A primer on reliability as applied to amateur radio packet networks.

T.C. McDermott, N5EG
Texas Packet Radio Society, Inc.

1.0 Scope

Many messages have been sent regarding linking of large number of packet radio switches, nodes, digipeaters, etc. And some have commented on the desirability of very long packet networks. This monograph will describe how to calculate the availability of such a system, given knowledge of the performance of the equipment.

2.0 Definitions

Let's define some terms, first. There are 3 basic parameters that need to be known in order to make suitable calculations about network availability.

MTBF = Mean Time Between Failures. This is the average (mean) time between failures of a particular piece of equipment. For example: an MTBF of 1680 hours would equate to a piece of equipment failing once every 10 weeks, on average.

MTTR = Mean Time To Restore. This is the average (mean) time to restore a failed piece of equipment to service. For example: a piece of equipment with an MTTR of 8 hours implies that it takes 8 hours to: 1) notice that there is a problem, 2) diagnose the problem, 3) drive to the site, 4) repair the equipment, and 5) place it back in service. Of course the actual series of events, and the time to restore all depend on whether the equipment is accessible at any time, backup equipment is available, etc.

A = Availability. This is the portion of time that a piece of equipment (on average) is available for use. This can be calculated as follows:

$$A=MTBF/(MTBF+MTTR) \qquad (1)$$

3.0 Some basic probability

Let's use some of the basic rules of probability to derive the availability of networks of equipment that each has availability, A. There are two basic configurations of multiple pieces of equipment: 1) series, and 2) parallel. By this is meant the following: two pieces of equipment are in series if both are required to be operating correctly in order to get the job done. For example: suppose that you wanted to transmit a packet message across a 100-mile path, and there were two switches in the middle that were linked, and that the failure of either would prevent your packets from traversing the path.

Then those switches, from a reliability point of view, are in series. The failure of either one of them would make the path not usable. In contrast, two pieces of equipment are in parallel when either alone is capable of getting the job done. For example: suppose that you wished to send a packet between two points that are 50 miles apart, and you had a choice of either of two switches, each of which alone was capable of making the path. Then those switches, from a reliability point of view, are in parallel.

3.1 Availability of things in series

We can calculate the availability of 'n' items, all with the same availability, 'A', that are in series. The combined availability is:

$$A_{n(ser)} = A^{n} \quad \text{(A raised to the 'n'-th power)} \qquad (2)$$

For example, suppose that we have a packet network consisting of 20 nodes, that each individual node has an MTBF of 4368 hours (6 months), and an MTTR of 168 hours (1 week). Then the availability of a single node is 4368 / (4368 + 168) = 0.963 (96.3 percent of the time it works). The availability of a network of 20 of these nodes would be: 0.963^{20} = 0.470 (47.0 percent of the time it works). We can see that, in general, putting items in series degrades the availability.

3.2 Availability of things in parallel

We can calculate the availability of 'n' items, all with the same availability, ‘A’, that are in parallel. The combined availability is:

$$A_{n(par)} = 1 - ((1 - A)^{n}) \qquad (3)$$

For example. suppose that we have a packet network consisting of 2 nodes, with MTBF = 4368 hours, and MTTR = 168 hours, and these two nodes are in parallel. Then the individual availability is 0.963 (as in 3.1 above) and the combined availability is 1 - (l-0.963)* = 0.9986 (99.86 percent of the time it works). We can see that, in general, putting items in parallel improves the availability.

3.3 More complex models

We can calculate the availability of more complex networks many times by reducing them to series and parallel combinations that we now know how to handle. Sometimes, the combinations are not reducible to series-parallel combinations, but these cases are not common in amateur packet networks. The general procedure is to break up a network into subsections that can be described as being in series or parallel. Then each subsection can in turn be broken up into smaller subsections that are in parallel and or series, until the remaining network segments are entirely parallel or series.

The availability of each subsection can be computed, and the subsection availability’s can be combined using (2) and (3) above to derive the network availability.

4.0 Some examples

Lets look at two example networks. Network one consists of 40 packet switches, all in series. It's a long haul network, and skinny (i.e.: no alternative routes exist within the network from end-to-end). Each node has an MTBF of 4368 hours, and the MTTR is 332 hours (2 weeks, since the sysop left on vacation yesterday, and does so frequently! - nice work if you can get it). Then:

$$A = 4368 / (4368+332) = 0.929$$

$$A_n = 0.929^{40} = 0.053$$

This network will function, end-to-end, 5 percent of the time, and will not work end-to-end 95 percent of the time. Hmmmm. OK, let's assume that our sysop loses his cushy job, his extravagant vacation policy, etc., and can get to the site within 72 hours. Then the network availability would be:

$$A = 4368 / (4368+72) = 0.984$$

$$An = 0.984^{40} = 0.520$$

Well, quite an improvement. The network actually works, end-to-end, 52% of the time. We can draw some conclusions about the level of service our poor sysops are going to have to provide if we want this stuff to really work. Alternatively, we could do some work up front, and build dual-redundant nodes. Those are ones with hot-standby equipment that takes over the failed equipment with no loss of service (even after lightning hits!). So, for network example number two, let's double the investment in our network by providing dual-redundant nodes at each of the 40 sites. Incidentally, building dual-redundant equipment without common (joint) failures can be no small task in itself. The availability of this network, assuming our intrepid sysop finds out that he now takes 2 week business trips all the time, can be calculated by breaking our network into some subsections. Each dual-redundant node is a subsection, and we have 40 of those subsections in series. So, first we calculate the availability of the subsection:

$$A_{2(par)} = (1 - (1 - 0.929)^2) = 0.995$$

and then the availability of all subsections would be:

$$A_{40(ser)} = 0.995^{40} = 0.817$$

Well, that's more like it. This network works 81.7 percent of the time, end-to-end, and the poor sysop can actually hold down a real job now. Ahhh, wait a minute. We have twice as much equipment in the network now, and thus it seems like twice as many things would break. Well, yes. Welcome to the dark side of the force - err, the dark side of high availability. In order to achieve this level of availability, we have to fix any failed equipment within 2 weeks - even if the failure does not take the node out of service. If we don't fix it, then the remaining part that still works now determines the node's availability, and we are no better off than before (at this node). So, there is no free lunch. Also, we have to be able to detect that something at the node has failed, even though it is still working. OK, so let's just put up two different packet networks each one of which reaches the two endpoints, but without any of this dual-redundant nonsense. In this case, we can model the two subsections as 40-element series connections of non-redundant nodes, and then we have two of these long strings in parallel.

$$A_{40(ser)} = 0.053 \text{ (from above)}$$

$$A_{2(par)} = (1 - (1 - 0.053)^2) = 0.103$$

Well, this strategy didn't work very well compared to making each node dual-redundant. So it seems like our poor sysop is stuck in engineering dual-redundant nodes if we want our networks to work reliably. Commercial telecom equipment is generally engineered this way.

5.0 Conclusions

Some conclusions we can draw about the results from our two examples are:

1) Engineering of fault-tolerant nodes is essential for long packet routes based on a number of packet nodes. Redundancy can be provided at the equipment level, or with alternate nodes having the same connectivity to the network.

2) Or, alternatively, we should focus on long-hop technologies such as satellite, HF, scatter or land-line telephone(?) connections. One drawback of such long-haul technologies is that we may lose real-time communications between end operators. Usually, gateway stations perform store-and-forward routing over HF, satellite, and land-line connections. And effective scatter communications probably would require specialized high-power gateway stations also. These techniques usually lose real-time capability since he media may not support propagation 100% of the time (e.g.: HF).