Staynerd (['ʃtʌɪnə]) Code User Manual

September 20, 2016

This user manual describes a solver implementation originally being developed as a contribution for the 11th DIMACS Implementation Challenge, dedicated to the study of Steiner tree problems. The code provides algorithms for exact and heuristic solution of the following problem types: Steiner tree problem (STP), prize-collecting Steiner tree problem (PCSTP), rooted prize-collecting Steiner tree problem (RPCSTP), maximum-weight connected subgraph problem (MWCS) and degree-constrained Steiner tree problem (DCST).

For implementation details the reader is referred to the corresponding paper:

Thinning out Steiner trees: A node-based model for uniform edge costs. M. Fischetti, M. Leitner, I. Ljubić, M. Luipersbeck, M. Monaci, M. Resch, D. Salvagnin and M. Sinnl.

1 Installation

The provided executable is a 64-bit Linux binary (compiled with GCC 4.9.2, requiring GLIBC version $\geq$ 2.3). For running the solver, the user is required to provide dynamic CPLEX libraries.

By default, a CPLEX installation will only include static libaries (i.e., the files `libconcert.a`, `libcplex.a` and `libilocplex.a`), so the corresponding dynamic libary files (`libconcert.so`, `libcplex.so` and `libilocplex.so`) must be generated manually from the set of static library files. For this purpose the following software is required:

- GCC version 4.9.2 or greater

- IBM ILOG CPLEX Optimization Studio (tested with version 12.6, other versions might work but are not officially supported)

Note that running the program requires a valid license file, which can be obtained from the authors on request.

The following procedure lists all steps necessary to build the dynamic libraries and verify that the solver works correctly:

1. Set the environment variable `CPLEX_DIR` to the base directory of the CPLEX installation on your system (e.g., `/opt/ibm/ILOG/CPLEX_Studio126`).

2. Run the script `make_cplex_dynamic.sh` provided with the binary to create the dynamic CPLEX library files `libconcert.so`, `libcplex.so` and `libilocplex.so`.

3. Make sure that the dynamic libraries are placed in the same directory as the solver binary `staynerd` and the license file `staynerd.license`.

4. A simple way to run the solver is as follows (see Section 3 for more options):

   ```
   staynerd [inputfile] [timelimit] [threads] [outputfile]
   ```

2 File formats

2.1 Input format

Problem instances files must adhere to the SteinLib instance format. For a detailed description and examples, the reader is referred to the SteinLib website (`http://steinlib.zib.de/format.php`). Test instances can also be downloaded at `http://dimacs11.zib.de/downloads.html`.

Input instances should be of one of the following problem types: STP, PCSTP, RPCSTP, MWCS and DCST. By default, the input file's problem type is inferred from the line starting with "`Problem`" in "`Comments`" section. The solver searches specifically for the following strings (quotes included):

Problem Type	Line
STP	`Problem "Classical Steiner tree problem in graph"`
PCSTP	`Problem "Prize-Collecting Steiner Problem in Graphs"`
RPCSTP	`Problem "Rooted Prize-Collecting Steiner Problem in Graphs"`
MWCS	`Problem "Maximum Node Weight Connected Subgraph"`
DCST	`Problem "Degree-Constrained Steiner Trees in Graphs"`

2.2 Output format

The output file format adheres to the specification provided by the organizers of the 11th DIMACS implementation challenge. It contains the best solution found within the specified time limit as well as addional information about the run (e.g., bounds, total time, times at which improving solutions have been found, etc.). A detailed description is available at `http://dimacs11.zib.de/docs/CompetitionRules-141119.pdf`.

3 Parameters

Aside from specifying parameters by position as described in Section 1, a larger set of parameters is acessible through the following program options:

Name	Default value	Range	Description
`-h`			print help message
`-f`			instance file to solve
`-i`			read starting solution from file
`-o`			solution file to write
`--seed`	0	{-1,0,...}	random seed, set to -1 to use system time
`--cplex.memory`	12288 (12 GB)		limit for CPLEX working memory
`-T`	# of execution units		number of threads
`-t`	86400 (24h)		total time limit (in seconds)
`--gap`	0		relative optimality gap (in percent)
`--heuristic`	1	{0,1}	enable/disable primal heuristics during branch-and-bound
`--ymodel`	0	{0,1}	use node variable model (only for instances with uniform edge weights)
`--filter`	1	{0,1,2}	choose parameters automatically based on instance properties: 0...no filter (no automatic selection of node model or heuristics) 1...setting for exact solution 2...setting for heuristic solution
`--localbranching`	0	≤ total time limit	use local branching heuristic for the specified amount of time
`--setcover`	0	≤ total time limit	use set-covering heuristic for the specified amount of time

Note that CPLEX runs may be aborted manually through the user by entering 'Q'.

4 Licenses

4.1 Staynerd Solver

```
Copyright (c) 2015, M. Fischetti, M. Leitner, I. Ljubic, M. Luipersbeck,
M. Monaci, M. Resch, D. Salvagnin and M. Sinnl.
All rights reserved.

Redistribution and use in source and binary forms, with or without
modification, are permitted provided that the following conditions are met:
1. Redistributions of source code must retain the above copyright
   notice, this list of conditions and the following disclaimer.
2. Redistributions in binary form must reproduce the above copyright
   notice, this list of conditions and the following disclaimer in the
   documentation and/or other materials provided with the distribution.
3. All advertising materials mentioning features or use of this software
   must reference the following publication:
   Thinning out Steiner trees: A node-based model for uniform edge costs.
   M. Fischetti, M. Leitner, I. Ljubic, M. Luipersbeck, M. Monaci,
   M. Resch, D. Salvagnin and M. Sinnl.
4. Neither the name of the Staynerd Team nor the
   names of its contributors may be used to endorse or promote products
   derived from this software without specific prior written permission.

THIS SOFTWARE IS PROVIDED BY STAYNERD TEAM ''AS IS'' AND ANY
EXPRESS OR IMPLIED WARRANTIES, INCLUDING, BUT NOT LIMITED TO, THE IMPLIED
WARRANTIES OF MERCHANTABILITY AND FITNESS FOR A PARTICULAR PURPOSE ARE
DISCLAIMED. IN NO EVENT SHALL THE AUTHORS BE LIABLE FOR ANY
DIRECT, INDIRECT, INCIDENTAL, SPECIAL, EXEMPLARY, OR CONSEQUENTIAL DAMAGES
(INCLUDING, BUT NOT LIMITED TO, PROCUREMENT OF SUBSTITUTE GOODS OR SERVICES;
LOSS OF USE, DATA, OR PROFITS; OR BUSINESS INTERRUPTION) HOWEVER CAUSED AND
ON ANY THEORY OF LIABILITY, WHETHER IN CONTRACT, STRICT LIABILITY, OR TORT
(INCLUDING NEGLIGENCE OR OTHERWISE) ARISING IN ANY WAY OUT OF THE USE OF THIS
SOFTWARE, EVEN IF ADVISED OF THE POSSIBILITY OF SUCH DAMAGE.
```

4.2 dtree

```
Copyright (c) 2012 David Eisenstat <eisenstatdavid@gmail.com>

Permission is hereby granted, free of charge, to any person obtaining a
copy of this software and associated documentation files (the "Software"),
to deal in the Software without restriction, including without limitation
the rights to use, copy, modify, merge, publish, distribute, sublicense,
and/or sell copies of the Software, and to permit persons to whom the
Software is furnished to do so, subject to the following conditions:

The above copyright notice and this permission notice shall be included
in all copies or substantial portions of the Software.

THE SOFTWARE IS PROVIDED "AS IS", WITHOUT WARRANTY OF ANY KIND, EXPRESS OR
IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF MERCHANTABILITY,
FITNESS FOR A PARTICULAR PURPOSE AND NONINFRINGEMENT. IN NO EVENT SHALL
THE AUTHORS OR COPYRIGHT HOLDERS BE LIABLE FOR ANY CLAIM, DAMAGES OR
OTHER LIABILITY, WHETHER IN AN ACTION OF CONTRACT, TORT OR OTHERWISE,
ARISING FROM, OUT OF OR IN CONNECTION WITH THE SOFTWARE OR THE USE OR
OTHER DEALINGS IN THE SOFTWARE.
```

4.3 Open Graph Drawing Framework

```
This program is free software; you can redistribute it and/or
modify it under the terms of the GNU General Public License
Version 2 or 3 as published by the Free Software Foundation
and appearing in the files LICENSE_GPL_v2.txt and
LICENSE_GPL_v3.txt included in the packaging of this file.

In addition, as a special exception, you have permission to link
this software with

  - Tulip (http://www.tulip-software.org)

  - the libraries of the COIN-OR Osi project (see
    http://www.coin-or.org/projects/Osi.xml);

  - all libraries required by Osi;

  - and all LP-solver libraries directly supported by the
    COIN-OR Osi project,

and distribute executables, as long as you follow the requirements
of the GNU General Public License in regard to all of the software
in the executable aside from these third-party libraries.

See also: http://www.ogdf.net/license.html

Contact: license@ogdf.net
```

4.4 Boost

```
Boost Software License - Version 1.0 - August 17th, 2003

Permission is hereby granted, free of charge, to any person or organization
obtaining a copy of the software and accompanying documentation covered by
this license (the "Software") to use, reproduce, display, distribute,
execute, and transmit the Software, and to prepare derivative works of the
Software, and to permit third-parties to whom the Software is furnished to
do so, all subject to the following:

The copyright notices in the Software and this entire statement, including
the above license grant, this restriction and the following disclaimer,
must be included in all copies of the Software, in whole or in part, and
all derivative works of the Software, unless such copies or derivative
works are solely in the form of machine-executable object code generated by
a source language processor.

THE SOFTWARE IS PROVIDED "AS IS", WITHOUT WARRANTY OF ANY KIND, EXPRESS OR
IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF MERCHANTABILITY,
FITNESS FOR A PARTICULAR PURPOSE, TITLE AND NON-INFRINGEMENT. IN NO EVENT
SHALL THE COPYRIGHT HOLDERS OR ANYONE DISTRIBUTING THE SOFTWARE BE LIABLE
FOR ANY DAMAGES OR OTHER LIABILITY, WHETHER IN CONTRACT, TORT OR OTHERWISE,
ARISING FROM, OUT OF OR IN CONNECTION WITH THE SOFTWARE OR THE USE OR OTHER
DEALINGS IN THE SOFTWARE.
```