

Let there are several paths from  $i$  to  $j$ . Let them denote as  $P = P_1, P_2, \dots, P_k, \dots, P_n$ .

The message will be broadcasted, but the time it takes from  $i$  to  $j$  will be unique, because only one message will reach the destination first even if interval between messages are milliseconds or less due to processing rules. Let the probability of choosing result fastest path be  $p_k$ . That means that  $\sum_{k=1}^n p_k = 1$ . Delay == time taken for a signal to traverse a network from point A to point B.

But let's consider  $p_k$  in detail.  $p_k$  is the probability of that this path will be the fastest who transport message to destination.

But what it means - the shortest path? The shortest path is minimum of all sums for each link cost of all paths. But how do we count link cost? Each network (provider, actually) has own metric that identifies as the result "link cost". (in non oriented graph, but we consider message going in one direction to reach destination and negotiate messages going in opposite one). So that each link has own metric that identifies how much it costs to go through this (more expensive, cheaper, longer, faster, etc. ). In this metric many parameters are included. (see [https://en.wikipedia.org/wiki/Metrics\\_\(networking\)](https://en.wikipedia.org/wiki/Metrics_(networking)) ).

So depending on size of message, period of incoming, path reliability, bandwidth, mtu, throughput, delay, this metric is calculated. Here even fault tolerance for the link and nodes are included, so it has impacts on availability and partition tolerance of full network. That means that having a metric, we can negotiate other considerations about partition tolerance and availability. The only thing is that each network provider has own metric calculation, own formula for that.

Let's remember this. Let's also remember that administrator of network (local network) knows the metric of each link. This means if datastore is limited to this network, the shortest path is known. Note, that link cost is calculated dynamically depending on the network settings and dynamic changing of devices.

Let this network be in London. Amazon Web Services has good infrastructure, that is separated by regions and availability zones. Region can be european west 1 or usa west 2 etc. Each region has about three availability zones (or more). If the datastore has similar infrastructure and gateways between each other that are synced periodically (due to changing of time belts moment syncing is not needed, but syncing on demand is left along with syncs twice per day, or once per two hours or more often, for example).

So let's consider separate network since we don't sync between regions and do not need instant global consistency. We want full consistency in just one network where we know the metric for each link.

Again, having this metric we may not consider partition tolerance and availability for the datastore since these values may be included in metric of network provider (depending on provider). We have a graph with known weights and shortest path will be indeed the shortest considering latency, availability of path, data loss etc. Along with this metric another metric can be implemented to the chosen shortest path. These both metrics will be dynamic. In time the metric of time through the shortest path will have statistical values to calculate average

and this will give the average coming through the diameter of the graph. This will mean the maximum period between messages incoming to such datastore.

Now let's consider multiple messages to the datastore. As we already know shortest path in given network does not depend on number of messages, it only identifies the limit of messages in size of bits.