Problema de asignación de aulas en universidades: un enfoque basado en programación de metas

Serena Ruiz Díaz[1], and Marian G. Marcovecchio[1,2]

[1] Facultad de Ingeniería Química, Universidad Nacional del Litoral. Santa Fe, Argentina
[2] INGAR, Instituto de Desarrollo y Diseño, CONICET-UTN. Santa Fe, Argentina
sere.ruizdiaz@gmail.com; mariangm@santafe-conicet.gov.ar

Resumen. En este trabajo se aborda el problema de asignación de aulas en universidades, que consiste en un problema particular de asignación de recursos limitados persiguiendo un cierto objetivo. El problema planteado por día puede formularse como: dado un conjunto de clases con sus respectivos horarios, cantidad de alumnos y requerimientos específicos, el objetivo es asignar un aula con capacidad suficiente a cada clase, satisfaciendo ciertos criterios de calidad para la asignación. Formulado como un problema de programación matemática, en muchos casos este problema resulta infactible. En consecuencia, se propone un abordaje basado en programación de metas, dividiendo las restricciones en restricciones duras y restricciones blandas. Se resuelve aplicando el método preventivo, a través de dos problemas de programación matemática del tipo mixto entero lineal. Se resuelven dos casos reales de aplicación, que permiten ilustrar la eficiencia de la metodología propuesta.

Palabras claves: optimización, problema de asignación, programación de metas

1 Introducción

La programación de horarios y la asignación de recursos limitados para satisfacer cierto objetivo son problemas que surgen en contextos muy variados y pueden plantearse como modelos de programación matemática. Si bien estos problemas tienen un planteo matemático general, cada aplicación presenta sus propias particularidades que deben ser modeladas matemáticamente de forma precisa. En este trabajo se aborda el problema de asignación de aulas en universidades, que consiste en una aplicación particular de estos problemas.

El problema abordado es parte del problema conocido en inglés como "University Course Timetabling Problem" (UCTP). En su formulación general, el problema consiste en la asignación de clases en aulas para intervalos de tiempo dados durante un semestre, considerando un conjunto de restricciones de modo de impedir conflictos. En el problema más amplio, se determinan tanto los horarios como las aulas asignadas a cada clase. Las restricciones del problema se dividen en restricciones duras que deben cumplirse obligatoriamente y restricciones blandas que describen preferencias y deben satisfacerse tanto como sea posible. Estos problemas son altamente combinatorios, clasificados como “NP-hard” y por lo tanto su resolución a optimalidad puede ser dificultosa [1]. Es por eso que se han propuesto numerosos métodos de resolución, en su mayoría heurísticos, para proporcionar soluciones en un tiempo de cómputo eficiente. A continuación, se comentan algunos trabajos representativos presentes en la literatura que abordan este problema.

En [2] se aborda la resolución del UCTP a través de una implementación particular de un algoritmo genético. Los autores emplean tres operadores genéticos: cruce, mutación y selección. Luego realizan una simulación para testear la metodología propuesta.

En [3] se propone un algoritmo de búsqueda local iterativo para encontrar una solución factible al problema general de UCTP. El algoritmo cuenta con tres fases: inicialización, intensificación y diversificación. En el enfoque propuesto se construye un cronograma inicial parcialmente factible, luego se realiza una búsqueda local basada en recocido simulado y finalmente se implementa un procedimiento de perturbación que busca mejorar la solución. Este procedimiento se repite de forma iterativa hasta alcanzar una condición de parada deseada. El algoritmo es testado en un conjunto amplio de datos.

En [4] también se aborda el problema de encontrar una solución factible para el UCTP. Para esto, los autores proponen un algoritmo basado en Búsqueda Tabú, que integra la estrategia de aleatoriedad controlada y el mecanismo de umbral. Testean la metodología de búsqueda propuesta en un conjunto de 60 instancias, y el algoritmo puede encontrar soluciones factibles para casi todos los casos estudiados.

En [5] plantean que el UCTP no debe resolverse cada año o semestre desde cero, sino que la programación debe basarse en la hecha para semestres o años anteriores. Para ello estudian el Problema de la Perturbación Mínima. Esto es, dado un conjunto de modificaciones que hacen que un cronograma ya no sea factible, se busca la solución factible que sea la más cercana respecto al cronograma inicial. Los autores proponen dos modelos diferentes de programación matemática entera. Para testear el abordaje propuesto, generan disrupciones de forma aleatoria a un cronograma original, considerando datos históricos del Instituto Superior Técnico, de la Universidade de Lisboa.

En [6] los autores presentan un modelo del tipo mixto entero para resolver el problema de horarios universitarios y asignación de aulas. La contribución del trabajo se basa en un estudio que permite reducir el número de variables y restricciones para conseguir un mejor rendimiento computacional en la resolución del problema. El método propuesto busca en primer lugar una solución factible inicial, luego se implementa un procedimiento de búsqueda local para iterativamente mejorar el valor de la solución, a través de la resolución de problemas de programación matemática mixtos enteros.

En [7] se presenta una metaheurística con procesamiento distribuido para encontrar soluciones al UCTP. La técnica aplica las llamadas comunicación colectiva y comunicación punto a punto, que permiten la cooperación entre los distintos procesos. La metaheurística propuesta está basada en recocido simulado, y la comunicación colectiva comunica al proceso maestro cuando se obtiene una mejor solución, mientras que la comunicación punto a punto transmite esa información a los subprocesos.

Se pueden mencionar algunos trabajos que resuelven en particular el problema de asignación de aulas. En [8] se introduce una formulación basada en patrones para resolver el problema de asignación de aulas, que generaliza modelos previos de planificación con y sin intervalos. Con esta formulación, los autores resuelven el problema de asignación de aulas para la Universidad de Auckland, Nueva Zelanda, considerando varias medidas de calidad de la solución. Además, identifican situaciones en las que pueden surgir soluciones fraccionarias en la relajación LP, lo que dificulta la resolución del problema con variables enteras.

En [9] se desarrolla un sistema completo que permite asignar las aulas en la Universidad Nacional del Centro de la Provincia de Buenos Aires. Este trabajo, constituye el trabajo final de carrera del autor. En el mismo se desarrolla una metodología y se muestra la implementación práctica como sistema que asiste al personal encargado de la distribución de aulas en dicha Universidad.

En [10] se estudia el problema de asignación de aulas en universidades con la preferencia adicional de que clases que tienen lugar distintos días y corresponden a una misma materia sean asignadas a una misma aula. Para abordar este problema, los autores proponen una heurística y comparan su eficiencia con la resolución exacta del problema.

Un resumen sobre los distintos enfoques propuestos para el estudio de la programación de cursos universitarios puede encontrarse en [11]. El trabajo incluye aportes desde la investigación operativa, métodos metaheurísticos y enfoques basados en sistemas multiagentes distribuidos. Además, se reportan datos confiables que podrían utilizarse para probar y evaluar nuevos enfoques.

En la práctica, la programación de horarios de cursado y asignación de aulas es un problema al que se enfrentan todas las instituciones educativas de nivel secundario, terciario o universitario, semestral o anualmente. El planteo del problema para los distintos niveles educativos tiene muchas similitudes y algunas particularidades. Considerando las instituciones universitarias, en la práctica la programación de horarios y asignación de aulas usualmente no se resuelve de forma conjunta, sino que se determina en dos niveles de decisiones. En una primera instancia se determinan los horarios de cada clase, y esto se hace de forma coordinada entre los docentes de materias correspondientes a un mismo año académico de cada carrera, para evitar superposiciones y convenir horarios de cursado conveniente para los alumnos. En una segunda instancia, se asignan aulas a cada clase de todos los años académicos de todas las carreras impartidas por la institución, considerando las características de las aulas disponibles y los requerimientos de las distintas clases. Esta descomposición es conocida como: "tiempos primero, aulas segundo" y, según el conocimiento de las autoras de este trabajo, es la más usada en la práctica en las instituciones universitarias, ya que se debe considerar la disponibilidad de horarios de los docentes que usualmente tienen obligaciones horarias en otras instituciones. Por esta razón, en la mayoría de las instituciones universitarias, el problema de asignación de aulas es la única instancia de toma de decisiones en la programación de horarios de cursado semestral que podría ser asistido por métodos computacionales.

En ese trabajo, se aborda el problema de asignación de aulas para el cursado semanal de un semestre en una institución universitaria.

Este trabajo surge en el ámbito de la Facultad de Ingeniería Química (FIQ) de la Universidad Nacional del Litoral (UNL), Santa Fe. En esta institución, la distribución de aulas, lleva un tiempo realizándose a partir de consideraciones y cronogramas pasados, muchas veces ineficientes o incluso erróneos, que ocasionan imprevistos en el desempeño diario del cronograma universitario. En este contexto, el trabajo surge como una forma de proporcionar una metodología eficiente para la tarea de distribución y asignación de aulas en la planificación semestral de la facultad. El objetivo es proveer de una herramienta que optimice esta tarea, brindando soluciones confiables en un tiempo de cálculo competente, que permitan un aprovechamiento óptimo de los recursos disponibles de la institución.

La propone una metodología basada en programación de metas y se resuelven dos casos de estudios reales para la asignación de aulas de un semestre en la FIQ-UNL.

El trabajo está organizado como sigue. En la sección 2 se describe el problema y su formulación matemática. En la sección 3 se presenta la metodología propuesta para resolver el problema de asignación de aulas para el cursado semestral de una institución educativa. En la sección 4 se presentan los casos de estudios resueltos con el abordaje propuesto. Finalmente, en la sección 5 se describen las conclusiones generales y los trabajos futuros que se desprenden del presente.

2 Descripción y formulación del problema

El problema abordado para la programación de horarios de cursado y asignación de aulas para un semestre dado puede formularse como un problema de asignación. En efecto el planteo del problema es particularmente sencillo, aunque las características combinatorias del conjunto de soluciones hacen que su resolución para instancias reales pueda resultar compleja y computacionalmente demandante.

El planteo del problema puede resumirse como: dado un conjunto de clases a dictarse en días y horarios dados, cada una con una cantidad dada de alumnos y con ciertos requerimientos específicos sobre las características del aula en donde dictarse; y dado un conjunto de aulas, cada una con una capacidad máxima conocida y con características dadas sobre equipamiento y recursos disponibles; se desea asignar a cada clase un aula disponible que satisfaga sus requerimientos específicos y que tenga capacidad suficiente para sus alumnos. En general, si no hay otros requerimientos especiales, este problema se resuelve para cada día de la semana de forma independiente.

Al planteo básico del problema, suelen incorporarse requerimientos particulares. Para algunas instituciones es deseable que clases correspondientes a una misma materia que son dictadas en días diferentes, sean asignadas a una misma aula. Algunas programaciones intentan que materias correspondientes a un mismo año académico sean asignadas a la menor cantidad posible de aulas diferentes, de manera que cada alumno no tenga gran variación en las aulas en la que debe tomar clases.

Además, un problema similar se plantea para la asignación de aulas para exámenes (Examination Timetabling Problems: ETP). Para este problema, dadas las limitaciones en cantidad de aulas y en el caso de exámenes escritos, muchas instituciones permiten asignar más de una materia a una misma aula.

Tanto en el problema de asignación de aulas para el cursado de un semestre como en el de asignación de aulas para exámenes, hay restricciones que deben satisfacerse obligatoriamente. Por ejemplo, no puede asignarse más de una clase a una misma aula en un mismo horario; no puede asignarse una clase a un aula con capacidad insuficiente para su número de alumnos. Estas restricciones son denominadas ‘restricciones duras’. Por otro lado, hay restricciones que deben satisfacerse tanto como sea posible. Por ejemplo, que clases correspondientes a una misma materia dictadas en distintos días sean asignadas a una misma aula. Estas restricciones son denominadas ‘restricciones blandas’.

En el presente trabajo, y considerando los casos reales resueltos, se asume que no hay superposición en los horarios de clases correspondientes a un mismo año académico.

En la siguiente sección de plantea la formulación matemática general adoptada para resolver el problema de asignación de aulas de cursado en un semestre de una institución educativa. Mientras que en la sección 3 que comentarán las restricciones adicionales consideradas para los casos particulares resueltos.

2.1 Formulación matemática

Las restricciones de la formulación matemática general, como problema de programación matemática para el problema de asignación de aulas para cursado en una institución educativa es presentada en esta sección. La asignación de aulas se programa por cada día de semana, de lunes a viernes, y se aplica a un semestre de cursado. La siguiente formulación corresponde a un día de cursado.

A continuación, se describen los conjuntos, índices, parámetros, variables y restricciones empleados en la definición del modelo matemático:

- Conjuntos:

  H: Conjunto de bloques horarios en un día de cursado

  C: Conjunto de clases a dictarse en un día de cursado

  A: Conjunto de aulas disponibles

- Índices:

  h: bloques horarios, $h \in H$

  c: clases a dictarse, $c \in C$

  a: aulas, $a \in A$

- Parámetros:

  CA_a: Capacidad máxima de alumnos del aula a, $a \in A$

  CC_c: Cantidad de alumnos que cursan la clase c, $c \in C$

- Subconjuntos:

  $A_c = \{a \in A: CA_a \geq CC_c\}$ conjunto de aulas factibles para la clase c, $\forall c \in C$

  $H_c = \{h \in H: c\ debe\ dictarse\ en\ h\}$ conjunto de bloques horarios en los que debe dictarse la clase c, $\forall c \in C$

- Variables:

  $x_{a,c}$: variable binaria que asume valor 1 si la clase c es asignada al aula a, $\forall (a,c) \in A_c \times C$

- Restricciones:

$$\sum_{a \in A_c} x_{a,c} = 1 \qquad \forall c \in C \tag{1}$$

$$\sum_{c/(h \in H_c \wedge a \in A_c)} x_{a,c} \leq 1 \qquad \forall h \in H, \forall a \in A \tag{2}$$

Por tratarse de un problema cuya mayor complejidad surge de la naturaleza combinatoria del conjunto de soluciones, se redujo al mínimo la cantidad de variables binarias definidas para el problema. Para esto se define un conjunto (A_c) para cada clase, que incluye sólo las aulas con capacidad mayor o igual a la cantidad de alumnos de la clase, y que además cumplan con los requerimientos específicos para su dictado. De esta manera, la condición que establece que el aula asignada a una clase debe tener capacidad suficiente no es incluida como una restricción, sino que limita las variables binarias a definir, reduciendo su número.

La restricción (1) impone que se asigne un aula para cada clase. Mientras que la restricción (2) previene de asignar un aula a más de una clase en cada bloque horario.

Las restricciones (1) y (2) son las únicas restricciones a satisfacer para determinar la asignación de aulas para el cursado de un día en una semana típica de un semestre. Si no hay otros requerimientos o disposiciones a satisfacer, el problema se reduce a encontrar valores de las variables binarias que satisfagan el conjunto de restricciones (1) y (2). Sin embargo, las instituciones tienen ciertas preferencias respecto a la distribución de aulas que pueden considerarse. Por ejemplo, podría buscarse soluciones que minimicen el exceso de capacidad en las aulas asignadas, considerando que las aulas que queden libres tengan la mayor capacidad posible para atender los pedidos de aulas particulares que puedan surgir por fuera del cursado fijo. En otros casos se prefiere el uso de ciertas aulas por sobre otras, siempre que sea posible. Los requerimientos extras pueden en algunos casos incorporarse como restricciones, y en otros, como funciones objetivo a optimizar.

Sin embargo, en muchos casos el sistema de ecuaciones (1) y (2) no tiene solución. Ese fue el caso para la programación de clases proporcionada por la institución objeto de estudio. Por lo tanto, se propuso abordar el problema a través de la programación de metas, que intenta satisfacer tanto como sea posible las restricciones que no pueden satisfacerse completamente. La metodología propuesta es explicada en la siguiente sección.

3 Metodología

Se propone abordar la asignación de aulas para la programación del cursado de una institución educativa a través de la programación de metas. Para esto, se distingue entre restricciones duras que deben satisfacerse obligatoriamente y restricciones blandas que serán satisfechas tanto como sea posible.

Como restricciones duras se consideran las restricciones (2), que impiden que un aula sea asignada a dos clases en un mismo bloque horario.

La restricción (1) que busca asignar un aula a cada clase será tratada como restricción blanda, ya que en algunos casos no es posible satisfacerla.

Por lo tanto, las restricciones duras del problema serán la restricción (2) y la restricción (1) en forma de desigualdad, de manera de impedir que se asigne más de un aula a una misma clase:

$$\sum_{a \in A_c} x_{a,c} \leq 1 \qquad \forall c \in C \tag{3}$$

Las restricciones blandas son tratadas como metas. En el problema resuelto, se determinaron las metas y sus prioridades a partir de los intereses planteados por la gestión de la FIQ - UNL. La prioridad que fue planteada por la gestión fue asignar aulas a la mayor cantidad posible de clases. Por lo tanto, se incorporó la siguiente restricción blanda:

$$\sum_{c \in C} \sum_{a \in A_c} x_{a,c} + z_1 = N \tag{4}$$

donde N es el total de clases a dictarse en el día. La variable no negativa z_1 indica la cantidad de clases para las cuales no fue posible asignar un aula. De forma que la meta 1 a minimizar será la variable z_1.

En segunda instancia, se busca minimizar la cantidad total de alumnos en las clases que no tienen asignada un aula. Para ello se incorpora la restricción:

$$z_2 = \sum_{c \in C} \left(CC_c \left(1 - \sum_{a \in A_c} x_{a,c}\right)\right) \tag{5}$$

La variable no negativa z_2 será la meta 2 a minimizar, e indica la cantidad total de alumnos en clases sin aula asignada. Esta meta fue formulada a partir de los intereses prácticos formulados por la gestión de la FIQ-UNL. En general, en el caso de no contar con aulas suficientes para las clases a impartir en cierto horario, es preferible solicitar un cambio de horarios a asignaturas que cuenten con menos alumnos, ya que resultará menos dificultoso encontrar un horario con disponibilidad de docentes, alumnos y aulas.

Existen diversas metodologías para resolver problemas con múltiples metas. En el abordaje propuesto en este trabajo, que considera dos metas, se optó por el método preventivo ya que es el que mejor ajusta y da respuesta a las inquietudes planteadas desde la gestión de la FIQ-UNL y del personal encargado de la distribución de aulas. Además, las dos metas consideradas cuantifican distintas medidas de eficiencia, que no serían comparables de forma apropiada en una única función objetivo que pondere ambas metas.

Como prioridad se optimizó la meta 1; y una vez obtenida una solución óptima para esta meta, se fijó su valor y se resolvió el problema que optimiza la meta 2.

De esta manera se formuló el primer problema para la meta 1:

P1: $$minimizar \quad z_1$$

Sujeto a: ecuaciones (**2**)-(**5**)

Este problema de programación matemática es del tipo Mixto Entero Lineal: MIP, es siempre factible y busca la solución que consiga asignar aulas a la mayor cantidad de clases posibles. Por lo tanto, su valor objetivo óptimo: z_1^{opt} indica la cantidad de clases a las que no es posible asignarles aulas en el día. Este problema, por lo general, tendrá soluciones óptimas alternativas. Entonces, si la solución óptima indica que hay una cantidad positiva de clases en esa condición, habrá distintas alternativas para determinar la distribución de aulas, dejando z_1^{opt} clases sin aula asignada. Entre esas alternativas, se buscará la que minimice la meta 2.

Por lo tanto, el segundo paso de la resolución propuesta consiste en fijar la variable z_1 en el valor obtenido en el problema P1, y resolver el siguiente problema que minimiza la meta 2:

P2: $$minimizar \quad z_2$$

Sujeto a: ecuaciones (**2**)-(**5**)

$$z_1 = z_1^{opt}$$

El problema 2 también es un problema del tipo MIP y es siempre factible. La solución óptima del problema P2 indicará una distribución de aulas que deje la menor cantidad posible de clases sin aula asignada y a la vez minimice la cantidad de alumnos en esas clases, facilitando la coordinación de nuevos horarios para las mismas.

Si bien es posible hallar una distribución de aulas eficiente omitiendo el problema 1 y resolviendo directamente el problema P2 con la variable z_1 libre, la solución obtenida no minimizará, necesariamente, la cantidad de clases que deban reprogramarse. Esto es particularmente el caso en la FIQ-UNL, que cuenta con carreras con cantidad muy dispares de alumnos. Mientras algunas clases tendrían un máximo de 15 alumnos, otras tendrán más de 200. Si bien es posible que las clases afectadas sean las mismas que con la metodología propuesta, no se asegura estar satisfaciendo las prioridades fijadas por la gestión y el personal del área encargada.

Una vez resueltos los problemas 1 y 2 en forma secuencial, el paso siguiente consistirá en reprogramar las clases a las que no se les haya asignado un aula. Sin embargo, la institución para la cual se desarrolló el presente trabajo, considera que la reprogramación de clases debe ser definida por los actores involucrados (docentes y alumnos), considerando las aulas que queden disponibles luego de la distribución dada por la solución propuesta. Las autoras han desarrollado un modelo de programación matemática que optimizaría la reprogramación de clases. En efecto, el modelo desarrollado minimiza la cantidad total de horas que deberían modificarse las clases a reprogramar, pero de forma de distribuir las horas modificadas de manera equitativa entre las clases envueltas. Este modelo se resolvería en una tercera instancia, manteniendo la asignación obtenida a partir de los problemas P1 y P2. El modelo para la reprogramación de clases no es presentado aquí, ya que no fue implementado para la resolución de los casos planteados por la institución objeto de estudio.

En la siguiente sección se aplica la metodología presentada para resolver la distribución de aulas de dos programaciones de cursado reales de la FIQ-UNL.

4 Resultados

La metodología propuesta fue aplicada para determinar la asignación de aulas para el cursado semanal en un semestre representativo de la FIQ- UNL

El modelo matemático parte de contar con un número de pedidos de clases a las que se les debe asignar aula, que cuentan con una hora de inicio y finalizado de cursado. Dentro de la información de entrada, se cuenta con la cantidad de alumnos que asistirán a cada clase y el día de cursado semanal correspondiente. De esta manera, el modelo se resuelve por día y considera únicamente las clases que pretenden utilizar un aula el día a resolver.

Se considera que los datos de entrada no suponen ninguna interferencia o superposición de horarios para materias de un mismo curso académico.

Los datos suministrados por la institución fueron transformados a archivos de extensión '.xlsx' para facilitar la lectura de información. El suministro consistía en un listado de materias con pedidos de utilización de aulas, en un determinado día y horario.

La metodología propuesta fue implementada en el software Pyomo [12] dentro del ambiente Python. El procesamiento de la información proporcionada por la gestión de la universidad se llevó a cabo utilizando bibliotecas como pandas.py, num.py, openpyxl.py y otras.

Los problemas de programación matemática MIP fueron resueltos con el solver CPLEX 20.10, en una notebook con procesador Intel Core i3-6100 U CPU 2.30GHz, RAM 4GB, sistema operativo de 64 bits.

A continuación, se presentan dos casos de estudios reales resueltos con la metodología propuesta, correspondientes a la planificación de un semestre típico en la FIQ-UNL. Esta facultad cuenta con 22 aulas para cursado, con capacidades que varían entre 15 y 280 alumnos. El horario de cursado comprende el período entre las 8:00hs y las 22:00hs, dividido en bloques horarios de 30 minutos. En la FIQ-UNL se dictan 2 carreras de pregrado, 7 carreras de grado, 2 carreras de grado compartidas con otras facultades de la UNL, 3 especializaciones, 3 maestrías y 6 doctorados. En total, se deben considerar más de 40 años académicos de carreras, para cada uno de los cuales se debe evitar superposiciones en horarios de clases.

4.1 Caso de estudio 1

Los datos del primer caso de estudio resuelto se resumen en la siguiente Tabla 1.

Tabla 1. Características del caso de estudio 1 y resultados óptimos obtenidos.

	Lunes	Martes	Miércoles	Jueves	Viernes
Características del problema					
Número de clases	37	47	49	46	34
Mínimo-máximo de alumnos sobre todas las clases	19-90	15-280	15-90	15-90	19-120
Características de los problemas MIP P1 y P2					
Número de variables binarias (P1 y P2)	814	1034	1078	1012	748
Número de restricciones	670	690	672	688	664
Soluciones óptimas obtenidas					
Meta 1 (z_1)	0	3	1	0	6
Meta 2 (z_2)	0	214	90	0	613

El problema fue resuelto con la metodología presentada en la sección 3. Esto es, en primer lugar, se resuelve el problema P1 que minimiza la meta 1, z_1: cantidad de clases para las cuales no es posible asignar un aula. En segundo lugar, se fija la variable z_1 en el valor óptimo obtenido y se resuelve el problema 2 que minimiza la meta 2, z_2: cantidad total de alumnos en clases sin aula asignada. El tiempo de cómputo total para la resolución de este caso fue de 0.25 CPUs.

La Tabla 1 sintetiza los resultados obtenidos. Como puede verse, en dos de los cinco días se consiguió realizar la asignación de aulas satisfaciendo todos los pedidos. Mientras que en los restantes días no fue posible asignar aulas a algunas clases (meta 1). La meta 2 indica la cantidad mínima de alumnos en dichas clases.

4.2 Caso de estudio 2

Para este segundo caso de estudio, se modificó el horario de algunas clases en los días para los cuales no fue posible recudir a 0 la meta 1 del caso de estudio 1; es decir, no fue posible asignar aulas a todas las clases. Por lo tanto, los datos del segundo caso de estudio en cuanto a características del problema y cantidad de variables de los problemas P1 y P2 son los mismos que en el caso anterior.

La Tabla 2 reporta los resultados óptimos obtenidos para el caso de estudio 2.

Tabla 2. Resultados óptimos obtenidos para el caso de estudio 2.

	Lunes	Martes	Miércoles	Jueves	Viernes
Soluciones óptimas obtenidas					
Meta 1 (z_1)	0	3	1	0	6
Meta 2 (z_2)	0	205	83	0	608

El tiempo de cómputo total para la resolución de este caso fue de 0.29 CPUs, y al igual que en el caso de estudio 1, no fue posible recudir a 0 la meta 1 en tres de los cinco días de la planificación.

4.3 Análisis de los resultados y aplicación a la programación de exámenes

Como puede observarse a partir de la comparación de las Tablas 1 y 2 para las metas óptimas obtenidas, existen 3 días de la programación semanal para los cuales es necesario una reprogramación de clases. Como se mencionó anteriormente, la gestión de la institución objeto de estudio, define la reprogramación de clases junto con los docentes involucrados y considerando los horarios de clases de los años académicos de las materias.

Sin embargo, las autoras han desarrollado un modelo de programación matemática que analiza una posible reprogramación de clases en el caso de no haber podido reducir a 0 la meta 1, es decir, cuando no se consigue asignar aulas a todas las clases programadas para un cierto día. El modelo tiene por objeto minimizar las modificaciones hechas, minimizando la cantidad de horas a modificar para cada clase reprogramada, considerando la disponibilidad de los docentes y teniendo en cuenta los horarios de clases de cada año académico. El modelo se resolvería luego de haber aplicado la metodología presentada en el presente trabajo, manteniendo la asignación obtenida luego de optimizar secuencialmente las metas 1 y 2.

A partir de los resultados obtenidos, surgió la solicitud de resolver la asignación de aulas para la programación de exámenes de un turno dado. La implementación fue muy similar a la presentada aquí, respetando la cantidad de horas estipuladas para cada examen. En este caso, se consideró el tipo de examen de cada materia, permitiendo que, en caso de exámenes escritos, se asigne más de una materia a una misma aula, dentro de los límites de capacidad de la misma. La metodología resultó satisfactoria, pudiendo determinar una distribución de aulas que hace un uso eficiente del espacio disponible.

4 Conclusiones

En este trabajo se presentó una metodología que permite resolver el problema de asignación de aulas para el cursado semestral en una institución educativa. La metodología propuesta está basada en la programación de metas y se resuelve con el método preventivo.

El trabajo estuvo motivado por la necesidad de brindar una herramienta de soporte eficiente para el personal encargado de la distribución de aulas en la FIQ-UNL, que previamente realizaba esta distribución de forma manual.

Para implementar la metodología desarrollada en casos reales de la facultad, se trabajó de forma coordinada con la gestión de la institución, que facilitó los datos en formatos determinados. Sin embargo, fue necesario diseñar programas que permitan capturar los datos recibidos e ingresarlos como parámetros para los modelos desarrollados. Esto fue posible dentro del ambiente de Python.

La técnica propuesta fue implementada de forma eficiente para la resolución de dos casos de estudio. Mostró ser eficiente para los objetivos perseguidos, obteniendo soluciones óptimas en muy bajos tiempos de cómputo.

Como futuros trabajos se abordará la reprogramación de clases en los casos en que el espacio disponible no sea suficiente para satisfacer todos los pedidos de aulas. Esto constituiría un soporte computacional adicional, para las tareas que usualmente se realizan de forma manual en las dependencias encargadas y podría agilizar la coordinación para la reprogramación de clases con los docentes involucrados.

Agradecimientos

Las autoras agradecen la facilitación de los datos proporcionada por la gestión de la Facultad de Ingeniería Química, UNL, y por el personal encarado de la distribución de aulas.

El trabajo fue financiado por el proyecto CAI+D 2020 UNL 50620190100101LI.

Referencias

1. Burke, E.K., Marecek, J., Parkes, A.J., Rudová, H.: Decomposition, reformulation, and diving in university course timetabling. Computers & Operations Research 37 (2010) 582-597.
2. Khonggamnerd, P., Innet, S.: On Improvement of Effectiveness in Automatic University Timetabling Arrangement with Applied Genetic Algorithm. 2009 Fourth International Conference on Computer Sciences and Convergence Information Technology, IEEE Computer Society.
3. Song, T., Liu, S., Tang, X., Peng, X., Chen, M.: An iterated local search algorithm for the University CourseTimetabling Problem. Applied Soft Computing 68 (2018) 597-608.
4. Chen, M., Tang, X., Song, T., Wu, C., Liu, S., Peng, X.: A Tabu search algorithm with controlled randomization for constructing feasible university course timetables. Computers and Operations Research 123 (2020) 105007.
5. Lemos, A., Monteiro, P.T., Lynce, I.: Disruptions in timetables: a case study at Universidade de Lisboa. Journal of Scheduling 24 (2021) 35-48.
6. Rappos, E., Thiémard, E., Robert, S., Hêche, J.F.: A mixed-integer programming approach for solving university course timetabling problems. Journal of Scheduling (2022), en prensa.
7. Cruz-Rosales, M.H., Cruz-Chávez, M.A., Alonso-Pencina, F., Peralta-Abarca, J.C., Ávila-Melgar, E.Y., Martínez-Bahena, B., Enríquez-Urbano, J.: Metaheuristic with Cooperative

Processes for the University Course Timetabling Problem. Applied Sciences 12-2 (2022) 542.
8. Phillips, A.E., Waterer, H., Ehrgott, M., Ryan, D.M.: Integer programming methods for large-scale practical classroom assignment problems. Computers & Operations Research (2015) 42-53.
9. Antunez, M.A.: Sistema para optimizar asignación de aulas en UNICEN. Trabajo final de la carrera de Ing. en Sistemas. Tandil, Argentina, 2015.
10. Tacchini, L.D., Martínez-Viademonte, J., Braga, M.A.: Evaluación de heurísticas para el problema de asignación de aulas. 50 Jornadas Argentinas de Informática e Investigación Operativa, 50 JAIIO-SIIIO (2021), ISSN: 2618-3277, 86.
11. Babaei, H., Karimpour, J., Hadidi, A.: A survey of approaches for university course timetabling problema. Computers & Industrial Engineering 86 (2015) 43-59.
12. Hart, W.E., Laird, C.D, Watson, J.P., Woodruff, D.L., Hackebeil, G.A., Nicholson, B.L., Siirola, J.D.: Pyomo – Optimization Modeling in Python. Second Edition. Vol. 67. Springer, 2017.