Lesson 10/11 HW: Combinatorial Optimization Models

ITSD is attempting to backup data on some external harddrives and have asked you for help. The data files have sizes 240, 462, 117, 560, 379, 110, 341, 294, 503, 469, 90, 65, 617, 500, 550, and 400 GB.

a) If the capacity of a harddrive is 780 GB, write a concrete integer programming model to determine the minimum number of harddrives needed to backup all files. *Hint: You need two types of variables. One is a $x_{f,h}$ variable to which equals 1 if a file is assigned to a harddrive. The other is a z_h which equals 1 if that harddrive is used. Make sure you include logical constraints to relate these variables.*

b) Convert your concrete model to a parameterized model.

c) How many total variables does your model have?

d) **Optional:** Implement your model in Python. You may have to start with a smaller number of harddrives and gradually increase. This is an interesting example of how for even a "small" problem, a solver can be slowed down with some difficult constraints/variables.

Solution

Part (a)

In this problem, we don't know the total number of Harddrives that we will need ahead of time. We can make a rough guess: the sum of all sizes is:

$$240 + 462 + \cdots + 400 = 5697$$

$$5697/780 = 7.3$$

This tells us that we will need at least 8 Harddrives (why?). I'll model it with 12 CDs to be safe. I'll do an abbreviated concrete model.

Sets

Let F be the set of files (specifically $F = \{1, 2, 3, 4 \cdots 16\}$
Let H be the set of Harddrives (specifically $H = \{1, 2, 3, \cdots, 12\}$

Decision Variables

Let $x_{f,h} = 1$ if file $f \in F$ is placed on harddrive $h \in H$ or 0 otherwise.
Let $z_h = 1$ if harddrive $h \in H$ is used.

Parameters
none

Objective Function

Our goal is to minimize the total number of harddrives used, so I'll minimize the sum of z_h

$$\text{minimize: } z_1 + z_2 + \cdots z_{12}$$

Constraints

Logically, we need three types of constraints:

1. Every file is placed on a harddrive
2. No harddrive is overloaded
3. If a file is placed on a harddrive, that harddrive is used.

To be super clear in the concrete model, I'll formulate it with these three types of constraints. Note that it is possible to combine constraints 2 and 3 and only have two types of constraints in your final model. That's what I'll do for the parameterized model

$$\begin{aligned}
x_{1,1} + x_{1,2} + \ldots x_{1,12} &= 1 && \text{(File 1 goes on a harddrive)}\\
x_{2,1} + x_{2,2} + \ldots x_{2,12} &= 1 && \text{(File 2 goes on a harddrive)}\\
&\vdots\\
x_{16,1} + x_{16,2} + \ldots x_{16,12} &= 1 && \text{(File 16 goes on a harddrive)}\\
240x_{1,1} + 462x_{2,1} + \ldots 396x_{16,1} &\leq 780 && \text{(harddrive 1 is not overloaded)}\\
240x_{1,2} + 462x_{2,2} + \ldots 396x_{16,2} &\leq 780 && \text{(harddrive 2 is not overloaded)}\\
&\vdots\\
240x_{1,12} + 462x_{2,12} + \ldots 396x_{16,12} &\leq 780 && \text{(harddrive 12 is not overloaded)}\\
x_{1,1} + x_{2,1} + \ldots + x_{16,1} &\leq 16z_1 && \text{(harddrive 1 must be used if a file is added t}\\
x_{1,2} + x_{2,2} + \ldots + x_{16,2} &\leq 16z_2 && \text{(harddrive 2 must be used if a file is added t}\\
&\vdots\\
x_{1,12} + x_{2,12} + \ldots + x_{16,12} &\leq 16z_{12} && \text{(harddrive 12 must be used if a file is added}\\
x_{f,h}, z_h &\in \{0,1\} \quad \text{for all } f \in F, h \in H && \text{(binary)}
\end{aligned}$$

Part (b)

Sets

Let F be the set of files
Let H be the set of harddrives

Decision Variables

Let $x_{f,h} = 1$ if file $f \in F$ is placed on harddrive $h \in H$ or 0 otherwise.
Let $z_h = 1$ if harddrive $h \in H$ is used.

Parameters

Let s_f be the size of file f

Let M be the maximum capacity of a harddrive

Objective Function

$$\text{minimize: } \sum_{h \in H} z_h$$

Constraints

Here in the parameterized model, I will combine the overloaded and linking constraints so my final model will only have 2 types of constraints (not including binary)

$$\begin{aligned}
\sum_{h \in H} x_{f,h} &= 1 && \text{for all } f \in F && \text{(file } f \text{ goes on a harddrive)} \\
\sum_{f \in F} s_f x_{f,h} &\leq M z_h && \text{for all } h \in H && \text{(Capacity and linking)} \\
x_{f,h}, z_h &\in \{0, 1\} && \text{for all } f \in F, h \in H && \text{(binary)}
\end{aligned}$$

Part c

I have one x variable for every file/harddrive combination. Since I used 12 harddrives, this leaves me with

$$16 * 12 = 192 \text{ total } x \text{ variables}$$

Additionally, I have 1 z variable for every harddrive, thus I have a total of 12 z variables. Thus, I have a grand total of 204 variables.

Bonus: How many constraints? In my abstract model, I only have $16 + 12 = 28$ total constraints (excluding binary constraints). However, this is a hard problem to solve due to the massive number of variables (imagine how much a problem like this can blow up for a large problem).

Part d I did not implement this in python.