Un algoritmo basado en la descomposición combinatoria de Benders para un problema de programación de tareas

Santina Bardengo[1 [0009-0009-4234-0947]]
Javier Marenco[1 [0000-0003-2694-4758]]

Escuela de Negocios, Universidad Torcuato Di Tella, Av. Figueroa Alcorta 7350
(1428) Ciudad de Buenos Aires, Argentina
sbardengo@mail.utdt.edu, javier.marenco@utdt.edu

Resumen En este trabajo consideramos un problema de programación de tareas para una institución deportiva. Se debe programar un conjunto de tareas a lo largo de varios días, y se tiene un conjunto de empleados que pueden realizar estas tareas. Se supone que todos los empleados realizan el mismo horario de trabajo. Cada tarea tiene su duración y una cantidad requerida de empleados. Si una tarea debe ser realizada por dos o más empleados, entonces los dos empleados deben estar trabajando en la tarea al mismo tiempo. Un empleado no puede trabajar en dos tareas al mismo tiempo, y las tareas no se pueden interrumpir una vez comenzadas. El objetivo es maximizar la importancia de las tareas programadas, respetando las restricciones mencionadas. Presentamos dos modelos de programación lineal entera para este problema, junto con un enfoque basado en la descomposición combinatoria de Benders. Los experimentos computacionales muestran que este último algoritmo es efectivo para resolver instancias reales del problema.

Keywords: programación de tareas · programación entera · descomposición de Benders

En este trabajo consideramos el siguiente problema de programación de tareas, motivado por la problemática de una institución deportiva. Tenemos un conjunto N de *tareas*, que deben ser programadas. Para cada tarea $i \in N$, tenemos la *duración* $l_i \in \mathbb{Z}_{>0}$ en horas de la tarea (suponemos $l_i \leq 8$) y la *cantidad de empleados* $e_i \in \mathbb{Z}_{>0}$ que deben realizar la tarea. Además, tenemos la *prioridad* $p_i \in \mathbb{Z}_{>0}$ de la tarea, de modo tal que un mayor valor de prioridad corresponde a las tareas más importantes. Finalmente, tenemos un parámetro $f_i \in \mathbb{Z}_{>0}$ que especifica la mínima cantidad de días entre dos repeticiones de la tarea i. Además, tenemos un conjunto J de *empleados*, que estarán a cargo de realizar las tareas del conjunto N. Para cada tarea $i \in N$, tenemos el conjunto $J_i \subseteq J$ de empleados con las habilidades necesarias para realizar la tarea i. Asimismo, tenemos un conjunto D de *días*, que corresponde al intervalo de programación. Para cada día $d \in D$, tenemos el conjunto $F_d \subseteq J$ de empleados que tienen *franco* (día de descanso) en el día d, de modo tal que los empleados disponibles en el día d

son los empleados del conjunto $J \backslash F_d$. Finalmente, tenemos el conjunto $R \subseteq N$ de tareas que se deben realizar repetidamente en el mismo día de la semana. El problema a resolver consiste en determinar para cada día $d \in D$ (a) qué conjnuto $N_d \subseteq N$ de tareas se programa, (b) qué empleados realizan cada tarea de N_d y (c) a qué hora se realiza cada tarea de N_d, de modo tal de maximizar la suma de las prioridades de las tareas realizadas, y respetando además las siguientes restricciones.

1. Se deben asignar e_i empleados de $J_i \backslash F_d$ a la tarea i, para todo $i \in N_d$ y todo $d \in D$.
2. La hora de inicio de la tarea i debe estar entre 0 y $8 - d_i$, para todo $i \in N_d$ y todo $d \in D$. Se supone que todos los empleados asignados a la tarea i realizan esta tarea al mismo tiempo, y sin interrupciones desde la hora de inicio y hasta la finalización de la tarea.
3. Un mismo empleado no puede encontrarse realizando dos tareas al mismo tiempo (junto con la restricción anterior, esto implica que un empleado no puede recibir tareas por más de ocho horas en un mismo día).
4. Las tareas de R se deben programar repetidamente en el mismo día de la semana.
5. No se deben tener dos repeticiones de la tarea i con menos de f_i días entre ellas, para $i \in N$.

En este trabajo presentamos dos modelos de programación lineal entera para este problema, y analizamos la posibilidad de resolver estos modelos con el solver SCIP [1]. Mostramos que no es posible encontrar soluciones de buena calidad para estos modelos en tiempos de cómputo acotados para instancias reales, debido al tamaño de los modelos. Por este motivo, presentamos un enfoque basado en la descomposición combinatoria de Benders [2], aprovechando la estructura en dos niveles de las soluciones de este problema.

1. El primer nivel de decisiones corresponde a determinar qué tareas se programan en cada día, teniendo en cuenta una relajación de las restricciones sobre las asignaciones posibles para cada día.
2. El segundo nivel de decisiones corresponde a determinar qué empleados realizan cada tarea y a qué hora comienza cada tarea.

Definimos a continuación los modelos utilizados en la descomposición. En el modelo maestro se busca asignar tareas a días. Para cada $i \in N$ y cada $d \in D$, la variable binaria T_{id} toma el valor 1 si la tarea i se programa el día d. En este contexto, el modelo para el problema maestro es el siguiente.

$$\max \sum_{i \in N} \sum_{d \in D} p_i \, T_{id}$$

$$\sum_{i \in N} T_{id} \, l_i \, e_i \leq 8 |J \backslash F_d| \qquad \forall d \in D$$

$$T_{id} \, e_i \leq |J \backslash F_d| \qquad \forall i \in N, \forall d \in D$$

$$\sum_{d \in D: d' \leq d \leq d'+f_i} T_{id} \leq 1 \qquad \forall i \in N, \forall d' \in D$$
$$T_{id} = T_{i,d+7} \qquad \forall i \in R, \forall d \in D : d+7 \in D$$
$$T_{id} \in \{0,1\} \qquad \forall i \in N, \forall d \in D$$

Por su parte, el subproblema busca optimizar la asignación de empleados a tareas y horas para cada día. Llamamos $T \subseteq N$ al conjunto de tareas asignadas al día en cuestión. Para cada $j \in J$, cada $i \in T$ y cada $h \in H$, definimos la variable binaria W_{jih} que toma valor 1 si el empleado j realiza la tarea i comenzando a la hora h y la variable binaria Z_{jih} que especifica si el empleado j se encuentra trabajando en la tarea i en la hora h. Además, para cada tarea $i \in T$ y cada hora h, tenemos la variable binaria P_{ih} que toma valor 1 si la tarea i comienza a la hora h. En este contexto, el modelo del subproblema es el siguiente.

$$\sum_{j \in J} W_{jih} = e_i \ P_{ih} \qquad \forall i \in T, \forall h \in H$$
$$\sum_{i \in T} \sum_{h \in H} W_{jih} \ l_i \leq 8 \qquad \forall j \in J$$
$$Z_{jih} = \sum_{h' \in H: h-l_i+1 \leq h' \leq h} W_{jih'} \qquad \forall i \in T, \forall j \in J, \forall h \in H$$
$$\sum_{i \in T} Z_{jih} \leq 1 \qquad \forall j \in J, \forall h \in H$$
$$\sum_{h \in H} P_{ih} = 1 \qquad \forall i \in T$$
$$P_{ih} = 0 \qquad \forall i \in T, \forall j \in J, \forall h \in H : h + l_i > 8$$
$$W_{jih} = 0 \qquad \forall i \in T, \forall j \in J \backslash J_i, \forall h \in H$$
$$W_{jih} \in \{0,1\} \qquad \forall j \in J, \forall i \in T, \forall h \in H$$
$$Z_{jih} \in \{0,1\} \qquad \forall j \in J, \forall i \in T, \forall h \in H$$
$$P_{ih} \in \{0,1\} \qquad \forall i \in T, \forall h \in H.$$

De acuerdo con nuestros experimentos computacionales, este enfoque es adecuado para resolver en forma óptima las instancias reales que motivaron este estudio. Presentamos además experimentos para evaluar la performance de este enfoque sobre instancias realistas, que muestran que las ideas sobre las cuales se basa el algoritmo pueden adaptarse a situaciones similares.

Referencias

1. Bestuzheva, K., Besançon, M., Chen, W.K., Chmiela, A., Donkiewicz, T., van Doornmalen, J., Eifler, L., Gaul, O., Gamrath, G., Gleixner, A., Gottwald, L., Graczyk, C., Halbig, K., Hoen, A., Hojny, C., van der Hulst, R., Koch, T., Lübbecke, M., Maher, S.J., Matter, F., Mühmer, E., Müller, B., Pfetsch, M.E., Rehfeldt, D., Schlein, S., Schlösser, F., Serrano, F., Shinano, Y., Sofranac, B., Turner, M., Vigerske, S., Wegscheider, F., Wellner, P., Weninger, D., Witzig, J.: Enabling research through the scip optimization suite 8.0 **49**(2) (2023). https://doi.org/10.1145/3585516

2. Codato, G., Fischetti, M.: Combinatorial benders' cuts for mixed-integer linear programming. Operations Research **54**(4), 756–766 (2006). https://doi.org/10.1287/opre.1060.0286