# Online supplement to "A closest Benders cut selection scheme for accelerating the Benders decomposition algorithm"

Kiho Seo
Department of Industrial and Systems Engineering, KAIST, Daejeon, Republic of Korea, emffp1410@kaist.ac.kr

Seulgi Joung
Department of Industrial Engineering, Chonnam National University, Gwangju, Republic of Korea, sgjoung@jnu.ac.kr

Chungmok Lee
Department of Industrial and Management Engineering, Hankuk University of Foreign Studies, Youngin-si, Republic of Korea, chungmok@hufs.ac.kr

Sungsoo Park
Department of Industrial and Systems Engineering, KAIST, Daejeon, Republic of Korea, sspark@kaist.ac.kr

## Appendix A: Two-stage Stochastic Programming Formulation

The mathematical formulation for the `SNDlib` depends on the "model" of the problem. As explained in Section 6.1, the model determines the direction of demands (Demand model), link capacity (Link model), the existence of fixed charge (Fixed-charge model), demand routing model (Routing model), and other characteristics (Admissible path model, Hop limit model, etc.). For example, a problem with model specification "DBMNCANN" means that the demands are directed (D), arc facilities are installed bi-directionally (B), link capacities are given as modular (M), there are no facilities at nodes (N), the fractional demand flows are allowed (C), all simple routes are allowed (A), there is no fixed-charge cost (N), there are no survivability requirements (N). For the sake of simplicity, here we provide the mathematical formulation for the model "DBMNCANN" while the required modifications for the different models can be found in Orlowski et al. (2010)[1]. The formulation is defined with the following parameters and decision variables:

Sets and parameters:

$\mathcal{V}$ : set of nodes

$\mathcal{E}$ : set of (directed) edges

$\mathcal{D}$ : set of demands

$\mathcal{T}_e$ : set of installable capacity modules for each edge $e \in \mathcal{E}$

$\delta_i^+$ : set of out-ward edges from node $i \in \mathcal{V}$

$\delta_i^-$ : set of in-ward edges into node $i \in \mathcal{V}$

$K_e$ : unit routing cost for each edge $e \in \mathcal{E}$

$k_e^t$ : installation cost for capacity module $t \in \mathcal{T}_e$ for edge $e \in \mathcal{E}$

$h_d$ : demand value for demand $d \in \mathcal{D}$

[1] Orlowski S, Wessäly R, Pióro M, Tomaszewski A (2010) SNDlib: 1.0—Survivable Network Design Library. Networks 55(3):276–286

$O(d)$ : origin node for demand $d \in \mathcal{D}$

$D(d)$ : destination node for demand $d \in \mathcal{D}$

$C_e$ : pre-installed capacity on edge $e \in \mathcal{E}$

$o(e)$ : opposite direction edge of edge $e \in \mathcal{E}$

Decision variables:

$x_e^d$ : amount of demand flow for demand $d \in \mathcal{D}$ on edge $e \in \mathcal{E}$

$y_e^t$ : number of modules of type $t \in \mathcal{T}_e$ on edge $e \in \mathcal{E}$

$f_e$ : total amount of flows induced on edge $e \in \mathcal{E}$

$Y_e$ : total capacity on edge $e \in \mathcal{E}$

$$(\texttt{SNDlib}_{\text{DBMNCANN}}) \quad \min \quad \sum_{e \in \mathcal{E}} \left( K_e f_e + \sum_{t \in \mathcal{T}_e} k_e^t y_e^t \right) \tag{1}$$

$$\text{s.t.} \quad \sum_{e \in \delta_i^+} x_e^d - \sum_{e \in \delta_i^-} x_e^d = \begin{cases} h_d, \text{ if } i = O(d), \\ -h_d, \text{ if } i = D(d), \\ 0, \text{ otherwise.} \end{cases}, \quad \forall i \in \mathcal{V}, d \in \mathcal{D}, \tag{2}$$

$$Y_e = C_e + \sum_{t \in \mathcal{T}_e} c_e^t y_e^t, \quad \forall e \in \mathcal{E}, \tag{3}$$

$$f_e \geq \sum_{d \in \mathcal{D}} x_e^d, \quad \forall e \in \mathcal{E}, \tag{4}$$

$$f_e \geq \sum_{d \in \mathcal{D}} x_{o(e)}^d, \quad \forall e \in \mathcal{E}, \tag{5}$$

$$f_e \leq Y_e, \quad \forall e \in \mathcal{E}, \tag{6}$$

$$x_e^d \geq 0, \quad \forall e \in \mathcal{E}, d \in \mathcal{D}, \tag{7}$$

$$y_e^t \in \mathbb{Z}_+, \quad \forall e \in \mathcal{E}, t \in \mathcal{T}_e, \tag{8}$$

$$f_e \geq 0, \quad \forall e \in \mathcal{E}, \tag{9}$$

$$Y_e \geq 0, \quad \forall e \in \mathcal{E}. \tag{10}$$

Objective function (1) minimizes the sum of the total routing costs and total installation costs of capacity modules on edges. Constraints (2) are the so-called flow-balance constraints between the origin and destination nodes. The total capacity of edge $e \in \mathcal{E}$ is defined as the sum of the pre-installed capacity and installed modules' capacity by constraints (3). The bi-directional link model implies that the maximum of opposite direction flows should be considered as the link flow, which is ensured by constraints (4) and (5). The capacity constraints (6) relate the flow to installed module capacity on edge $e \in \mathcal{E}$. Constraints (6)-(10) ensure the domains of the decision variables.

The two-stage stochastic programming problem considers a set of demand scenarios $\mathcal{S}$. The first-stage decision determines the capacity of modules on edges (i.e., $y_e^t$ and $Y^e$) while the second-stage decision involves the demand routing (i.e., $x_e^d$ and $f_e$). For a given demand $d \in \mathcal{D}$, edge $e \in \mathcal{E}$, and scenario $s \in \mathcal{S}$, let $h_e^s$, $x_e^{d,s}$, and $f_e^s$ denote the demand, amount of demand flow on edge, and the maximum (bi-directional) total flow on edge, respectively. The two-stage stochastic programming problem for `SNDlib` can be formulated as:

$$(\texttt{SNDlib}^{sp}_{\text{DBMNCANN}}) \quad \min \quad \sum_{e \in \mathcal{E}} \left( \frac{1}{|\mathcal{S}|} \sum_{s \in \mathcal{S}} K_e f_e^s + \sum_{t \in \mathcal{T}_e} k_e^t y_e^t \right) \tag{11}$$

$$\text{s.t.} \quad \sum_{e \in \delta_i^+} x_e^{d,s} - \sum_{e \in \delta_i^-} x_e^{d,s} = \begin{cases} h_d^s, \text{ if } i = O(d), \\ -h_d^s, \text{ if } i = D(d), \\ 0, \text{ otherwise.} \end{cases} \quad \forall i \in \mathcal{V}, d \in \mathcal{D}, s \in \mathcal{S}, \tag{12}$$

(3),

$$f_e^s \geq \sum_{d \in \mathcal{D}} x_e^{d,s}, \quad \forall e \in \mathcal{E}, s \in \mathcal{S}, \tag{13}$$

**Table 1  Detailed results of Cplex for `SNDlib` with multiple scenarios**

| #Scenario | #Solved | %Gap | Time | Node |
|---|---|---|---|---|
| 2 | 24/25 | 0.91 (±0.05) | 417.0 (±1432.4) | 10838.0 (±15713.9) |
| 3 | 23/25 | 1.72 (±0.06) | 864.3 (±1987.7) | 7853.6 (±9124.4) |
| 4 | 23/25 | 1.72 (±0.06) | 1339.6 (±2361.5) | 12939.6 (±26407.8) |
| 5 | 21/25 | 2.59 (±0.08) | 1697.8 (±2662.9) | 12204.0 (±18406.9) |
| 6 | 20/25 | 3.04 (±0.08) | 1994.3 (±3022.1) | 8090.1 (±8873.8) |
| 7 | 20/25 | 2.91 (±0.08) | 2258.4 (±3114.8) | 9714.8 (±15054.1) |
| 8 | 19/25 | 2.99 (±0.07) | 2251.2 (±3064.2) | 13928.1 (±41608.0) |
| 9 | 18/25 | 3.69 (±0.09) | 2737.1 (±3214.2) | 108851.0 (±485190.9) |
| 10 | 17/25 | 3.71 (±0.08) | 2547.7 (±3281.5) | 14208.4 (±42204.9) |
| 25 | 10/16 | 7.20 (±0.10) | 3453.5 (±3411.8) | 5500.3 (±6971.9) |
| 50 | 8/16 | 10.50 (±0.14) | 3922.2 (±3405.6) | 2329.9 (±2153.3) |
| 100 | 7/16 | 14.97 (±0.19) | 5020.3 (±2863.3) | 2593.4 (±4055.8) |

$$f_e^s \geq \sum_{d \in \mathcal{D}} x_{o(e)}^{d,s}, \quad \forall e \in \mathcal{E}, s \in \mathcal{S}, \tag{14}$$

$$f_e^s \leq Y_e, \quad \forall e \in \mathcal{E}, s \in \mathcal{S}, \tag{15}$$

$$x_e^{d,s} \geq 0, \quad \forall e \in \mathcal{E}, d \in \mathcal{D}, s \in \mathcal{S}, \tag{16}$$

(8),

$$f_e^s \geq 0, \quad \forall e \in \mathcal{E}, s \in \mathcal{S}, \tag{17}$$

(10).

The problem minimizes the sum of the expected routing costs and installation costs by assuming all scenarios have the same realization probability.

## Appendix B: Detailed Results for `SNDlib` with Multiple Scenarios

Detailed results for `SNDlib` with multiple scenarios are given in the following Tables 1 and 2. For each number of generated scenarios (column "#Scenario"), we report the number of solved instances/the number of all instances tested (column "#Solved"), the remaining relative objective gap (column "%Gap"), the computational time (column "Time"), the number of explored nodes (column "Node"), the time spent in solving the subproblems (column "Sub"), and the number of feasibility cuts and optimality cuts (columns "Feas" and "Opt"). All values are average results obtained within the time limit (two hours). Also, one standard error (i.e., one standard deviation) is given in parentheses. We categorized the results into two groups: the problems with small numbers of scenarios (up to 10) and the problems with larger numbers of scenarios (#Scenario = 25, 50, and 100). We tested instances of networks "pdh" and "polska" for the larger number of scenarios cases because other networks result in an out-of-memory error with `Cplex`.

For the problems with a small number of scenarios, `Closest-BnC` solved more instances within the time limit and reduced the average computational times and relative objective gap values than `Cplex`. Also, the computational time has a smaller deviation in `Closest-BnC` than `Cplex`. For many cases, the standard deviation values are larger than the mean values, which implies that the computational times have long right tails up to the time limit. Note that `Closest-BnC` explored more branch-and-bound nodes than `Cplex` because it solves the small size master problems compared to the straightforward MIP formulations. Similar results were shown when the number of scenarios was 25 or 50. However, for the problems with 100 scenarios, `Cplex` showed better results on average. This seems to be because the size of the subproblem of `Closest-BnC` increases considerably as the number of scenarios increases. Especially, the results clearly show that the solving time of the subproblem (column "Sub") becomes dominant as the number of scenarios increases, which strongly implies that the solving subproblem is a bottleneck for those problems. We should emphasize that our Benders implementation employs no problem-specific algorithms to solve the Benders subproblem efficiently.

**Table 2 Detailed results of Closest-BnC for SNDlib with multiple scenarios**

| #Scenario | #Solved | %Gap | Time | Node | Sub | Feas | Opt |
|---|---|---|---|---|---|---|---|
| 2 | 25/25 | 0.00 (±0.00) | 141.4 (±335.1) | 837166.0 (±2113242.0) | 20.0 (±25.1) | 133.3 (±87.9) | 8.1 (±20.2) |
| 3 | 24/25 | 0.01 (±0.00) | 363.7 (±1425.8) | 1076279.0 (±4023846.9) | 37.1 (±42.9) | 135.4 (±93.5) | 8.2 (±19.5) |
| 4 | 25/25 | 0.00 (±0.00) | 254.9 (±708.9) | 1116709.2 (±3996061.6) | 66.3 (±80.4) | 126.4 (±82.3) | 10.4 (±25.1) |
| 5 | 24/25 | 0.01 (±0.00) | 440.7 (±1414.9) | 1344528.0 (±4165184.3) | 90.6 (±91.7) | 127.2 (±81.6) | 8.3 (±20.1) |
| 6 | 24/25 | 0.00 (±0.00) | 463.5 (±1411.0) | 1273910.0 (±4854868.6) | 133.0 (±124.7) | 131.4 (±86.0) | 9.1 (±21.7) |
| 7 | 25/25 | 0.00 (±0.00) | 357.7 (±629.6) | 998941.0 (±3154855.7) | 176.4 (±157.7) | 130.7 (±84.2) | 9.0 (±21.8) |
| 8 | 25/25 | 0.00 (±0.00) | 446.2 (±695.8) | 1200902.9 (±3792555.6) | 247.6 (±267.8) | 129.4 (±86.7) | 11.2 (±27.3) |
| 9 | 24/25 | 0.01 (±0.00) | 662.0 (±1399.2) | 1121818.8 (±3219888.2) | 309.6 (±303.7) | 130.2 (±84.1) | 10.0 (±24.4) |
| 10 | 24/25 | 0.01 (±0.00) | 788.2 (±1419.4) | 1025860.6 (±3907219.2) | 471.5 (±484.6) | 134.9 (±82.4) | 11.7 (±27.5) |
| 25 | 16/16 | 0.00 (±0.00) | 2180.3 (±2128.5) | 1669281.3 (±3860578.1) | 1929.2 (±2099.7) | 150.6 (±87.2) | 0.0 (±0.0) |
| 50 | 9/16 | 9.61 (±0.15) | 4479.9 (±2673.9) | 1400948.1 (±3710384.3) | 4253.2 (±2623.6) | 117.5 (±49.4) | 0.0 (±0.0) |
| 100 | 5/16 | 30.81 (±0.32) | 6551.7 (±1681.6) | 115708.2 (±252145.5) | 6467.1 (±1706.2) | 68.0 (±16.5) | 0.0 (±0.0) |