

SNDlib 1.0–Survivable Network Design Library*

Sebastian Orlowski, *Zuse Institute Berlin (ZIB), Germany*

Michał Pióro, *Warsaw University of Technology, Poland, and Lund University, Sweden*

Artur Tomaszewski, *Warsaw University of Technology, Poland*

Roland Wessäly, *atesio GmbH, Berlin, and Zuse Institute Berlin (ZIB), Germany*

Abstract

We provide basic information on Survivable Network Design Library (SNDlib), a data library for fixed telecommunication network design that can be accessed at <http://sndlib.zib.de>. In version 1.0, the library contains data related to 22 networks which, combined with a set of selected network models, result in 830 problem instances. In this paper, we describe the data concepts of SNDlib and the data sets included in the library.

Keywords: telecommunication network design, data library, optimization

Introduction

Many different approaches have been proposed for solving telecommunication networks design problems, like mixed-integer linear programming methods, heuristics, approximate methods, or combinations of these. For drawing credible conclusions on the competitiveness of a new model or algorithm, it is indispensable to have a set of representative, challenging, and referenceable test instances on which the new approach can be tested and compared to previous ones. Unfortunately, such comparisons are rare in the network design literature, which is partially due to unavailability of realistic data sets. This is in contrast to other areas where libraries of standardized benchmark instances are available. Examples are TSPLib [1] for the Traveling Salesman Problem, MIPLIB [2] for mixed-integer linear programs, SteinLib [3] for Steiner tree problems, and FAP Web [4] for frequency assignment problems in GSM mobile phone networks.

Since December 2005, the Survivable Network Design Library (SNDlib), a library of standardized test instances for the design of survivable fixed telecommunication networks, has been available online at <http://sndlib.zib.de>. The instances, together with the best known solutions and dual bounds, can be viewed and downloaded from the SNDlib website in various formats.¹ In addition, SNDlib contains a collection of references (BiBTeX entries) to relevant publications in network design, a list of conferences related to this area, and a mailing list. Currently, the library is deliberately limited to designing survivable single-layer networks with a given set of installable link capacities. In future versions, various extensions related to, e.g., node equipment configuration and multi-layer scenarios, are envisaged.

The purpose of this paper is to present the data concepts and the problem instances of the library together with their background and statistical information. A precise mathematical description of the SNDlib problems will be given in a future full paper.

SNDlib data concepts

Similar to several modelling languages for mixed-integer programming like Zimpl [5] or AMPL [6], models and data are separated from each other in SNDlib. An SNDlib network planning problem consists of two parts:

- a *network* (called *SNDlib network instance* or shortly *SNDlib network* in the sequel), which describes nodes, links, demands, capacities, cost, and some other planning data, and
- a *network planning model* (called *SNDlib model instance* or shortly *SNDlib model* in the sequel), which specifies the planning parameters, e.g., whether links are directed or undirected, whether the routing of

*Supported by the German ministry of science (BMBF) project Eibone.

¹XML, GML, and a native ASCII format

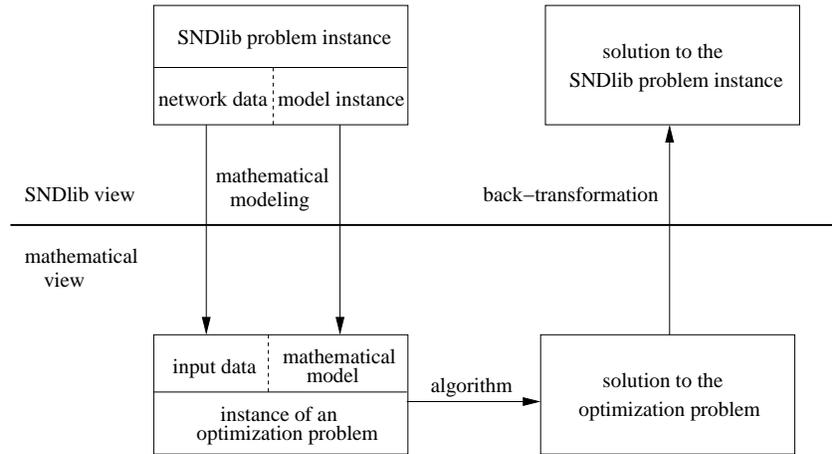


Figure 1: The planning process.

a demand may be split on several paths or not, or which capacity model or survivability concept should be used.

A particular SNDlib *network* can usually be combined with various SNDlib *models* to form different SNDlib network planning *problems (problem instances)*, for example, one without survivability and another one with 1+1 protection, or one with a multi-commodity flow routing and another one with a single-path routing. A *solution* to a particular SNDlib problem consists of a routing of the demands specified in the SNDlib network and a suitable capacity installation on the links that satisfy the side constraints specified in the SNDlib model.

The SNDlib problems and their solutions are independent of any mathematical model or algorithm. To solve such a problem, a suitable *mathematical model* has to be chosen to reflect its SNDlib model. Often, the same SNDlib model can be represented by several mathematical models, like an edge-flow formulation or an equivalent path-flow formulation. Such a mathematical model combined with input data derived from an SNDlib network forms a particular problem instance in the mathematical sense, as illustrated in the left part of Figure 1. After solving the resulting mathematical optimization problem, its solution has to be transformed back into an SNDlib solution, as shown in the right part of Figure 1. The data provided by the library belongs to the upper part of the figure, whereas users of the library are invited to fill in the lower part by investigating different mathematical models and solution methods.

The following section characterizes the network data and the possible instantiations of the SNDlib model informally. A precise mathematical formulation will be given in the full paper. Examples and a detailed definition of the I/O formats can be found on the SNDlib website <http://sndlib.zib.de>.

SNDlib network An *SNDlib network* describes the network structure, the traffic to be routed, and a set of admissible routing paths. The set of links defines potential connections between the nodes that may be used to carry traffic; parallel links are allowed. For every link, capacity and cost information is specified by the amount of pre-installed capacity and its cost, and a list of capacity modules with associated cost, which can be installed on that link. Additionally, fixed-charge cost for using the link is defined, as well as flow cost incurred by routing traffic through that link.

The set of communication demands describes the traffic to be routed. Each demand is characterized by its end-nodes and the volume of traffic that has to be routed through the network and is given in multiples of some base routing unit. The routing unit of a demand defines the amount of link capacity consumed by one unit of the demand's traffic (corresponding, for instance, to 2 Mbit/s, 155 Mbit/s, or 2.5 Gbit/s).

Eventually, a list of admissible routing paths has to be specified. Either the list is empty, meaning that every possible path is admissible, or a non-empty set of paths must be specified for each demand. A hop limit

Model parameter	Admissible values
Demand model	DIRECTED (D), UNDIRECTED (U)
Link model	DIRECTED (D), UNDIRECTED (U), BIDIRECTED (B)
Link capacity model	LINEAR_LINK_CAPACITIES (L), MODULAR_LINK_CAPACITIES (M), EXPLICIT_LINK_CAPACITIES (E)
Fixed-charge model	YES (Y), NO (N)
Routing model	SINGLE_PATH (S), CONTINUOUS (C), INTEGER (I)
Admissible path model	EXPLICIT_LIST (E), ALL_PATHS (A)
Hop limit model	INDIVIDUAL_HOP_LIMITS (Y), IGNORE_HOP_LIMITS (N)
Survivability model	NO_SURVIVABILITY (N), ONE_PLUS_ONE_PROTECTION (P), SHARED_PATH_PROTECTION (S), UNRESTRICTED_FLOW_RECONFIGURATION (U)

Table 1: SNDlib model parameters and their admissible values

may be imposed for each demand, i.e., a maximum number of links for each admissible routing path. If the SNDlib model specifies to use this hop limit, it further restricts the set of admissible paths.

SNDlib model An *SNDlib model* specifies a set of selected design parameters. These are defined by the set of attributes and their permissible values given in Table 1; exactly one value has to be selected for each attribute. Currently, the only considered design objective is to minimize total link cost. This includes fixed-charge cost, cost of pre-installed capacity, cost of selected capacity modules, and routing cost. A complete model instance in the native ASCII format of SNDlib might look as follows:

```

DEMAND_MODEL           = UNDIRECTED           # U
LINK_MODEL              = UNDIRECTED           # U
LINK_CAPACITY_MODEL    = MODULAR_LINK_CAPACITIES # M
FIXED_CHARGE_MODEL     = NO                    # N
ROUTING_MODEL          = CONTINUOUS            # C
ADMISSIBLE_PATH_MODEL = ALL_PATHS             # A
HOP_LIMIT_MODEL        = IGNORE_HOP_LIMITS    # N
SURVIVABILITY_MODEL    = NO_SURVIVABILITY     # N

```

Naming conventions We have introduced a naming convention for the SNDlib problems that reflects both the name of the network and the characteristics of the model. Such a name consists of an abbreviation of the SNDlib network name and a sequence of letters specifying the selected attribute values of the model. These letters are given in Table 1, and in the name they are ordered as in the example above, i.e., first the demand model, then the link model, and so on. That is, the problem instance built from the network EXAMPLE and the above model would be named EXAMPLE-U-U-M-N-C-A-N-N.

SNDlib solution Basically, a solution to an SNDlib problem instance consists of a link configuration and a routing for the demands such that all side constraints are satisfied. A detailed description of the available solution formats, as well as an example with a detailed cost analysis can be found at the SNDlib website.

Network design instances

SNDlib currently contains 22 networks. Each of these networks has been combined with several SNDlib models (basically, all reasonable SNDlib models for each network) to construct 830 problem instances.

network	$ \mathcal{V} $	$ \mathcal{E} $	$ \mathcal{D} $	$ I $	DL	CAP	surv	pre	fix	rcost	exp	hop
ATLANTA	15	22	210	20	UU/DB	M	×	×	-	×	-	-
COST266	37	57	1332	18	UU/DB	ELM	-	-	-	×	-	-
DFN-BWIN	10	45	90	20	UU/DB	E	×	-	-	-	-	-
DFN-GWIN	11	47	110	24	UU/DB	EM	-	×	-	-	-	×
DI-YUAN	11	42	22	20	UU/DB	E	×	-	-	-	-	-
FRANCE	25	45	300	80	UU/DB	LM	×	-	×	-	-	-
GERMANY50	50	88	662	40	UU/DB	LM	×	-	-	-	-	-
GIUL39	39	172	1471	10	DD	E	×	-	-	-	-	-
JANOS-US	26	84	650	40	DD	LM	×	-	-	-	-	-
JANOS-US-CA	39	122	1482	20	DD	LM	×	-	-	-	-	-
NEWYORK	16	49	240	60	UU/DB	ELM	×	-	-	×	-	-
NOBEL-EU	28	41	378	20	UU/DB	E	×	-	-	-	-	-
NOBEL-GERMANY	17	26	121	20	UU/DB	E	×	-	-	-	-	-
NOBEL-US	14	21	91	6	UU/DB	E	-	-	-	-	-	-
NORWAY	27	51	702	60	UU/DB	ELM	×	-	-	-	-	-
PDH	11	34	24	60	UU/DB	ELM	×	-	-	-	-	-
PIORO40	40	89	780	80	UU/DB	M	×	-	×	-	×	-
POLSKA	12	18	66	80	UU/DB	M	×	-	×	-	×	-
SUN	27	102	67	10	DD	M	×	-	-	-	-	-
TA1	24	55	396	120	UU/DB	ELM	×	-	-	-	-	×
TA2	65	108	1869	36	UU/DB	ELM	-	×	-	-	-	×
ZIB54	54	81	1501	6	UU/DB	E	-	-	-	-	-	-

Table 2: Overview on the SNDlib problem instances

Table 2 provides an overview of the networks together with statistical information and with an indication on which SNDlib models have been used for each network. The name of the network is followed by the number of its nodes ($|\mathcal{V}|$), links ($|\mathcal{E}|$), and demands ($|\mathcal{D}|$). The next column ($|I|$) provides the total number of problem instances constructed from each network. More information about the used SNDlib models is given in the next two columns. The column DL specifies the combinations of demand model and link model applied to this network: UU (both UNDIRECTED), DD (both DIRECTED), DB (DIRECTED demands and BIDIRECTED links), and combinations of these. This is followed by the used link capacity models (E=EXPLICIT_LINK_CAPACITIES, L=LINEAR_LINK_CAPACITIES, M=MODULAR_LINK_CAPACITIES).

The columns in the last block indicate whether the network supports survivability (*surv*), whether links with positive preinstalled capacity (*pre*), fixed-charge cost (*fix*), or routing cost (*rcost*) exist, whether explicit lists of paths are provided (*exp*), and whether demands with hop limits exist (*hop*). The table shows that the problem instances cover the whole range from small to relatively large problem instances, with different densities, cost structures, and routing constraints.

The network instances included in SNDlib have different backgrounds. Some of them stem from industrial projects, others have been defined as reference networks in international research projects involving a variety of industrial and academic partners. A separate group are network instances whose background is not publicly available, for instance, due to non-disclosure agreements. In what follows, we will summarize the background and specific features of the network instances as close as possible.

Instances with industrial background The following network instances have been provided by network operators or by manufacturers of network equipment, either directly or via academic partners. In some cases, we know that the data has been modified by the industrial or academic partners in order to avoid revealing internal information. From our experience with industry projects, we suspect this to be the case for other network instances as well. Nevertheless, we assume that the cost and demand structures provided in these data sets are not too far away from reality.

DFN-BWIN, DFN-GWIN	Derived from IP/OSPF planning problems on the German research network, which is operated by DFN Verein e.V. and connects most universities and research institutes. The demands are based on traffic measurements in the network.
FRANCE	WDM planning data provided by a network operator.
GIUL39	Derived from a VPN planning problem in a two-layer metropolitan network with SDH as the underlying transport technology. Provided by an equipment vendor. The installable capacities have been constructed for SNDlib.
PDH	PDH planning data provided by a network operator.
POLSKA	SDH transport network of the Polish Telecom from the early 1990's.
TA1, TA2	SDH planning problems provided by the network operator Telekom Austria.
ZIB54	Slightly modified real-world instance from a telecom network operator.

Instances from international research projects The network instances belonging to this group have been defined as reference networks in large research projects funded by the European Union. Those projects involved major telecom network operators and equipment manufacturers, as well as academic partners. Hence, even if the reference networks defined in the projects do not represent real networks, they were carefully constructed to correspond to realistic planning scenarios.

COST266	Originating from the project <i>COST266–Advanced Infrastructure for Photonic Networks</i> [7] of the European Cooperation in the Field of Scientific and Technical Research (COST).
NOBEL-US, NOBEL-EU, NOBEL-GERMANY, GERMANY50	The NOBEL networks are reference networks originating from the European project NOBEL [8], where a detailed cost model was developed for various kinds of SDH and WDM equipment, like fibers, transponders, repeaters, multiplexers, and line cards, see for instance [9]. The network GERMANY50, provided by T-Systems International AG, is an extension of the NOBEL-GERMANY network.

Other instances The following network instances either do not fit into one of the above categories, or we do not know their original source. Several of these network instances have been used in the network design literature before being sent to us; we give the reference to their first appearance in the literature if available. For a number of old data sets, solving a linear problem was nearly sufficient to solve the integer problem because the capacity granularity was small in comparison to the demands. In such cases, we have scaled the demands or capacities to make the problem instances more challenging.

ATLANTA, NEWYORK, NORWAY	The ATLANTA network represents an ATM network in Los Angeles, the NEWYORK network a telecommunication network in the greater New York area, and the NORWAY network a backbone network in Norway. These networks have been used in [10], among others. Demands and installable capacities have been scaled for SNDlib.
DI-YUAN	An optical city network with unknown background.
JANOS-US, JANOS-US-CA	The JANOS-US network was first presented in [11]. The JANOS-US-CA instance is an extension of that network published in [12].
PIORO40	An unpublished randomly generated network with an SDH cost structure.
SUN	The network, first presented in [10], have been constructed from NEWYORK by bidirecting all links and changing the demand pattern. We have scaled all demands by 10 to make them integer.

Final remarks

The purpose of SNDlib is to provide realistic network planning data to the network design community, which can be used to compare different mathematical models and optimization algorithms. At the same time, we wanted to create a platform for sharing not only network planning data but also other information. To this end,

SNDlib provides a collection of references on fixed network design, a list of upcoming conferences in the field, a list of relevant journals, and a mailing list² related to fixed network design. Everybody in the network design community is cordially invited to use the SNDlib networks in publications, to submit solutions or dual bounds of SNDlib problems, as well as references to new papers, or conference announcements, either via the web interface at <http://sndlib.zib.de> or at the e-mail address sndlib-webmaster@zib.de.

We hope that SNDlib will help to develop network planning and design methodologies, and in consequence contribute to improving the quality of future communication networks.

Acknowledgement

Special thanks to the contributors of SNDlib who have helped build up the library by providing network instances, solutions or bibliography information, or by doing a lot of programming work. In particular, this includes Edoardo Amaldi, Pieter Audenaert, Dan Bienstock, Andreas Bley, Giuliana Carello, Tibor Cinkler, Mateusz Dżida, Claus Gruber, Oktay Günlük, Ralf Hülsermann, Peter Jonas, Roman Klähne, Mariusz Mycek, Mats Petter Petterson, Christian Raack, Janos Tapolcai, Di Yuan, Michał Zagożdżon, and Adrian Zymolka. We also thank Marc Pfetsch for useful comments on a draft of this paper.

References

- [1] G. Reinelt. TSPLIB—A Traveling Salesman Problem library. *ORSA Journal on Computing*, 3:376–384, 1991. <http://www.iwr.uni-heidelberg.de/groups/comopt/software/TSPLIB95>.
- [2] Tobias Achterberg, Thorsten Koch, and Alexander Martin. MIPLIB 2003. *Operations Research Letters*, 34(4):1–12, 2006. <http://miplib.zib.de>.
- [3] T. Koch, A. Martin, and S. Voß. SteinLib: An updated library on Steiner tree problems in graphs. Technical Report ZR-00-37, Konrad-Zuse-Zentrum für Informationstechnik Berlin, 2000. <http://elib.zib.de/steinlib>.
- [4] FAP web—A website devoted to frequency assignment. <http://fap.zib.de>, 2000–2007.
- [5] T. Koch. Zimpl user guide. ZIB Report ZR-01-20, Konrad-Zuse-Zentrum für Informationstechnik Berlin, 2001. <http://zimpl.zib.de>.
- [6] R. Fourer, D. M. Gay, and B. W. Kernighan. A modeling language for mathematical programming. *Management Science*, (36):519–554, 1990. <http://www.ampl.com>.
- [7] S. De Maesschalck, D. Colle, I. Lievens, M. Pickavet, P. Demeester, C. Mauz, M. Jäger, R. Inkret, B. Mikac, and J. Derkacz. Pan-European optical transport networks: An availability-based comparison. *Photonic Network Communications*, 5(3):203–225, 2003.
- [8] NOBEL—Next generation Optical networks for Broadband European Leadership. <http://www.ist-nobel.org>, 2004–2007.
- [9] M. Gunkel, R. Leppla, M. Wade, A. Lord, D. Schupke, G. Lehmann, C. Fürst, S. Bodamer, B. Bollenz, H. Haunstein, H. Nakajima, and J. Martensson. A cost model for the WDM layer. In *International Conference on Photonics in Switching (PS 2006)*, Herakleion, Crete, Greece, October 2006.
- [10] D. Bienstock, O. Günlük, S. Chopra, and C.Y. Tsai. Minimum-cost capacity installation for multicommodity flows. *Mathematical Programming*, 81:177–199, 1998.
- [11] M. De, V. Mariappan, V. Chandramouli, and S. Kuppusamy. US national network design. Presentation held at CREWMaN, University of Texas, Arlington, 2002.
- [12] J. Tapolcai. *Routing Algorithms In Survivable Telecommunication Networks*. PhD thesis, Budapest University of Technology and Economics, March 2005.

²The list is moderated to avoid spam mail.