Fixing timeout intervals for lost packet detection in computer communication networks

by ROBERT J. T. MORRIS
Bell Laboratories
Holmdel, N.J.

INTRODUCTION

In a packet-switched data communication network which provides internal packet accountability via an end-to-end positive acknowledgment protocol, it is necessary to include a mechanism for detection and retransmission of missing packets. In particular, a decision must be made as to how long a sending element should reasonably wait before declaring an unacknowledged packet lost and initiating a recovery action. This waiting time is one of a class of system parameters which have come to be known as timeout intervals. The choice of a timeout interval is a delicate problem. If it is too short, network capacity is wasted by frequent unnecessary actions for packets which are not lost but merely delayed. If the timeout interval is too long, lost packets cause needlessly prolonged delays before recovery is initiated.

Several recent papers have considered this problem. Sunshine considered a retransmission scheme where a packet is retransmitted every $T$ seconds, until an acknowledgment of the packet is received. He considered the quantities mean delay including retransmission and mean number of transmissions. He illustrated the analysis by considering an Erlangian delay distribution with a defect representing packet loss. As $T$ is varied, the curve mean delay versus mean number of transmissions exhibits a definite "knee."

Fayolle, Gelenbe and Pujolle conducted an analysis of a related protocol. A packet is retransmitted every $T$ seconds until an acknowledgment of the last retransmitted packet is received. They derive individually optimal values of $T$ for three separate performance measures: buffer throughput (maximized), response time (minimized) and loss rate due to buffer overflow (minimized).

Buttò, Colombo, Tagliasco and Tonietti conducted an analytic and simulation study of a nontrivial network and demonstrated that overall network performance could be severely degraded by employing too short a retransmission timeout interval. They identified the following interesting network phenomenon: if a network reaches a critically busy state, delays may be sufficiently long that timeouts continually occur, resulting in a flooding of the network with retransmission traffic which in turn increases delays, causing unstable traffic growth until network saturation occurs. We will discuss ways of averting this type of behavior in a later section.

McQuillan and Cerf gave the rule of thumb: the timeout interval should be set equal to the round-trip delay plus a "fudge factor" to account for variance.

The treatment we give here is distinguished by the following features:

1. We begin by identifying what appear to be the performance objectives of major importance in normal network operation. We use a simple model to produce a design method for the timeout interval which simultaneously satisfies each objective as well as possible.

2. We are particularly interested in networks using fixed routes or virtual calls. In contrast to References 2 and 3, we do not assume independently distributed packet round-trip times, since a packet and any retransmissions will normally follow the same network path causing highly correlated round-trip times.

3. Despite the inevitably subjective nature of the problem, we defend our design rules by demonstrating the presence of an inherent generality in the approach.

4. We consider the practical implications and caveats of the results. Of particular importance are the additional considerations which arise when we allow the possibility of abnormal network conditions caused by faults or congestion. It is shown that in order to obtain satisfactory operation, it is highly desirable that the timeout mechanism be made adaptive in an autonomous manner to the network and connection condition.

A SIMPLE MODEL

In this section we construct a simple model comprised of a network path characteristic and a rudimentary retransmit protocol. The model serves to identify some important performance issues which should be taken into account when fixing the timeout interval $T$ in normal network operation.

Consider a path between two nodes $A$ and $B$ of a packet-switched network. Packets are transmitted from $A$ to $B$, and $B$ returns individual acknowledgment packets to $A$. Suppose the probability that a packet is not acknowledged (forward
packet or acknowledgment lost) is \( L \) and if the packet is not lost, its round-trip delay (transmission to receipt of acknowledgment) is governed by a continuous distribution \( F \) with finite mean \( \mu \) and complement \( F_0 = 1 - F \).

Next, consider a retransmit protocol wherein a packet \( P \) is retransmitted every \( T \) seconds until an (any) acknowledgment of \( P \) is received.

The main performance issues in choosing \( T \) are as follows:

1. \( T \) should not be so short that frequent "false alarms" occur, i.e., frequent retransmissions of packets which are not lost but merely delayed.
2. \( T \) should not be so long that a lost packet causes an unnecessarily prolonged delay in system recovery. A long recovery delay affects system performance in two ways. Firstly, delayed recovery causes prolonged retention of unacknowledged packets in the retransmit buffer, and consequently a wastage of shared buffer space. Secondly, the delayed eventual packet receipt at node \( B \) caused by a packet loss can cause a delay which is directly apparent to an application such as terminal to host computer interaction.

Other performance issues related to abnormal network conditions are discussed later.

We now quantify the above considerations. Rather than assume that the round-trip delays of a packet and its successive retransmissions are independent random variables, we adopt the following assumption which is appropriate for fixed-route or virtual call paths.

**Assumption A:**

No retransmitted packet will pass an earlier version of the same packet.

Now let

- \( D \) = time from packet first sent until packet acknowledgment first received, i.e., effective round-trip delay.
- \( N \) = number of transmissions required to successfully deliver one packet.
- \( F \) = maximum round-trip delay of a packet which is not lost = sup \{ \( F(i) \) \mid \( i \leq F \) \}
- \( X_i \) denote the \( i^{th} \) retransmission of packet \( X \), \( i = 1, 2, \ldots \).

Then

\[
E(N) = \sum_{i=1}^{\infty} i \cdot \text{Pr}\{X_1, \ldots, X_i \text{ only sent}\} = \sum_{i=1}^{\infty} i \cdot \text{Pr}\left\{ \bigcup_{j=0}^{i-1} \{X_j, \ldots, X_{i-1} \text{ lost}; X_i \text{ acknowledged in } [(i-1)T, iT]\} \right\},
\]

by Assumption A

\[
= \frac{1}{1-L} + \sum_{i=1}^{\infty} F(iT)
\]

and

\[
E(D) = \sum_{i=0}^{\infty} (\mu + iT) \cdot \text{Pr}\{X_i \text{ first to be acknowledged}\}
= \sum_{i=0}^{\infty} (\mu + iT) \cdot \text{Pr}\{X_0, \ldots, X_{i-1} \text{ lost}; X_i \text{ acknowledged}\},
\]

by Assumption A

\[
= \sum_{i=0}^{\infty} (\mu + iT)L^i(1-L) = \mu + TL/(1-L) \tag{2.2}
\]

We now go on to show how \( E(N) \) and \( E(D) \) are related to considerations 1 and 2, stated previously.

Let us define transmission efficiency \( \eta \) as the reciprocal of the number of packets transmitted per successfully delivered packet. A lowering of \( \eta \) causes an increase in effective network load which in turn leads to a depletion of the capacity of the network to do useful work. Consideration 1 is embodied in the requirement that \( \eta = E(N)^{-1} \) be kept high. Note that the maximum network "throughput" allowable for a given delay figure will be proportional to \( \eta \). Thus our first stated objective is to choose \( T \) to keep \( E(N) \) small.

According to (2.1), \( T \) should be chosen large.

We now turn our attention to consideration 2. First note that \( E(D) \) describes the mean holding time in the sender's retransmit buffer. Assuming that the path throughput is constant, then it follows from Little's results that mean retransmit buffer occupancy is proportional to \( E(D) \). Attention to mean buffer occupancy is reasonable since buffer space at the sender is shared amongst many independent calls. Next assume (although this can be relaxed) that both packet round-trip delay and loss are approximately equally shared between forward and return paths and that \( L \) is small. Then the mean time to successful receipt of a forward packet is approximately \( E(D)/2 \). Thus both aspects of consideration 2 are taken into account by requiring that \( E(D) \) be kept small. Hence our second stated objective is to choose \( T \) to keep \( E(D) \) small. According to (2.2), \( T \) should be chosen small, in conflict with our earlier requirement.

In order to study this demonstrated conflict between the requirements of avoiding false alarms yet retaining short delays in the presence of loss, we examine more closely expressions (2.1) and (2.2).

Assume for simplicity that \( L \) is non-zero and \( F(i) \) is strictly increasing for \( i \leq F \). Then \( E(N) \) and \( E(D) \) are shown diagrammatically as \( T \) varies in Figures 1a,b. The "tradeoff curve" \( E(N) \) versus \( E(D) \) is shown in Figure 1c.

Considering the multiobjective optimization problem of minimizing both \( E(N) \) and \( E(D) \) by choice of \( T \), we see immediately that any choice of \( T \leq F \) is pareto-optimal (also known as efficient, noninferior, etc.). The specification of an "optimal" value for \( T \) requires precise knowledge about the relative costs of \( E(N) \) and \( E(D) \). In the absence of any such knowledge, one way of identifying the knee of the tradeoff curve, Figure 1c, is to demand that the operating point be such that the relative degradations of both objec-
E(D)
E(N)
\[ E(D) = L \]
\[ E(N) = \frac{1}{1-L} \]

The value of \( T_e \) is shown in Figure 1c and always exists by our assumption that \( F \) is continuous and \( L \) is non-zero.

We will be particularly interested in the case where \( L \) is small, so it is of interest to see what happens to \( T_e \) as \( L \) approaches zero.

**Proposition I:**

\[ \lim_{L \to 0} T_e(L) = F' \]

**Proof:**

Assume first that \( F' < \infty \). Then \( T_e \neq F' \) or else (2.4) would be contradicted. Letting \( L \to 0 \), the left-hand side of (2.4) approaches zero. The only way for the right-hand side of (2.4) to approach zero is for \( T_e \) to approach \( F' \).

Now consider the case \( F' = \infty \). Pick \( L_0 \in (0, 1) \). Then for \( L < L_0 \), \( T_e(L) \) must be unbounded, for if it were bounded the right-hand side of (2.4) would be bounded below, but the left-hand side of (2.4) cannot be bounded below unless \( T_e(L) \) is unbounded. The observation that \( T_e(L) \) increases as \( L \to 0 \) completes the proof.

The effectiveness of the design decision \( T = T_e \) naturally depends on the relative importance of the two objectives \( E(N) \) and \( E(D) \) and in turn the considerations 1 and 2. However, if the value of equation (2.4) is small then little improvement of either objective is possible, and then only at the expense of a worsening of the other objective. Thus when the value of equation (2.4) is small it would be difficult to find grounds on which to criticize the choice \( T = T_e \) and for this reason the design can be considered sound.

Hence it is of interest to note that for small loss probability \( L \), the value of equation (2.4) is also small.

**Proposition II:**

\[ \lim_{L \to 0} \frac{L}{1-L} \frac{T_e(L)}{\mu} = 0 \]

**Proof:**

Assume first that \( F' < \infty \). By Proposition I as \( L \to 0 \), \( T_e(L) \to F' \) and the result follows.

If \( F' = \infty \), then as \( L \to 0 \), \( T_e(L) \to \infty \). But

\[ \sum_{i=1}^{n} F^e(iT_e) \leq \frac{1}{T_e} \int F^e(t) dt = \frac{\mu}{T_e} \]

which also approaches zero.

A natural question which now arises concerns the sensitivity of the performance objectives \( E(D) \) and \( E(N) \) to a variation of \( T \) away from \( T_e \). We shall answer this question briefly for the case where \( L \) is small. By observation of Figures 1a and 1b, it is seen that if \( T \) is increased above \( T_e \), the degradation of \( E(D) \) will be negligible (since \( L \) is small) and the improvement of \( E(N) \) slight. On the other hand, if \( T \) is decreased below \( T_e \), there is a possibility of a serious worsening of \( E(N) \) caused by excessively many false alarms. We consider this possibility further in a later section. An observation of some general value is that for systems with low loss probability, if there is any uncertainty, \( T \) should be made longer rather than shorter.
ALTERNATIVE APPROACH AND COMPARISON

Given a delay distribution and loss probability, we have given a technique for choosing a timeout interval. This technique was based on an examination of overall system objectives. We now develop an alternative approach.

We adopt the same model as in the previous section and continue to employ Assumption A.

Consider the first transmission of a packet. Denote by \( q(t) \) the conditional probability that the packet is lost given that it has not been acknowledged by time \( t \). Thus

\[
q(t) = \frac{L}{L + (1-L)F(t)}
\]

The curves \( q(t) \) and \( 1-q(t) \) are plotted in Figure 2. The significance of the point \( T_d \) shown on the figure is that

\[
\Pr\{\text{packet is lost | packet not acknowledged by } T_d\} = \Pr\{\text{packet is not lost | packet not acknowledged by } T_d\}
\]

and for \( t>T_d \), presently available information implies that the packet is more likely lost than delayed. Assumption A now implies that if the packet is delayed but not lost, there is nothing to be gained by retransmission.

Thus, in the absence of further information, an appropriate time to declare the packet lost is at \( T_d \). Note that \( T_d \) satisfies

\[
q(T_d) = 1 - q(T_d)
\]  

or

\[
F^c(T_d) = \frac{L}{L} = \frac{L}{1-L}
\]  

Equation 3.2 always has a solution if \( L \leq \frac{1}{2} \) and \( F \) is continuous and it is clear that \( \lim_{L \to 0} T_d(L) = F' \).

The choice \( T = T_d \) is, of course, inherently no less or more arbitrary than the choice \( T = T_e \). What is of interest is that, as we now show, there is a degree of consistency between the values \( T_e \) and \( T_d \).

We begin with a numerical example.

**Example**

Consider the case where \( F \) is an Erlangian distribution with 16 phases and mean \( \mu \). The values \( T_e \) and \( T_d \) obtained from equations (2.3) and (3.1) are shown as a function of \( L \) in Figure 3.

A reasonable agreement between \( T_e \) and \( T_d \) is observed. Note that for loss probabilities smaller than one percent, the value \( LT_e/(\mu(1-L)) \) is sufficiently small that it may be argued that \( T_e \) provides a sound design, in that little improvement of \( E(D) \) or \( E(N) \) is possible with a different decision.

For many delay distributions of interest (e.g., Erlangian, hyperexponential, exponential), it can be shown that

\[
\lim_{L \to 0} T_e(L) = 1
\]

Sufficient conditions for (3.3) to be true are that

a. \( F \) has density \( f \) satisfying \( \lim_{t \to \infty} \inf F(t)/F'(t) > 0 \).

b. \( F' \) is log-asymptotic to a log-concave complementary distribution.

Thus, as well as the reasonable agreement evidenced in the example, it is observed that there is an intrinsic consistency between the two approaches.

It is of interest to observe that if Equation 2.4 is replaced by

\[
\frac{1}{1 - \frac{L}{\mu}} = a(1-L) \sum_{i=1}^{n} F'(iT_e)
\]

where \( a \) is arbitrarily positive, then under the above conditions (3.3) remains true. This implies that the asymptotic (small \( L \)) behavior of \( T_e \) is independent of the relative costs attributed to the delay and efficiency objectives.

Similarly, if Equation 3.1 is replaced by

\[
q(T_d) = b[1 - q(T_d)]
\]

where \( b \) is positive and allows a solution to (3.4), then under the above conditions, (3.3) also remains true.

**ABNORMAL NETWORK CONDITIONS**

In the two previous sections we have considered that a network path is characterized by a fixed delay distribution...
F and a fixed loss probability L. While this assumption is reasonable under normal network operation, F and L may be considerably altered during periods of network congestion or partial failure. A poor choice of timeout intervals can even contribute to a change in F and L.

In the introduction we described an effect where too short a timeout interval T led to continual retransmission of a packet causing increased traffic load and in turn increased delay. Such an effect can lead to artificially-induced network saturation and it is necessary to incorporate some adaption into the timeout mechanism to prevent this hazard. The measurement of packet round-trip times by the sender for the purposes of detecting timeouts provides valuable information on the current network condition and T can be set using the method of the two previous sections, but taking into account recent observations of round-trip delay. However, this does not provide complete protection, since a sudden change in path or network condition could result in a sudden increase in mean round-trip time. A transmitter should interpret the occurrence of a timeout as meaning that a packet has been lost, or, quite possibly, that an increase in round-trip delay has occurred. Thus the transmitter's response to a timeout should include a lengthening of the timeout interval as well as retransmission. This latter mechanism provides an autonomous adaptation to a network condition that does not rely on the receipt of recent round-trip delay information. Automatic deflation of T occurs when the network problem subsides and round-trip times resume normal values. Similar comments to the above apply in the case where a change in network condition causes an increase in L (perhaps to unity) and the appropriate action is a lengthening of T.

Thus we can identify three classes of inputs to an adaptive timeout setting algorithm:

1. Setup parameters—Number and type of nodes in path; call priority.
2. Observational corrections—Observed round-trip delays; congestion indicators.
3. Autonomous corrections—Number of recent timeouts.

One issue which has not been discussed concerns the relationship between timeouts and fault detection strategies. If the occurrence of timeouts is used as an input to a fault detection/diagnosis mechanism, this will provide another reason for not choosing T too large.

SUMMARY AND CONCLUSION

In this paper we have tried to isolate some of the most important underlying issues which arise when choosing a timeout interval.

We have observed in the second section the presence of several conflicting goals and that the solution we have proposed should be viewed as a pleasing compromise. It was demonstrated in the next section that our solution yields a result similar to that of a quite different design technique and in fact, that for small loss probabilities the solution is insensitive to the parameter representing the compromise. In the fourth section we have considered how the basic design should be modified to take into account the possibility of abnormal network conditions and to ensure that timeouts themselves cannot act as a source of network problems.

It is likely that any real network and protocol will possess its own features and idiosyncrasies which require additional factors to be included in the timeout strategy. In the author's application, a number of additional objectives and protocol details have been taken into account. For example, an efficient protocol will acknowledge groups of packets rather than individual packets. As a general network strategy, when a timeout occurs it may be desirable for the transmitter to send an inquiry requesting status of the receiver rather than simply retransmitting the original packet. In this case the essence of the model remains valid since these inquiries need to be repeated periodically so that recovery from a short duration fault (e.g., trunk fade) is accomplished automatically. Another modification which can be accommodated in the above analysis arises from the consideration of block retransmit as opposed to the selective retransmit scheme we assume here.

ACKNOWLEDGMENT

The author is grateful to Jack Holtzman of Bell Laboratories for suggesting a study of this problem and providing many suggestions and comments.

REFERENCES
