Relying on Autonomous Multipath Routing to Achieve Global Load Balancing in the Internet

Robert Löfman

Åbo Akademi University, Department of Computer Science

Abstract: In this study we are interested in the impacts of autonomous multipath routing and we investigated a situation where every autonomous system implements a multipath routing (MPR) protocol and where the inter-domain protocol has been updated to support MPR. In other words, we wanted to know if the current autonomic position of the domains can be retained while providing maximum global load balancing without cooperation between domains. It is the autonomic and hierarchical structure that has provided the scalability of the Internet and it is interesting to know whether global multipath routing will contradict this structure by needing cross-domain cooperation. By simulation we found that un-cooperative global MPR can improve the throughput of UDP but not TCP due to low throughput incurred by a small congestion window when utilizing high delay routes.

1. Introduction to the Simulation.

By setting up a partial Internet (with generators and real WANs) with global paths having realistic length and alternative routes, we were able to obtain results of internetwork scale. We realize the two MPR protocols MPR-2 and MPR-3 with EIGRP. In MPR-2 all routers with more than one nexthop for the destination will forward packets from one flow to two certain nexthops. MPR-3 forwards to three nexthops, and shortest path routing is denoted SPR. We test these with both UDP and TCP (MPR-UDP and MPR-TCP). We put no constraints on paths being disjoint, only that routers can forward packets to 3, 2 or 1 nexthops (thus with MPR-#, much more than # routes are used) on the route to one of the possibly multiple egress router of its domain. We also assume that BGP supports MPR. In order to eliminate the problem of spurious fast retransmits (FR) of TCP we set the maximum segment size (MSS) to 15000B as this value seems to guarantee no spurious FRs for many routes. The background traffic is also constructed to mimic the well known tri-modal packet size distribution, in that every link has a mean packet size of one of the peaks. The 25 runs plus further test were done with OPNET. Due to space limitation we omit the introduction, a detailed description of the simulation, related work and change suggestions to TCP, but these can be found in the complete paper [1].

2. Simulation of MPR-UDP.

With MPR we can balance the load between the shortest path (SP) and, for simplicity while discussing, one alternative path (ALT). Assuming that the ALT has less bandwidth (BW) than SP, then throughput will not improve when sharing the load. If, however, other packets can be transmitted on the SP instead of the shared packets (the data rate is increased), and no regard for packets being reordered is needed, then we can always experience an increase in throughput after the propagation delay of the ALT. Within the inter-domain of the Internet it is not possible to “source route” nor to perform proportional routing. Thus, assuming round robin packet dispersion, two routes will receive equal amount of packets, meaning that if we increase the data rate too high, the ALT will drop packets. This only happened during 96% background load in our simulation and we believe this did not occur during lower loads because of the number of ALTs used. UDP displayed significant improvement in throughput when using MPR (figure 2.1a). The statistic looks positive for MPR, but we cannot always guarantee that this is the “real” improvement (see “in-sequence throughput” below).

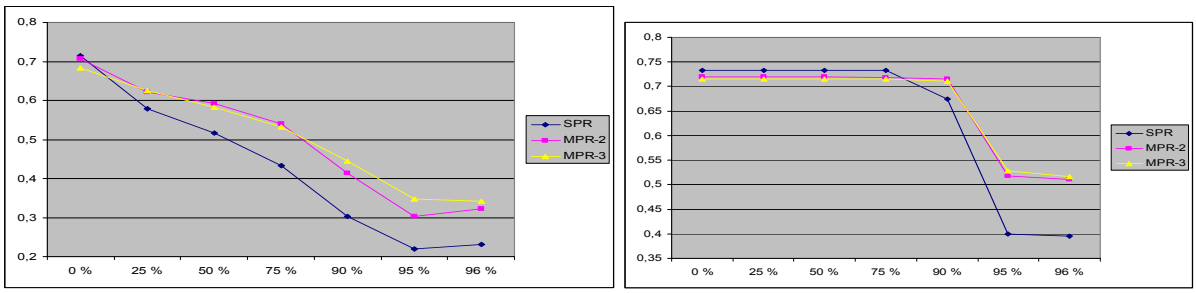

Fig.2.1a left: Average throughput (fraction of data sent) as a function of background load when transmission rate is 300000000 pkt/h of size 1460B each. Fig.2.1b, right: Throughput when transmission rate is 200000000 pkt/h.

When there is no background traffic in the network SPR performs better. This is because distributing traffic over sub-optimal routes while the SP still has enough residual BW, or residual bottleneck BW to be precise, to qualify

as the SP (in terms of transmission time of a packet), will not improve throughput. As the background load is increased, MPR-2 and MPR-3 begins to perform significantly better. Figure 2.1b shows an average statistic from a set of simulations where the transmission rate is much lower than in 2.1a, which results in MPR-# being less advantageous. This indicates that the level of improvement of MPR depends on the transmission rate. The alternative paths can thus be used when the flow itself saturates the SP due to its high send rate (or low residual BW) as happens already at a much lower load with higher transmission rate in figure 2.1a. The SP has, at that point itself become sub-optimal for that transmission rate compared to a possible aggregated BW of MPR. In the worst case this means that packets have to be dropped or at least buffered and thus experience extra delay. It is the *transmission rate to residual bandwidth ratio* which determines if the SP still transmits packets faster than MPR on aggregated paths. It can be noted that MPR-3 always performs the worst at low load and always the best at high load. The "switch" of best scheme happens at the crossing point of the graphs where the SP is saturated by the transmission rate (i.e. the ratio is one). Thus, as the ratio approaches one, it is appropriate to use MPR. If it is allowed to go above one, without adding alternative paths, the throughput will not improve, and the only way to increase the transmission rate and thus the throughput from that point on is to add a new alternative path. Assuming that the added path has residual BW, it will allow the sender to increase its send rate and achieve a higher throughput after the initial propagation delay of the alternative path. More alternative paths can be added with the same reasoning when the current aggregate BW is saturated. The initial propagation delay of the added path will determine at which point throughput is boosted. A problem with using multiple routes is indeed reordering. When adding a route, reordering proportional to the delay of the alternative path will occur. After that the "lag" between the highest received sequence number and the lowest missing out-of-order packet will either increase or decrease according to the throughput of the alternative routes. The factors to take into account when providing global autonomous multipath routing are the following: (A) The transmission rate of the sender, (B) The residual BW of the SP and alternative paths. We place little importance in the initial propagation delay, other than that packets will be reordered according to this amount in beginning. Assuming that (1) transmission on the SP and alternative path starts at the same time and (2) every router performs round robin forwarding of packets to nexthop in the order they are received, we can reason with help of figure 2.2 in the following way (what we guarantee with the third assumption is that every n^{th} packet in the sender's packet sequence will traverse the same, n^{th} route):

Fig.2.2: Packet lag

By using the aggregate BW of the SP and the alternative path (Alt.P) the receiver will have packets 1 through 5, in-sequence (=without missing packets having smaller sequence number than 5), by the time T4. When comparing this to transmission with the SP only, packets 1-5 will have been received later at T5. The in-sequence BW is the amount of packets received per second (pps) that do not have a previous packet outstanding in the sequence. The in-sequence BW for the aggregate path at time T4 is 1.25 pps and 1 pps for SP. The out-of-sequence BW for the aggregate path is 1.5 pps. The delay of the alternative path is two time units which is the lag that will be present throughout the transmission, if the transmission time for packets is the same. If, however, the alternative path has more residual bandwidth, will the transmission on this path start to gain on the SP after the initial propagation delay. If the path with the longest initial propagation delay does not have the least residual BW, and provided that there is enough data to send, this path will at some later point have transmitted more packets than the path with the least residual BW and smaller initial propagation delay. At that point the in-sequence throughput will start to depend on the other path with the least residual BW. What is worse is that this lag will then increase since "packets travel faster" on routes with more residual BW. The problem of lagging routes adheres from the fact that with inter-domain routes it is impossible to guarantee proportional routing, that is, to assign a smaller amount of packets to lagging routes (and thus reduce waiting for out-of-sequence packets). This can be done in intra-domain routing, but an ignorant BGP speaker may assign just as much traffic to its peer in AS1 as it does to its peer in AS2. Unfortunately, the route through AS2 may be congested. The initial propagation delay can be ignored, assuming that we have enough data to send that this delay becomes insignificant compared to the transmission times of the "real" lagging route in the long run.

Theorem 1: *If all n ALTs in MPR have a BW greater than or equal to (SP_{BW}/ n) pps, where SP_{BW} is the BW of the SP, then MPR will always have a throughput greater or equal to SPR..*

Proof: *Since the slowest ALT has a BW of ALT_{BW} , equaling or more than (SP_{BW}/ n) pps, it is possible to transmit at least n packets in parallel, simultaneously on n routes by the time (n/ SP_{BW}), which is the time the n^{th} packet would have arrived when using the SP only. Thus (1/ ALT_{BW}) ≤ (n/ SP_{BW}) holds, as it must* □

Assumption (2) ensures that packets on the lagging route arrive with a constant rate and thus the in-sequence BW of the aggregate route is constant and dependant of the rate of the lagging route. If assumption (2) is not included then a packet p that was put before on the lagging path may now be put on another, "faster" path and thus the BW may increase. This is only temporary if another packet is put in p's place on the lagging path. It is only the in-sequence BW that is limited by the route with the smallest residual BW. It can be noted that in MPR it does not matter how much BW the other alternative routes have, if the ALT with least BW have less BW than what is specified in theorem 1, since then the n^{th} packet cannot be received by the time n packets have been transmitted on the SP and SPR will thus be advantageous. The great challenge is to know what transmission rate to use for how many alternative routes. This is of importance in the future as gigabit and 10-gigabit Ethernet last-mile links become more common or if the edge to core bandwidth ratio changes for some other reason.

3. Simulation of MPR-TCP.

It is known that TCP performs badly with MPR due to the reaction of FR-algorithm to unordered packets [2]. With UDP we saw improvements in performance due to the fact that we could freely choose a high transmission rate. From these simulations of MPR-TCP, it will become evident that there are more problems than that of the FR-algorithm. Let us first take a look at the statistic of transferring a file with FTP (which uses TCP) in figure 3.1.

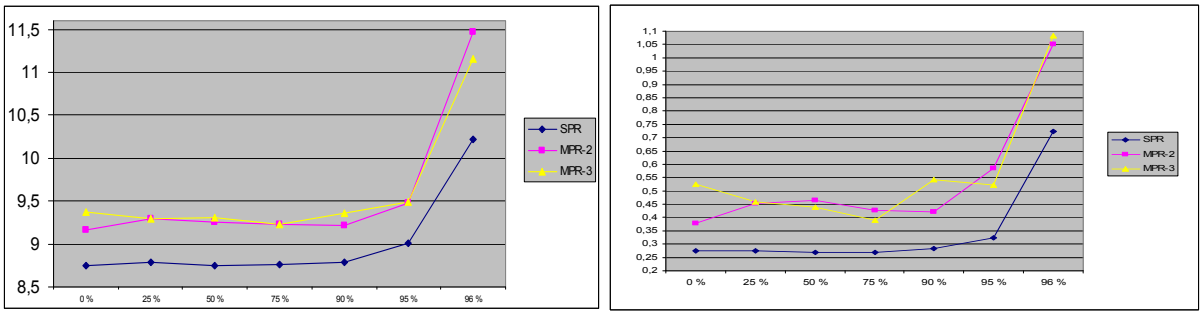

Figure 3.1, left: Average response time (seconds) of FTP. Figure 3.2, right: Average cumulative pause time caused by a full CWind

This is the time it takes to transmit a file and receive an acknowledgement for it, as a function of the background traffic load. SPR always performs the best. The cause of bad MPR-# performance was traced to a small CWind (figure 3.2 indicates that MPR pauses transmission more) for MPR-# (figure 3.3).This graph is a representative sample from one run with each MPR-#/Load combination and it shows the size of the CWind at measuring point in time. The graph is time-skewed for readability. In all gathered statistics the CWind grow more slowly, and to a smaller max, for MPR-#. Tests with a normal 1.5K maximum segment size showed even smaller CWind as the standard deviation for 1.5K MSS is about one magnitude larger. The reason for a slowly growing CWind is obviously long packet transfer times. It was evident that with MPR, much smaller amounts of data were acknowledged. This is displayed in a representative[1] sample in figure 3.4. We can see that SPR is able to transmit much larger burst (fewer and higher sequence numbers). Since at the beginning of transmission, all settings are equal for SPR and MPR, it must be the bad throughput of MPR (either long propagation delay, see equation 1, or the fact that some fragments face low residual BW) that spoils the growth of the CWind. This is curious when reminiscing about the improvement of MPR-UDP and the fact that MPR-TCP has a much greater bandwidth. When the transmission of packets on alternative paths take too long, the receiver will send acknowledgements for smaller bursts, therefore increasing the CWind for the sender only by a little, meaning that the following maximally allowable burst will also be smaller for MPR-#. With the large MSS size we do not raise the throughput to the level of out-of-sequence throughput, since the fragments still deviate as much as normal sized segments. See the full paper [1] for statistics which shows that the average fragment transmission times of MPR-# is somewhat longer and that the jitter of fragments is significantly larger for MPR-#. With large MSS we merely ensure that all segments deviate roughly the same amount by increasing the chance that all segments experience the same transmission time (some fragments of every segment traverse low BW paths), but in this way more partial segments will have to wait (be buffered) for slowly arriving fragments. Thus, the throughput of large MSS still follow the rule of lagging routes proved above (think of the fragments from different segments as a sequence of packets). It would seem logical that MPR could take advantage of this bandwidth and have a higher throughput at some point, although the delay is inevitably longer for MPR-# with suboptimal alternative routes.

[1] The sample is typical to the degree that neither MPR scheme ever performed better than SPR.

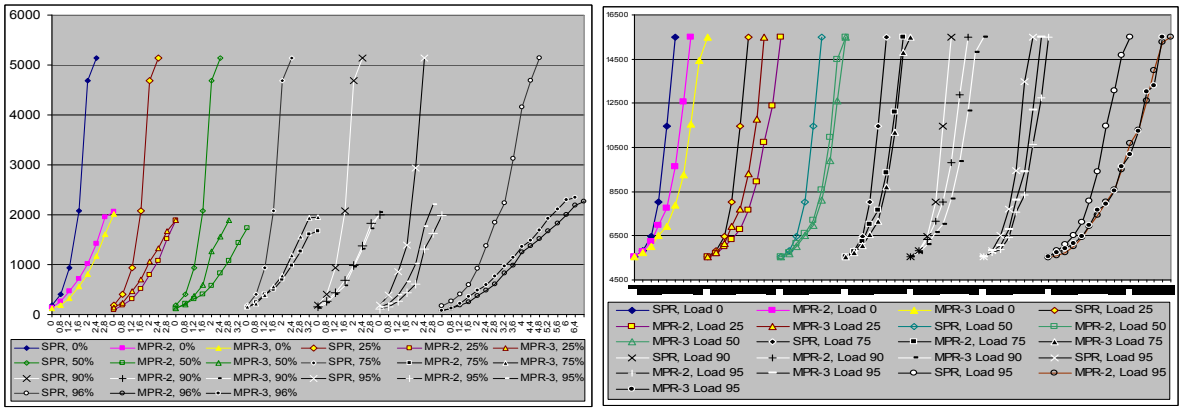

Figure 3.3, left: The congestion windows (in thousands) for MSS=15000 as a function of time (seconds). Figure 3.4, right: Sent segment sequence numbers (in thousands) as a function of time (seconds).

To take an example proportional to our simulations, an MPR route with a BW of 2 Gbps and a SPR path with BW of 1.5 Gbps and the former has a delay of 70 msec and the latter 50 msec. In order for MPR to be advantageous a burst size of at least 15000 KB would have to be sent. The problem is now that the CWind portions the burst into, starting from one segment and then increasing the portions exponentially per round trip time (RTT). This means that until the CWind reaches the point were burst size is large enough for MPR to be advantageous, we will loose throughput proportionally to the propagation delay, every RTT. The amount of lost throughput, can be calculated as equation 1 (assuming that burst_size is smaller than what is needed for MPR to be advantageous):

$$\sum_{n=1}^{\#bursts_{MPR}} \left(\frac{burst\_size\_MPR_n}{BW_{MPR}} + delay_{MPR} \right) - \sum_{n=1}^{\#bursts_{SPR}} \left(\frac{burst\_size\_SPR_n}{BW_{SPR}} + delay_{SPR} \right) \quad (1)$$

As the burst size is enlarged by CWind, MPR is able to win more and more time per burst, but in our simulations for instance, even with a file size of 20MB, the CWind never becomes large enough to even have some burst transmitted faster. This means that TCP-MPR will always suffer from the large propagation delay without being able to take advantage of its high BW during every burst. Since then, the CWind of SPR will grow faster (all statistics of the CWind showed a significant jump SPR already at the beginning), SPR will be able to transmit at an even higher rate. It seems counter-intuitive to claim that MPR could improve the throughput of TCP when it is in fact the scheme itself that ruins the growth of the CWind. It is however plausible, that if the transmission rate would not be constrained by the CWind, a high enough rate could be found which only MPR can facilitate.

3. Conclusions.

It seems that it is the lag of some fragments which spoils the growth of the CWind for MPR-TCP as this scheme has a massive bandwidth-delay product. This relates to the fact that proportional routing is not possible on a global scale at the moment and too much fragments are transmitted on “slow” paths.

References.

[1] “Relying on Autonomous Multipath Routing to Achieve Global Load Balancing in the Internet”, Robert Löfman, http://www.tucs.fi/publications/insight.php?id=tLofman05a&table=techreport, 2005.

[2] “TCP Performance over Multipath Routing in Mobile Ad Hoc Networks”, Haejung Lim, Kaixin Xu, and Mario Gerla, In Proceedings of ICC 2003.