CMSC 858F: Algorithmic Game Theory
Fall 2010
BGP and Interdomain Routing

Instructor: Mohammad T. Hajiaghayi
Scribe: Yuk Hei Chan

November 3, 2010

1 Overview

In this lecture, we cover *BGP* (*Border Gateway Protocol*) and *interdomain routing*, and discuss their relation to game theory.

2 Introduction

The Internet can be viewed as a collection of "clouds", where each cloud is an *autonomous system* (AS), a network (or a set of networks) that is under a single administration. For example, the whole UMD network can be an AS, while for larger corporations like AT&T, the whole corporate network can be divided into several ASes, say AT&T California, AT&T Texas, and so on. Each AS has an officially registered *AS number* (ASN). Currently (as of 2010) there are over 35,000 registered ASNs in use.

IGP (Interior Gateway Protocol) is used inside one AS. It is used to figure out how do to route within an AS. *IGRP* (Interior Gateway Routing Protocol) and *OSPF* (Open Shortest Path First) are used as an IGP. They are distance-based policies.

IGRP, together with its enhanced version, *EIGRP* (Enhanced IGRP), are Bellman-Ford based protocols. They are *distance-vector* routing protocols. In each router, there is a routing table which is a vector of size n (n is the number of routers in the AS). This vector indicates the distance to other routers in the AS, as well as the next hop neighbor to send the packet to. The router does not possess the full information about the network topology. The distance-vector is advertised to its neighbors to perform updates to the routing table. IGRP and EIGRP are supported by Cisco routers.

OSPF, on the other hand, is a *link-state* routing protocol. It saves the current state (adjacency matrix) of the whole "world" (which is the whole AS).

It is a Dijkstra based protocol and is more dominant. The adjacency matrix is updated with the distance / capacity of the router's links. It is more global compared to EIGRP. If there are several shortest paths, it divides the payload into parts and utilize all shortest paths so it is faster. OSPF is also supported in Cisco routers.

In updating the distance metrics, we assume *metricity* (or *triangle inequality*), i.e. $d(a, c) \leq d(a, b) + d(b, c)$ for all router a, b and c in the AS.

BGP (Border Gateway Protocol) is the sole protocol used for routing between domains. It replaced *EGP* (Exterior Gateway Protocol) since 1994 and BGP version 4 is now the accepted standard. It allows each AS to define its own preference for routing policy. The policy does not need to adhere to a distance-based policy like the ones in *Interior Gateway Protocol* (IGP) metrics and *Exterior Gateway Protocol* (EGP). In BGP, each AS becomes a node. By making sure the AS number does not appear in a path more than once, loops are prevented.

The analysis of the robustness of policy choices in BGP has implication towards the overall efficiency and the functionality of the Internet. Also, stability correspond to Nash equilibria and that's why we study BGP.

3 Theoretical Model

We model the whole system as an undirected graph on ASes, where each AS is a node and we want to find some route to the destination d. The goal is to design mechanism so as to have good behavior. Consider the BGP routing mechanism where d is the destination:

1. d advertise itself
2. For all router $v \neq d$:
   - Iteratively receives updates about path to d
   - Receives status updates
   - Choose the best path and update the forwarding table (according to some policy)
   - Announce the best path to its neighbors similar to IGRP

As a simple example, node i ($i = 1, 2, 3$) prefers the route $i(i+1)d$ to id to the destination d:

Let's say at the beginning, 3 picks the route $3d$ (as there are currently no paths chosen). Then, 2, on seeing the choice of 3, picks the route $23d$ (since $23d$ is preferred by 2 to the route $2d$). 1 picks the route $1d$ (Figure 3, left). Then 3, on seeing the choice of 1, changes its mind and picks the route $31d$. Next, seeing that the route $23d$ is gone, 2 has to pick the route $2d$ (Figure 3, right). In turn, 1 sees the opportunity to change is route to $12d$, which forces 3 to change its choice. This switching among route on each router's preference list goes on forever which means the system is not stable.

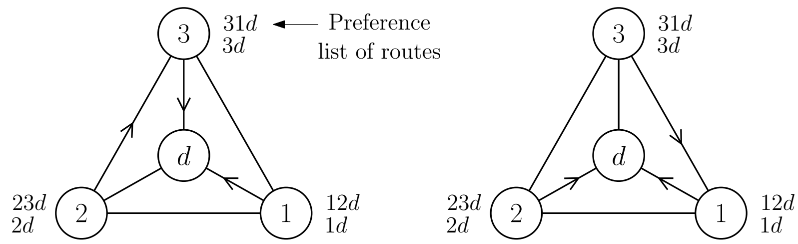

Figure 1: Not all choices of preference list lead to stable paths.

4 The Stable Path Problem (SPP)

The *stable path problem* (SPP) is as follows:
Input: an undirected graph $G = (V, E)$.

- For each $v \in V$, P^v is the set of permitted (simple) paths from v to the destination vertex d.
- For each $v \in V$, there is a ranking function λ^v defined over P^v. If $\lambda^v(P_1) < \lambda^v(P_2)$ then P_2 is a more preferred permitted path than P_1.
- Empty path $\epsilon \in P^v$ is permitted and ranked lowest: $\lambda^v(\epsilon) = 0$, while $\lambda^v(P) > 0$ for $P \neq \epsilon$.
- $P_1, P_2 \in P^v \implies \lambda^v(P_1) \neq \lambda^v(P_2)$. (each path receives a distinct ranking)
- If the end point of a path P is the same as the start point of a path Q, the *concatenation* of the two paths is denoted by PQ. If $P = (1, 2)$ and $Q = (2, 3)$ then $PQ = (1, 2, 3)$. Note that $\epsilon P = P\epsilon = P$.

A *path assignment* is a function Π that maps each node $u \in V$ to a path $\Pi(u) \in P^u$ (with the special case $\Pi(d) = (d)$). Initially $\Pi(u) = \epsilon$ which means u is not assigned a path to the destination. The set of paths $\mathsf{choices}(\Pi, u)$ for $u = d$ is $\{(d)\}$ and $\{(u, v)\Pi(v) \mid \{u, v\} \in E\} \cap P^u$ otherwise. This represents all possible permitted paths at u that can be formed by extending the paths assigned to neighbors of u. Given a node u, suppose W is a subset of the permitted paths P^u such that each path in W has a distinct next hop. Then the best path in W is defined to be $\mathsf{best}(W, u) = P \in W$ with maximal $\lambda^u(P)$ for $W \neq \emptyset$ and ϵ otherwise. The path assignment Π is *stable at node* u if $\Pi(u) = \mathsf{best}(\mathsf{choices}(\Pi, u), u)$. Note that if Π is *stable at node* u and $\Pi(u) = \epsilon$, then the set of choices at u must be empty. The path assignment Π is *stable* if it is stable at each node u. Essentially, $\Pi = (P_1, \ldots, P_n)$ where $\Pi(u) = P_u$ (assume 0 is the destination). If Π is stable and $\Pi(u) = (u, w)P$, then $\Pi(w) = P$. Hence, any stable assignment defines a tree rooted at the destination (each node has one outgoing edge that points towards the destination), although it is not always the case that it is a shortest path tree.

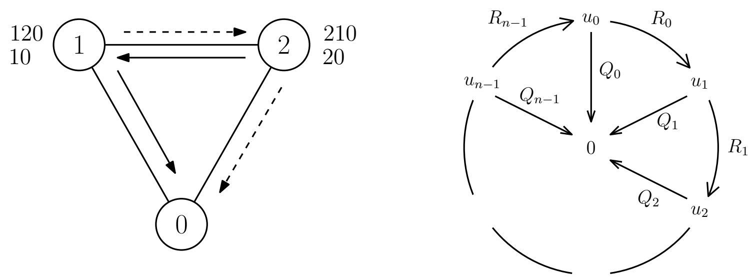

Figure 2: (Left) A disagree gadget; (Right) A dispute wheel.

The stable paths problem $\mathsf{S} = (\mathsf{G}, \mathcal{P} = \cup_v \mathsf{P}_v, \lambda)$ is solvable if there is a stable path assignment. The example in Figure 3 does not have a stable path assignment.

Theorem 1 *(Griffin, Shepherd, Wilfong, TON 2002) SPP is NP-Complete (proved by reducing it to 3-SAT).*

When does SPP admit a unique solution? Let's look at a configuration with more than one solution. A *disagree gadget* is shown in Figure 2, left. In this configuration, there are 2 stable paths, namely $(2, 1, 0)$ and $(1, 2, 0)$. The disagree gadget can be generalized into a *dispute wheel*, as shown in Figure 2, right:

1. A *rim path* R_i is a path from u_i to u_{i+1}
2. *Spoke path* $Q_i \in P^{u_i}$
3. $R_i Q_{i+1} \in P^{u_i}$
4. $\lambda^{u_i}(Q_i) < \lambda^{u_i}(R_i Q_{i+1})$

Theorem 2 *If the stable path problem* S *has no dispute wheel configuration, then there is a unique solution.*

Note that if S has no dispute wheel, then it is solvable. Roughly speaking, having no dispute wheel implies safety and robustness.

There are two types of relationship between entities on the Internet, namely provider-customer and peer-to-peer (Figure 3). *Gao-Rexford condition* is a reasonable assumption in the context of connectivity of the Internet, which states if there is no customer-provider (directed) cycle, then there is no dispute wheel configuration, which implies a stable assignment. We can achieve this by filtering of paths, e.g. not advertising your peers to your provider.

We can define a game such that Nash equilibria of the game are precisely the stable solutions in the equivalent SPP formulation.

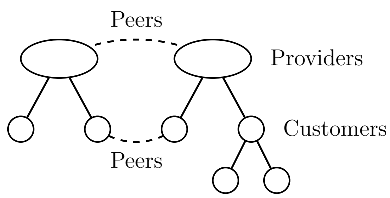

Figure 3: Provider-customer relationship (solid line) and peer-to-peer relationship (dashed line).

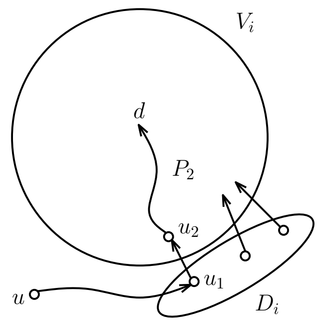

Figure 4: A consistent path.

5 Greedy Algorithm in the Absence of Dispute Wheel

In the absence of dispute wheel, a greedy algorithm that is based on the idea of "expanding the tree" can be used to find the solution to the SPP. Starting with V_0, which contains only the destination 0, we construct larger and larger set V_i, such that $\{0\} = V_0 \subset V_1 \subset \cdots \subset V_k$, and for $v \in V_i$, $\Pi(v) \in V_i$ (i.e. the whole path stays inside V_i).

If $u \in V - V_i$ and $P \in P^u$, then P is said to be consistent with current (partial) Π_i if $P = P_1(u_1, u_2)P_2$, where P_1 is a path in the digraph induced by $V - V_i$, $u_2 \in V_i$, $u_1 \in V - V_i$, $\{u_1, u_2\} \in E$ and $P_2 \in \Pi(u_2)$ (Figure 4). Call a path P a *direct path to* V_i if $P_1 = \emptyset$ (in this case $u = u_1$). Let D_i be the set of nodes with direct path to V_i. Assuming every node has a non-empty permitted path to the origin, D_i is non-empty. Let $H_i \subseteq D_i$ be the set of vertices u such that a direct path of u is preferred among all consistent paths of u. For the greedy algorithm to work, we want H_i to be non-empty at each iteration.

What if $H_i = \emptyset$ but $D_i \neq \emptyset$? Since $u_0 \in D_i - H_i$, there is a vertex $u_1 \in V - V_i$ and path $R_0 \in V - V_i$ from u_0 to u_1, such that $\{u_1, v_1\} \in E$, $v_1 \in V_i$ and $P_i(v_1) \in \Pi(v_1)$. Let the path from u_0 to a vertex in V_i and on to the destination be Q_0. Since Q_0 is not the preferred path, we have $\lambda(R_0Q_1) > \lambda(Q_0)$, where R_0 is a path in $V - V_i$ that leads from u_0 to u_1, and Q_1 is a path from u_1 to v_1 and on to the destination (Figure 5). Then consider u_1. Again since

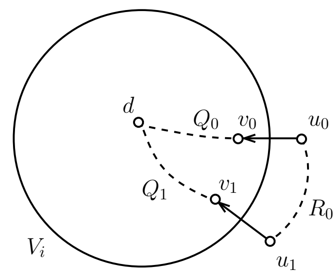

Figure 5: The first step to a dispute wheel.

$u_1 \in D_i - H_i$, there exists $u_2 \in V - V_i$, $R_1 \in V - V_i$ from u_1 to u_2, Q_2, $v_2 \in V_i$ with $\{u_2, v_2\} \in E$ and $\lambda(R_1 Q_2) > \lambda(Q_1)$. The argument continues and eventually forms a dispute wheel (although u_0 may not involved in the dispute wheel). This shows that $H_i \neq \emptyset$ and proves that the greedy algorithm works in the absence of dispute wheel.

References

[1] Timothy G. Griffin, F. Bruce Shepherd, Gordon Wilfong. *The stable paths problem and interdomain routing.* IEEE/ACM Transactions on Networking (TON), Vol. 10(2), 2002.

[2] L. Gao, J. Rexford. *Stable Internet Routing Without Global Coordination.* ACM SIGMETRICS, June 2000