Benders Decomposition

A Mixed Integer Programming Technique

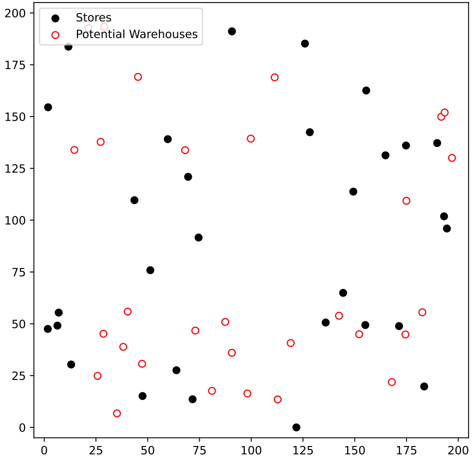
Ben Varley

12/09/2024

Uncapacitated Facility Location Problem

Imagine you own n grocery stores in some geographical area, and you want to build warehouses to supply them.
After some surveying you identify m possible locations to build your warehouses and you calculate the cost of building a warehouse at each location, and the cost of supplying each store from each possible warehouse.

Setup for $n = m = 30$



Question: what is the cheapest way to supply all the stores?

Linear Programming Problems

In brief, a linear programming problem (or LP) is of the following form.

$$\min_{\boldsymbol{x} \in \mathbb{R}^n} \boldsymbol{c}^\intercal \boldsymbol{x}$$

subject to,

$$A\boldsymbol{x} \leq \boldsymbol{b}$$
$$\boldsymbol{x} \geq 0$$

where $\boldsymbol{c} \in \mathbb{R}^n, \boldsymbol{b} \in \mathbb{R}^m, A \in \mathbb{R}^{m \times n}$

What is so cool about these problems.

If we can convert our problem into this form then we can solve it (to optimality) using the Simplex method.
Often we want our variables to be integers (creating an IP) or we have a mix of integer and continuous variables (creating a MIP).
We solve these using a branch and bound algorithm.

IP Formulation of the UFL

Sets:

- I - set of stores
- J - set of possible warehouse locations

Data:

- $b_j, \forall j \in J$ - cost of building warehouse m
- $c_{ij}, \forall i \in I, j \in J$ - cost of supplying store i with warehouse j.

Variables:

- $y_j \in \{0, 1\}$ - 1 if we build a warehouse in location j, 0 otherwise.
- $x_i j \in \{0, 1\}$ - 1 if we supply store i with warehouse j, 0 otherwise.

MIP Formulation of the UFL

Objective:

$$\min \sum_{j \in J} b_j y_j + \sum_{i \in I} \sum_{j \in J} c_{ij} x_{ij}$$

Constraints:

- Each store has to be supplied.

$$\sum_{j \in J} x_{ij} = 1, \forall i \in I \qquad \text{(Supply)}$$

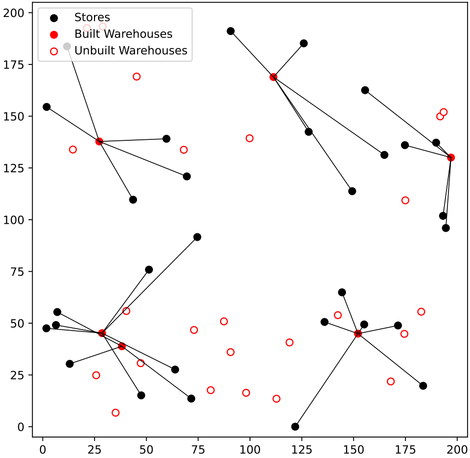
- Warehouses have to be built to supply stores.

$$x_{ij} \leq y_j, \forall i \in I, j \in J \qquad \text{(Build)}$$

Code time

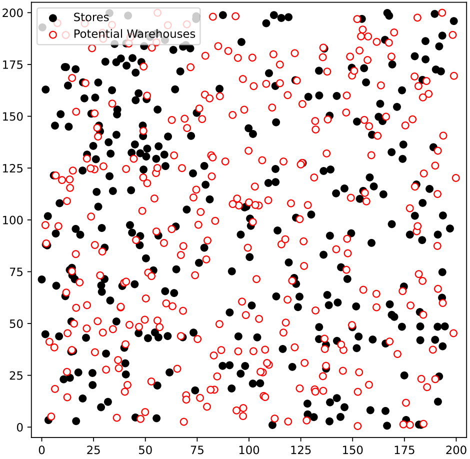
- Locations are picked in a 200×200 square.
- Connection costs are equal to Euclidean distance plus $x \sim U[0, 70]$.
- Build costs are equal to $y \sim U[75, 125]$.

Solution for small n.

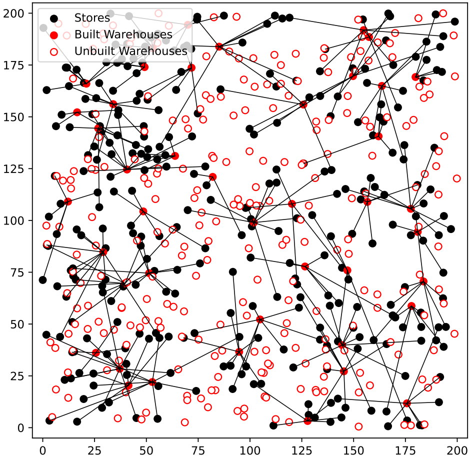


Solves in about 0.02 seconds. $\ddot\smile$

Setup for $n = m = 300$



Solution for big n.



Solves in about 4 minutes. $\ddot{\frown}$

Analysis of the IP Formulation

We increased the problem size by a factor of 10 but our time increased by a factor of 12650.
Furthermore, our data makes the problem a lot easier. If we use completely random connection costs by the four minute mark the MIP gap is still 13.1%.

What happened?

Our current formulation is doing a lot of work to allocate stores to warehouses. We have $n \times m$, X variables and $n \times m$ constraints purely to manage store allocation.
In reality allocating the stores is an incredibly easy problem once we have a set of built warehouses.

For each store we simply choose the built warehouse with the lowest connection cost (i.e. the *closest* warehouse).

Abstract Benders Decomposition

Suppose we have some integer programming problem,

$$\min \boldsymbol{c}^{\mathsf{T}}\boldsymbol{y} + \sum_{\omega\in\Omega} \Theta_\omega(\boldsymbol{y})$$

subject to,

$$\begin{aligned} A\boldsymbol{y} &\leq \boldsymbol{b} \\ \boldsymbol{y} &\geq 0 \end{aligned}$$

where each Θ_ω is some complicated function of $\boldsymbol{y}$ (called a BSP) and $\Theta_\omega(\boldsymbol{y}^\star) = \infty \implies \boldsymbol{y}^\star$ is not a feasible solution.

Benders Master Problem

By introducing new variables θ_ω we can construct a Benders Master Problem (BMP)

$$\min \boldsymbol{c}^\mathsf{T}\boldsymbol{y} + \sum_{\omega\in\Omega}\theta_\omega$$

Subject to,

$$\begin{aligned} A\boldsymbol{y} &\leq \boldsymbol{b} \\ \text{OptCuts} \\ \text{FeasCuts} \end{aligned}$$

where OptCuts and FeasCuts are sets of constraints which are initially empty.

Algorithm

1. Solve the BMP to get a solution $\boldsymbol{y}^\star$
2. For each subproblem $(\omega \in \Omega)$
   2.1 Calculate $\Theta_\omega(\boldsymbol{y}^\star)$
   2.2 If $\Theta_\omega(\boldsymbol{y}^\star) = \infty$ add a feasibility cut to the BMP.
   2.3 If $\Theta_\omega(\boldsymbol{y}^\star) > \theta_\omega$ add an optimality cut to the BMP.
3. If cuts were added go to Step 1, else break.

Feasibility cuts are of the form,

$$0 \geq \Gamma + \boldsymbol{\gamma}^\intercal \boldsymbol{y}$$

Optimality cuts are of the form,

$$\theta_\omega \geq \Gamma + \boldsymbol{\gamma}^\intercal \boldsymbol{y}$$

Constraints on our constraints

$$0 \geq \Gamma + \boldsymbol{\gamma}^\intercal \boldsymbol{y} \qquad \text{(Feasibility Cut)}$$

$$\theta_\omega \geq \Gamma + \boldsymbol{\gamma}^\intercal \boldsymbol{y} \qquad \text{(Optimality Cut)}$$

A valid feasibility cut must satisfy,

- $0 < \Gamma + \boldsymbol{\gamma}^\intercal \boldsymbol{y}^\star$
- $0 < \Gamma + \boldsymbol{\gamma}^\intercal \boldsymbol{y} \implies \Theta_\omega(y) = \infty$

A valid optimality cut must satisfy,

- $\Theta_\omega(\boldsymbol{y}^\star) = \Gamma + \boldsymbol{\gamma}^\intercal \boldsymbol{y}^\star$
- $\Theta_\omega(\boldsymbol{y}) \geq \Gamma + \boldsymbol{\gamma}^\intercal \boldsymbol{y}$

Bending the UFL

Our Benders Master Problem is the following,

$$\min \sum_{j \in J} b_j y_j + \sum_{i \in I} \theta_i$$

subject to,

$$\sum_{j \in J} y_j \geq 1$$

and where our θ_i variables will approximate

$$\Theta_i(\boldsymbol{y}) = \min_{j \in J | y_j = 1} c_{ij}$$

i.e. we have a subproblem for each store.

Bending the UFL

And for each subproblem we will add the following cut if $\Theta_i(\boldsymbol{y}^\star) > \theta_i$,

$$\theta_i \geq \Theta_i^\star - \sum_{j \in J | c_{ij} < \Theta_i^\star} \left(\Theta_i^\star - c_{ij}\right) y_j$$

where $\Theta_i^\star = \Theta_i(\boldsymbol{y}^\star)$.

Code time again

- Random Data (with $n = m = 200$) - (we're still mathematicians, we would like our methods to work in the general case).
- Benders ran in 7 minutes 26 seconds and added 1979 cuts.
- Naive MIP ran in 37 minutes, 43 seconds.

Constructing Cuts

Abstract Benders Decomposition requires you to invent cuts based on specific knowledge about Θ_ω.
Can get very complex as Θ_ω gets more interesting.
However, if Θ_ω is another Mixed Integer Program, we have,

1. A full Benders Decomposition (not abstract)
2. A closed form solution for master problem cuts.

Benders Decomposition

$$\min_{\boldsymbol{x},\boldsymbol{y}} \boldsymbol{c}^\intercal \boldsymbol{x} + \boldsymbol{f}^\intercal \boldsymbol{y}$$

Subject to,

$$\begin{aligned} A\boldsymbol{x} + B\boldsymbol{y} &\leq \boldsymbol{b} \\ D\boldsymbol{y} &\leq \boldsymbol{d} \\ \boldsymbol{x} \geq 0, \boldsymbol{y} &\geq 0 \end{aligned}$$

Is a problem in our general form with $\Theta(\boldsymbol{y}^\star)$ given by the following MIP,

$$\min_{\boldsymbol{x}} \boldsymbol{c}^\intercal \boldsymbol{x}$$

Subject to,

$$\begin{aligned} A\boldsymbol{x} &\leq (b - B\boldsymbol{y}^\star) \\ \boldsymbol{x} &\geq 0 \end{aligned}$$

Duality

The Dual of the BSP is,

$$\max_{\boldsymbol{u}} (\boldsymbol{b} - B\boldsymbol{y}^\star)^\intercal \boldsymbol{u}$$

Subject to,

$$A^\intercal \boldsymbol{u} \leq \boldsymbol{c}$$
$$\boldsymbol{u} \leq 0$$

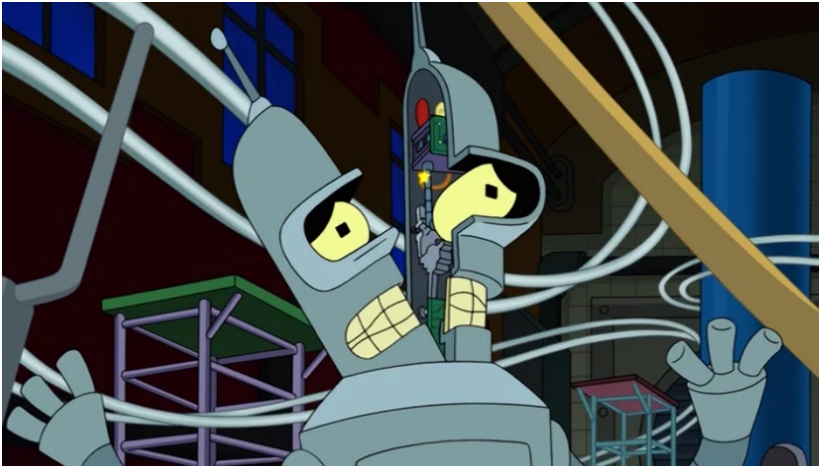
Duality theory says the optimal objective values of the dual and primal problem are equal. And that for all feasible solutions the objective of the primal is $\geq$ the objective of the dual. Which implies that,

$$\boldsymbol{c}^\intercal \boldsymbol{x} = (b - B\boldsymbol{y}^\star)^\intercal \boldsymbol{u}^\star$$
$$\boldsymbol{c}^\intercal \boldsymbol{x} \geq (b - B\boldsymbol{y})^\intercal \boldsymbol{u}^\star$$

so the below is a valid feasibility cut.

$$\theta \geq (b - B\boldsymbol{y})^\intercal \boldsymbol{u}^\star$$

Thank you



I have never watched a complete episode of Futurama in my life. I condone the actions of this character if they are good and condemn them if they are bad.