo: Matriz de Horarios por Sección (HC_{cid})

Curso	Dias-Horas Sección	L-6	L-7	L-8	L-9	M-6	M-7	M-8	M-9
		A	1	1	0	0	1	1	0
	2	0	0	1	1	0	0	1	1
B	1	1	0	1	1	0	0	0	0
	2	0	1	0	0	0	0	1	1

El anterior ejemplo muestra que la matriz contiene los cursos A y B. La materia A, cuenta con dos secciones en las cuales la sección 1 tiene programadas clases en los días – horas L-6, L-7, M-6 y M-7, es decir que la sección 1 de la materia A, debe ser cursada en el horario de Lunes de 6:00 a 8:00 pm y de Martes de 6:00 a 8:00 pm. Igual interpretación se da para la B.

Vector de Prioridades (Prc): Tiene tantas posiciones como materias tenga el plan de estudios, cada posición contiene la prioridad de la materia, una materia con prioridad 2, indica que es más importante que la de prioridad 1. Las materias con mayor prioridad, se definen con base al modelo propuesto por (Nahmias, 2009) para planificación de proyectos y selección de ruta crítica.

$$Pr_c = \begin{cases} 2 & \text{Si la materia tiene una alta prioridad, por ser de la ruta crítica} \\ 1 & \text{Si la materia tiene una prioridad normal, por estar fuera de la ruta crítica} \end{cases}$$

Vector de Horas por Curso (Oics): Horas semanales que se dictan para cada curso en su respectiva sección

Vector de Créditos por Curso (Ccc): Indica los créditos que posee cada curso.

Vector de Estudiantes Máximos por Sección (IMax): Indica la capacidad máxima de estudiantes que pueden quedar inscritos en una sección.

Máximo de Cursos (CMax): Número máximo de cursos que puede tomar un estudiante al semestre.

Máximo de Créditos (RMax): Número máximo de créditos que puede tomar un estudiante al semestre.

Máximo de Horas (WMax): Indica las horas máximas semanales de clase para los estudiantes

Variables

La variable para el presente modelo es:

$$X_{sci} = \begin{cases} 1 & \text{si el estudiante “s” es asignado al curso “c” en la sección “i”} \\ 0 & \text{De lo contrario} \end{cases}$$

Modelo Propuesto SSP/P

Ecuaciones:

Función Objetivo

$$Max Z = \sum_{s \in S} \sum_{c \in C} \sum_{i \in I(c)} CP_{cs} Pr_c X_{sci} \quad (1)$$

Sujeto a:

$$\sum_{c \in C} \sum_{i \in I(c)} X_{sci} \leq C^{Max} \quad \forall s = \{1 \dots S\} \quad (2)$$

$$\sum_{c \in C} \sum_{i \in I(c)} Cc_c X_{sci} \leq R^{Max} \quad \forall s = \{1 \dots S\} \quad (3)$$

$$\sum_{s \in S} X_{sci} \leq I^{Max} \quad \forall c = \{1 \dots C\} \quad \forall i = \{1 \dots I\} \quad (4)$$

$$O_{ics} - X_{sci} \geq 0 \quad \forall s = \{1 \dots S\} \quad \forall c = \{1 \dots C\} \quad \forall i = \{1 \dots I\} \quad (5)$$

$$\sum_{c \in C} \sum_{i \in I(c)} HC_{cid} X_{sci} \leq 1 \quad \forall s = \{1 \dots S\} \quad \forall d = \{1 \dots D\} \quad (6)$$

$$\sum_{c \in C} \sum_{i \in I(c)} O_{ici} X_{sci} \leq W^{Max} \quad \forall s = \{1 \dots S\} \quad (7)$$

$$X_{sci} \leq CP_{cs} \quad \forall s = \{1 \dots S\} \quad \forall c = \{1 \dots C\} \quad \forall i = \{1 \dots I\} \quad (8)$$

$$\sum_{i \in I(c)} X_{sci} \leq 1 \quad \forall s = \{1 \dots S\} \quad \forall c = \{1 \dots C\} \quad (9)$$

$$X_{sci} \in \{1,0\} \quad (10)$$

Se muestran a continuación las siguientes definiciones de la función objetivo y las restricciones.

Función Objetivo (1): Maximizar para cada estudiante, el número de materias vistas al semestre y la prioridad de las mismas.

Restricción (2): Establece que un estudiante no puede exceder un número máximo de cursos a tomar en un semestre.

Restricción (3): Establece que un estudiante no puede exceder un número máximo de créditos a tomar en un semestre.

Restricción (4): Evaluar en cada sección, el número máximo de alumnos que se pueden inscribir.

Restricción (5): Establece que cuando una sección no tiene horas asignadas para una semana, un estudiante no puede ser inscrito en dicha sección.

Restricción (6): Define que un estudiante asignado al curso c en la sección i, no puede cursar en el mismo horario otra sección del mismo curso ó uno distinto.

Restricción (7): Define la carga horaria máxima que puede cursar el estudiante en una semana.

Restricción (8): Restringe que la variable sólo tome valor "1" para los cursos en los cuales el estudiante cumple los requisitos.

Restricción (9): Define que las variables relacionadas con un mismo estudiante, sólo tome valor "1" para una sola sección de un mismo curso.

Restricción (10): Define que las variables son binarias

5. Solución del Modelo

Resultados del Modelo Principal SSP/P

La información para la ejecución del modelo, fue tomada del plan de estudios actual de la carrera de Ingeniería Industrial, el cual cuenta con 56 materias repartidas en 10 semestres académicos (Tabla 1.). Para implementar el modelo se tomaron los datos del semestre 2011-I brindados por la Universidad Central. Con base a esto, se utilizó el modelo referenciado por (Nahmias, 2009), para definir las materias que hacen parte de la ruta crítica, a partir de ahí, se obtuvieron los datos de la figura 2., asignando una prioridad de “2”, las materias de la Ruta Crítica, mientras que las materias fuera de la ruta, obtuvieron como valor de prioridad el número “1”.

Tabla 1. Materias del plan de estudios

Nombre	Abreviación	Nombre	Abreviación
Matemáticas I	M1	Práctica de Ingeniería III	PI3
Química I	Q1	Constitución Nacional	CN
Teoría de Sistemas	TS	Investigación de Operaciones I	IO1
Inglés I	I1	Estadística II	E2
Práctica de Ingeniería I	PI1	Higiene y Seguridad Industrial	HYS
Contexto I	C1	Desarrollo de Producto y Proceso	DPP
Física I	F1	Dinámica Organizacional	DO
Matemáticas II	M2	Costos y Análisis Financiero	CAF
Álgebra Lineal	AL	Investigación de Operaciones II	IO2
Algoritmia y Programación	AP	Producción I	P1
Inglés II	I2	Calidad	C
Contexto II	C2	Gestión del Talento Humano	GTH
Física II	F2	Inteligencia de Mercados	IM
Matemáticas III	M3	Práctica de Ingeniería IV	PI4
Físico-Química	FQ	Modelamiento y Simulación	MS
Ciencia de los Materiales	CM	Producción II	P2
Inglés III	I3	Gestión de Servicios	GS
Práctica de Ingeniería II	PI2	Ingeniería Económica	IE
Lógica	L	Contexto III	C3
Matemáticas IV	M4	Logística	LS
Probabilidad y Estadística	PE	Electiva de Profundización I	EP1
Procesos Industriales	PI	Sistemas Integrados de Gestión	SIG
Sistemas Naturales	SN	Proyecto de Grado	PG
Economía	E	Práctica de Ingeniería V	PI5
Estadística I	E1	Electiva de Profundización II	EP2
Ingeniería de Procesos	IP	Electiva de Profundización III	EP3
Gestión para la Innovación	GI	Electiva de Profundización IV	EP4
Microeconomía	ME	Proyecto de Grado II	PG2

Con la idea de encontrar una solución cercana a la situación real de la Universidad Central, para probar el funcionamiento del modelo SSP/P se desarrollaron un conjunto de 10 instancias en las cuales se varía el número de estudiantes conforme se incrementaba el número de cursos; así mismo, Las instancias fueron enunciadas según la tipología del problema (SSP), seguida por el número de alumnos y de materias a evaluar, la instancia SSP120056 (1.200 estudiantes, 56 materias), representa la situación real del departamento de Ingeniería Industrial, las demás instancias se desarrollan con el ánimo de medir el comportamiento general del modelo. Las materias con las que se corrió cada instancia se muestran a continuación y su tamaño en términos de número de variables y restricciones se presentan en las tablas 2 y 3. Además de los resultados obtenidos para cada instancia en términos de valores de la función objetivo y tiempos

computacionales, se evalúa un índice que permite medir el porcentaje de ocupación que presenta la sección, respecto al total de estudiantes que la pueden cursar, este elemento se nombrará como “Índice de ocupación de secciones”, por otro lado, también se propone medir el porcentaje de estudiantes que lograron de acuerdo a su programación de materias, tomar todos los cursos que tenían permitidos, dicho elemento se nombrará como “Índice de asignación total de materias” algunos de los resultados obtenidos se muestran a continuación.

Figura 2. Red de materias del programa de Ing. Industrial

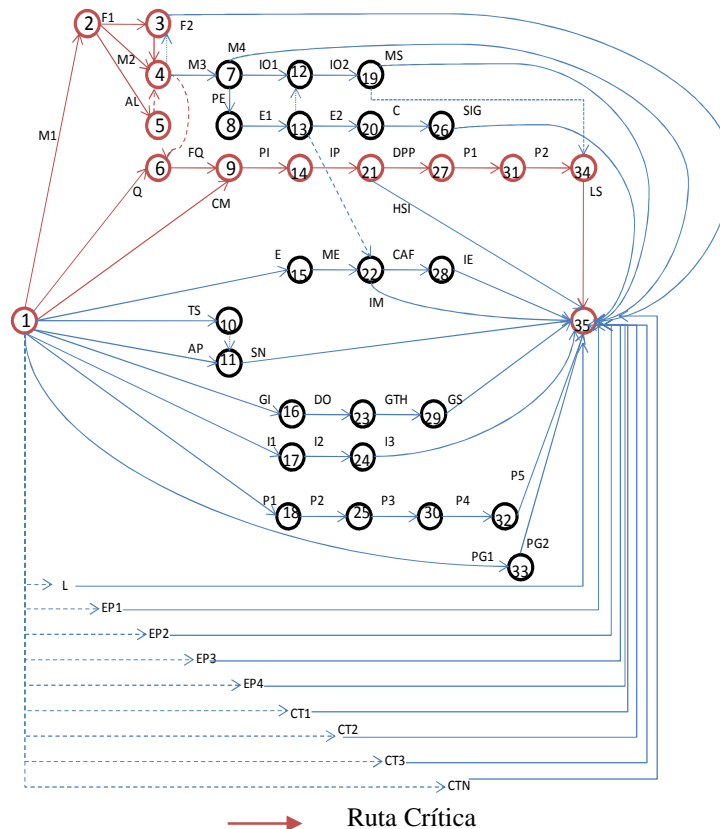


Tabla 2. Materias usadas para la ejecución de cada instancia

Instancias	Materias									
SSP20029/ SSP30029/ SSP40029	M1	Q1	TS	I1	PI1	C1	F1	M2		
	AL	AP	I2	C2	F2	M3	FQ	CM		
	PI2	L	M4	PE	PI	SN	E	E1		
	IP	ME	PI3	CN	E2					
SSP50040/SSP55040/SSP60040	M1	Q1	TS	I1	PI1	C1	F1	M2		
	AL	AP	I2	C2	F2	M3	FQ	CM		
	I3	PI2	L	M4	PE	PI	SN	E		
	E1	IP	GI	ME	PI3	CN	IO1	E2		
	HYS	DPP	DO	CAF	IO2	P1	C	GTH		
SSP65056/SSP80056	M1	Q1	TS	I1	PI1	C1	F1	M2		
SSP100056/SSP0120056*	AL	AP	I2	C2	F2	M3	FQ	CM		
	I3	PI2	L	M4	PE	PI	SN	E		
	E1	IP	GI	ME	PI3	CN	IO1	E2		
	HYS	DPP	DO	CAF	IO2	P1	C	GTH		
	IM	PI4	MS	P2	GS	IE	C3	LS		
	EP1	SIG	PG	PI5	EP2	EP3	EP4	PG2		

*Instancia real del problema

Los valores obtenidos para las instancias evaluadas se muestran a continuación:

Tabla 3. Resultados computacionales del modelo SSP/P

Instancias	Función Objetivo	Tiempo (Segundos)	Índices	
			Ocupación de secciones	Asignación total de materias
SSP20030	302	10,3	17,86%	20,00%
SSP30030	487	17,1	11,24%	21,67%
SSP40030	596	20,6	14,55%	20,50%
SSP50040	1.595	27,2	19,27%	43,60%
SSP55040	1.730	33,2	24,65%	42,55%
SSP60040	1.880	32,7	21,37%	43,83%
SSP65056	4.126	43,2	37,90%	93,38%
SSP80056	4.814	46,1	43,82%	93,50%
SSP100056	5.728	58,4	52,88%	93,40%
SSP0120056*	6.551	73,3	61,07%	93,33%

*Instancia real del problema

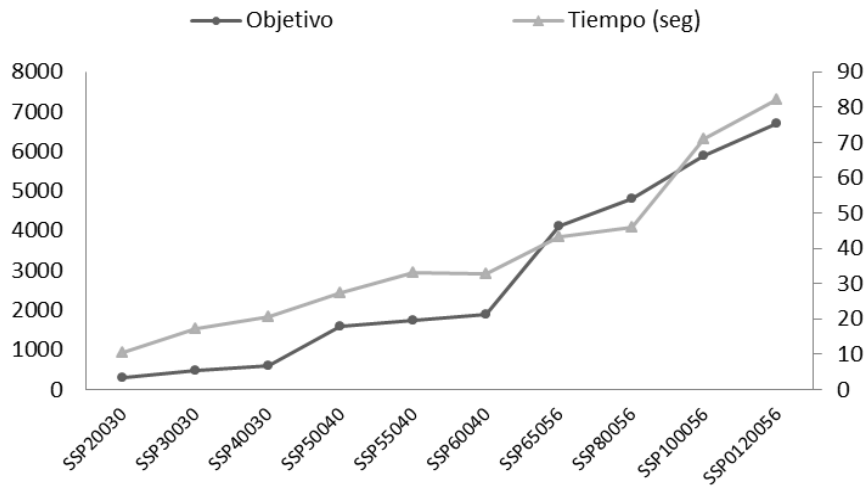


Figura 3. Resultados función objetivo y tiempo computacional SSP/P

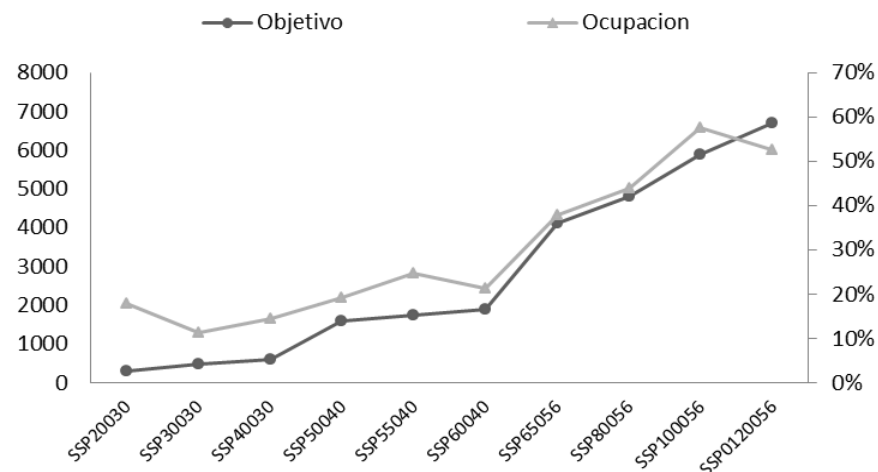


Figura 4. Resultados función objetivo e índice de ocupación SSP/P

Las instancias evaluadas se implementaron el programa de modelamiento Xpress hallando soluciones óptimas con bajos tiempos computacionales. Los mejores resultados en términos de ocupación de secciones se dieron para aquellas instancias en las cuales el número de

estudiantes y de materias programadas para ser dictadas por semestre fue mayor, así mismo, el gran número de estos dos parámetros, favoreció el índice de estudiantes con asignación total de materias, debido a que apenas con un incremento del 86% en el número de materias se logró que dicho índice se ubicara en 93.33% para un valor de 1200 estudiantes, lo anterior, indica una gran nivel de eficacia del modelo ya que en la última instancia, solo 6.67 de cada 100 alumnos no lograron encontrar cupo para todas las materias que podían cursar.

Con la idea de medir que tan sensible es el modelo al cambio prioridades en las materias de la ruta crítica, se realizaron un conjunto de pruebas haciendo variaciones en el parámetro, obteniendo los siguientes resultados:

Tabla 4. Análisis de Sensibilidad

Instancias SSP120056					
Prioridad			SSP/P		
En Ruta Crítica	Fuera de Ruta Crítica	Tiempo (Segundos)	Ocupación de secciones	Asignación de Materias	
2	1	82,1	53,5%	93,4%	
50	1	83,2	51,4%	93,4%	
100	1	80,1	50,7%	93,4%	
200	1	78,6	52,0%	93,4%	

Se evidencia que variaciones significativas en el valor de las prioridades de las materias de la ruta crítica, no generan variaciones altamente medibles en los resultados del modelo.

6. Conclusiones

Aunque de manera a priori, y con base a los resultados obtenidos en las instancias SSP100056 y SSP120056, se creía que una mayor oferta de cursos, podría lograr mayores índices de asignación de materias totales, tal vez causadas a base de menores índices de ocupación, los resultados de la instancia SSP5000280 mostraron lo contrario, esto explicado en gran parte por que los parámetros de entrada para las materias no obedecen a un análisis concienzudo de la posible oferta de cursos, sino al resultado de una configuración aleatoria de datos para el ejercicio; sin embargo, en aplicaciones prácticas, se hace necesario entender que bajos índices de ocupación de secciones y de asignación total de materias para cada estudiante, evidencia la necesidad dentro de la organización, de que los procesos de programación de clases (Courses Timetabling) obedezcan a un estudio estructurado que puede ser integrado con el modelo de programación de estudiantes propuesto por esta investigación (Student Scheduling Problem), todo a fin de lograr que este tipo de procesos administrativos sean más eficientes y optimicen la distribución de los recursos para la prestación de servicios a los estudiantes.

Partiendo de los resultados del modelo propuesto, se puede generar un conjunto de horarios iniciales para cada alumno del programa y a partir de estos, el estudiante puede ajustar su carga académica (reconfigurar su horario), según su preferencia de profesores, materias o inclusive jornadas académicas, elementos que este trabajo no contempla por ser decisión propia del alumno, y cuyos gustos y preferencias no fueron analizadas para el desarrollo de la presente investigación. En el análisis de variación de parámetros de prioridades, aunque el índice de ocupación de secciones varía levemente, los índices de asignación total de materias se mantienen constantes para todas las pruebas, lo que hace suponer que el problema atacado presenta varios óptimos posibles.

Lista de Referencias

Beligiannisa G.N., Moschopoulos C.N., Kaperonis G.P., Likothanassis S.D., Applying evolutionary computation to the school timetabling problem: The Greek case, *Computers & Operations Research* 35 (2008) 1265 – 1280

- Burke E.K., Petrovic S.**, 2002, Recent research directions in automated timetabling, *European Journal of Operational Research* 140 (2002) 266–280
- Burke, E.K. Kingston, J., Jackson, K., Weare, R.**, 1997. Automated university timetabling: The state of the art. *The Computer Journal* 40 (9), 565–571.
- Busam, V.A.**, An Algorithm for Class Scheduling With Section Preference. *Computer Sciences Corp* 10 (1967), 567-569.
- De Causmaecker P., Demeester P., Vanden Berghe G.**, A decomposed metaheuristic approach for a real-world university timetabling problem, *European Journal of Operational Research* 195 (2009) 307-318
- Carrasco M.P., Pato M.V.**, A comparison of discrete and continuous neural network approaches to solve the class/teacher timetabling problem, *European Journal of Operational Research* 153 (2004) 65–79
- Castro E;** Heurística basada en programación entera binaria para el problema de asignación de salones en la Universidad de los Andes, Bogotá, 2004.
- Carter M.W., Laporte G.**, Recent development in practical course timetabling, in: *Selected and Revised Papers of the Second International Conference on Practice and Theory of Automated Timetabling, (PATAT 1997)*, LNCS, vol. 1408, Springer, Toronto, 1998, pp. 3–19.
- Cheng E; Kruk S; Lipman M;** Flow Formulations for the Student Scheduling Problem, *International Conference on the Practice and Theory of Automated Timetabling*, 2002
- Daskalaki S., Birbas T., Housos E.**, An integer programming formulation for a case study in university timetabling, *European Journal of Operational Research* 153 (2004) 117–135
- Head C., Shaban S.**, A heuristic approach to simultaneous course/student timetabling, *Computers & Operations Research* 34 (2007) 919–933
- Parker, R.**, *Deterministic Scheduling Theory*, London : Chapman & Hall, 1995.
- Pongcharoen P., Promtet W., Yenradee P., Hicks C.**, Stochastic Optimisation Timetabling Tool for university course scheduling. *International Journal of Production Economics* 112 (2008), 903-918
- Nahmias S.**, *Production and Operations Analysis*. New York, NY : McGraw-Hill/Irwin, 2009.
- Mulvey J.M.**, A classroom/time assignment model, *European Journal of Operations Research* 9 (1982), 64 – 70
- Saldaña Crovo A., Oliva C., Pradenas Rojas L.**, Models of integer programming for an university timetabling problem, *Revista Chilena de Ingeniería*, vol. 15 N° 3, 2007, pp. 245-259
- The new wave, *Selected and Revised Papers of the First International Conference on Practice and Theory of Automated Timetabling, (PATAT 1995)*, Edinburgh, LNCS, vol. 1153, Springer, 1996, pp. 22–45.
- Van den Broek J., Hurkens Cor, W.**, Timetabling problems at the TU Eindhoven, *European Journal of Operational Research* 196 (2009) 877–885.
- Valoux C., Housos E.**, Constraint programming approach for school timetabling, *Computers & Operations Research* 30 (2003) 1555–1572
- Wren, A.**, Scheduling, timetabling and rostering – A special relationship In: *Burke and Ross (1996)* pp. 46–75. 1996.
- Zhang D., Liu Y. M’Hallah R., Leung Stephen C.H.**, A simulated annealing with a new neighborhood structure based algorithm for high school timetabling problems, *European Journal of Operational Research* 203 (2010) 550–558.