

Un modelo de programación entera para la generación de horarios universitarios: Un caso de estudio

Luis E. Urbán-Rivero^{*,1}, Myrna H. Lezama-León²,
Eduardo Cruz-Aldana², Noé D. Mares-Ortega³,
Luisa F. Loera-Díaz³

¹ Universidad Autónoma Metropolitana Azcapotzalco,
Departamento de Sistemas,
México

² Universidad Politécnica Metropolitana de Hidalgo,
México

³ Centro de Investigación en Matemáticas,
Unidad Aguascalientes,
México

leur@azc.uam.mx, {mlezama, ealdana}@upmh.edu.mx,
{noe.mares, luisa.loera}@ciamat.mx

Resumen. En este trabajo se presenta un modelo de programación entera, específicamente de programación entera binaria para la asignación de profesor a curso y horario, para un caso de estudio en una universidad mexicana siguiendo los requerimientos de operación de la institución. Se sabe que esta actividad requiere de un tiempo considerable para su desarrollo debido a distintos factores (internos como externos a la institución). En este artículo se propone un modelo en dos etapas para la generación de horarios dentro de un programa de estudios específico. Este modelo se aplicó usando datos reales de la universidad y los resultados fueron comparados con los requisitos solicitados por la universidad. Se obtuvieron horarios en menos tiempo que el implicado en una asignación manual.

Palabras clave. Generación de horarios, investigación de operaciones, programación entera.

An Integer Programming Model for University Timetable Generation: A Case Study

Abstract. This article shows an integer programming model, specially a binary programming model for teacher-course-schedule assignment, in a case study of

a Mexican university for schedule planning (timetabling), according to organizational requirements of the institution, because this activity demands a high investment of time for its development, since there are various factors (internal and external to the institution) that must be met. This article proposes a two-stage optimization model for the allocation of schedules for a specific educational program. This model is applied to a course and its results are compared with the requested requirements, obtaining the model results in less time than the obtained by manual assignment, in addition to complying with the restrictions established by the institution, as well as with the requirements of the teachers.

Keywords. Timetabling, operations research, MILP.

1. Introducción

Uno de los problemas más ampliamente estudiados en el área de la optimización combinatoria es el conocido como Timetabling (Planificación de horarios) [25, 3, 32]. En dicho problema se pretende hacer la planeación de los cursos que se imparten en una institución

educativa durante un periodo lectivo, incluyendo detalles como el horario de impartición de cada curso, el profesor que imparte el curso y el salón donde se llevara a cabo dicho curso [25, 3]. La planeación depende ampliamente de la organización que tiene cada institución educativa.

Dependiendo del nivel educativo, los problemas de organización pueden ser distintos, por ejemplo en [8, 24, 26] se muestran casos particulares de instituciones de educación media superior en Estados Unidos y Dinamarca respectivamente, mientras que en [10, 20, 4, 7, 17, 21, 1, 18, 5, 27] se muestran casos de instituciones de educación superior (IES).

Existen también distintos esquemas de organización de los cursos a impartir dependiendo de las decisiones que se toman por las instituciones educativas sobre como gestionar la demanda de los planes de estudio en dichas instituciones. La siguiente lista muestra algunas de las características consideradas en la planeación de los cursos:

1. Disponibilidad infraestructura física.
2. Disponibilidad del capital humano (profesores).
3. Preferencias y capacidad de los profesores para la participación de los cursos.
4. Subdivisión de los cursos en sesiones.
5. Horario laboral.
6. Gestión de la demanda.
7. Forma de impartir la clase.
8. Reglas laborales.
9. Preferencias de los alumnos.

Cabe destacar que los puntos del 7 al 9 pudiesen tener características de operación heterogéneas lo que complica su generalización, razón por la cuál se pueden observar múltiples trabajos en la literatura, donde cada uno aborda las características específicas de una institución en particular [3].

En el caso de la gestión de la demanda, esta depende ampliamente de las características específicas de los planes de estudio, la forma de inscribirse de los alumnos y si existen cursos comunes entre planes de estudio.

En este sentido, tanto en instituciones de educación media superior como en instituciones de educación superior estadounidenses o europeas se tienen modelos homogéneos de la gestión de la demanda, donde algunos cursos tienen claves comunes, no solo a nivel Institucional, si no incluso a nivel nacional, esto para facilitar el intercambio o la transferencia de créditos.

En cuanto a la forma de que se imparte la clase también existe heterogeneidad ya que mientras lo común es que un curso sea impartido por un solo profesor, existen instituciones donde algunos cursos como laboratorios o talleres son impartidos por hasta tres profesores de manera simultánea o diferida.

Por último, las reglas laborales son las que hacen la mayor diferencia entre casos de estudio, ya que pueden variar, entre instituciones, sistemas educativos, estados y países. Por lo que se tiene que modelar y resolver la situación particular.

Es claro que algunas universidades mexicanas tienen reglas de operación comunes o fácilmente generalizables [3], pero dados los tres aspectos heterogéneos antes mencionados es necesario adaptar los modelos a las condiciones de las IES de México.

En [10, 20, 4, 21, 1] se presentan casos de estudio en IES de México que incluyen reglas de operación específicas de Universidades Estatales con un sistema de Facultades, Universidades Privadas, Universidades Politécnicas e Institutos Tecnológicos.

El modelo que se presentará en este trabajo está basado en dichas referencias, tomando los aspectos que se tienen en común. La gestión de la demanda de los cursos es un elemento distintivo de una institución a otra. Es posible clasificar la forma en que se gestiona la demanda en IES de México en 4 diferentes formas:

- Demanda fija y no compartida.
- Demanda fija y compartida.
- Demanda flexible y no compartida.
- Demanda flexible y compartida.

La demanda fija se basa en lo que se conoce como los modelos basados en la estructura curricular como se puede observar en [5], donde se planean los cursos basándose en los distintos planes de estudio y asegurando que los alumnos regulares tengan garantizado su avance siempre que se cumplan los prerrequisitos de los cursos.

En México se da el caso que las IES más grandes imparten todos los cursos obligatorios en los planes de estudio durante todos los periodos lectivos y se asegura que al menos una vez al año se impartan los cursos optativos.

Por otro lado, universidades de un tamaño más reducido, deben limitar su oferta debido a los limitados recursos físicos y de capital humano. A continuación describiremos cada uno de los mecanismos de gestión de demanda.

Demanda fija: La demanda está determinada por el plan de estudios, los alumnos regulares están obligados a llevar los cursos en el orden que se establece en el plan de estudios. Los alumnos irregulares se adaptan a la oferta generada por los alumnos regulares. En este esquema lo común es que la IES asigne al alumno su carga académica obligatoria sin tomar en cuenta sus preferencias por algún profesor u horario.

Demanda flexible: La demanda está determinada por el plan de estudios, los alumnos deciden el orden en que llevan las materias únicamente respetando los requisitos de cada curso. En este esquema lo común es que el alumno se inscriba en los cursos que desee, en el horario y con el profesor que prefiera, sólo evitando traslapes y respetando los prerrequisitos de los cursos.

Demanda no compartida: Los cursos se imparten a los alumnos asociados a un plan de estudios en particular, es decir, si hay cursos en común con otro plan de estudios, estos no pueden tener alumnos de diferente plan de estudios.

Por ejemplo en donde los cursos podrían tener el mismo nombre pero el contenido y profundidad es diferente es necesario hacer dicha distinción como un Curso de Álgebra Lineal para Matemáticos y el mismo curso para Ingenieros, en apariencia son el mismo pero el contenido es diferente y podría provocar dificultades en los alumnos.

Demanda compartida: Los planes de estudio con cursos en común permiten que existan cursos con alumnos de diferentes planes de estudio. Por ejemplo, en una IES con sólo ingenierías se tiene como curso común Cálculo por lo que es una buena opción que un curso de Cálculo pueda aceptar a alumnos de cualquier plan de estudios de ingeniería de dicha IES.

En el caso particular de México existen IES con los cuatro formas de gestión de demanda antes mencionadas. En este trabajo nos concentraremos en un caso de estudio que presenta el caso de demanda fija y no compartida.

Utilizaremos como base el modelo presentado en [1] que corresponde con el mismo esquema de gestión demanda y con el mismo sistema educativo, es decir, es una Universidad Politécnica de México (UPM).

Algunos elementos del modelo presentado en [1] serán retomados y otros particulares del caso de estudio se agregarán en nuestro modelo. Por otro lado, se debe destacar que en lo que se refiere a problemas de generación automática de horarios, existen dos tipos de requerimientos:

Requerimientos duros: Son aquellos que forzosamente se deben cumplir para que se considere un horario factible, por ejemplo, un curso debe tener asignado un profesor.

Requerimientos blandos: Son aquellos que se intentan cumplir en medida de lo posible, por ejemplo, las preferencias de cursos de los profesores.

Lo más común para manejar los distintos requisitos de un horario, es que los requerimientos duros se modelen como restricciones, mientras que los requerimientos blandos, se modelen como parte de la función objetivo [3].

2. Características del problema

A continuación, se describen las características de la operación de las UPM:

- Los periodos lectivos, son cuatrimestrales.

- Cada plan de estudios está vinculado a un único departamento que se encarga de planear los horarios de dicho plan para cada periodo.

Por lo tanto, la asignación de horarios para curso en una UPM, se puede realizar por plan de estudios de manera independiente (Demanda no compartida).

- Cada curso es impartido por solo un profesor.
- Existen cursos especiales que se deben dar en laboratorios o talleres.
- Los salones estándar pueden tener distintas capacidades.
- Existen profesores de tiempo completo (PTC) que tienen un horario establecido y una carga académica mínima y una carga máxima.
- Existen profesores de Asignatura (PA) ellos establecen su disponibilidad y tienen una carga máxima y una carga mínima opcional.
- Los profesores dan preferencias sobre los cursos que pueden impartir basándose en su perfil.
- Un curso tiene un número de horas en las que se debe impartir en una semana.
- Un curso se divide en sesiones, las sesiones tienen una duración mínima y una duración máxima.
- Las sesiones duran horas completas, no pueden ser interrumpidas e inician y finalizan en horas enteras. Por ejemplo de 7:00-9:00 es una sesión de 2 hrs valida.
- En un día puede haber a lo mas una sesión de un curso y las horas que conforman la sesión deben ser consecutivas.
- Un grupo es un conjunto de alumnos que acude a un conjunto de cursos establecidos por la coordinación de acuerdo al plan de estudios.

3. Caso de estudio

El presente trabajo se realizó con datos de la Universidad Politécnica Metropolitana de Hidalgo (UPMH), localizada en el municipio de Toluca, en el Estado de Hidalgo, México.

La UPMH pertenece al sistema UPM por lo que la demanda es fija y no compartida junto con las características previamente mencionadas parte del sistema de las UPM.

Adicional a las características mencionadas se tienen las siguientes situaciones específicas del caso de estudio que corresponde a la UPMH, ya que dicha universidad opera bajo el modelo Bilingüe Internacional y Sustentable (BIS), donde se priorizan las actividades relacionadas a un segundo idioma y el seguimiento de alumnos mediante actividades de tutoría.

Por tanto, las características de operación específicas son las siguientes:

- La coordinación de inglés (segundo idioma) reserva los horarios en los que sus profesores impartirán clase en cada grupo. Por lo que los horarios elegidos para los cursos de inglés de cada grupo no están disponibles para el resto de los cursos del grupo.
- Los PTC están en un horario laboral de 8 horas continuas.
- Tanto los PA como los PTC acuerdan con la coordinación el número mínimo y máximo de horas frente a grupo.
- Los PTC y PA seleccionados deben tener una hora para tutoría grupal y una hora para tutoría individual con al menos 1 grupo y no mas de dos grupos.
- Los cursos del turno matutino se imparten de las 7:00 hrs a las 18:00 hrs.
- Los cursos del turno vespertino se imparten de las 12:00 hrs a las 21:00 hrs.
- El turno de cada grupo es establecido por la coordinación de acuerdo con el avance general del grupo y las actividades adicionales a realizar.

4. Metodología

Realizar este tipo de planeaciones es un tema ampliamente estudiado. En [13] se puede encontrar un resumen reciente de los avances en metodologías metaheurísticas y como estas han ayudado en algunos casos de estudio en concreto.

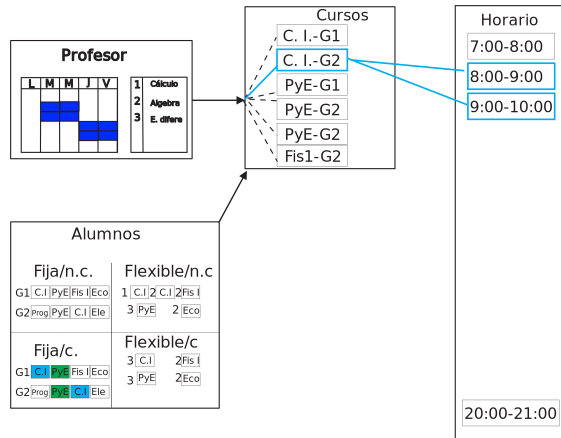


Fig. 1. Asignación de profesor-curso-horario. Se muestra cómo la demanda de cursos se genera de acuerdo a como se gestione la misma

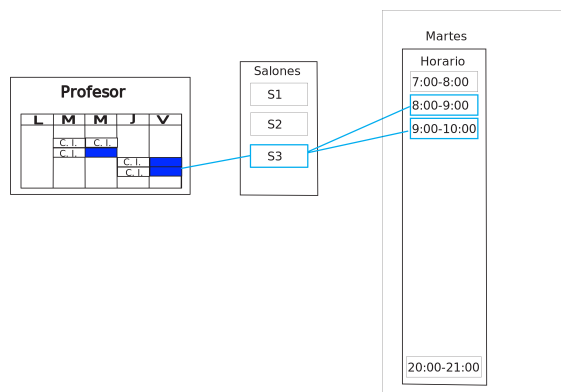


Fig. 2. Asignación de salón. En este caso el profesor ya tiene una carga horaria y se le asigna el salón de acuerdo a criterios de capacidad y preferencia

En [23, 31, 12, 19, 22], se pueden ver casos específicos, resueltos con técnicas metaheurísticas. Por último, en [29] se da un análisis profundo de como afecta en el desempeño de los modelos el dividir un problema de asignación multi-dimensional en asignaciones parciales tanto en la optimalidad como en el tiempo de ejecución.

En este trabajo se optó por la modelación programación entera y su resolución con un solver comercial, para el tamaño de las instancias, dicha opción fue suficiente. Se desarrollaron dos modelos de programación entera.

Dadas las características de la instancia se hicieron unas modificaciones al modelo propuesto en [1]. En este caso se dividió el problema en dos etapas. En la primera etapa se realizará una asignación de profesor-curso-horario como se muestra en la Figura 1. La asignación de salón se hará en una segunda etapa de manera similar a como se realiza en [28] como se observa en la Figura 2.

5. Modelo de programación binaria para la asignación profesor-curso-horario

Variables

$$x_{i,j}^{t,d} = \begin{cases} 1, & \text{Si el profesor } i \text{ es asignado al curso } j \text{ en la hora } t \text{ del día } d. \\ 0, & \text{En otro caso.} \end{cases}$$

$$y_{i,j} = \begin{cases} 1, & \text{Si el profesor } i \text{ es asignado al curso } j. \\ 0, & \text{En otro caso.} \end{cases}$$

$$z_{j,d} = \begin{cases} 1, & \text{Si el curso } j \text{ se imparte el día } d. \\ 0, & \text{En otro caso.} \end{cases}$$

$$s_{j,t,d} = \begin{cases} 1, & \text{Si el curso } j \text{ comienza a la hora } t \text{ en el día } d. \\ 0, & \text{En otro caso.} \end{cases}$$

Parámetros

- P : Conjunto de profesores.
- C : Conjunto de cursos.
- T : Conjunto de carriles horarios.
- D : Conjunto de días.
- G : Conjunto de grupos.
- $C(g)$: Cursos del grupo g , $g \in G$.
- C_{fijo} : Cursos con horario fijo, $C_{\text{fijo}} \subset C$.
- P_{TC} : Profesores de tiempo completo.
- P_A : Profesores de asignatura.
- P_{Tut} : Profesores que imparten tutorías.
- P_{ing} : Profesores de inglés.

- P_F : Profesores ficticios.
- T_{am} : Carriles horarios de las 7:00 a las 18:00 hrs.
- T_{pm} : Carriles horarios de las 12:00 a las 21:00 hrs.
- $\tau(j)$: Horas a la semana del curso j .
- P : $P_{TC} \cup P_A \cup P_{ing} \cup P_F$.
- $P_{Tut} \subseteq$: $P_{TC} \cup P_A \cup P_F$.
- $\gamma_{min}(i)$: Horas mínimas que puede impartir el profesor i .
- $\gamma_{max}(i)$: Horas máximas que puede impartir el profesor i .
- $\psi_{min}(j)$: Duración mínima de la sesión del curso j .
- $\psi_{max}(j)$: Duración máxima de la sesión del curso j .
- $$\beta_i^{t,d} = \begin{cases} 1, & \text{Si el profesor } i \text{ esta disponible} \\ & \text{en la hora } t \text{ el día } d. \\ 0, & \text{En otro caso.} \end{cases}$$
- $$H_j^{t,d} = \begin{cases} 1, & \text{Si el curso fijo } j \text{ esta programado} \\ & \text{en la hora } t \text{ el día } d. \\ 0, & \text{En otro caso.} \end{cases}$$
- $$\alpha : P \times C \rightarrow \mathbb{Z}^+, \quad (1)$$

donde:

$\alpha(i, j)$ = Preferencia del profesor i por el curso j .

Se tienen tres objetivos. El primero es asignar en medida de lo posible un horario al profesor dentro de su disponibilidad. Por lo tanto minimizamos la suma de las diferencias entre lo que solicitan los profesores y los horarios en los que se les asignan los cursos:

$$f_1 = \sum_{i \in P} \sum_{j \in C} \sum_{t \in T} \sum_{d \in D} (1 - \beta_i^{t,d}) \cdot x_{i,j}^{t,d}. \quad (2)$$

Por otro lado se busca asignarle al profesor el curso que prefiera de una lista de preferencias previamente otorgadas, donde la preferencia mas alta tiene el valor mas bajo:

$$f_2 = \sum_{i \in P} \sum_{j \in C} \alpha(i, j) \cdot y_{i,j}. \quad (3)$$

Para finalizar la función objetivo se agrega un conjunto de profesores ficticios que pueden impartir cualquier asignatura y que tienen disponibilidad completa, pero que su uso genera un costo alto. Esto último evita que el problema sea infactible, es claro que se deben agregar tantos profesores ficticios como sea necesario:

$$f_3 = \sum_{i \in P_F} \sum_{j \in C} y_{i,j}. \quad (4)$$

Con lo que la función objetivo global queda como sigue:

$$\text{mín } w = f_1 + f_2 + Mf_3, \quad (5)$$

donde M es una constante con un valor muy grande. El profesor i puede atender a lo mas un curso durante la hora t en el día d :

$$\sum_{j \in C} x_{i,j}^{t,d} \leq 1, \quad i \in P, \quad t \in T, \quad d \in D. \quad (6)$$

El profesor i debe impartir al menos $\gamma_{min}(i)$ horas a la semana y no mas de $\gamma_{max}(i)$ horas a la semana:

$$\gamma_{min}(i) \leq \sum_{j \in C} \sum_{t \in T} \sum_{d \in D} x_{i,j}^{t,d} \leq \gamma_{max}(i), \quad i \in P. \quad (7)$$

El curso j debe tener asignado un profesor:

$$\sum_{i \in P} y_{i,j} = 1, \quad j \in C. \quad (8)$$

El profesor i debe impartir todas las horas del curso j , si el es seleccionado para impartirlo:

$$\sum_{t \in T} \sum_{d \in D} x_{i,j}^{t,d} = \tau(j) \cdot y_{i,j}, \quad i \in P, \quad j \in C. \quad (9)$$

Las sesiones del curso j deben tener una duración de al menos $\psi_{\min}(j)$ hrs y no mas de $\psi_{\max}(j)$ hrs:

$$\psi_{\min}(j) \cdot z_{j,d} \leq \sum_{i \in P} \sum_{t \in T} x_{i,j}^{t,d} \leq \psi_{\max}(j) \cdot z_{j,d}, \quad (10)$$

$$j \in C, d \in D.$$

Para lograr que una sesión de un curso tenga horas consecutivas, sin tiempos muertos, se necesitan las restricciones (10) y (11) originalmente propuestas por [2] en otro contexto pero se pueden adaptar a nuestras necesidades. El curso i sólo puede tener un inicio el día d :

$$\sum_{t \in T} s_{j,t,d} \leq z_{j,d}, \quad j \in C, d \in D, \quad (11)$$

$$s_{j,t,d} \geq x_{i,j}^{t,d} - x_{i,j}^{t-1,d}, \quad (12)$$

$$i \in P, j \in C, t \in T \setminus 1, d \in D.$$

Las restricciones (13–22) corresponden con las condiciones particulares del caso de estudio de la UPMH. Un grupo g puede tener a lo mas un curso con un profesor i , es decir, un profesor no puede impartir dos o mas cursos a un mismo grupo:

$$\sum_{q \in C(g)} y_{i,q} \leq 1, \quad i \in P, g \in G. \quad (13)$$

Los cursos de un grupo g no pueden programarse de manera simultánea:

$$\sum_{q \in C(g)} x_{i,q}^{t,d} \leq 1, \quad i \in P, g \in G, t \in T, d \in D. \quad (14)$$

Los grupos del turno matutino toman clases de 7:00 a 18:00 hrs:

$$x_{i,j,t,d} \leq 0, \quad i \in P, j \in C, t \in T_{pm}, d \in D. \quad (15)$$

Los grupos del turno vespertino toman clases de 12:00 a 21:00 hrs:

$$x_{i,j,t,d} \leq 0, \quad i \in P, j \in C, t \in T_{am}, d \in D. \quad (16)$$

Los cursos no fijos de un grupo g ($C(g) \setminus C_{\text{fijo}}(g)$) no pueden programarse en las horas reservadas para los cursos fijos de ese grupo.

En nuestro caso particular los únicos cursos fijos son los cursos de inglés que son determinados previamente por la coordinación correspondiente y enviados como restricción a la coordinación de cada carrera. Los profesores de los cursos fijos son provistos por otra coordinación, en nuestro caso los representamos mediante el conjunto P_{ing} :

$$x_{i,q,t,d} \leq 0, \quad i \in P \setminus P_{\text{ing}}, q \in C(g) \setminus C_{\text{fijo}}(g), \quad (17)$$

$$(t,d) | H_j^{t,d} = 1, j \in P_{\text{ing}}, t \in T, d \in D.$$

Por otro lado, debemos asegurar que los profesores de los cursos fijos no sean asignados a cursos no fijos:

$$x_{i,q,t,d} \leq 0, \quad i \in P_{\text{ing}}, q \in C \setminus C_{\text{fijo}}, \quad (18)$$

$$(t,d) | H_j^{t,d} = 0, j \in P_{\text{ing}}, t \in T, d \in D.$$

Es necesario asegurar que los profesores de los cursos fijos (P_{ing}) se asignen a cursos fijos solamente. Esto se puede hacer dos maneras. Se puede colocar costos altos a la asignación de profesores de cursos fijos con cursos no fijos, o bien, se puede asignar directamente a los profesores de cursos fijos solamente a dichos cursos. Para asegurar la factibilidad se crean tantos profesores como cursos fijos existan.

Además de que se le da una disponibilidad total a estos profesores y que su única preferencia sean los cursos fijos dado que estos profesores son gestionados de manera externa. Un curso fijo j debe tener asignado un profesor del conjunto de profesores para cursos fijos (P_{ing}) en el horario $(t,d) | H_j^{t,d} = 1$:

$$\sum_{i \in P_{\text{ing}}} x_{i,j,t,d} = 1, \quad j \in C_{\text{fijo}}, (t,d) | H_j^{t,d} = 1. \quad (19)$$

Un profesor i de cursos fijos puede tener a lo mas un curso fijo asignado en el horario $(t,d) | H_j^{t,d} = 1$:

$$\sum_{j \in C_{\text{fijo}}} x_{i,j,t,d} \leq 1, \quad i \in P_{\text{ing}}, (t,d) | H_j^{t,d} = 1. \quad (20)$$

El grupo de profesores para dar tutoría debe dar al menos una hora de tutoría grupal y un hora

de tutoría individual con el mismo grupo y no se le pueden asignar más de dos horas de tutoría (grupal e individual) a estos profesores. Para este propósito las horas de tutoría se agregarán como un curso de tutoría grupal e individual de dos horas a la semana ($C_{TGI} \subset C \setminus C_{fijo}$) y con sesiones de duración mínima y máxima de una hora:

$$1 \leq \sum_{j \in C_{TGI}} y_{i,j} \leq 2, \quad i \in P_{Tut}. \quad (21)$$

Por último, en una hora t no puede haber más de λ cursos simultáneos:

$$\sum_{i \in P} \sum_{j \in C} x_{i,j}^{t,d} \leq \lambda \quad t \in T, \quad d \in D. \quad (22)$$

Con el modelo de programación entera (2–22) se asegura una asignación de profesores a cursos basándose en su preferencia y disponibilidad de horario. Adicional a esto el modelo incluye las reglas de operación específicas de la universidad, por lo que su solución genera un horario factible o bien que requiere cambios mínimos.

Resultado de este modelo se podrán obtener, el horario global de todos los cursos, el horario de cada profesor y los horarios de los grupos. Dado que un grupo tiene al mismo conjunto de alumnos, bastaría con hacer la asignación por grupos y no por cursos a los salones. Por lo que el modelo para asignar grupo salón horario queda como sigue:

6. Modelo de programación binaria para la asignación de curso-salón-horario

Variables

$$x_{i,j}^{t,d} = \begin{cases} 1, & \text{Si el curso } i \text{ es asignado al salón } j \\ & \text{en la hora } t \text{ del día } d. \\ 0, & \text{En otro caso.} \end{cases}$$

$$x_{i,j}^{t,d} = \begin{cases} 1, & \text{Si el curso } i \text{ es asignado al salón } j \\ & \text{en el día } d. \\ 0, & \text{En otro caso.} \end{cases}$$

$$y_{i,j} = \begin{cases} 1, & \text{Si el salón } j \text{ es usado en el día } d. \\ 0, & \text{En otro caso.} \end{cases}$$

donde:

r_d = Número de salones utilizados en el día d .

Parámetros

C : Conjunto de cursos.

S : Conjunto de salones.

G : Conjunto de grupos.

T : Conjunto de carriles horarios.

D : Conjunto de días.

$CT(t, d)$: Cursos que se traslapan en la hora t del día d .

$C(g)$: Cursos del grupo g .

$|\mathcal{H}_i^d|$: Número de horas que se imparten del curso i el día d .

$\rho(g, s)$: Factor de utilización del grupo g en el salón s .

$\phi(c, s)$: Preferencia del salón s para el curso c .

$$\mathcal{H}_i^{t,d} = \begin{cases} 1, & \text{Si el curso } i \text{ esta programado en la} \\ & \text{hora } t \text{ el día } d. \\ 0, & \text{En otro caso.} \end{cases}$$

En este caso se tienen cuatro objetivos que se mencionan a continuación:

1. Maximizar la permanencia de un grupo en un salón (\mathcal{F}_1).
2. Minimizar el número de salones utilizados por día (\mathcal{F}_2).
3. Asegurar que el salón asignado tenga el tamaño adecuado para el tamaño del grupo (\mathcal{F}_3).
4. Asegurar que se asignen los salones de acuerdo a las preferencias que da la coordinación (\mathcal{F}_4).

Tabla 1. Resultados de la asignación profesor-curso-horario

Profesor	Gurobi		CPLEX		MOSEK		HIGHS	
	P_H	P_C	P_H	P_C	P_H	P_C	P_H	P_C
1	1.00	1.00	1.00	1.00				
2	1.00	1.00	1.00	1.00				
3	1.00	1.00	1.00	1.00				
4	1.00	1.00	0.95	1.00				
5	1.00	1.00	1.00	1.00				
6	1.00	1.00	1.00	1.00				
7	1.00	1.00	0.96	1.00				
8	1.00	1.00	1.00	1.00				
9	1.00	1.00	1.00	1.00				
10	0.90	1.00	0.90	1.00	Infactible		Infactible	
11	1.00	1.00	1.00	1.00				
12	0.93	1.00	0.93	1.00				
13	1.00	1.00	1.00	1.00				
14	1.00	1.00	1.00	1.00				
15	0.92	1.00	1.00	1.00				
16	1.00	1.00	1.00	1.00				
17	1.00	1.00	1.00	1.00				
Total	16.75	17.00	16.74	17.00				
Proporción	0.99	1.00	0.98	1.00				
Tiempo (s)	266		3600		3600		3600	

donde \mathcal{F}_1 se define como a continuación:

$$\mathcal{F}_1 = \sum_{g \in G} \sum_{j \in S} \sum_{p, q \in C(g), p \neq q} y_{p,j}^d \cdot y_{q,j}^d \quad (23)$$

En (23) se busca que los cursos del mismo grupo se lleven en el mismo salón en la medida de lo posible, por lo que equivalente a minimizar los cambios de salón, es maximizar el número de cursos en un mismo salón, es necesario linealizar este objetivo, el procedimiento es sencillo y se puede encontrar en [14] o bien se puede incluir directamente en Gurobi [9]. Para el objetivo (2), se define \mathcal{F}_2 como sigue:

$$\mathcal{F}_2 = \sum_{d \in D} r_d \quad (24)$$

Para el caso del objetivo (3) es necesario definir el factor de utilización como:

$$\rho(g, s) = \frac{|g|}{|s|}, \quad (25)$$

donde $|g|$ es el número de alumnos inscritos en el grupo g , mientras que $|s|$ es el cupo máximo del salón s . Por lo que \mathcal{F}_3 queda como sigue:

$$\mathcal{F}_3 = \sum_{g \in G} \sum_{i \in C(g)} \sum_{j \in S} \sum_{d \in D} \rho(g, j) \cdot y_{i,j}^d \quad (26)$$

Por último se debe asegurar que se da preferencia a los salones específicos para los cursos que así lo necesiten, como por ejemplo los cursos que requieren un laboratorio o taller en especial:

$$\mathcal{F}_4 = \sum_{i \in C} \sum_{j \in S} \sum_{d \in D} \phi(i, j) \cdot y_{i,j}^d \quad (27)$$

Por lo tanto la función objetivo global queda como sigue:

$$\text{mín } z = -A_1 \cdot \mathcal{F}_1 + A_2 \cdot \mathcal{F}_2 + A_3 \cdot \mathcal{F}_3 + A_4 \cdot \mathcal{F}_4, \quad (28)$$

donde A_1, A_2, A_3 y A_4 son constantes positivas. En cuanto a las restricciones se tienen las que se enuncian enseguida. Un curso i debe tener asignado un salón en una hora t el día d :

$$\sum_{j \in S} x_{i,j}^{t,d} = 1, \quad i \in G, \quad t \in T, \quad d \in D. \quad (29)$$

Un salón j puede albergar a lo más un curso durante una hora t , en el día d :

$$\sum_{i \in G} x_{i,j}^{t,d} \leq 1, \quad j \in S, \quad t \in T, \quad d \in D. \quad (30)$$

Cursos que se traslapan en la hora t del día d , a lo más uno puede ocupar el salón j :

$$\sum_{i \in CT(t,d)} y_{i,j}^d \leq 1, \quad j \in S. \quad (31)$$

Se deben impartir todas las horas de una sesión del día d en el mismo salón:

$$\sum_{t \in \mathcal{H}_i^{t,d}=1} x_{i,j}^{t,d} = |\mathcal{H}_i^d| \cdot y_{i,j}^d, \quad i \in C, \quad j \in S, \quad d \in D. \quad (32)$$

Tabla 2. Resultados de la asignación de salones

Día	Gurobi		CPLEX		MOSEK		HIGHS	
	P_T	P_S	P_T	P_S	P_T	P_S	P_T	P_S
Lunes	0.83	0.85	0.83	0.87	0.83	0.85	0.83	0.85
Martes	0.90	0.98	0.90	0.98	0.90	0.98	0.90	0.98
Miércoles	0.85	0.87	0.85	0.87	0.85	0.87	0.85	0.87
Jueves	0.88	0.93	0.88	0.93	0.88	0.93	0.88	0.93
Viernes	0.87	0.95	0.87	0.95	0.87	0.95	0.87	0.95
Tiempo (s)	0.85		1.33		18.73		2.61	

En un día d no se pueden usar más de r_d salones:

$$\sum_{j \in S} w_{j,d} \leq r_d \quad d \in D. \quad (33)$$

7. Resultados

Se resolvió una instancia con datos del cuatrimestre 2022-3 de septiembre a diciembre de 2022 usando Python [30], la biblioteca PULP [15] y los solvers Gurobi 10.0 [9], CPLEX 22.1 [6], MOSEK 10.1 [16] y HIGHS 1.6.0 [11] en un equipo Con CPU Intel Core i5 a 3.40GHz con 16 GB de RAM.

La instancia consta de la programación de 75 cursos de los cuales 10 tienen horario fijo (cursos de inglés), repartidos entre 10 grupos (cinco matutinos y cinco vespertinos), tomando en cuenta la disponibilidad y preferencias de de 17 profesores, lo cuál genera un modelo con 784,354 restricciones y 183,450 variables enteras.

Para medir los resultados de dicha instancia en la etapa de de asignación profesor-curso-horario, se utilizaron los indicadores P_H que es la proporción de horario asignado respecto al horario solicitado por un profesor. El indicador P_C es la proporción de cursos asignados dentro las preferencias de un profesor.

En la Tabla 1 Se muestran los resultados de la instancia, también se muestra la suma de la proporciones para P_H y P_C , además de la proporción total de profesores y el tiempo de ejecución en segundos.

Como se puede observar se cubrió casi en su totalidad la demanda de los cursos dentro de la disponibilidad de los profesores salvo en cuatro casos donde se pudo llegar a un acuerdo con ellos o el cambio se hizo de manera manual.

En cuanto a las preferencias, a todos los profesores se les asignaron cursos dentro de su lista proporcionada y no se asigno a nadie alguna materia que estuviera fuera de sus aptitudes.

Por último, cabe destacar que en los resultados de la instancia se hizo necesario que se contratará a un profesor adicional para impartir dos cursos que no pudieron ser asignados a ningún profesor de la plantilla laboral actual.

Cabe destacar que los solvers MOSEK y HIGHS no pudieron encontrar una solución factible en un tiempo máximo de 3600 segundos. Por otro lado, en la segunda etapa.

La instancia cuenta con 75 cursos con horario y profesor establecido que debieron programarse en siete salones y dos laboratorios. El modelo resuelto de esta segunda etapa se resolvió con las mismas herramientas que en la primera etapa, el modelo resultante cuenta con 74,540 restricciones y 72,726 variables enteras.

En la Tabla 2. Se muestran por día los indicadores P_T que representa la proporción de cursos que fueron asignados al salón del tamaño apropiado, tomando en cuenta que cada curso está asociado a un grupo y dicho grupo tiene una cantidad de alumnos establecida. En este caso se dejaron fuera los cursos asignados a los laboratorios ya que se tiene que asignar el curso a dicho laboratorio a pesar de que el tamaño no sea el adecuado.

Por otro lado P_S representa la proporción de cursos que fueron asignados a salones dentro de su lista de preferencias. En este caso los valores de las A_k , $k = 1, 2, 3, 4$ se ajustaron de manera experimental siguiendo las necesidades del tomador de decisiones. Los archivos con las instancias se pueden encontrar en la página ¹ en formato csv. Cabe destacar que todos los solvers pudieron resolver a optimalidad la instancia pero en tiempo distintos.

¹github.com/lurbanrivero/tt_upmh

8. Conclusiones y trabajo futuro

Se logró construir dos modelos de programación lineal entera que son capaces de generar horarios escolares en el caso particular de la UPMH. Cabe destacar que aunque se resolvió un caso particular, algunas características son generalizables a todo el sistema de UPM y sólo se requieren ajustes específicos para el caso en particular. El proceso de manera normal se llevaba a cabo de manera manual por la coordinación de cada carrera. Dicho proceso es susceptible a errores humanos e inconformidades, además de que requiere de dos a tres semanas de trabajo para consolidar el horario.

Con los horarios generados por esta metodología se tiene una propuesta que requiere ajustes mínimos y permite consolidar una propuesta de horario en menos de una semana.

Esto último ya sin la generación manual. Cabe resaltar que existe una limitación para esta metodología que es que no permite cambios puntuales y mantener toda la demás asignación, de hacer esto último se podría generar un horario completamente diferente.

La metodología generada en este trabajo está limitada a un esquema de demanda. Una dirección de investigación sería explorar otro modelo de demanda, generar un modelo y aplicarlo en un caso de estudio donde se puedan mostrar las ventajas de generar horarios de manera automatizada y adecuados a cada caso.

Referencias

1. **Arratia-Martinez, N. M., Maya-Padron, C., Avila-Torres, P. A. (2021).** University course timetabling problem with professor assignment. *Mathematical Problems in Engineering*, Vol. 2021, pp. 1–9. DOI: 10.1155/2021/6617177.
2. **Bixby, E. R., Felon, M., Gu, Z., Rothberg, E., Wunderling, R. (2000).** MIP: Theory and practice — closing the gap. *International Federation of Information Processing. Advances in Information and Communication Technology*, pp. 19–49. DOI: 10.1007/978-0-387-35514-6.2.
3. **Chen, M. C., Sze, S. N., Goh, S. L., Sabar, N. R., Kendall, G. (2021).** A survey of university course timetabling problem: Perspectives, trends and opportunities. Vol. 9, pp. 106515–106529. DOI: 10.1109/access.2021.3100613.
4. **Chávez-Bosquez, O., Hernández-Torruco, J., Hernández-Ocaña, B., Canul-Reich, J. (2020).** Modeling and solving a Latin American university course timetabling problem instance. *Mathematics*, Vol. 8, No. 10, pp. 1833. DOI: 10.3390/math8101833.
5. **Colajanni, G., Daniele, P. (2020).** A new model for curriculum-based university course timetabling. *Optimization Letters*, Vol. 15, No. 5, pp. 1601–1616. DOI: 10.1007/s11590-020-01588-x.
6. **CPLEX, IBM ILOG (2022).** V22. 1: User's manual for CPLEX. *International Business Machines Corporation*, Vol. 46, No. 53, pp. 157.
7. **Cruz-Chávez, M. A., Flores-Pichardo, M., Martínez-Oropeza, A., Moreno-Bernal, P., Cruz-Rosales, M. H. (2016).** Solving a real constraint satisfaction model for the university course timetabling problem: A case study. *Mathematical Problems in Engineering*, Vol. 2016, pp. 1–14. DOI: 10.1155/2016/7194864.
8. **Dorneles, A. P., de Araújo, O. C. B., Buriol, L. S. (2014).** A fix-and-optimize heuristic for the high school timetabling problem. *Computers and Operations Research*, Vol. 52, pp. 29–38. DOI: 10.1016/j.cor.2014.06.023.
9. **Gurobi Optimization, LLC (2023).** Gurobi optimizer reference manual.
10. **Hernández-Vázquez, J. I., Hernández-González, S., Baltazar-Flores, M. R., Jiménez-García, J. A., Hernández-Vázquez, J. O. (2020).** Programación matemática binaria por etapas en la elaboración de un horario universitario. *Entreciencias: Diálogos en la Sociedad del Conocimiento*, Vol. 8, No. 22. DOI: 10.22201/enesl.20078064e.2020.22.70018.

11. **Huangfu, Q., Hall, J. A. J. (2017).** Parallelizing the dual revised simplex method. *Mathematical Programming Computation*, Vol. 10, No. 1, pp. 119–142. DOI: 10.1007/s12532-017-0130-5.
12. **Iqbal, Z., Ilyas, R., Chan, H. Y., Ahmed, N. (2021).** Effective solution of university course timetabling using particle swarm optimizer based hyper heuristic approach. *Baghdad Science Journal*, Vol. 18, No. 4, pp. 1465. DOI: 10.21123/bsj.2021.18.4(suppl.).1465.
13. **Kaur, M., Saini, S. (2020).** A review of metaheuristic techniques for solving university course timetabling problem. *Lecture Notes in Networks and Systems*, pp. 19–25. DOI: 10.1007/978-981-15-5421-6.3.
14. **MirHassani, S. A., Hooshmand, F. (2019).** *Methods and models in mathematical programming*. Springer. DOI: 10.1007/978-3-030-27045-2.
15. **Mitchell, S., O’Sullivan, M., Dunning, I. (2011).** PuLP: A linear programming toolkit for python. *The University of Auckland, Auckland, New Zealand*, Vol. 65.
16. **MOSEK ApS (2023).** *The mosek optimization toolbox for MATLAB manual*. Version 10.1.
17. **Prabodanie, R. A. R. (2017).** An integer programming model for a complex university timetabling problem: A case study. *Industrial Engineering and Management Systems*, Vol. 16, No. 1, pp. 141–153. DOI: 10.7232/iems.2017.16.1.141.
18. **Pratiwi, M., Rosyidi, C. N., Yuniaristanto (2021).** An optimization model for course scheduling in undergraduate industrial engineering program of universitas sebelas maret. *Institute of Physics Conference Series: Materials Science and Engineering*, Vol. 1072, No. 1, pp. 012008. DOI: 10.1088/1757-899x/1072/1/012008.
19. **Sakaliuk, O., Trishyn, F. (2021).** Using a genetic algorithm to solve the courses timetabling creation problem. *Automation of Technological and Business Processes*, Vol. 13, No. 2, pp. 22–28. DOI: 10.15673/atbp.v13i2.2053.
20. **Sánchez-Partida, D., Baquela, E. G., Mora-Vargas, J., Smith, N. R. (2016).** Case study: Simulated annealing for improving the educational timetable. *Nova Scientia*, Vol. 8, No. 17, pp. 340–360.
21. **Sánchez-Partida, D., Martínez-Flores, J. L., Olivares-Benítez, E. (2014).** An integer linear programming model for a university timetabling problem considering time windows and consecutive periods. *Journal of Applied Operational Research*, Vol. 6, No. 3, pp. 158–173.
22. **Song, T., Liu, S., Tang, X., Peng, X., Chen, M. (2018).** An iterated local search algorithm for the university course timetabling problem. *Applied Soft Computing*, Vol. 68, pp. 597–608. DOI: 10.1016/j.asoc.2018.04.034.
23. **Sylejmani, K., Gashi, E., Ymeri, A. (2022).** Simulated annealing with penalization for university course timetabling. *Journal of Scheduling*, Vol. 26, No. 5, pp. 497–517. DOI: 10.1007/s10951-022-00747-5.
24. **Sørensen, M., Dahms, F. H. W. (2014).** A two-stage decomposition of high school timetabling applied to cases in Denmark. *Computers and Operations Research*, Vol. 43, pp. 36–49. DOI: 10.1016/j.cor.2013.08.025.
25. **Tan, J. S., Goh, S. L., Kendall, G., Sabar, N. R. (2021).** A survey of the state-of-the-art of optimisation methodologies in school timetabling problems. *Expert Systems with Applications*, Vol. 165, pp. 113943. DOI: 10.1016/j.eswa.2020.113943.
26. **Tassopoulos, I. X., Iliopoulou, C. A., Beligiannis, G. N. (2019).** Solving the greek school timetabling problem by a mixed integer programming model. *Journal of the Operational Research Society*, Vol. 71, No. 1, pp. 117–132. DOI: 10.1080/01605682.2018.1557022.
27. **Tavakoli, M. M., Shirouyehzad, H., Lotfi, F. H., Najafi, S. E. (2020).** Proposing

- a novel heuristic algorithm for university course timetabling problem with the quality of courses rendered approach: A case study. *Alexandria Engineering Journal*, Vol. 59, No. 5, pp. 3355–3367. DOI: 10.1016/j.aej.2020.05.004.
28. **Urbán-Rivero, L. E., Benítez-Escárcega, M. R., Alvarez, J. V. (2020).** An integer linear programming model for a case study in classroom assignment problem. *Computación y Sistemas*, Vol. 24, No. 1, pp. 97–104. DOI: 10.13053/cys-24-1-3191.
29. **Valencia, C. E., Alfaro, C. A., Zaragoza-Martinez, F. J., Vargas-Magaña, M. C., Perez-Perez, S. L. (2018).** Outperforming several heuristics for the multidimensional assignment problem. 15th International Conference on Electrical Engineering, Computing Science and Automatic Control (CCE), pp. 1–6. DOI: 10.1109/ICEEE.2018.8533978.
30. **Van-Rossum, G., Drake, F. L. (2009).** Python 3 reference manual.
31. **Yusoff, M., Roslan, N. (2019).** Evaluation of genetic algorithm and hybrid genetic algorithm-hill climbing with elitist for lecturer university timetabling problem. *Advances in Swarm Intelligence*, pp. 363–373. DOI: 10.1007/978-3-030-26369-0_34.
32. **Zhu, K., Li, L. D., Li, M. (2021).** A survey of computational intelligence in educational timetabling. *International Journal of Machine Learning and Computing*, Vol. 11, No. 1, pp. 40–47. DOI: 10.18178/ijmlc.2021.11.1.1012.

Article received on 11/05/2023; accepted on 06/11/2023.

** Corresponding author is Luis E. Urbán-Rivero.*