

A Linear Programming Representation of the 1 Dimensional PMCLP-PCR Problem

Nathan Wycoff

July 14, 2018

1 Intuitive Introduction of Solution Space

In order to formulate the PMCLP-PCR in canonical optimization format, we abandon the vector-space-valued solution-space in favor for a different representation.

To build intuition, let's begin by taking a journey to 2D problems. We'll go back to the 1D case shortly.

The bottom and left sides of each Demand Zone (DZ) act as axes. Instead of the natural axes for Cartesian space, these axes are of finite length.

Let w_j and h_j represent the width and height of the j^{th} DZ. The solution set consists of the following Cartesian product:

$$x \in [0, w_1]^2 \times [0, h_1]^2 \times [0, w_2]^2 \times [0, h_2]^2 \times \dots \times [0, w_p]^2 \times [0, h_p]^2$$

Each pair of coordinates represents the “from” and “to” of the rectangle along each axis.

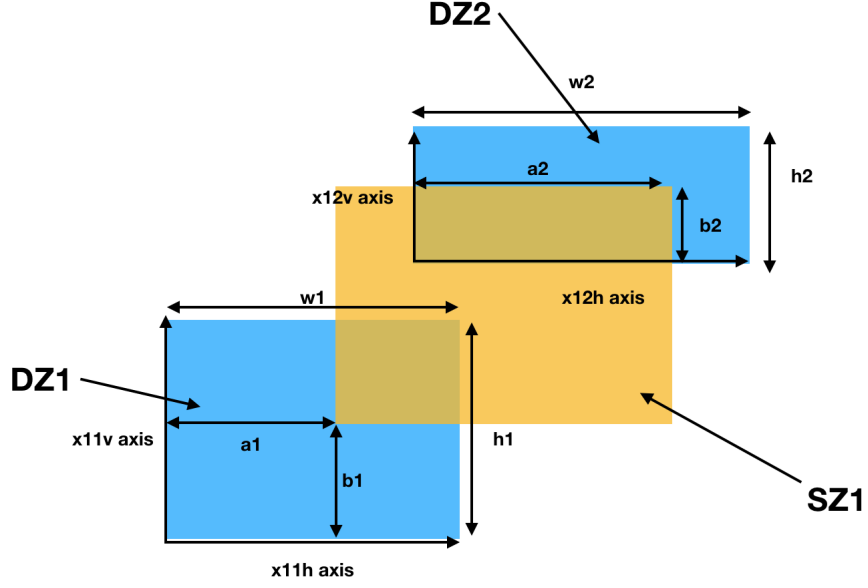
The natural basis on such a space is the $2p$ elements with 0's in all components except for a 1 in a certain component.

For the purposes of nomenclature, this space will be referred to as Demand Space.

Some elements in this space can be mapped back to candidates for optimal rectangles, by creating the rectangle that passes through all the points.

We denote our decision variables as $x_{i,j,\{h,v\},\{f,t\}}$ to denote the location of the i^{th} SZ on the j^{th} DZ. If there is no overlap, all variables are 0. If there is overlap along a certain axis (v for vertical and h for horizontal, though this could be generalized for higher dimensional space), the $x_{...f}$ variable denotes the “from”, where the overlap begins, and the $x_{...t}$ variable denotes the “to”, where the SZ and DZ no longer meet, again, all in terms of the “axes” of demand space: the bottom and left sides for each DZ.

Examples will be illuminating:



In this case, the supply zone (orange) partially overlaps two demand zones. We can represent this SZ in demand space:

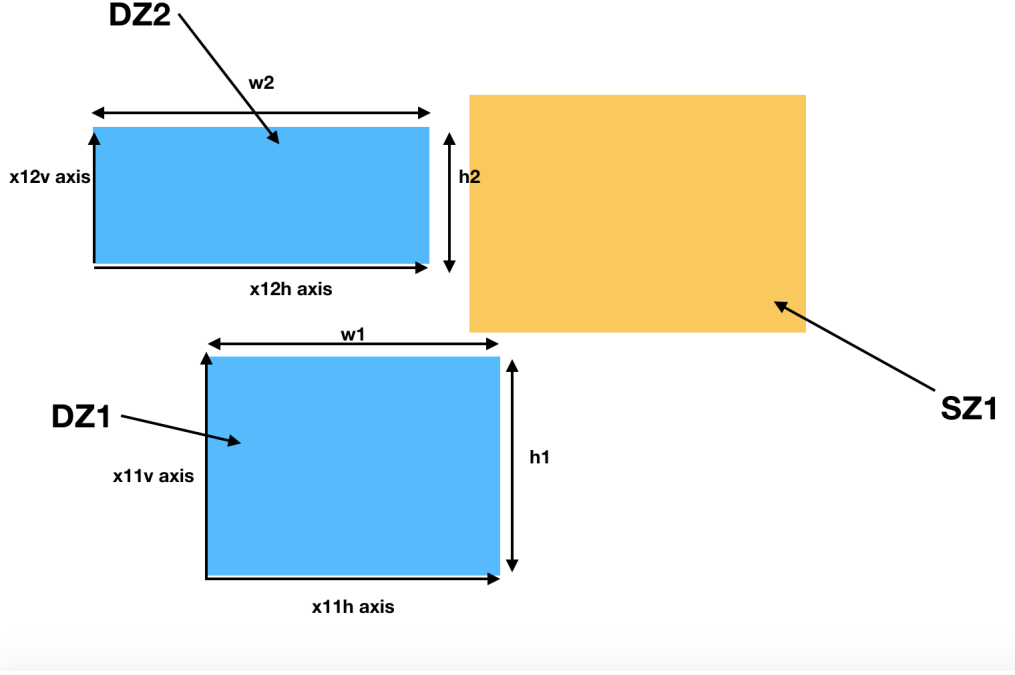
Since the SZ begins $a1$ along DZ1's horizontal axis, we set $x_{1,1,h,f} = a1$, and since it goes all the way off the right side, we set $x_{1,1,f,t} = w1$, the width of the DZ, and the maximum value of the variable.

Similarly, it begins $b1$ along the vertical axis of DZ1, so we set $x_{1,1,v,f} = b1$, and again goes off the top, so we set $x_{1,1,v,t} = h1$.

There is also overlap with DZ2 which must be accounted for. Since the SZ covers the "origin" of DZ2, we have that $x_{1,2,h,f} = x_{1,2,v,f} = 0$. Since it doesn't cover the entire DZ, we have that $x_{1,2,h,t} = a2$ and $x_{1,2,v,t} = b2$.

Hypothetically, if there had been a 3rd DZ that was offscreen (not covered at all by the SZ), we would have that $x_{1,3,h,f} = x_{1,3,h,t} = x_{1,3,v,f} = x_{1,3,v,t} = 0$.

Note that this "alternative basis", demand space, does not have the ability to uniquely represent all feasible solutions to the original problem. For instance:



The SZ does not overlap with any DZ's. It, along with all other SZ solutions which do not overlap at all with a DZ, may be represented by the zero vector in demand space.

This nonuniqueness is not a problem, as we can pick any corresponding rectangle to get the same optimal value. A similar issue will occur if an SZ is larger than a DZ that it covers in either height or width.

Note also that many members of Demand Space do not have a rectangular representation. We add constraints to make sure that these elements are not also elements of our feasible space.

2 Index Sets

Back to the 1D problem:

1. Supply Zone index $i \in \{1, \dots, p\}$
2. Demand Zone index $j \in \{1, \dots, n\}$

3 Parameters

1. Weight of DZ j : v_j .
2. Length of DZ j : l_j .
3. Beginning point of DZ on axis j : c_j .
4. Length of SZ i : s_i .

4 Decision Variables

1. Where SZ i begins along DZ j : $x_{i,j,f}$.
2. Where SZ j ends along DZ j : $x_{i,j,t}$.

There are therefore $2np$ many variables at this stage. Note that unlike the 2D example, there is no “width” or “height” index, as there is only 1 axis in 1D. In general dimensions, we would have additional variables to record distance along each dimension.

5 Objective Function

The objective is to maximize weighted coverage. The coverage function is complicated by the fact that multiple coverage must be accounted for. We can determine the amount of coverage a configuration of SZ’s has for a particular DZ by taking the maximum extent, and subtracting from it the minimum extent (call γ_j the coverage of DZ j):

$$\gamma_j = \max_{\forall i} \{x_i, j, t\} - \min_{\forall i} \{x_i, j, f\}$$

The objective is to maximize the sum of these coverages, weighted by the DZ weights:

$$\sum_{j=1}^n v_j \gamma_j = \sum_{j=1}^n v_j (\max_{\forall i} \{x_i, j, t\} - \min_{\forall i} \{x_i, j, f\})$$

6 Linearization

Unfortunately, maximizing a maximum is inherently nonconvex. We can, however, write this as a MIP. Let’s see how to do this in general:

If our initial problem is of the form:

$$\text{Maximize: } \max_i x_i$$

S.T.

$$x_i \in [0, l_i]$$

We can rewrite it as such, adding continuous variables z_i and binary variables b_i :

$$\text{Maximize: } \sum_i z_i$$

S.T.

1. $z_i \in [0, x_i]$
2. $z_i \in [0, l_i b_i]$
3. $\sum b_i = 1$

4. b_i binary

In our case, the objective is of the form

$$\sum_{j=1}^n v_j \gamma_j = \sum_{j=1}^n v_j (\max_{\forall i} \{x_i, j, t\} - \min_{\forall i} \{x_{i,j,f}\})$$

First, rewrite it as such:

$$\sum_{j=1}^n v_j \gamma_j = \sum_{j=1}^n v_j (\max_{\forall i} \{x_i, j, t\} + \max_{\forall i} \{-x_{i,j,f}\})$$

Introduce variables $y_{i,j,t}$ and $y_{i,j,f}$ which will replace the two optima. Our new objective is as such:

$$\sum_{j=1}^n v_j \gamma_j = \sum_{j=1}^n v_j (\sum_{i=1}^p y_{i,j,t} + \sum_{i=1}^p y_{i,j,f})$$

Though we need to add constraints. To do so, also introduce binary variables $a_{i,j,t}, a_{i,j,f}$ corresponding to each term in each maximum. These will be constrained such that only one may be nonzero, which will correspond to the term which is maximal.

1. Max less than items:

- (a) $y_{i,j,t} - x_{i,j,t} \leq 0$
- (b) $y_{i,j,f} + x_{i,j,f} \leq 0$

2. Max zero if binary zero:

- (a) $y_{i,j,t} - s_i a_{i,j,t} \leq 0$
- (b) $-y_{i,j,f} - s_i a_{i,j,f} \leq 0$

3. Only 1 max active at a time:

- (a) $\sum_{i=1}^p a_{i,j,t} = 1$
- (b) $\sum_{i=1}^p a_{i,j,f} = 1$

4. Bounds on variables:

- (a) $a_{i,j,t} \in \{0, 1\}$
- (b) $a_{i,j,f} \in \{0, 1\}$
- (c) $y_{i,j,t} \in [0, \infty]$
- (d) $y_{i,j,f} \in [-\infty, 0]$
- (e) $x_{i,j,t} \in [0, s_i]$
- (f) $x_{i,j,f} \in [0, s_i]$

7 Constraints

1. Don't fall off the DZ axes ($2pn$ constraints):

$$x_{i,j,f} \leq l_j \forall i$$

$$x_{i,j,t} \leq l_j \forall i$$

2. We need DZ axis variables to map back to rectangles that are within specifications. This constraint may be phrased as such:

$$x_{i,j,t} - x_{i,j,f} > 0 \implies x_{i,j,t} + c_j - x_{i,j',f} - c_{j'} \leq s_i \forall i, j, j'$$

That is, if a DZ is “active” with respect to demand zone i , we need to make sure that it doesn't stretch the SZ more than it can bear.

We can partially linearize this constraint as follows. Introduce a binary variable $b_{i,j}$ for each DZ-SZ relationship. Then, introduce this constraint:

$$x_{i,j,t} - x_{i,j,f} \leq s_i b_{i,j}$$

This gives us that if DZ j is active wrt SZ i , $b_{i,j} = 1$ (Note: this is not an if and only if relationship. I believe that this is not problematic, since it will be iff at optimality. This deserves more investigation).

So now we have a switch for each DZ-SZ pair, giving 1 if the pair is active. We can use this to create conditional constraints which are linear. Let R denote the range of possible values in the original space, $R = \max_i \{c_i + l_i\} - \min_i \{c_i\}$.

$$x_{i,j',t} + c_{j'} - x_{i,j,f} - c_j + Rb_{i,j} + Rb_{i,j'} \leq s_i + 2R$$

Let's examine these constraints for each possible value of $b_{i,j}$ and $b_{i,j'}$. If one of the two is equal to 1, we can remove the corresponding R from both sides, but there will still be an R on the RHS. By design, a quantity of at least R on the RHS means this constraint will always be satisfied (and s_i , being a length, is nonnegative). The situation is similar if both b 's are zero. However, if both are one, all terms involving R cancel, and we are left with the original, desired constraint.

But as mentioned earlier, if $x_{i,j,t} - x_{i,j,f} = 0$, there's no guarantee that the corresponding $b_{i,j} = 0$; it could be 1. This means that we could have more constraints active than necessary. However, removing this constraint would lead to a decrease in the cost, and will therefore occur at optimality.

3. No “negative extent” rectangles (np constraints):

$$x_{i,j,t} - x_{i,j,f} \geq 0 \forall i, j$$

4. Nonnegativity ($2np$ constraints):

$$x_{i,j,t} \geq 0 \forall i, j$$

$$x_{i,j,f} \geq 0 \forall i, j$$

8 In General Dimension

In general, we are trying to maximize the hypervolume of DZ's covered by SZ's. In the rectangular case, hypervolume coverage is simply the product of marginal coverages. So extension to the general dimensional case should be easy from a mathematical perspective: simply take the product of D many 1 dimensional objectives, where D is the dimensionality of the space. Everything else may be handled marginally.