Two-Stage Stochastic Programming for Transportation Network Design Problem

Dušan Hrabec, Pavel Popela, Jan Roupec, Jan Mazal and Petr Stodola

Abstract The transportation network design problem is a well-known optimization problem with many practical applications. This paper deals with demand-based applications, where the operational as well as many other decisions are often made under uncertainty. Capturing the uncertain demand by using scenario-based approach, we formulate the two-stage stochastic mixed-integer linear problem, where the decision, which is made under uncertainty, of the first-stage program, is followed by the second-stage decision that reacts to the observed demand. Such a program may reach solvability limitations of algorithms for large scale real world data, so we refer to the so-called hybrid algorithm that combines a traditional optimization algorithm and a suitable genetic algorithm. The obtained results are presented in an explanatory form with the use of a sequence of figures.

Keywords Two-stage stochastic programming · Scenario-based approach · Transportation model · Network design problem · Genetic algorithm · Hybrid algorithm

D. Hrabec (✉) · P. Popela · J. Roupec
Faculty of Mechanical Engineering, Brno University of Technology, Technická 2, 616 69 Brno, Czech Republic
e-mail: hrabec.dusan@gmail.com

P. Popela
e-mail: pavel.popela@fme.vutbr.cz

J. Roupec
e-mail: jan.roupec@fme.vutbr.cz

J. Mazal · P. Stodola
Faculty of Military Leadership, University of Defence in Brno, Kounicova 65, 662 10 Brno, Czech Republic
e-mail: jan.mazal@unob.cz

P. Stodola
e-mail: petr.stodola@unob.cz

R. Matoušek (ed.), *Mendel 2015*, Advances in Intelligent Systems and Computing 378, DOI 10.1007/978-3-319-19824-8_2

1 Introduction

The transportation network design problem (TNDP) deals with optimization of transportation networks under specific constraints [1]. Specifically, TNDP is an important topic in transportation planning and development, where the objective is to minimize transporting costs or to maximize achieved profit. The discrete case of the network design problem (NDP) focuses on addition of new roads to a transportation network while the continuous case deals with capacity expansion of existing links. A mixture of both is referred to as the mixed NDP (see [2, 3]). See also [4, 5] for recent overviews of the NDP.

The decisions on production and transportation/supply planning must often be made without (exact) knowledge of the customer's demand. We will further emphasize the importance of uncertainty in problems of demand based management, that we interpret as a decision-making under uncertainty. Many real-world complex systems include these uncertainties, and they can be modeled in different ways. One of them is stochastic programming (SP), see [6]. The beginning of SP, and in particular stochastic linear programming, dates back to the 50's and early 60's of the last century. Although many ways have been proposed to model uncertain quantities, stochastic models have proved their flexibility and usefulness in diverse areas of science. Finally, integer programming deals with models where some of the variables are discrete, see [7] for general concepts.

The paper deals with two-stage stochastic TNDP. Two-stage programs have been named in various ways by different authors (e.g., two-stage programs, programs with recourse, two-stage programs with recourse). Such problem was firstly introduced by Dantzig [8] and Beale [9], independendtly, and has been studied extensively in the years since (see [6] or [10]). As a typical and famous example of the two-stage stochastic linear problem we have identified the so-called STORM model, which appears in paper by Mulvey and Ruszczyński [11]. It was used by the US Military to plan the allocation of aircrafts to routes during the Gulf War of 1991 (see also paper by Higle and Sen [12] for more details). In this original two-period problem, routes are scheduled to satisfy a set of demands at first-stage, demands occur, and unmet demands are delivered at higher costs in second-stage to account for shortcomings [13]. The STORM served well for tests of algorithms and evaluation of the solution quality (see, e.g., VSS fundamental concept in SP).

Various approaches have been taken to solve NDP as well as TNDP. For a detailed review of solution techniques see, e.g., [14, 15]. However, suitable algorithms for large scale problems are still under development as a challenging optimization field. This paper presents a hybrid algorithm for the solution of a two-stage scenario-based stochastic mixed integer linear program.

2 Two-Stage Stochastic Transportation Network Design Model

The important property of the problem considered in this paper is based on the fact that the decisions (at least some of them) must be made under uncertainty. In general, uncertainty may be included in the model in many ways. By computational purposes, we prefer to deal with the case of discrete random variable ξ with a finite support [16]. We model the situation by using a scenario-based (SB) approach to two-stage stochastic programs (see [16]) that helps us to capture the evolution of demand, and so, we can develop SP model for determining the operational and network design decisions.

In our two-stage problem, we will distinguish two different decisions that have to be made at two different time stages: (a) amount to be transported (b) new links to be added to the network. The first-stage decisions are made before the demand is observed, and so, we decide due to demand given by all considered scenarios. When the demand is observed, the second stage decisions are made to satisfy real customers' demand (with a higher transporting/penalty cost), see Fig. 1 for an illustrative example.

Then, we define the following two-stage SB stochastic mixed-integer linear program (SMILP):

$$
\begin{aligned}
\min \sum_{e} c_e x_e + \sum_{e_n} d_{e_n}\delta_{e_n} + \sum_{s} p_s \left(\sum_{e^*} q_{e^*} y_{e^*,s} + \sum_{i} (r^+_{i,s} w^+_{i,s} + r^-_{i,s} w^-_{i,s}) \right) \\
\text{s.t.} \quad \sum_{e} A_{i,e} x_e + \sum_{e^*} A^*_{i,e^*} y_{e^*,s} = b_{i,s} - w^+_{i,s} + w^-_{i,s}, \; \forall i \in I, \forall s \in S, \\
x_{e_n} \le \delta_{e_n} M, \qquad & \forall e_n \in E_n, \\
w^+_{i,s} \le b_{i,s}, \qquad & \forall i \in I, \forall s \in S, \\
x_e \ge 0, \qquad & \forall e \in E, \\
y_{e^*,s} \ge 0, \qquad & \forall e^* \in E^*, \forall s \in S, \\
w^+_{i,s}, w^-_{i,s} \ge 0, \qquad & \forall i \in I, \forall s \in S, \\
\delta_{e_n} \in \{0,1\}, \qquad & \forall e_n \in E_n,
\end{aligned} \tag{1}
$$

with the first-stage decision variables:

x_e : amount of a product to be transported on edge e,
δ_{e_n} : 1 if new edge e_n is built, 0 otherwise,

the second-stage decision variables:

$y_{e^*,s}$: amount of a product to be transported on edge e^* in scenario s,
$w^+_{i,s}$: shortages in a node i in scenario s,
$w^-_{i,s}$: leftovers in a node i in scenario s,

the sets of indices:

E : set of edges, $e \in E$,
E_n : set of new (built) edges, $e_n \in E_n,\ E_n \subset E$,
E^* : set of edges for the second-stage decision, $e^* \in E^*$, depending on the problem, e.g., $E^* \subset E$ or $E^* = E$,
I_1 : set of customers (or places with a non-zero demand), $i_1 \in I_1$,
I_2 : set of production places (or warehouses), $i_2 \in I_2$,
I_3 : set of traffic nodes, $i_3 \in I_3$,
I : set of all nodes in the network, $i \in I, I = I_1 \cup I_2 \cup I_3$,
S : set of all possible scenarios, $s \in S, s = 1, 2, \dots, m$,

and parameters:

$A_{i,e}$: incidence matrix, $A_{i,e} \begin{cases} 1 & \text{if edge } e \text{ from a node to node } i \text{ exists,} \\ -1 & \text{if edge } e \text{ from node } i \text{ to a node exists,} \\ 0 & \text{otherwise,} \end{cases}$
A^*_{i,e^*} : incidence matrix for the second stage,
$b_{i,s}$: the demand in a node i for a scenario s (alternatively denoted by $\xi_{i,s}$),
c_e : unit transporting cost on edge e,
d_{e_n} : cost of building of new edge e_n,
q_{e^*} : unit cost for the second-stage transporting on edge e^*,
$r^+_{i,s}$: unit penalty cost for shortages at node i in scenario s,
$r^-_{i,s}$: unit penalty cost for leftovers at node i in scenario s,
p_s : probability of observing a scenario s, $\sum_s p_s = 1$,
M : a large number, e.g. $M = \sum_{i_2}(-b_{i_2})$.

Analysing principal ideas from references introduced in the previous paragraphs, we can see that we have turned from static programs discussed in previous papers, e.g. [17, 18], to the programs having two decision stages. So, the decision $\mathbf{x}$, obtained as the solution of the first-stage program (master program), is followed by the decision $\mathbf{y}(\xi)$ that solves the second-stage program (see [16]). The network design decisions (when to make them, respectively), δ's, can be modified due to needs of the particular problem. We illustrate the decision sequence in Fig. 1.

3 Computational Example

In this section, we introduce our testing network example that was previously used in [17–20] (see Fig. 2) and solve the two-stage SMILP given by (1). We use next modification of our hybrid algorithm that we previously used for simpler cases with recourse in [17, 18]. We also shortly demonstrate the main idea of the algorithm (see Sect. 3.1). For an exhaustive description of the interface and further details see [17, 18].

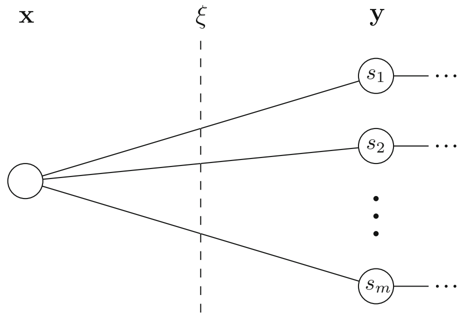


Fig. 1 Two-stage scenario tree illustrating sequence of transportation decisions

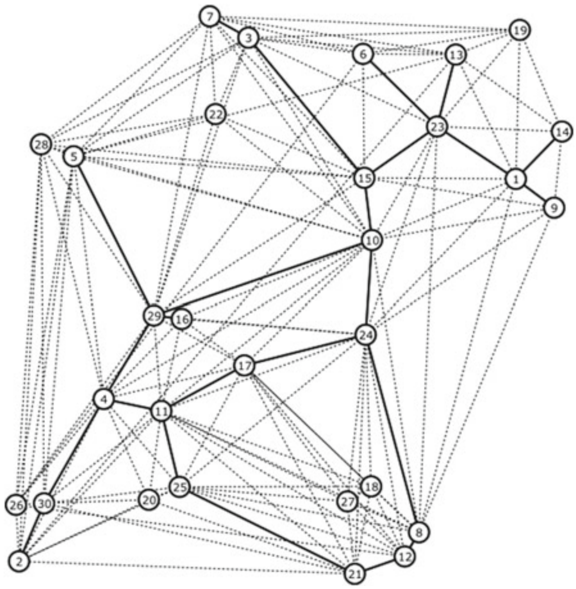


Fig. 2 Example of the network: $m = 30$, solid lines present existing network while dash lines are the potential network design variables/connections; moreover $b(i_1) > 0, I_1 = 1, 2, \dots 14$, $b(i_2) < 0, I_2 = 15, 16$, and $b(i_3) = 0, I_3 = 17, 18, \dots 30$

3.1 Hybrid Algorithm

We have implemented our model (1) in GAMS and we have solved it by the use of BARON, CONOPT, and CPLEX solvers for small test instances obtaining acceptable results (see [17, 18]). The attempt to solve larger test problems in the same way led to significant increase of computational time. Thus, we have modified and utilized our original hybrid computational technique that combines the GAMS code with genetic algorithm (GA). The algorithm is efficiently implemented in C++ with focus on GAMS-GA interface features.

Hybrid algorithm description

$\boxed{\mathbf{1}}$ Initialization of parameters for all procedures and memory allocation.

$\boxed{\mathbf{2}}$ Set up the scenario-based GAMS model (read model and data in *.gms files) for each scenario. Set up control parameters for the GA.

$\boxed{\mathbf{3}}$ Create an initial population for each GA instance, so initial values of $0-1$ variables must be generated and placed into the so called `$INCLUDE` files, where they can be read-in by GAMS. Several runs of random generation are needed, corresponding to the population size and number of scenarios.

$\boxed{\mathbf{4}}$ Repeatedly run the GAMS model by using the CPLEX solver. Each run solves the two-stage stochastic linear program for the fixed values of $0-1$ variables. Profit function values are calculated, also for new individuals created by means of the genetic operators, initially in $\boxed{\mathbf{3.}}$ and then in $\boxed{\mathbf{8.}}$

$\boxed{\mathbf{5}}$ Save the best results obtained from GAMS in $\boxed{\mathbf{4.}}$ for comparisons.

$\boxed{\mathbf{6}}$ Test the algorithm termination rules and stop in case of their satisfaction. Otherwise continue till the last scenario solution is obtained.

$\boxed{\mathbf{7}}$ Generate input values for the GA from GAMS results, see step $\boxed{\mathbf{4.}}$ Specifically, the profit function values for each member of population of the GA are obtained from results of the GAMS runs in $\boxed{\mathbf{4.}}$

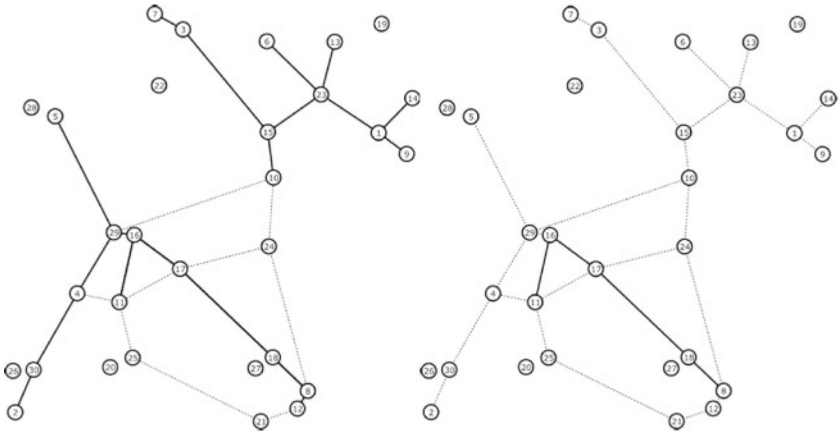
$\boxed{\mathbf{8}}$ Run GA to update the set of $0-1$ variables (population).
Switch of binary variables belonging to edges having zero flow and recalculate the objective function (by using GAMS). Return to step $\boxed{\mathbf{3.}}$

3.2 Results

In this subsection, we present results of the computations on Figs. 3 and 4. The first-stage decisions are realted to the network flow (Fig. 3a) and to the network design (Fig. 3b). When the real demands are known, we make the second-stage decisions: see the network flow (Fig. 4a) and the network design (Fig. 4b). Practically, if the second-stage decisions must be made as well as realized very quickly (e.g. in military applications, see similarities to aforementioned STORM ideas) then the network design decision from the second-stage should be realized in advance, i.e. in the first-stage. Then, the second stage decisions are restricted to only the transportation (network flow).

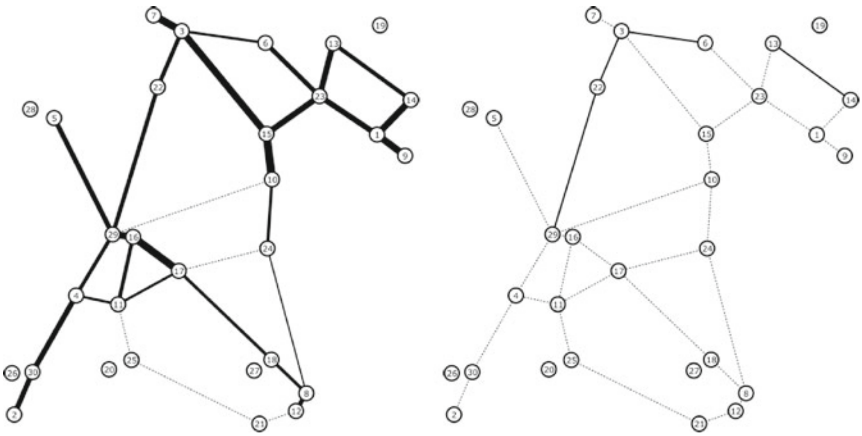
3.3 Conclusions and Further Research

The paper presents principle ideas behind the development of the original modification of hybrid algorithm involving GA and GAMS for the solution of the large-



(a) Network flow in the first-stage. (b) Network design in the first stage.

Fig. 3 First-stage decisions: network flow (x) and network design (δ).



(a) Network flow in the second stage: line thickness captures number of uses of the edge in $n = 100$ scenarios. (b) Network design in the second-stage.

Fig. 4 Second-stage decisions: network flow (y) and network design (δ).

scale stochastic TNDP. The introduced hybrid algorithm can be further applied to engineering optimization of design parameters in civil engineering applications and continuous casting, where both integer and continuous variables may appear in the large scale instances of modeled problems. The idea is to test these conclusions more carefully for large test cases and real world applications in, e.g., waste management problems especially in the related transportation network design. The approach is also portable to other problems leading to nonlinear integer programming formulations. The hybrid algorithm can be further tested comparing with some other evolutionary approaches (see, e.g., [21]).

Acknowledgments The present work has been supported by the specific research project "Modern Methods of Applied Mathematics for the Use in Technical Sciences", no. FSI-S-14-2290, id. code 25053. We would like to acknowledge the help of Petr Jindra with the visualization of achieved results.

References

1. Ghiani, G., Laporte, G., Musmanno, R.: Introduction to Logistic Systems Planning and Control, Wiley-interscience series in systems and optimization. Wiley, Chichester (2004)
2. Magnati, T.L., Wong, R.T.: Network design and transportation planning: models and algorithms. Transp. Sci. **18**(1), 1–55 (1984)
3. Yang, H., Bell, M.G.H.: Models and algorithms for road network design: a review and some new developments. Transp. Rev. **18**(3), 257–278 (1998)
4. Liu, Ch.: A Stochastic Programming Approach for Transportation Network Protection. University of California, Davis, Research Report, Institute of Transportation Studies (2009)
5. Chen, A., Zhou, Z., Chootinan, P., Ryu, S., Yang, Ch., Wong, S.C.: Transport network design problem under uncertainty: a review and new developments. Transp. Rev. **31**(6), 743–768 (2011)
6. Kall, P., Wallace, S.W.: Stochastic Programming, 2nd edn. Wiley, Chichester (1994)
7. Wolsey, L.A.: Integer programming. Wiley-interscience Series in Discrete Mathematics and Optimization (1998)
8. Dantzig, G.B.: Linear programming under uncertainty. Manag. Sci. **1**, 197–206 (1955)
9. Beale, E.M.L.: On Minimizing a convex function subject to linear inequalities. J. Roy. Stat. Soc. **17b**, 173–184 (1955)
10. Glynn, P.W., Infanger, G.: Simulation-based confidence bounds for two-stage stochastic programs. Math. Program. **138**(1–2), 15–42 (2013)
11. Mulvey, J.M., Ruszczyński, A.: A new scenario decomposition method for large-scale stochastic optimization. Oper. Res. **43**(3), 477–490 (1995)
12. Higle, J.L., Sen, S.: Stochastic Decomposition. A Statistical Method for Large Scale Stochastic Linear Programming. Kluwer Academic Publishers, Dordrecht (2004)
13. Holmes, D.: A Collection of Stochastic Programming Problems. Technical report, University of Michigan (1994)
14. Babazadeh, A., Poorzahedy, H., Nikoosokhan, S.: Application of particle swarm optimization to transportation network design problem. J. King Saud Univ. Sci. **23**, 293–300 (2011)
15. Poorzahedy, H., Rouhani, O.M.: Hybrid meta-heuristic algorithms for solving network design problem. Eur. J. Oper. Res. **182**, 578–596 (2007)
16. Popela, P., Novotný, J., Roupec, J., Hrabec, D., Olstad, A.: Two-stage stochastic programming for engineering problems. Eng. Mech. **21**(5), 335–353 (2014)
17. Roupec, J., Popela, P., Hrabec, D., Novotný, J., Olstad, A., Haugen, K.K.: Hybrid algorithm for network design problem with uncertain demands. In: Proceedings of the World Congress on Engineering and Computer Science, WCECS 2013, pp. 554–559. San Francisco, USA (2013)
18. Hrabec, D., Popela, P., Roupec, J., Jindra, P., Haugen, K.K., Novotný, J., Olstad, A.: Hybrid algorithm for wait-and-see network design problem. In: 20th International Conference on the Soft Computing MENDEL 2014, pp. 97–104. Brno, Czech Republic (2014)
19. Hrabec. D.: Stochastic programming for engineering design. Diploma thesis, Department of Mathematics, Faculty of Mechanical Engineering, BUT, 2011

20. Hrabec, D., Popela, P., Novotný, J., Haugen, K.K., Olstad, A.: The stochastic network design problem with pricing. In: 18th International Conference on the Soft Computing MENDEL 2012, pp. 416–421. Brno, Czech Republic (2012)
21. Stodola, P., Mazal, J., Podhorec, M., Litvaj, O.: Using the ant colony optimization algorithm for the capacitated vehicle routing problem. In: 16th International Conference on Mechatronics - Mechatronika (ME), pp. 503–510 (2014)
22. Matousek, R.: HC12: the principle of CUDA implementation. In Proceedings of 16th International Conference on Soft Computing MENDEL 2010, Mendel series, vol. 2010, pp. 303–308, Brno (2010). ISSN: 1803–3814

http://www.springer.com/978-3-319-19823-1

Mendel 2015
Recent Advances in Soft Computing
Matoušek, R. (Ed.)
2015, XII, 388 p. 111 illus., 26 illus. in color., Softcover
ISBN: 978-3-319-19823-1