Egde Flow Hypothesis Test (EFHT) Project, Datahub Requirement

Overview of the workflow:

Graph generation -> output edge txt file when each graph is generated -> use the external software to classify the subgraph we need by using the output edge txt file -> it will output two files for each output edge txt file used by the software -> Flow generation -> perform tests when each flow is generated

MATLAB functionality:

MATLAB Scripts: The following files form part of the core:

Flow\_Examples.m: This is the frame function which contains all the steps we need to do for the workflow, and most steps can’t be done sepreately.

Random\_Flow\_Sampler.m: This is the file to run the whole program. It contains all parameters that needed for the graph and flow generation. So basically we can change parameters, graph distributions, flow distributions, test choices, subgraph types from this file.

For exactly how the subgraph searching and extraction part works:

When generating the graph, we have two parameters loop for i and j.

So it will generate graph1.1,graph 1.2, graph1.3 and so on. What we do in this part is in this loop, once the graph i.j is generated, we need to store the graph as an txt file, which means it need a place to store it. Then we need to run the external software to searching for specific subgraph we need, it will use the txt file that just created and output a txt file to store the result. (to run the program we need to rely on a Linux system, basically is to run this “./gtrieScanner -s 4 -g {input\_file} -u -o {output\_file\_base}.txt -oc {output\_file\_base}\_sub.txt -f simple.”in the Linux system commend . More details is showed below.

subgraph\_selecting.m: A key part of the subgraph extraction logic, this script selects subgraphs based on a given set of input graphs and stores the selected subgraphs for further processing.

perform\_test.m: This script is used for doing tests when the flow is generated. It contains different tests. Given tests choices in the Flow\_Examples.m, it will call the related tests automatically.

connected\_Erdos\_sampler.m,EDITTED\_fixed\_degree\_distribution\_sampler.m, WattsStrogatz.m: These three files are used for Erdo, Small World, Scale Free graph distribution. It will be called in the Random\_Flow\_Sampler.m file when the graph generation step. Don’t have to do any modifications, just need to keep it in the same folder

randl.m :This file will be called in the Random\_Flow\_Sampler.file when the flow generation step. Don’t have to do any modifications, just need to keep it in the same folder

The most dependency on a Linux system part:

gtrieScanner\_mex.cpp, gtrieScanner\_mex.mexw64 : The mexw64 file is generated by complieing the cpp file in matlab. The software needs to be run in the Linux system. And the software is put in Linux Ubuntu. So the file is designed for calling the software and run it in the windows system by calling it in Ubuntu.

gtrieScanner: It is the software used for looking for specific subgraph in the graph. It needs to be run in the Linux system. And the commend we use is “./gtrieScanner -s 4 -g {input\_file} -u -o {output\_file\_base}.txt -oc {output\_file\_base}\_sub.txt -f simple.” And we may need to change the number 4 in this command when we need to look for different subgraph type. Because the software needs to be called when each graph is generated, so the input\_file\output\_file\_base is written in the file Random\_Flow\_Sampler.m before the graph starts to generated.



The software description link is :http://www.dcc.fc.up.pt/gtries/

gtrieScanner\_src\_01.zip: It’s the software zip file