README

This data set can be divided in two components: (i) the data sets used to train the delay/jitter RoutNet models (‘nsfnet.zip’, ‘geant2.zip’, ‘gbn.zip’), and (ii) the delay/jitter RouteNet models already trained that we used for the evaluation in the paper (‘trained_model_delay.zip’, ‘trained_model_jitter.zip’).

Description of the training data sets:

- The root directory contains a *.ned file that describes the network topology. We consider that all the links inthe topology have a capacity of 10 kbps.
- The /routing* directory contains the different routing schemes simulated. Each routing file includes a matrix that describes the forwarding of each node: matrix(src node , dst node) = output port of the src node to forward the traffic towards the dst node.
- The /delays* directory contains the results of the simulation. The file names use the following structure :

dGlobal_0_<lambda>_<routing_file>.txt, where <lambda> is the traffic intensity and <routing_file> indicates the routing used for the simulation. Each line in these dGlobal_0_*.txt files corresponds to a simulation with different traffic matrices probabilistically generated from a given traffic intensity (lambda). We describe below the data structure in each line (i.e., each simulation). Note that in a topology with ‘n’ nodes, nodes are enumerated in the range [0, n-1].

1-. Bandwidth (in kbps) generated for each source-destination pair in the network.

$$Index = src_{node} * n + dst_{node}$$

2-. Average delay over the packets transmitted for each source-destination pair in the network.

$$Index = n * n + (src_{node} * n + dst_{node}) * 7$$

3-. Percentile 10 over the packets transmitted for each source-destination pair.

$$Index = n * n + (src_{node} * n + dst_{node}) * 7 + 1$$

4-. Percentile 20 over the packets transmitted for each source-destination pair.

$$Index = n * n + (src_{node} * n + dst_{node}) * 7 + 2$$

5-. Percentile 50 over the packets transmitted for each source-destination pair.

$$Index = n * n + (src_{node} * n + dst_{node}) * 7 + 3$$

6-. Percentile 80 over the packets transmitted for each source-destination pair.

$$Index = n * n + (src_{node} * n + dst_{node}) * 7 + 4$$

7-. Percentile 90 over the packets transmitted for each source-destination pair.

$$Index = n * n + (src_{node} * n + dst_{node}) * 7 + 5$$

8-. Variance of the delay (jitter) over the packets transmitted for each source-destination pair.

$$Index = n * n + (src_{node} * n + dst_{node}) * 7 + 6$$

9-. Absolute number of packet losses in each source-destination pair.

$$Index = n * n + n * n * 7 + src_{node} * n + dst_{node}$$

10-. Total number of packets lost in the network. This is the sum of all the losses in the previous vector.

$$Index = n*n + n*n*7 + n*n$$

In order to use the delay/jitter RouteNet models already trained, you can find more information at https://github.com/knowledgedefinednetworking/net2vec/tree/RouteNet-SOSR/routenet.

If you have any doubts, or you want to discuss anything related to the data sets, you can subscribe and send an email to the mailing list kdn-users<at> knowledgedefinednetworking.org (Link: https://mail.knowledgedefinednetworking.org/cgi-bin/mailman/listinfo/kdn-users). New data sets are also notified through this mailing list.
If you want to contact our team for any other reason, you can send an email to kdn-contactus <at> knowledgedefinednetworking.org.