Project: Internet Connectivity

João Luís Sobrinho

September 28, 2020

1 Internet as interconnection of autonomous systems

The Internet is an interconnected collection of about $60,000$ networks, called Autonomous Systems (ASes). Two ASes are neighbors if they have a bidirectional link between them (one link in each direction). Every pair of neighbor ASes is associated with a commercial relationship. Ultimately, the commercial relationships determine how traffic flows across the Internet. With some oversimplification, the commercial relationship between two neighbor ASes belongs to one of two types. In a *provider-customer* relationship, the customer pays the provider to transit its traffic with the rest of the Internet, whereas in a *peer-peer* relationship, the two peers exchange traffic between themselves and their customers without monetary compensations. The customer-provider relationships create an hierarchy in the Internet. At the bottom of the hierarchy are ASes without customers, called stubs. All other ASes transit traffic; they are called Internet Service Providers (ISP). At the top of the hierarchy are ASes without providers, called Tier-1s. The majority of ASes are stubs, which include content provider and content distribution networks, such as those of Google, Facebook, Amazon, Microsoft, Apple, and Akamai, access networks, and many enterprise networks. There are only a dozen or so Tier-1s, including AT&T, CenturyLink, Verizon Enterprise Solutions, Deutsche Telekom Global Carrier, Telxius, and Tata Communications. The Center for Applied Internet Data Analysis (CAIDA) provides inferred topologies of the Internet annotated with the type of commercial relationships between neighbor ASes.

2 Connectedness and hierarchy

An internet (small 'i') is a network with bidirectional links where each pair of neighbor nodes has either a provider-customer or a peer-peer relationship. An internet is *connected* if there is a path from every one of its nodes to every other. A *bridge* in a internet is a bidirectional link whose failure splits a connected network into two connected components. A network is *link-biconnected*

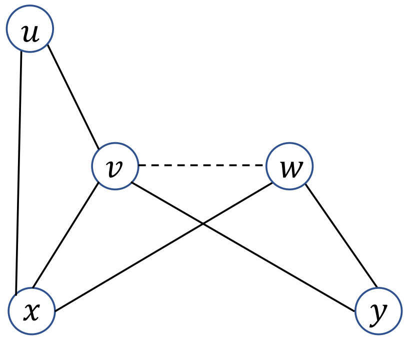

Figure 1: An internet. Solid lines join providers to customers, with providers placed higher than customers. Dashed lines join peers.

if it has no bridges. The concepts of connectedness and link-biconnectedness are independent of the commercial relationships.

A *provider-customer cycle* is a cycle such that every node is a customer of the next around the cycle; equivalently, every node is a provider of the next around cycle. An internet is *commercially acyclic* if it does not have provider-customer cycles. In a commercially acyclic internet, the hierarchy among nodes can be expressed by the concept of *tier*. The tier of a node without providers is one; the tier of a node with providers is one plus the tier of its provider of highest tier.

The commercial relationships constrain the routing paths in an internet so that every intermediate node along a path is monetarily compensated for transiting traffic. A *commercial path* in an internet is a path of the form PRC, where: (1) P is either the empty path or a path comprised solely of customer-to-provider links; (2) R is either the empty path or a single peer-to-peer link; (3) C is either the empty path or a path comprised solely of provider-to-customer links. An internet is *commercially connected* if there is a commercial path from every node to every other.

Figure 1 shows a small internet, where a provider is joined to a customer by a solid line, with the provider placed higher than the customer, and two peers are joined by a dashed line. For example, u is a provider of v and v and w are peers. The internet is connected and even biconnected. It is commercially acyclic with u and w of tier one, v of tier two, and x and y of tier three. Paths uvy, vwy, and xvy are commercial paths, but the internet is not commercially connected since there are no commercial paths between u and w.

3 Your assignment

What you have to do. You will be given an internet in the format illustrated in the table below. Each line represents a link. The first value is the identifier of the node at the tail of the link. The

4323	12122	1
12122	4323	3
7018	17228	1
17228	7018	3
29017	34309	2
34309	29017	2

second value is the identifier of the node at the head of the link. The third value is 1 if the tail of the link is a provider of the head of the link; it is 2 if the tail of the link is a peer of the head of the link; and it is 3 is the tail of the link is customer of the head of the link. There is a link from node i to node j with value 1 if, and only if, there is a link from node j to node i with value 3; there is link from node i to node j with value 2 if, and only if, there is a link from node j to node i also with value 2. For example, in the table above, the first two links imply that 4323 is a provider of 12122. The number of nodes in an internet is limited to $65,535$.

You are asked to do the following:

- Design and implement an algorithm to determine whether or not an input internet is connected.
- Design and implement an algorithm to determine whether or not an input internet is link-biconnected; if it is not, the algorithm should output a bridge.
- Design and implement an algorithm to determine whether or not an input internet is commercially acyclic; if it is not, the algorithm should output a provider-customer cycle.
- Design and implement an algorithm to determine whether or not an input internet is commercially connected.
- Depending on the evolution of the project, I may ask the design and implementation of one additional algorithm.

What you have to deliver, how, and when.

- You have to deliver your code and a report with a cover page and no more than three other pages containing: (1) a clear explanation of your algorithms; (2) a proof of their correctness; (3) their worst-case complexity; (4) a relevant discussion. (You do not need to explain elementary algorithms such as BFS and DFS.)
- The code and the report should be sent in a .zip file to my email address with subject p1.<group number>.zip where <group number> is your group number.

- The deadline is October 30, 2020, 23:59.

How I will evaluate your assignment.

- I will start by reading your report, which is the means for you to communicate your ideas, algorithms, and conclusions with others. Organize your report in sections; present high-level, but precise descriptions of your algorithms, highlighting their most subtle steps, if any; draw concise, but unambiguous conclusions.
- If the report is readable, I will run some tests on your programs.
- I will have a discussion with you about your report and will test your code at the end of the semestre, jointly with the other assignment. All students of the group must actively participate in the design and implementation of the algorithms.