Reliable Routing Contest 2017

TRB Network Modeling Committee (ADB30)

1 Introduction

Routing on a network is a fundamental problem in transportation systems, in logistics, in telecommunication systems, and many other critical domains. In many of these areas, the routing objective includes not only average performance, but also a measure of reliability or risk aversion. Research in route choice in transportation reveals that reliability is just as important as typical travel time (Pinjari and Bhat, 2006) – and furthermore roughly half of congestion can be traced to nonrecurring causes (Short, 2012). In particular, a natural objective is to minimize a linear combination of the mean and standard deviation of the chosen path's travel time. Unlike the canonical shortest path problem, this objective is not separable by link, and Bellman's principle does not apply.

Recent years have seen substantial improvements both algorithmically and in the availability of computational hardware. However, since Bellman's principle does not apply to the the mean-standard deviation with independent arc cost shortest path problem, one cannot rely on computational advances alone to render this problem tractable. Furthermore, comparing different algorithms proposed in the literature is notoriously difficult, due to differences in implementation, computer platform, test instances, and so forth. The purpose of this contest is to provide a platform to both (1) allow comparison of existing algorithms for this problem in a common environment and (2) spur further algorithmic innovation and research in this critical area.

Entrants in this contest must submit source code for a program which can read a network of arbitrary size and topology, and return a path between an arbitrary origin and destination. These programs will be ranked based on the extent to which the output paths minimize the sum of mean travel time, and standard deviation travel time under independent arc costs. A first prize of $1000 and a second prize of $500 will be awarded. The rules below fully specify the problem statement, eligibility rules, and judging criteria.

This contest is jointly sponsored by the Network Models in Practice Subcommittee and the Freight, Transit and Logistics Subcommittees of the Transportation Research Board, subcommittees of the Network Modeling Committee (ADB30). The organizers and judges of this competition are Stephen Boyles (The University of Texas at Austin), Avinash Unnikrishnan (Portland State University), and Xuesong Zhou (Arizona State University).

2 Problem statement

Consider a directed network $G = (N, A)$ with set of nodes N and set of links A. Two distinct nodes r and s are specified as the origin and destination, respectively. There are no parallel links, so the notation (i, j) is unambiguous. For each link (i, j), the expected link cost μ_{ij} and the standard deviation of the link cost σ_{ij} are given; both are nonnegative floating point numbers. Furthermore, for any two links (i, j) and (k, l), $\rho_{ij,kl}$ gives the correlation coefficient between the costs on link (i, j) and (k, l), a floating point number between -1 and $+1$. Given any simple path π, let δ_{ij}^{π} equal one if link (i, j) is part of path π, and zero otherwise. The cost of any path π is given by

$$C^{\pi} = \sum_{(i,j)\in A} \mu_{ij}\delta_{ij}^{\pi} + \sqrt{\sum_{(i,j)\in A}\sum_{(k,l)\in A} \rho_{ij,kl}\sigma_{ij}\sigma_{kl}\delta_{ij}^{\pi}\delta_{kl}^{\pi}}$$

The objective is to find the simple path π connecting nodes r and s with the minimal value of C^{π}, given the network topology, origin, destination, and link cost information. In some instances of the problem, link costs will be independent, so $\rho_{ij,kl}$ will be zero whenever $(i, j) \neq (k, l)$.

3 Contest rules and judging criteria

1. All entries will be submitted by email to `sboyles@mail.utexas.edu`, `avinashu@pdx.edu`, and `xzhou74@asu.edu`, no later than December 31, 2017. Entries received after this time will not be considered.

2. A first prize of \$1000 and a second prize of \$500 will be awarded based on the scoring system described below.

3. Entries may be submitted by a single participant, or by a team of multiple participants. A team may split the prize money amongst its members in any way.

4. No single person may participate on more than 3 submissions, in any combination of single entries or teams.

5. Current graduate students and postdoctoral advisees of the judging team are ineligible to submit entries or participate on teams.

6. Each entry will consist of one or more source code files, written in a programming language in common use, and for which a free, open source compiler is available.

7. Any additional libraries needed to run the code must be clearly specified in a readme file submitted with the source, and must also be available freely and in open source. This readme file should also contain any other instructions needed for compiling the code (e.g., makefiles, compiler flags).

Participants are discouraged from submitting entries with complicated compilation processes or library dependencies — if the judging team is unable to compile an entry, it will be disqualified.

8. This code should be written in a platform-independent way. Do not assume any specific hardware features (e.g., availability of a GPU) or operating system features.

9. All entries will be compiled and executed on the same desktop machine, with the same operating system.

10. The judging team will generate 210 problem instances (7 network topologies, each with 30 different means, standard deviations, correlation matrices, origins, and destinations). The network topologies will be consistent with large scale transportation networks (at least 25,000 nodes), and the average node connectivity will not exceed 4. Of the 30 instances corresponding to each network topology, ten will involve independent link costs (diagonal correlation matrix), ten will involve sparse correlation matrices (relatively few nonzero entries), and ten will involve general correlation matrices. These networks will not be made known prior to the contest submission deadline.

11. These problem instances will be formulated as text files. An example of input file is provided at the end of this document. Each program entry must read these text files, which will be placed in the same directory as the source code, and named `input.txt`. Entries which cannot read all of these instances successfully will be disqualified.

12. All program output will be placed in a text file named `output.txt`, in the same directory as the source code. The sample output format is given at the end of this document. This file consists of one or more paths, specified by the nodes in the path. Only the last path reported in this file will be used for judging purposes.

13. The score of each instance is the sum of the mean and standard deviation of the last path reported in the output file. Each program will run for 60 seconds (wall clock time) before being terminated by killing the process (do not assume output buffers will be written when the process is terminated). The value of the objective function will be calculated for the last path written to `output.txt`, and used to rank each submission. In the event of a tie, the allowed computation time will be halved, and the process repeated among tied submissions, until a winner is determined. If none of the entries has reported a path within 60 seconds, the allowed computation time will be increased by tripled, and this process repeated until at least one entry has produced a path.

14. The entry which wins the greatest number of of the 210 instances will be declared the winner. In the event of a tie in the top number of problem

instances won by an entry, the total first and second prize ($1500) will be divided equally among all top algorithms.

15. Each of the 210 problem instances must be solved completely from scratch — any submission which stores information from one instance to be re-used in another instance will be disqualified.

4 Sample file formats

4.1 `input.txt`

The first line of the file contains the number of nodes n, the number of links m, the origin node r, and the destination node s. The next m rows contain the tail and head nodes of each link, followed by μ_{ij} and σ_{ij} The next m rows contain the correlation matrix, containing the values $\rho_{ij,kl}$.

An example of this file:

```
3,5,1,4
1,2,1,5
1,4,4.3,3
2,3,2,1
2,4,1,1
3,4,1,7.15
1,0,0,0,0.5
0,1,0,0,0
0,0,1,0,0
0,0,0,1,0
0.5,0,0,0,1
```

4.2 `output.txt`

Each line in this file represents a path from r to s, specified by the nodes in the path. Only the last line in this file will be used for judging purposes. An example of this file:

```
1,2,3,4
1,4
1,2,4
```

This file will be scored using the two-link path consisting of (1,2) followed by (2,4). Using the sample input file above, its cost is given by

$$\mu_{12} + \mu_{24} + \sqrt{\sigma_{12}^2 + \sigma_{24}^2 + 2\rho_{12,24}\sigma_{12}\sigma_{24}} = 1 + 1 + \sqrt{25 + 1 + 5} = 7.57\,.$$

5 References

- Pinjari, A. R., and C. R. Bhat (2006). On the Nonlinearity of Response to Level of Service Variables in Travel Mode Choice Models. *Transportation Research Record*, Vol. 1977, pp. 67-74.

- Short, J., (2012). Measuring recurring and non recurring congestion,, NATMEC Conference, Available at: `http://onlinepubs.trb.org/onlinepubs/conferences/2012/NATMEC/Short.pdf`