



Trabajo Práctico - Programación Lineal Entera

Asignación de cuadrillas

Fecha de entrega: Lunes 24 de junio de 2024, hasta las **23:59 hs.**

Problema

Muchas empresas ofrecen productos y servicios que dependen de ciertos trabajos realizados en los domicilios de los clientes, como instalación, mantenimiento o inspección. Estos trabajos deben ser realizados por cuadrillas, que son asignadas específicamente a los diferentes requerimientos.

En este trabajo práctico nos interesa la planificación semanal de las cuadrillas de trabajo a las diferentes órdenes enlistadas. Para esto vamos a contar con un conjunto de trabajadores y una lista de trabajos a realizar. El objetivo es maximizar la ganancia total, teniendo en cuenta el beneficio que nos otorga resolver un trabajo y considerando los pagos a los trabajadores.

El problema formalmente es el siguiente:

- Contamos con una cantidad T de trabajadores para realizar los trabajos.
- Por otro lado, tenemos una lista de O órdenes de trabajo a realizar.
- Cada orden o_i requiere de una cantidad prefijada de trabajadores, t_i , y realizarla da un beneficio b_i .
- La semana a planificar tiene 6 días, de lunes a sábado.
- Cada día tiene 5 turnos de 2 horas. Se asume que cada orden de trabajo se puede realizar utilizando un turno, y un mismo trabajador no puede realizar más de una orden en un mismo turno.

La solución debe indicar cómo asignar los trabajadores a las órdenes, y a su vez cómo asignar las órdenes a los turnos, cumpliendo las siguientes restricciones necesarias para que la asignación sea factible:

- No toda orden de trabajo tiene que ser resuelta.
- Ningún trabajador puede trabajar los 6 días de la planificación.
- Ningún trabajador puede trabajar los 5 turnos de un día.
- Hay pares de órdenes de trabajo que no pueden ser satisfechas en turnos consecutivos de un trabajador (si bien en este problema no nos preocupamos por el ruteo sino sólo por la asignación, hay órdenes tan lejanas geográficamente que no se podrán satisfacer consecutivamente por el mismo trabajador).
- Una orden de trabajo o_i debe tener asignados sus t_i trabajadores en un mismo turno para poder ser resuelta.



- Existen algunos pares de órdenes de trabajo correlativas. Un par ordenado de órdenes correlativas A y B , nos indica que si se satisface A , entonces debe satisfacerse B ese mismo día en el turno consecutivo.
- Los trabajadores son remunerados según la cantidad de órdenes asignadas, por lo que la diferencia entre el trabajador con más órdenes asignadas y el trabajador con menos órdenes no puede ser mayor a 8. Para esto se consideran todos los trabajadores, aún los que no tienen ninguna tarea asignada esa semana.

Si se cumplen las anteriores restricciones, la asignación es factible. De todas maneras, existen dos restricciones más que son deseables. Es decir, vamos a querer evaluar si se pueden intentar cumplir o estar cerca de ello. Estas dos restricciones son:

- Hay conflictos entre algunos trabajadores que hacen que prefieran no ser asignados a una misma orden de trabajo.
- Hay pares de órdenes de trabajo que son repetitivas, por lo que será bueno que un mismo trabajador no sea asignado a ambas.

Una vez cumplidas las restricciones necesarias, el objetivo de nuestra solución es maximizar la ganancia de la asignación. La ganancia de la asignación se define como la suma de los beneficios de las órdenes satisfechas menos la remuneración otorgada a cada trabajador.

Los trabajadores son remunerados según el siguiente esquema:

- Por cada una de las primeras 5 órdenes que realizan obtienen una remuneración de \$1000.
- Por cada una de las órdenes entre la 6 y 10, obtienen una remuneración de \$1200.
- Por cada una de las órdenes entre la 11 y 15, obtienen una remuneración de \$1400.
- A partir de la orden 15, obtienen una remuneración de \$1500 por cada una.

Resolución

Para las instancias considerar un archivo con el siguiente formato:

- Primera línea: un entero que representa la cantidad de trabajadores, T .
- Segunda línea: un entero que representa la cantidad de órdenes, O .
- Línea 3 a $O + 2$: una orden por línea conteniendo:
 - id: un entero no repetido, i .
 - beneficio: un entero positivo, b_i .
 - trabajadores necesarios: un entero positivo, t_i
- Línea $O + 3$: un entero positivo que representa la cantidad de conflictos entre trabajadores, $cant_conf_trabajadores$.



- Siguiendo *cant_conf_trabajadores* líneas: dos enteros entre 0 y $T - 1$ que representan los trabajadores en conflicto.
- Línea $O + cant_conf_trabajadores + 4$: un entero positivo que representa la cantidad de órdenes correlativas, *cant_ord_correlativas*.
- Siguiendo *cant_ord_correlativas* líneas: dos enteros entre 0 y $O - 1$ que representan los pares ordenados de órdenes correlativas.
- Línea $O + cant_conf_trabajadores + cant_ord_correlativas + 5$: un entero positivo que representa la cantidad de órdenes conflictivas, *cant_ord_conflictivas*.
- Siguiendo *cant_ord_conflictivas* líneas: dos enteros entre 0 y $O - 1$ que representan los pares de órdenes conflictivos.
- Línea $O + cant_conf_trabajadores + cant_ord_correlativas + cant_ord_conflictivas + 6$: un entero positivo que representa la cantidad de órdenes repetitivas, *cant_ord_repetitivas*.
- Siguiendo *cant_ord_repetitivas* líneas: dos enteros entre 0 y $O - 1$ que representan los pares de órdenes repetitivos.

La resolución del trabajo consiste en la realización de modelos que se adecuen al problema, resolver diferentes instancias del problema mediante el paquete CPLEX y la entrega de un informe. Generar instancias que consideren adecuadas para poder sacar conclusiones sobre la experimentación.

Contamos con una implementación preliminar para el problema con la lectura de los datos de una instancia. Luego de realizar la lectura de los datos, estos se encuentran guardados en una instancia de la clase *InstanciaAsignacionCuadrillas* que contiene los siguientes campos:

- *cantidad_trabajadores*: un entero que representa la cantidad de trabajadores.
- *ordenes*: una lista que contiene todas las órdenes de la semana.
- *conflictos_trabajadores*: una lista con pares de trabajadores que conflictúan entre sí.
- *ordenes_correlativas*: una lista con pares ordenados de identificadores de órdenes correlativas.
- *ordenes_conflictivas*: una lista con pares de identificadores de órdenes conflictivas.
- *ordenes_repetitivas*: una lista con pares de identificadores de órdenes repetitivos.

A su vez, una orden de trabajo contiene los siguientes campos:

- *id*: un entero identificador de la orden.
- *beneficio*: un entero que representa el beneficio que nos da una orden si la satisfacemos.
- *cant_trab*: un entero que representa la cantidad de trabajadores necesarios para satisfacerla.



Se pide:

1. Modelo: Formular un modelo que cumpla con todas las restricciones para realizar una asignación factible.
2. Restricciones deseables: Reformular el modelo para tener en cuenta las restricciones deseables. Considerar su incorporación como nuevas restricciones y como términos de la función objetivo.
3. Implementación: Realizar la implementación de las alternativas propuestas utilizando CPLEX.
4. Experimentación algorítmica: Evaluar diferentes alternativas algorítmicas mediante los parámetros que provee el CPLEX y discutir los resultados obtenidos.
5. Experimentación sobre distintos escenarios: Considerar diferentes escenarios (función objetivo, restricciones deseables) para lograr una discusión tanto cuantitativa como cualitativa de los resultados obtenidos.