Project 3

Section one, paper review

Applications such as remote robotic surgery demands combinations of timeliness and reliability requirements in order for interaction between people in a natural way where round trip delay must be less than 130ms. However, the Internet only support reliable but not timely protocol TCP or timely with best-effort reliability protocol UDP. The paper have three main contribution.The first one is invention of dissemination graphs, providing a unified framework that makes it easy to spcify arbitray graphs. The second one is analysis of real world network data and finding out the problems which often occurs around the source or destination.The third one is coming up with targeted redundancy solution based network condition. The key idea is investing fewer resources in more reliable parts of the network and more resources in less reliable parts of the network in order to achieve the cost-reliability tradeoff. Due the network charactreristic ,analysised by the author, the approach uses two disjoint paths under normal conditions, and switches to use targeted redundancy dissemination graphs when network problems are detected. The testing results show that the approach achieve nearly optimal reliability, which is almost closed to time constrained flooding and is under the low and reasonable cost that is closed to cost of two disjoint paths routing.

Strength:

(1) It is a feasible solution and the protocol conducts fast recovery by switching to pre-compute dissemination graphs when detecting problems. And it uses bitmask to represent the graph.

(2) The experiment is very plausible. It lasts 4 weeks and based on global overlay network. The experiment gives us the result that using two disjoint paths performs well in most cases. It is a very useful result which can be used in later research.

Weakness:

(1) The approach given by the paper is only targeted to solve specific network problem where major failure cases occur around source or destination.

(2) The paper says: “overlay nodes are situated in well-provisioned data centers, where ISPs have invested in multiple independent fiber connections; overlay links are selected to follow the underlying network as much as possible (based on available ISP backbone maps); and overlay links are kept short to increase routing predictability”. I think it is not easy to provision. So I think this is a small weakness.

Justification: In previous work, researchers improve overlay routing performance only considering non-disjoint paths. However, this paper focuses on strict timely and reliable requirement and target specific problems ground on real world data. The paper makes the assumption that there is no limit of the amount of data sent on each link, but the network in real world could be more sophisticated. I doubt the data gathered during four months in some way could really represent the network situation of real world. As we all know, most of packets sent are came from big companies such as Google or Meta Platforms and some extreme case may happen during Chrismates or Thanks Giving. If the network condition is not the same as the author experience, there is no guarantee that the solution will work, which means there may be no precompute graph can solve the network problem. Overall, the idea is cool, precomputing the graph is fast and cheap given strict time constraint, and also convert the problem into a classify problem. Besides, using bitmask to represent the graph could be simpler and cheaper.

Section two, Analyzing claims and experiments

The paper claims that using two disjoint paths as routing approach performs well in most cases, and failure cases typically involve problems around a source or destination. And the paper concludes their approach can cover over 99% of the performance gap between a traditional single-path approach and an optimal scheme and this performance improvement is obtained at a cost increase of about 2% over two disjoint paths.

Authors collected data using Playback Network Simulator over four months on a global overlay network, including overlay nodes in twelve data center, spanning Asia North America and Europe through LTN global communications. Firstly, they conduct the initial dissemination graph construction approaches, including dynamic single path, static two paths, dynamic two paths, and time-constrained flooding, in order to see how the approaches would perform on their collected data. Then, after observing the performance, they develop targeted redundancy approach and use Playback Network Simulator to access the performance over the four weeks of data they collected over several months .

The paper presents 2 Tables. The table1 presents overall reliability and availability results for each of the dissemination-graph construction approaches. For every initial routing approach, the availability is at least 99.994% which is good and the unavailability is also good, ranging from 24.88 seconds /flow /week for Time-Constrained Flooding to 35 seconds /flow /week for single path approach. Based on the available, the author calculates the reliability and both time-constrained flooding and the targeted redundancy approach reach nearly 99.9999% and around 1.5 packets lost under these two approaches. The single path performs worse, its reliability reaches 99.9976%.

The table2 show the percent of benefit of time constrained flooding obtained by each approach and scaled cost given that the baseline is single path approach and the optimal is time constrained flooding. For the approach the paper given, it shows very good results for all 4 weeks, achieving nearly optimal reliability and covering 99.81% of gap between the baseline and optimal approach with scaled cost 2.098, which is very closed to cost of dynamic two disjoint Paths. It is obvious to see that two disjoint paths reaches dramatic improve on reliability compared with a single path, covering 70% of gap. Besides, the author list the distribution of time for targeted redundancy approach. The approach uses the static two node-disjoint paths dissemination graph 98.11% of the time, the source-problem graph 0.82% of the time, the destination-problem graph 0.68% of the time, and the source- destination-problem graph 0.39% of the time.

The results support most of the claims. Static two disjoint paths covers 44.58% of gap and takes up 98.11% of time of targeted redundancy approach, supporting the claims that using two disjoint paths as routing approach performs well in most cases. However, it is still not clear that failure cases typically involve problems around a source or destination, since no evidence support the claim, and I am expecting some chart to show the distribution of location where failure cases happened.

Section three, Assessing existing code and data.

1.My team uses spines to deploy a overlay architecture on Geni, and uses existing code for dissemination graph.

The code is available at http://spines.org/ .

2. Upon until now, my team have already set up a small overlay network with two nodes, and set 1000000kbps bandwith, 25ms latency and 1% loss in both directions.

And we test the program by sending 10000 traffic packets through the overlay network.

Section four. Experimental plan

We will emulate overlay network using Spines on Geni.

We will try bunches of experiments. For reproduce part, we want to show availability and reliability for each routing approach under 65ms latency constraint and the table of percent of the benefit of time constrained flooding obtained by each approach and scaled cost. Besides, we plan to show distribution of time of targeted redundancy approach to confirm the claims that two disjoint paths perform in major cases.

For extend part, firstly we plan to show the distribution of location where network problems (unavailable) may occur under two disjoint path protocol in order to confirm the key finding from authors, that is routing approach using two disjoint paths performs well in most cases, and that cases where two disjoint paths do not perform well typically involve problems around a source or destination.

Secondly, we plan to run the targeted redundancy approach under different network conditions. In order to do that, we will try to tune the parameter of loss rate and latency and also make another traffic generator which will sends may be 100 or 1000 times of traffic than usual at certain amount of time. Then we will see how it behaves, whether it still perform well or not. Besides, we also want to see how the algorithm will behave after tuneing the parameters of loss threshold, the parameters for detecting loss.