Table 1: Decision Variables

Notation	Description	Variable type
x_j	1 if CL j is opened, 0 otherwise	Binary
y_{kj}	1 if all individuals in PN k are assigned to CL j, 0 otherwise	Binary
z_s	1 if total cost meets the budget in scenario s, 0 otherwise	Binary
t_{js}	Amount of exceeded demand assigned to CL j in scenario s	Continuous

Table 2: Parameters

Parameter	Definition
w_k	Inhabitants in population node(PN) $k \in K$
p_s	Occurrence probability of scenario $s \in S$
c_j	Pre-defined capacity of candidate location(CL) $j \in J$
b_k^s $(0 \leq b_k^s \leq w_k)$	Demand occurs in PN k under scenario s
d_{kjs}	Distance between PN k and CL j in scenario s
η_j	Capacity expansion cost per unit for CL j
f_j	Establishment cost of a shelter in CL j
C	Budget that limits the total cost
α	service level

1 Formulation of the Model

The formulations below give the deterministic equivalents of our chance-constrained stochastic programs. Table 1 lists the decision variables of the models. Table 2 shows the parameters that used in model.

***Ex ante* problem:**

$$\min\ \mu + 2\lambda\mu G = \frac{1}{N} \sum_{k\in K}\sum_{s\in S}\sum_{j\in J} p_s\, b_k^s\, y_{kj} d_{kjs} + \frac{\lambda}{N^2} \sum_{k\in K}\sum_{k'\in K} (\tau^+_{k,k'} + \tau^-_{k,k'}) \quad \text{s.t.} \tag{1}$$

$$\tau^+_{k,k'} - \tau^-_{k,k'} = \sum_{s\in S} p_s \left(w_{k'}\, b_k^s \sum_{j\in J} y_{kj} d_{kjs} - w_k\, b_{k'}^s \sum_{j\in J} y_{k'j} d_{k'js} \right) \qquad \forall k,\ k' \in K \tag{2}$$

$$y_{kj} \le x_j \qquad \forall k \in K,\ j \in J \tag{3}$$

$$\sum_{j\in J} y_{kj} = 1 \qquad \forall k \in K \tag{4}$$

$$\sum_{k\in K} b_k^s\, y_{kj} \le t_{js} + c_j \qquad \forall j \in J,\ s \in S \tag{5}$$

$$\sum_{j\in J} f_j x_j + \eta_j\, t_{js} \le C + M_0(1 - z_s) \qquad \forall s \in S \tag{6}$$

$$\sum_{s\in S} p_s z_s \ge \alpha \tag{7}$$

$$x_j \in \{0,1\} \quad \forall j \in J, \quad z_s \in \{0,1\} \quad \forall s \in S$$

$$y_{kj} \in \{0,1\} \qquad \forall k \in K,\ j \in J$$

$$t_{js} \ge 0 \qquad \forall j \in J, s \in S \tag{8}$$

$$\tau^+_{k,k'}, \tau^-_{k,k'} \ge 0 \qquad \forall k,\ k' \in K \tag{9}$$

To reformulate the objective function in linear terms, we introduced auxiliary variables $\tau^+_{k,k'}$ and $\tau^-_{k,k'}$ as discussed above and added constraints (2). Constraint (30) ensures that a PN can only be served by an opened shelter. Constraint (31) guarantees that each PN is assigned to exactly one shelter. Constraint (32) calculates the exceeded capacity for each CL in each scenario. Constraint (33) defines the indicator variable z_s for the satisfaction of the budget constraint in scenario s. If we meet the budget in scenario s, then z_s has no reason to become zero; otherwise, z_s will be zero. Our chance constraint (34) imposes that the budget should be met with probability at least α. Finally, the large number M_0 can be replaced by $\sum_{j\in J} (f_j + \eta_j\,(N - c_j))$.

***Ex post* problem:**

$$\min\ \sum_{s\in S}\sum_{k\in K}\sum_{j\in J} \frac{1}{N'_s}\, p_s b_k^s\, y_{kj} d_{kjs} + \lambda \sum_{s\in S} p_s\, \frac{1}{N'^2_s} \sum_{k\in K}\sum_{k'\in K} b_k^s\, b_{k'}^s\, (\rho^+_{kk's} + \rho^-_{kk's}) \quad \text{s.t.} \tag{10}$$

$$\rho^+_{kk's} - \rho^-_{kk's} = \sum_{j\in J} y_{kj} d_{kjs} - \sum_{j\in J} y_{k'j} d_{k'js} \quad \forall s \in S,\ k, k' \in K \tag{11}$$

$$\rho^+_{kk's}\, \rho^-_{kk's} \ge 0 \quad \forall k,\ k' \in K,\ s \in S \tag{12}$$

constraints (30) – (35)

where $\rho^+_{kk's}$ and $\rho^-_{kk's}$ are three-indexed nonnegative auxiliary variables that help us to make the objective function linear.

Combined problem:

$$\min\ \gamma \left(\frac{1}{N} \sum_{k\in K}\sum_{s\in S}\sum_{j\in J} p_s\, b_k^s\, y_{kj} d_{kjs} + \frac{\lambda}{N^2} \sum_{k\in K}\sum_{k'\in K} (\tau^+_{k,k'} + \tau^-_{k,k'}) \right)$$
$$+ (1-\gamma) \left(\sum_{s\in S}\sum_{k\in K}\sum_{j\in J} \frac{1}{N'_s}\, p_s b_k^s\, y_{kj} d_{kjs} + \lambda \sum_{s\in S} p_s \frac{1}{N'^2_s} \sum_{k\in K}\sum_{k'\in K} b_k^s\, b_{k'}^s\, (\rho^+_{kk's} + \rho^-_{kk's}) \right) \tag{13}$$

s.t. constraints (2) – (9) and (29) – (36).

In the objective function, γ is the weight of the *ex ante* measure which is a constant parameter. The special cases $\gamma = 1$ and $\gamma = 0$ yield the *ex ante* and the *ex post* problem, respectively.

2 Solution method

We utilize the benders decomposition method to solve the MIP model. We will define mater problem and two sub-problems in the rest of this section. Master problem will decide on the x variables, first sub-problem deals with y variables to generate feasible assignment according to the objective function, and finally, the second sub-problem ensure the feasibility of current x and y variables by deciding about capacity extension variables (t) in order to meet the chance constraint.

2.1 Master Problem

In master problem we just decide about x variables as binary variables. In each iteration of the algorithm it is possible that one cut will be added to the master problem by each of the sub-problems. When the fist sub-problem is infeasible or when the second sub-problem is feasible. The algorithm continued until the master problem become infeasible.
Master **problem:**

$$\begin{aligned} \min\ & 0 \\ & \textbf{s.t.} \\ & x_j \in \{0,1\} \quad \forall j \in J \end{aligned}$$

Feasibility Cut to add to Master:

$$\sum_{j\in J_1} (1 - x_j) + \sum_{j\in J_0} x_j >= 1$$

Where J_1 is the set of indices that current x solution are one, and J_1 is the set of all indices that current x solution are zero. These cuts are called combinatorial benders cut introduced by Codato and Fischetti [2006]. Note that we can not use normal dual cuts because both sub-problem are mixed integer and we can't take their dual.

2.2 First sub-problem

***First sub-problem* :**

$$\min\ \gamma\left(\frac{1}{N}\sum_{k\in K}\sum_{s\in S}\sum_{j\in J}p_s\, b_k^s\, y_{kj}d_{kjs}+\frac{\lambda}{N^2}\sum_{k\in K}\sum_{k'\in K}(\tau^+_{k,k'}+\tau^-_{k,k'})\right)$$
$$+(1-\gamma)\left(\sum_{s\in S}\sum_{k\in K}\sum_{j\in J}\frac{1}{N'_s}\,p_s b_k^s\, y_{kj}d_{kjs}+\lambda\sum_{s\in S}p_s\,\frac{1}{N'^2_s}\sum_{k\in K}\sum_{k'\in K}b_k^s\, b_{k'}^s\,(\rho^+_{kk's}+\rho^-_{kk's})\right)$$

s.t.

$$\tau^+_{k,k'}-\tau^-_{k,k'}=\sum_{s\in S}p_s\left(w_{k'}\, b_k^s\sum_{j\in J}y_{kj}d_{kjs}-w_k\, b_{k'}^s\sum_{j\in J}y_{k'j}d_{k'js}\right)\qquad\forall k,\ k'\in K$$
$$\rho^+_{kk's}-\rho^-_{kk's}=\sum_{j\in J}y_{kj}d_{kjs}-\sum_{j\in J}y_{k'j}d_{k'js}\quad\forall s\in S,\ k,k'\in K$$
$$y_{kj}\le\bar{x_j}\qquad\forall k\in K,\ j\in J$$
$$\sum_{j\in J}y_{kj}=1\qquad\forall k\in K$$
$$0\le y_{kj}\le 1\qquad\forall k\in K,\ j\in J$$
$$\tau^+_{k,k'},\tau^-_{k,k'}\ge 0\qquad\forall k,k'\in K\tag{14}$$
$$\rho^+_{kk's},\rho^-_{kk's}\ge 0\qquad\forall k,k'\in K,\ s\in S\tag{15}$$

Dual of the First sub-problem

$$\max\ \sum_{k\in K}\sum_{j\in J}(v_{kj}\hat{x}_j+r_{kj})+\sum_{k\in K}w_k\tag{16}$$

s.t.

$$\sum_{k'\neq k}u^1_{kk'}(a_{kk'j}-a_{k'kj})+\sum_{s\in S}\sum_{k'\neq k}u^2_{kk's}(d_{k'js}-d_{kjs})-\tag{17}$$
$$v_{kj}+\sum_{k\in K}w_k-r_{kj}\le\gamma(\frac{1}{N}\sum_{s\in S}p_s\, b_k^s\, d_{kjs})+(1-\gamma)\left(\sum_{s\in S}\frac{1}{N'_s}\,p_s b_k^s\, d_{kjs}\right)\forall k\in K,\ j\in J$$
$$u^1_{kk'}\le\frac{\gamma\lambda}{N^2}\tag{18}$$
$$u^1_{kk'}\ge\frac{-\gamma\lambda}{N^2}\tag{19}$$
$$u^2_{kk's}\le\ \frac{(1-\gamma)\lambda\, p_s\, b_k^s\, b_{k'}^s}{N'^2_s}\tag{20}$$
$$u^2_{kk's}\ge\ \frac{-(1-\gamma)\lambda\, p_s\, b_k^s\, b_{k'}^s}{N'^2_s}\tag{21}$$
$$v_{kj},r_{kj}\ge 0\qquad\forall k\in K,\ j\in J$$
$$\tag{22}$$

Let's define

$$a_{kk'j}=\sum_{s\in S}\sum_{j\in J}p_s\,(w_{k'}\, b_k^s\, d_{kjs})$$

Before adding any cut to the first sub-problem it is always feasible. If there is at least one open facility then the first sub-problem just assigns the PNs to that open facility and make sure that all PNs are assigned. Furthermore the optimal solution of this sub-problem minimize the primal objective function. Therefor, if this solution is feasible according to the resulting second sub-problem then it well be the best possible PNs - CLs assignment. In this case the algorithm will add a new combinatorial cut to the master problem to explore other values for x variables. However, in case that the second sub-problem become infeasbile the algorithm add a combinatorial cut to the first sub-problem on y variables to avoid resulting the same solutions in future solve of first sub-problem. We will continue adding these cuts until we found a solution from first sub-problem that have feasible second sub-problem. Or the first sub-problem become infeasible due to these combinatorial cuts.
Finally the algorithm continues until the master problem became infeasible. After the termination of the algorithm the solution that yields best objective value for the first sub-problem and have feasible corresponding second sub-problem, will be the optimal solution.
Feasibility Cut to add to Subproblem:

$$\sum_{j \in I_1} (1 - y_{j,i}) + \sum_{j \in I_0} y_{j,i} >= 1$$

I_1 is the set of indices where current y variables are one,I_0 and is the set of indices that current y variables are zero.

2.3 Second sub-problem

The second sub-problem
Second sub-problem:

$$\begin{aligned}
\min\ & 0 \\
& \textbf{s.t.} \\
& \sum_{k \in K} b_k^s\, \bar{y_{kj}} \le t_{js} + c_j \qquad \forall j \in J,\ s \in S && (23) \\
& \sum_{j \in J} f_j \bar{x_j} + \eta_j\, t_{js} \le C + M_0(1 - z_s) \qquad \forall s \in S && (24) \\
& \sum_{s \in S} p_s z_s \ge \alpha && (25) \\
& t_{js} \ge 0 \qquad \forall j \in J, s \in S \\
& z_s \in \{0, 1\} \quad \forall s \in S
\end{aligned} \tag{26}$$

The second sub-problem is a mixed integer problem that we just need to find a feasible solution for it. In order to meet constraint (25) we need to minimize the number of z variables with zero value. Therefore, for each $s \in S$ we should minimize the left side of (24). The only variable in that expression is t variables furthermore notice that the coefficient of them are positive. As a result, we can clam that if the minimize value for the t variables lead to violating the 25 then the problem is infeasible. Using constraint 23 we can calculate the minimum value for each t as follow: $t_{js} = \max\left(0, \sum_{k \in K} b_k^s\, \bar{y_{kj}} - c_j\right)$

In case that we found a feasible solution for the second sub-problem we have to add a combinatorial cut to master problem. In other hand if the second sub-problem is infeasible then we have to add a combinatorial cut to first sub-problem.

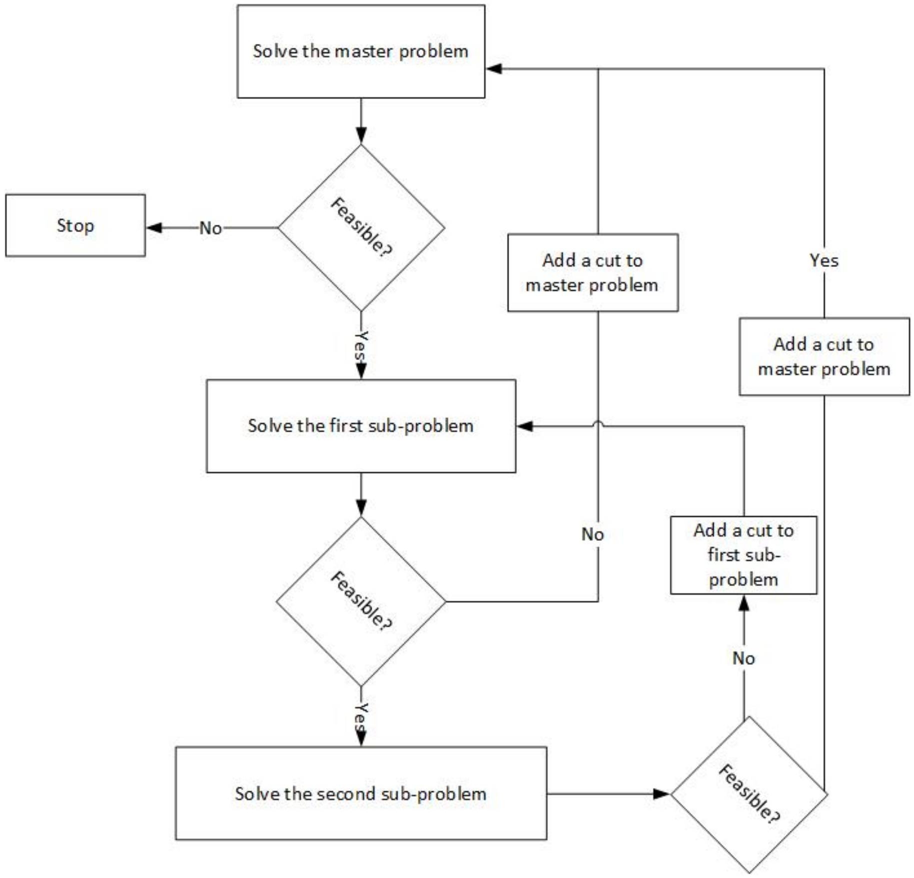

Figure 1: Flowchart of the algorithm

The bellow flowchart shows the overall structure of the algorithm.

References

G. Codato and M. Fischetti. Combinatorial benders' cuts for mixed-integer linear programming. *Operations Research*, 54(4):756–766, 2006.