Massachusetts Institute of Technology
Department of Electrical Engineering and Computer Science

6.899 Fall 2002 | **Problem Set 1** | September 10, 2002

This problem set has nine questions, each with several parts. Answer them as clearly and concisely as possible. You may discuss ideas with others in the class, but your solutions and presentation must be your own. Do not look at anyone else's solutions or copy them from anywhere. Turn in your solutions in on **Tuesday, September 24, 2002** in class.

1 Multiplexing

In this problem, we will compare statistical multiplexing to time-division multiplexing (TDM) to understand the differences between packet switching and circuit switching.

In our statistical multiplexing scheme, packets of all sessions are merged into a single queue and transmitted on a first-come first-served (FCFS) basis. Our TDM scheme is the same as the one described during the first lecture (see the L1 notes).

A switch is said to be *work conserving* if the only time it is idle is when there are no frames waiting for service.

1. Is our TDM scheme work conserving? What about our statistical multiplexing scheme?
2. Let's study the impact of statistical multiplexing on queuing delays. Suppose there are N concurrent sessions each with a Poisson traffic stream with rate λ frames/second. Also suppose that frame lengths are exponentially distributed, such that the average rate at which frames are serviced at the switch is μ frames per second ($\mu > N\lambda$). What is the average delay seen by a frame in TDM and in statistical multiplexing? What is the physical interpretation of your result?
3. Assume that all the sessions send frames at a simple constant bit rate and that there are A active sessions at a given point in time out of a possible N, sharing an output link. What is the utilization of the output link when the aggregate input rate for the A active sessions is μ frames/second. Sketch this as a function of A for both statistical multiplexing and TDM.
4. Explain why TDM has smaller variation in the delay of a frame through a switch, compared to statistical multiplexing. (This delay variation is sometimes called the *jitter*.)

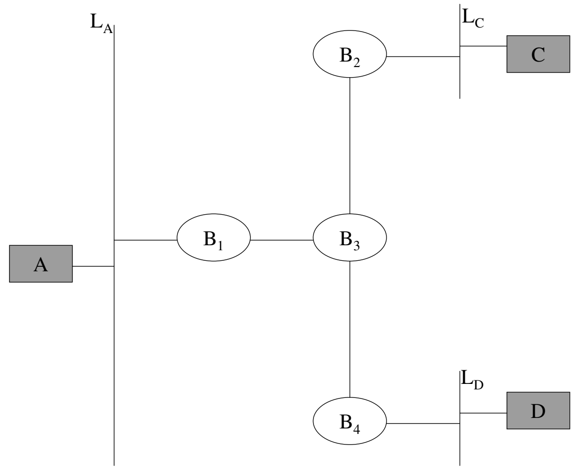


Figure 1: Bridge topology.

2 Learning (about) bridges

Consider the bridge topology shown in Figure 1. Assuming that all the forwarding tables are initially empty, write out the forwarding tables at each of the four bridges $B1$ through $B4$ at the conclusion of the following transmissions:

1. A sends to C.
2. C sends to A.
3. D sends to C.

In the forwarding table at each node, identify the port by the unique LAN segment ($L_A, L_C,$ *or* L_D) reachable using that port, unless there isn't one, in which case use the identifier of the neighboring bridge to identify the port.

3 Link packet traversals

Suppose source S sends a packet to destination D in a packet-switched network. Suppose the network topology and state in the switches do not change. Clearly explain why each of these statements below is true or false.

1. If datagram routing is used, correct forwarding can occur even if the packet traverses the same network link (and switch pair) in opposite directions.

2. If virtual circuit switching is used, correct forwarding can occur even if the packet traverses the same link (and switch pair) in opposite directions.

4 TCP retransmission timers

1. TCP computes an average round-trip time (RTT) for the connection using an exponential weighted moving average (EWMA) estimator:

   $$y(n) \leftarrow \alpha r(n) + (1-\alpha)y(n-1)$$

   where $r(n)$ is the n^{th} RTT sample and $y(n)$ is the average estimate updated after the arrival of the n^{th} sample. Suppose that at time 0, the initial estimate, $y(0)$ is equal to the true value, r_0. Suppose that immediately after this time, the RTT for the connection increases to a value R and remains at that value for the remainder of the connection. You may assume that $R >> r_0$.

   Suppose that the TCP retransmission timeout value at step n, $RTO(n)$, is set to $\beta y(n)$. Calculate the number of RTT samples before we can be sure that there will be no spurious retransmissions. Old TCP implementations used to have $\beta = 2$ and $\alpha = 1/8$. How many samples does this correspond to before spurious retransmissions are avoided, for this problem? (Today's TCPs use the mean linear deviation rather than $\beta y(n)$ as the RTO formula.)

2. Suppose that, instead of the EWMA estimator, TCP computed the average RTT by averaging over a fixed amount of past history. I.e.,

   $$y(n) \leftarrow \frac{\sum_{i=n-k}^{n-1} r(i)}{k}; k > 1.$$

   Now suppose that that the previous k samples are all equal to r_0, the true value, and that the RTT for the connection increases to a value $R(>> r_0)$ and remains at that value for the remainder of the connection. Using the same RTO as in part 1, calculate the number of RTT samples before we can be sure that there will be no spurious retransmissions.

3. In your opinion, which estimator is a better one for TCP? Why?

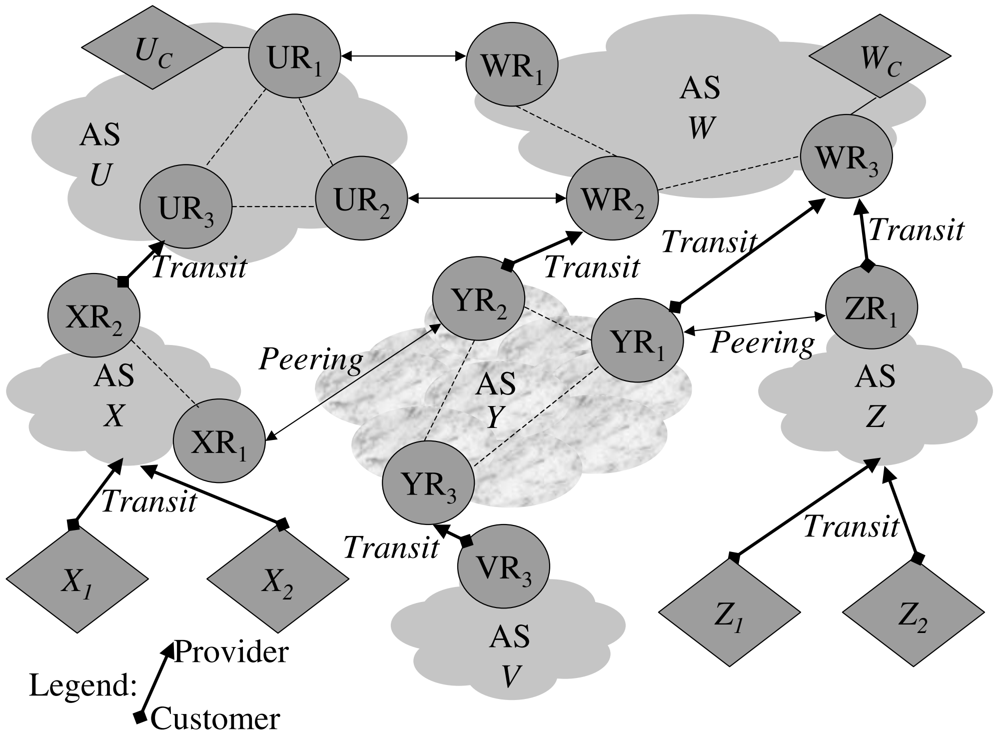
5 TCP checksums

TCP has an end-to-end checksum that covers part of the IP header, in addition to the TCP header and data. When the receiver receives a data segment whose checksum doesn't match, it can do one of two things:

1. Discard the segment and send an ACK to the data sender with the cumulative ACK field set to the next in-sequence byte it expects to receive, or

2. Discard the segment and do nothing else.

Is one action preferable to the other (or are they both equivalent)? Why? (You might look up the TCP and headers in any standard networking textbook; note that the header formats in Cerf & Kahn's paper aren't used any more.)



6 AS interconnections

There are six AS's shown in the picture above: U, W, X, Y, Z, V. The diamond-shaped boxes are customers of the AS's (ISP's) they're connected to. Some relationships are marked: the *Transit* ones involve the higher AS (in the picture) providing Internet service to the lower one(s) for money. The X-Y and Y-Z interconnections are standard peering relationships. The circles with names like WR_1 stand for BGP routers; WR_1 refers to a BGP router in AS W. Within each AS, the dotted lines show IBGP interconnections.

1. Which AS does not have correct IBGP interconnections?

2. Suppose an AS has k routers. What is the minimum number of IBGP sessions required for correct configuration, assuming no use of route reflectors or confederations?

3. Consider the peering relationship between Y and Z. Which of these statements are true?

   (a) Z will hear routes to V announced by Y, *and* may also hear routes to V announced by W.

   (b) Z *must* use a route to V announced to it by Y, since that's a route announced from a peering relationship.

   (c) Y will usually not announce routes to Z_1 and Z_2 to W.

   (d) Y will usually not announce routes to Z_1 and Z_2 to V.

4. U wants to ensure that packets sent to U_C from W are sent to it via UR_1 and not UR_2.

(a) Clearly explain how it might try to do this. (A small picture explaining routing messages will help.)

(b) Can it always ensure that the desired behavior happens? Why (not)?

5. W would like to ensure that packets sent to W_C from X reach it via AS's X and U, rather than via AS Y. How can this be done? (Describe the BGP routing messages involved.)

6. W would like to ensure that packets sent to W_C from X_1 reach it via AS's X and U, and packets sent to W_C from X_2 reach it via AS Y. Can this be done with BGP? If so, how? (Describe the BGP routing messages involved.)

7 Understanding BGP using table dumps

For this question, you will need to download the Routeviews routing table from `http://nms.lcs.mit.edu/6.829/ps/ps1/route-views.bgp.20020903.gz`

This file contains a Cisco BGP-4 routing table snapshot, taken at Oregon Route Views (`http://www.routeviews.org/`) on September 3, 2002. If you are curious about what other snapshots look like, you can find daily snapshots at `ftp://ftp.routeviews.org/pub/routeviews/bgpdata/`.

1. To start with, find the routing table entry for the MIT network.

   (a) What is the IP address of the best next hop from this router to MIT? How does this router know how to reach that next hop IP address?

   (b) How many AS's must a packet traverse between the time it leaves the router and the time that it arrives at MIT?

   (c) Use `traceroute` today to trace the route from MIT to the router that took the snapshot. Is the current route from MIT to the router the same as the reverse route in the trace data?

   (d) On September 3, 2002 at 4 pm EDT, the AS path to `route-views2.oregon-ix.net` from MIT was `10578 11537 4600 3701`. Why is this path not simply the reverse of the path from MIT to Routeviews? Why does this traceroute (which was run at the same time), not match the AS path?[1]

```
running /usr/local/bin/traceroute -A  198.32.162.102...
 1  anacreon (18.31.0.1) [AS3]  1 ms  1 ms  1 ms
 2  radole (18.24.10.3) [AS3]  6 ms  2 ms  1 ms
 3  B24-RTR-1-LCS-LINK.MIT.EDU (18.201.1.1) [AS3]  2 ms  2 ms  1 ms
 4  EXTERNAL-RTR-2-BACKBONE.MIT.EDU (18.168.0.27) [AS3]  185 ms  19 ms  2 ms
 5  192.5.89.89 (192.5.89.89) [AS1742]  1 ms  2 ms  3 ms
 6  ABILENE-GIGAPOPNE.NOX.ORG (192.5.89.102) [AS1742]  6 ms  7 ms  7 ms
 7  clev-nycm.abilene.ucaid.edu (198.32.8.29) [(null)]  20 ms  20 ms  24 ms
 8  ipls-clev.abilene.ucaid.edu (198.32.8.25) [(null)]  25 ms  25 ms  27 ms
 9  kscy-ipls.abilene.ucaid.edu (198.32.8.5) [(null)]  34 ms  36 ms  34 ms
10  dnvr-kscy.abilene.ucaid.edu (198.32.8.13) [(null)]  47 ms  45 ms  44 ms
11  pos-6-3.core0.eug.oregon-gigapop.net (198.32.163.13) [AS4600]  80 ms  78 ms  80 ms
12  nero.eug.oregon-gigapop.net (198.32.163.151) [AS4600]  77 ms  77 ms  78 ms
13  198.32.162.102 (198.32.162.102) [AS3582]  79 ms  79 ms  78 ms
```

[1] You can try this for yourself at `http://bgp.lcs.mit.edu/diag.html`.

(e) From the routing table file, what is the AS number for MIT?

(f) How many routes are there to get from this router to MIT?

(g) From the routing table, what is the best route to MIT? Why was this route selected as the best route?[2]

(h) What AS's do all of the different routes to MIT have in common? Which occurs most frequently? What is the likely relationship between the dominating AS and MIT?

(i) What IP network does the above AS correspond to? Again, all the information you need to answer this question is contained in the routing table. You can use `nslookup` to some host on this network to find out which company this is.

2. Several of the IP prefixes in the table are formatted as $w.x.y.z/m$. The mask field, m, specifies the length of the network mask to use when matching input destination addresses to entries in the table.

   (a) Write down the bit-wise operation to determine whether a destination address, A_i, matches a prefix A/m in the routing table. A_i and A are 32 bits each.

   (b) Find the first "Class C" CIDR address in the table (address prefix $\geq$ 192.0.0.0). How many class C networks does this address correspond to? What is the maximum number of routing table entries that this single CIDR address saves? Why is it that we can only infer the maximum, and not the actual, number of addresses that this CIDR address saves?

   (c) In the table, there are examples of groups of prefixes that have the same advertised AS path, but show up as separate entries in the routing table.[3]

      (i) Provide an example of non-contiguous prefixes (and the corresponding AS path) for which this is true. Why might non-contiguous prefixes have the same AS path?

      (ii) Provide an example of contiguous prefixes (and the corresponding AS path) for which this is true. This practice is often called *deaggregation*. Why might this be done?

3. Ben Bitdiddle is interested in studying the characteristics of the Internet using routing table snapshots. The Oregon Exchange has agreed to give Ben Bitdiddle some partial routing table snapshots from 1995 to the current day, including some snapshots from before the upgrade to BGP-4. They will give him snapshots containing the following:

   (a) Only the destination addresses.

   (b) Only the lines marked `*>`.

   (c) Only the paths, with best next-hops marked.

   Ben doubts that these partial snapshots could tell him anything interesting, but you disagree. What information about the evolution of the Internet could you infer from each type of partial snapshot?

[2] If you're interested, see the L4 notes or `http://www.cisco.com/warp/public/459/25.shtml` for an overview of the BGP decision process. Note that the process is slightly vendor-specific.

[3] For both parts of this problem, it's sufficient to find the existence of one AS path that is advertised more than once. It is *not* necessary to find two prefixes for which *all* advertised paths are the same.

8 Inferring AS Relationships

As you know, the Internet is composed of about 14,000 distinct origin AS's that exchange routes to establish global connectivity, and that business relationships determine which routes are exchanged between each pair of AS's.

Recall that one network will re-advertise its customer routes to its peers and providers, but will not re-advertise routes heard from a peer to other peers or providers. With the knowledge of these rules and a view of a default-free routing table (or multiple tables), one can deduce relationships between AS pairs based on links that exist in the AS graph.

In *On Inferring Autonomous System Relationships in the Internet*[4], Lixin Gao observes that, because of these constraints, AS paths must adhere to one of the following patterns:

1. a series of customer-provider links (an *uphill path*)
2. a series of provider customer links (a *downhill path*)
3. an uphill path followed by a downhill path
4. an uphill path followed by a peering link
5. an peering edge followed by a downhill path
6. an uphill path followed by a peering link, followed by a downhill path

This is called the "valley free" property of AS paths. The hard question, of course, is: where is the "top of the hill"? Gao suggests using the AS in the path that contains the larges degree: that is, the AS that connects to the most other AS's.

We have provided a Routeviews routing table for you at `http://nms.lcs.mit.edu/6.829/ps1/route-views.bgp.20020903.gz`. (Note that the file is 8MB.) Your task is to produce a good guess about relationship between each AS pair in the table.

1. Produce CDF of AS degree (i.e., plot the fraction of AS's that have an degree of $< n$, for all $n > 0$). Also include a table of the "top 10" AS's for degree and the value of their degrees. Do not count a link from an AS to itself as an edge. Also, consider *all* AS paths that are given in the table (about 2.4 million paths), not just the best path for each prefix.

2. For each of the following AS paths, list the transit relationships inferred for each pair, based on that path. *This is a two-step process.*

   First, for each AS path, note the transit relationships. For example, for the path $ABCD$, if C were the AS with the highest degree, you would write "Transit relationships: $A \to B$, $B \to C$, $D \to C$". This will give you a list of AS pairs that have transit relationships.

   Once you have scanned all AS paths, you may find that you have a commutative transit relationship: i.e., A transits B *and* B transits A. This is called a sibling relationship. For all pairs in the following paths, note which AS transits for the other, or if the two pairs have a sibling relationship.

[4] You can find a copy of this paper at `http://www-unix.ecs.umass.edu/~lgao/ton.ps`. While you don't need to read the paper to solve this problem, you may find it helpful and interesting.

(a) 3130 2914 701

(b) 8121 19151 3356 18566

(c) 16150 8434 3549

(d) 6539 701 7018

(e) 7911 209 19092 3908 10947

3. Finding the "top of the hill" by using the AS with the highest degree sometimes produces the wrong answer. Another way to do this is to view the AS paths from one vantage point as a directed graph, and using a reverse pruning algorithm to the AS graph in order to assign ranks to each AS.

   First, leaf nodes of the AS graph are assigned the lowest rank. Then, these nodes and their incident edges are removed from the graph. The nodes that are leaves in this new graph are assigned the next highest rank. The process repeats until the graph is strongly-connected (i.e., there are no leaves); each node in the strongly-connected component of the graph receives the highest rank. AS relationships are inferred by comparing rankings from AS graphs as visible from *multiple vantage points*[5].

   (a) What are advantages of using this type of ranking scheme over a power-law based scheme? What are the disadvantages?

   (b) Why does this scheme require multiple vantage points to be effective?

9 Traffic Flow Patterns

People often want to know how much traffic they are sending to each neighboring AS. Network operators use traffic volumes to detect congestion, determine if they are violating peering agreements, or sending too much traffic on an expensive transit link. Typically, network operators use Netflow[6] to calculate these volumes. In the absence of Netflow, packet monitoring and routing information can provide a crude approximation of traffic patterns.

In this problem, you will answer the question: "How much traffic from the MIT Lab for Computer Science flows toward Internet2?". This requires two pieces of information: how much traffic is destined for each IP address, and which routes correspond to which prefixes. Note that the latter requires doing a longest prefix match for each IP address.

To answer this question, you will need the routing table as seen from MIT to determine which prefixes have routes via Internet2 and which have routes via Genuity. The routing table was collected at LCS on September 9, 2002 via an IBGP session with MIT's border router; the machine has no other BGP sessions. This routing table is available at: `http://nms.lcs.mit.edu/6.829/ps/ps1/mit.bgp.20020909.gz`.

Additionally, you will need to know how many bytes were destined for each IP address. These byte counts, produced from a trace on December 6, 2000 are available at: `http://nms.lcs.mit.edu/6.829/ps/ps1/20001206.byte.summary.gz`.

[5] You can find the paper that describes this algorithm in detail at `http://www.ieee-infocom.org/2002/papers/594.pdf`.

[6] `http://www.cisco.com/warp/public/732/netflow/`

1. Why is there only one route per prefix in this table?

2. Produce the longest-prefix match for each of the following IP addresses from the routing table we provided. (*Hint:* You don't have to implement the most efficient longest-prefix match; a simple sorting-based scheme should be sufficient for this problem.[7]

   (a) 150.65.236.70
   (b) 24.218.254.226
   (c) 20.138.0.10
   (d) 47.249.128.12

3. How much traffic leaves MIT from LCS via Internet2? Via Genuity?

4. List at least two potential sources of inaccuracy that may result from using this method to measure traffic volumes.

5. Now suppose MIT wanted to send less traffic outbound via Internet2 (typically this would be a bad idea since the Genuity link is more expensive, etc., but assume for the sake of the problem that this is reasonable). What would be a good way for MIT's network operator to adjust the outbound traffic volumes? What if MIT wanted to adjust inbound traffic volumes?

[7]However, you are welcome to use existing longest-prefix match implementations. For example, Perl has a convenient Patricia module for IPv4 route lookups that you may consider using if you do this problem by writing a Perl program.