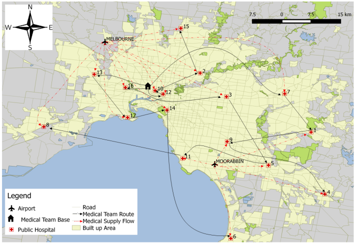
Informal title: Optimisation and Disaster Management

N. Sukhorukova and Behrooz Bodaghi (Swinburne University of Technology)



Application: Disaster management

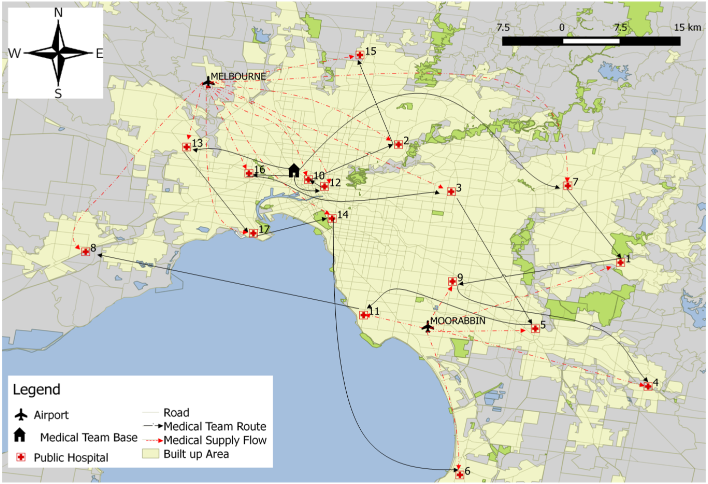


Figure: Hospitals.

Formulation and difficulties

Introduce binary variables: $x_{ij} = \{0, 1\}$. $x_{ij} = 1$ if location i is covered by hospital j.

Convenient, natural, but not so easy to solve. It is much easier to solve problems where the values of the variables are not restricted to integers.

Mixed-Integer Linear programming problems. What can we do?

- Reformulate it differently.
- Cluster the locations to have smaller size data.

There is a better way.

Transportation Problem

Goods are produced at m factories (sources)

$$S_1, \ldots, S_m$$

and sold at n markets (destinations):

$$D_1, \ldots, D_n.$$

The supply available at source S_i is $s_i \geq 0$ units, the demand at destination D_j is $d_j \geq 0$ units and the transportation cost of one unit from S_i to D_j is $c_{ij} \geq 0$. We have to identify which sources should supply which destinations to minimise total transportation costs. Let x_{ij} be the number of units to be sent from S_i to D_j. Then the corresponding optimisation problem can be formulated as follows:

$$\min \sum_{i=1}^{m} \sum_{j=1}^{n} c_{ij} x_{ij}$$

$$\sum_{j=1}^{n} x_{ij} \leq s_i, \ \ i = 1, \ldots, m; \sum_{i=1}^{m} x_{ij} \geq d_j, \ \ j = 1, \ldots, n;$$

$$x_{ij} \geq 0, \ \ i, j = 1, \ldots, n.$$

Transportation problem: integer solution

It is well known that if all supplies and demands are integers, then there exists an optimal solution x_{ij}, which is integer. This is important for many applications where the units (for example, computers, cars, people) can not be split. In general, integer and mixed-integer programming problems are much harder to solve than linear programming problems. It is also well known that the simplex method applied to a transportation problem, terminates at an optimal solution that is also integer.
This can be proved, for example, by demonstrating that the constraint matrix is totally unimodular.

Integer programming

A comprehensive overview on integer programming: Alexander Schrijver (1986). Theorem 19.3 of this book (p. 268) covers the conditions when the vertices of the feasible sets are integers and therefore an optimal solution found at a vertex is integer. In this paper, we only use conditions (i), (iii) and (iv) of the theorem (originally proved by Hoffman and Kruskal (1956) and Ghouila (1962). A simplified version of this theorem, formulated for this study, is as follows (next slide).

Theorem

Theorem

Let A be a matrix with entries $\{0, 1, -1\}$. Then the following are equivalent:

1. *matrix A is totally unimodular, that is each square submatrix of A has determinant 0, 1 or -1;*
2. *for all integral vectors a, b, c and d the polyhedron*
   $$\{x | c \leq x \leq d,\ a \leq Ax \leq b\}$$
   has only integral vertices;
3. *each collection of columns of A can be split into two parts so that the sum of the columns in one part minus the sum of the columns in the other part is a vector with entries $\{0, 1, -1\}$;*

More results

The first condition of Theorem 1 is usually used as a definition for totally unimodular matrices. The class of totally unimodular matrices is closed under a number of operations. We need the following ones:

- transposition;
- multiplication a row (column) by -1.

Also, matrix A is totally unimodal if and only if matrix $[IA]$ (where I is an identity matrix of the corresponding dimension) is totally unimodular.

Comsumable resources

It is enough to think about incident points as "Markets'', while the processing centres are "Factories''. The transportation costs are "processing and transportation time''.

The feasible set (without sign constraints and integer requirement):

$$\begin{bmatrix} -I_n & -I_n & -I_n & \dots & -I_n \\ e_n & 0_n & 0_n & \dots & 0_n \\ 0_n & e_n & 0_n & \dots & 0_n \\ 0_n & 0_n & e_n & \dots & 0_n \\ \vdots & \vdots & \vdots & \ddots & \vdots \\ 0_n & 0_n & 0_n & \dots & e_n \end{bmatrix} X \leq b, \tag{1}$$

where

- $b \in \mathbb{R}^{(m+n)}$ represents the corresponding demands and supplies and therefore b is integral;
- $X \in \mathbb{R}^{mn}$ is the vector of decision variables;
- I_n is an identity matrix of size m;
- $e_n = (1, 1, \dots, 1) \in \mathbb{R}^n$; $0_n \in \mathbb{R}^n = (0, 0, \dots, 0)$;
- the system matrix $A \in \mathbb{R}^{(n+m)\times(mn)}$.

Consumable: Theorem

Theorem

The system matrix A from (1) is totally unimodular.

Proof: Consider matrix B obtained from A^T by multiplying the first n columns of A^T by -1. Matrix A is totally modular if and only if matrix B is totally unimodular.

Assign the first n columns of B to part I and the remaining columns to part II and assume that one or more columns may be removed from the total collection of columns. The sum of the columns in part I is an (mn)-dimensional vector S_1 whose components are 0 or 1. The sum of the columns in part I is an (mn)-dimensional vector S_2 whose components are 0 or 1. Therefore the components of $S_1 - S_2$ are 0, 1 or -1 and hence, by Theorem 1, we conclude that matrices B and A are totally unimodular and all the vertices of the feasible set have integer coordinates.

$\square$

An optimal integer solution to this problem can be found by applying the simplex method.

Non-consumable

In the case of non-consumable resources, the problem can also be formulated as an integer programming problem, where some of the summation constraints from a classical transportation problem constraints are replaced with maximisation. A mixed-integer formulation for the case of non-consumable resources is as follows

$$\min \sum_{i=1}^{m} \sum_{j=1}^{n} c_{ij} x_{ij} \tag{2}$$

subject to

$$\sum_{i=1}^{m} x_{ij} \geq d_j, \ j = 1, \ldots, n; \tag{3}$$

$$\max_{j=1,\ldots,n} x_{ij} \leq s_i, \ i = 1, \ldots, m; \tag{4}$$

$$x_{ij} \geq 0, \ i = 1, \ldots, n, \ j = 1, \ldots, m, \tag{5}$$

$$x_{ij} \text{ are integers}, \ i = 1, \ldots, m, \ j = 1, \ldots, n, \tag{6}$$

where $d_i, \ i = 1, \ldots, m$ are incident point demands and $s_j, \ j = 1, \ldots, n$ are processing centre capacities.

Non-consumable: LP

A relaxation of this problem, obtained by removing the last constraint (6), can be formulated as an LPP by replacing constraints (4) with equivalent systems of linear inequalities:

$$x_{ij} \leq s_j, \ j = 1, \dots, n, \ i = 1, \dots, m. \tag{7}$$

The feasible set of this problem can be formulated as follows (without sign constraints and integer requirement):

$$\begin{bmatrix} -I_n & -I_n & -I_n & \dots & -I_n \\ & & I_{mn} & & \end{bmatrix} X \leq b, \tag{8}$$

where

- $b \in \mathbb{R}^{(n(m+1))}$ represents the corresponding demands and supplies and therefore b is integral;
- $X \in \mathbb{R}^{mn}$ is the vector of decision variables;
- I_n is an identity matrix of size n;
- I_{mn} is an identity matrix of size mn;
- the system matrix $A \in \mathbb{R}^{n(m+1)\times(mn)}$.

Non-consumable: Theorem

Theorem

The system matrix A from (15) is totally unimodular.

Proof: A is totally unimodular if and only if matrix

$$B = \begin{bmatrix} I_n & I_n & I_n & \dots & I_n \end{bmatrix} \tag{9}$$

is totally unimodular and matrix B is totally unimodular if and only if I_n is totally unimodular: one can assign any collection of columns of I_n to part I and the remaining columns to part II. The difference of the corresponding columns sums contains 1 and -1 as the components.

□

Therefore, similar to the case of consumable resources, we can reduce an integer linear programming problem to an LPP whose vertices are integral.

k-medoid or k-median

In this application, the distance matrix between all the incident points is given. The goal is to select k points in such a way that, after assigning all the remaining points to the nearest selected point (cluster centre), the total sum of distances between the points and centres is minimal. Each cluster centre is a relief centre, whose optimal location (selection among the incident points) is the objective. In this application, we assume that the demand of the incident points can be covered regardless of the allocation, since the main objective is to minimise the total distance. This kind of clustering is called k-medoid, was first proposed by Kaufman and Rousseeuw.

Relief centres

Assume that there are n demand points in total and the distance matrix

$$\mathbf{D} = \{d_{ij}\},\ i = 1, \dots, n,\ j = 1, \dots, n.$$

It is easy to see that this matrix is symmetric and its main diagonal consists of zeros. The goal is to select k points as relief centres. The decision variables are binary:

$$x_{ij} \in \{0, 1\},\ i = 1, \dots, n,\ j = 1, \dots, n$$

and $y_i \in \{0, 1\},\ i = 1, \dots, n$. Variable y_i is 1 if incident point i is treated as a relief centre, otherwise, this variable is zero. Variable x_{ij} is 1 if incident point i was assigned to point j.

Formulation

The corresponding optimisation problem is as follows:

$$\min \sum_{i=1}^{n} \sum_{j=1}^{n} d_{ij} x_{ij} \tag{10}$$

subject to

$$\sum_{i=1}^{n} x_{ij} = 1, \; j = 1, \ldots, n; \tag{11}$$

$$x_{ij} \leq y_i, \; i, j = 1, \ldots, n; \tag{12}$$

$$\sum_{i=1}^{n} y_i = k; \tag{13}$$

$$x_{ij}, \; y_i \in \{0, 1\}, \; i, j = 1, \ldots, n. \tag{14}$$

Constraints (11) ensure that each incident point is assigned to a single relief centre. Constraints (12) ensure that an incident point i can only be assigned to an incident point j if this point is also a relief centre. Finally, constraint (13) ensures that exactly k points are selected as relief centres.

k-medoid: Theorem

Theorem

All the vertices of the feasible set in k-medoid method have integer coordinates.

Proof: The set of constraints contains $(n+1)$ equalities and n^2+n inequalities (not counting sign constraints and integer requirements). Use these equalities to reduce the number of variables and obtain a simpler constraint matrix A.

Theorem: cont

Then the feasible set is as follows (without sign constraints and integer requirement):

$$\begin{bmatrix} & & & & \dots & e_n & e_n & \dots & e_n \\ & & I_{(n-1)n} & & \dots & -e_n & 0_n & \dots & 0_n \\ & & & & \dots & 0_n & -e_n & \dots & 0_n \\ -I_n & -I_n & \dots & -I_n & \dots & 0_n & 0_n & \dots & -e_n \end{bmatrix} X \leq b, \tag{15}$$

where

- $b \in \mathbb{R}^{(n^2-1)}$ contains integral numbers only (1, -1, 0 or k);
- $X \in \mathbb{R}^{n^2-1}$ is the vector of decision variables;
- $e_n = (1, 1, \dots, 1)^T \in \mathbb{R}^n$;
- I_n is an identity matrix of size n;
- $I_{(n-1)n}$ is an identity matrix of size $(n-1)n$;
- the system matrix $A \in \mathbb{R}^{n^2 \times (n^2-1)}$.

Theorem: cont

Matrix A contains n blocks of rows (n rows in each block). Add the first $(n-1)$ blocks to the final block of rows (keeping the same order of rows in each block). By doing this, the obtained right hand side vector remains integer, while the last block of rows consists of zeros. If the problem is feasible (that is $k \geq 1$), the final block of rows can be removed. Then the remaining system matrix is

$$B = \left[I_{n(n-1)} C \right], \tag{16}$$

where

$$C = \begin{bmatrix} e_n & e_n & e_n & \dots & e_n & e_n \\ -e_n & 0_n & 0_n & \dots & 0_n & 0_n \\ 0_n & -e_n & 0_n & \dots & 0_n & 0_n \\ 0_n & 0_n & 0_n & \dots & -e_n & 0_n \end{bmatrix},$$

where $e_n = (1, 1, \dots, 1)^T \in \mathbb{R}^n$ and $0_n = (0, 0, \dots, 0)^T \in \mathbb{R}^n$, $C \in \mathbb{R}^{n(n-1) \times n-1}$.

Theorem: cont

To complete the proof, it is enough to show that matrix C is totally unimodular. Indeed, for any collection of m columns in C ($m \leq n$), the columns can be split into two parts: it is enough to assign any $(m-1)/2$ columns (when m is odd) or $m/2$ columns (when m is even) to one of the parts and the remaining columns in the other part. Then the sum of the columns in part I minus the sum of the columns in part II contains 1, -1 or 0.

$\square$

Therefore, it is enough to apply the simplex method to solve this problem and obtain an integer optimal solution.

Has someone discovered it before?

Note that Theorem 4 is an important result, since it allows one to avoid integer solvers when applying k-medoid method. This results is of interest of cluster analysis and allocation and therefore has many other potential applications.

To our best knowledge, there is no result in the literature confirming that all the vertices of the linear relaxations of k-medoid formulations are integers. One relevant study (**Relax, no need to round: integrality of clustering formulations**) conducts a comprehensive numerical study on k-means and k-medoid, where most experimental results confirm that the relaxation produce integer (or nearly integer) results: "LP relaxation remains integral with high probability". In the same paper, the authors talk about "generically unique solutions", since "no constraint is parallel to the objective function ".

In our paper, we provide an analytical proof that the vertices are integral and therefore the classical implementation of the simplex method always terminates at an integer solution and therefore the corresponding problems can be solved efficiently.

Computational experiments, difficulties and discussions

- RELAX, NO NEED TO ROUND: INTEGRALITY OF CLUSTERING FORMULATIONS
- Disaster management.
- Signal processing.

How can you confirm that the Simplex method is implemented in the classicall way?