

Tarea 4.

Las preguntas 1, 2 y 3 requieren desarrollos computacionales usando el problema de planificación de cosecha de fruta que apareció en la Interrogación 2. Usted tendrá que desarrollar una implementación de Relajación Lagrangeana para el problema y también de una meta-heurística de las estudiadas en el curso.

Los datos para el problema se generan en forma aleatoria. Para esto, le entregamos, junto a esta tarea, un código Python que genera datos.

Pregunta 1 (8 pts):

El problema a tratar es el mismo que se presentó en la Interrogación 2. Para completitud, lo repetimos aquí: Un empresa del sector agropecuario necesita planificar sus actividades de cosecha y procesamiento de frutas sobre un horizonte de T períodos de tiempo (días, para fijar ideas). Existen n parcelas que pueden ser cosechadas, y la disponibilidad de fruta en la parcela j es L_j (kilos). Hay un costo por kilo cosechado en cada período, c_{jt} y también un costo fijo cada vez que se decide cosechar una parcela, f_{jt} . Además, hay personal empleado en la cosecha y α_{jt} es el requerimiento unitario de mano de obra (en horas) en la parcela j en el periodo t , y el total disponible en el periodo t es igual a B_t . Además, una parcela sólo se puede cosechar una vez en todo el horizonte, aunque sea parcialmente. Una vez cosechada la fruta, debe ser procesada en una de m plantas de procesamiento, para lo cual la fruta cosechada en un día es acumulada y enviada a alguna de las plantas (todo está cerca, así que el transporte lo consideraremos despreciable). La planta de procesamiento i puede recibir un máximo de K_i kilos sobre todo el horizonte de tiempo. La empresa vende la fruta procesada de la planta i a un precio p_i . El costo de procesamiento es e_{it} para la planta i en el día t . La empresa ha construido el siguiente modelo de optimización para encontrar la planificación más rentable, el que usa las siguientes variables:

x_{jt} : kilos cosechados en la parcela j el día t .

y_{jt} : vale 1 si se cosecha la parcela j en t , 0 en caso contrario.

z_{it} : kilos procesados en la planta i en t .

El modelo es:

$$\begin{aligned} \max \quad & \sum_{t=1}^T \sum_{i=1}^m p_i z_{it} - \sum_{t=1}^T \left\{ \sum_{j=1}^n c_{jt} x_{jt} + \sum_{j=1}^n f_{jt} y_{jt} + \sum_{i=1}^m e_{it} z_{it} \right\} \\ \text{s.a.} \quad & \sum_{j=1}^n \alpha_{jt} x_{jt} \leq B_t, \quad t = 1, \dots, T \quad (1) \\ & x_{jt} \leq L_j y_{jt}, \quad j = 1, \dots, n; t = 1, \dots, T \quad (2) \\ & \sum_{t=1}^T y_{jt} \leq 1, \quad j = 1, \dots, n \quad (3) \\ & \sum_{j=1}^n x_{jt} = \sum_{i=1}^m z_{it}, \quad t = 1, \dots, T \quad (4) \\ & \sum_{t=1}^T z_{it} \leq K_i, \quad i = 1, \dots, m \quad (5) \\ & x_{jt} \geq 0, z_{it} \geq 0, y_{jt} \in \{0, 1\}, j = 1, \dots, n; t = 1, \dots, T; i = 1, \dots, m \quad (6) \end{aligned}$$

Codifique este problema en Python, usando el esquema de generación de datos que se le ha entregado, para una instancia con $m = 8$, $n = 140$, $hor = 12$. Cambie la semilla de generación del generador de números

aleatorios. Esa línea actualmente dice: `np.random.seed(16)`; cámbie el “16” por un número de 6 ó 5 dígitos formado a partir de la fecha de nacimiento del o la líder del grupo, en el formato DDMMYY (por ejemplo, 161001, 50202, etc). Reporte en su solución de la tarea la semilla empleada y también indique el tipo de hardware que estará usando para resolver estos problemas.

Trate de resolver el problema mediante Gurobi, para obtener una solución óptima. Es posible que, dependiendo de los datos que se le generaron, no pueda llegar a una solución óptima dentro de un tiempo razonable (o llegue a una muy rápido). En ese caso, deberá tomar una decisión respecto a cuál es la mejor estimación que le entrega Gurobi. Registre este valor, y también la información sobre el proceso de solución: tiempo de CPU usado (el reportado por Gurobi), número de nodos de Ramificación y Acotamiento, etc. Revise la documentación de Gurobi para saber qué información se despliega y su significado, así como para apoyarse en el funcionamiento del software.

Pregunta 2 (10 pts):

Ahora le pedimos que desarrolle una implementación completa de Relajación Lagrangeana para el problema. Para esto, deberá decidir qué conjunto de restricciones dualizar. Justifique con precisión su elección. Deberá implementar una relajación que haga uso de una cota adecuada para la determinación del paso (no está permitido hacer la “trampa” que mostramos en clases, aunque tenga un valor óptimo de la Pregunta 1). Entonces, tendrá que proponer una Heurística Lagrangeana para el problema e incorpore eso en su código poder calcular, en cada iteración, una cota inferior (observe que el problema es de maximización) al valor óptimo. Explique con precisión la lógica de su heurística. Corra la relajación lagrangeana y registre el valor dual al que llega, así como el mejor valor de cota superior al que llega. Reporte también el tiempo de computación, número de iteraciones. Compare estos resultados con lo que resultó en la pregunta anterior al correr el problema completo mediante ramificación y Acotamiento. Si recurre a literatura adicional (y está bien que lo hagan) no olvide referenciarla.

Pregunta 3 (8 pts + 12 puntos adicionales máximos según la calidad de la solución):

Evaluación de la pregunta: Esta pregunta se evaluará con 8 puntos que se obtienen si el desarrollo formal y corridas del algoritmo que se pide son totalmente correctos. Los 12 puntos restantes serán según el éxito que obtenga en obtener una buena solución. Para esto, debe reportar el error porcentual al que pudo llegar, comparado con la mejor estimación del valor de la solución óptima, que se puede obtener de la Pregunta 1 o de la 2. Les pedimos que incluyan un “screenshot”, captura de pantalla, del momento de la solución final del software. También registre el tiempo computacional en el que obtuvo esta solución (en los códigos de Newton, FISTA y similares, hay código para medir tiempo de ejecución, les pueden servir). Para otorgar el puntaje, procederemos a ordenar esta información, de menor a mayor según el error alcanzado, y otorgaremos el puntaje según la siguiente tabla:

Percentil (%)	5	15	40	60	100
Puntos	12	8	6	4	2

De este modo, los grupos que lograron el menor error (dentro del 5% superior del curso) se llevan los 12 puntos y así sucesivamente.

Lo que tiene que hacer es programar una meta-heurística para el problema. Su procedimiento debe ser o Simulated Annealing o Búsqueda Tabú, lo que ustedes prefieran. Su misión es obtener la mejor solución posible. Para esto puede cambiar los parámetros de la heurística (temperatura, número de iteraciones, largo de lista tabú, según sea el caso, etc.) También puede cambiar la forma de la solución inicial, la forma en que se generan las vecindades, etc. Tiene libertad de hacer lo que considere adecuado, mientras se mantenga dentro de lo que se entiende por la meta-heurística que eligió. De los resultados de las preguntas anteriores, debiera tener una estimación del valor óptimo del problema, así que eso le debe servir. En su informe de la tarea, reporte con claridad lo que hizo, lo que implementó, los experimentos que hizo, etc. Puede recurrir a literatura adicional, por supuesto, hay muchas variantes, extensiones y mejoras a ambas metaheurísticas y las pueden encontrar en artículos y libros. Con eso pueden guiarse respecto a cómo usar mejor la meta-

heurística elegida, pero no olvide poner las referencias.

Pregunta 4 (10 pts):

En esta pregunta le pedimos que investigue el uso de métodos heurísticos en aplicaciones relevantes. Para esto, busque en la literatura un artículo que describa alguna aplicación de alguno de los métodos metaheurísticos estudiados en el curso: Simulated Annealing, Búsqueda Tabú, Algoritmo genéticos, en alguno de estos ámbitos:

1. Problemas de logística, ruteo de vehículos o relacionados.
2. Problemas de planificación o programación o secuenciamiento de actividades, ya sea en proyectos o en ámbitos manufactureros.
3. Problemas de asignación de recursos o planificación de actividades en el área de gestión de sistemas de salud.

Una vez que haya hecho su elección, deben preparar un resumen de máximo dos planas con una descripción del desarrollo, los resultados y otros. También debe comentar, en forma crítica los resultados y la aplicación. Si encuentra algún artículo de aplicación que no esté exactamente en las categorías anteriores, o tiene dudas al respecto, puede preguntar al profesor o ayudante de cátedra o coordinadora.