'31

BGP (Border Gateway Protocol) and Interdomain Routing:

BGP is the sole protocol used for routing between domains. BGP allows each autonomous system (AS) (a collection of connected Internet protocol (IP) routing prefixes under the control of one or more network operators), to define its own preference for routing policy rather than strictly adhering to a distance-based policy such as traditional Interior Gateway Protocol (IGP) metrics and Exterior Gateway Protocol (EGP).
IGP is used within an AS while EGP is for determining network reachability between AS and makes use of IGPs to resolve routes with an AS (because of this within one AS we have metrics

1 — IGP is ~~mainly~~ based on Distance-vector routing protocol in which each router does not possess information about the full network topology. It advertises its distance to other routers and receive similar advertisment from other routers. Using these routing advertisments each router populates its routing table and in the next advertisement cycle, a router advertises updated information from its routing table. This process continues until the routing tables of each router converge to stable values which might be a slow convergence. A famous protocol of this type is Interiour Gateway Routing Protocol (IGRP) which is doing the process above and it is used by Cisco routers. Also there is an Enhanced version of that (EIGRP) Similar to Bellman-Ford

2 — IGP is also based on Link-state routing protocol in which each node possesses information about the complete network topology (i.e. connectivity of the graph) and this the only step in which each node share its routing table with its neighbor. Then each node independently calculates the best next hop from it for every possible destination in the network using local information of the topology and the collection of best next hops froms the routing table for the node. Network topology and connectivity of the graph is called the adjacency data base.

☆ A famous protocol of this type is Open Shortest Path First (OSPF). The weight of each edge can be changed by the network operator and can be set proportional to their physical distances or inversely proportional to its capacity (assuming we know them) OSPF is also very common and is supported by cisco routers. OSPF is more popular than EIGRP
In each router the demand going in the router is sent to its destination by splitting the flow between the links that are on the shortest paths to the destination. The exact mechanics of the splitting can be somewhat complicated, depending on the implementation, but as a simplifying approximation, we can assume that it is even split. OSPF is using Dijkstra's algorithm for finding shortest paths

OSPF gets the whole adjacency matrix from the neighbors while IGRP router gets only the distance to other routers from its neighbor (due to this you can find link failures faster in OSPF)

As aforementioned, BGP allows each AS to define its own preference for routing policy rather than strictly adhering to a distance-based policy like OSPF or IGRP. BGP is esp. more useful when you have more options like multiple connections to the destinations rather than just one connection. BGP has a rich set of features to allow influence traffic paths. Though potentially BGP can be used instead of traditional Gateway protocols (IGP), but it was created to replace EGP (exterior gateway protocol) which is now obsolete and it was a simple reachability protocol to interconnect autonomous systems. Currently BGP v4 is the accepted standard for Internet routing and among autonomous systems (since 1994). Each AS has an officially registered Autonomous System Number (ASN) which is used in BGP routing. The number of registered ASN is above 35000 by now (2010). An autonomous system is one network or sets of networks under a single administrative control. An AS might be the set of all computer networks owned by a company, or a college. Companies and organizations might own more than one AS but the idea is that each AS is managed independently with respect to BGP (e.g. a company may have one AS for its European branch and one for its U.S. branch). BGP makes certain that an AS number does not appear in a path more that once and thus preventing routing loops. Analysis of the robustness of policy choices in BGP has implications towards the overall efficiency and the functionality at the Internet. and because of that we consider BGP.

The model

We have an undirected graph $G(V,E)$ where the nodes are ASs and in which we want to find some route to destination d. The originator of a packet sent on the network announces a destination and each node in the network has preferences for routing policies through which to send the packet en route to d. The main question we seek to answer is when can we expect good behavior in the system to arise and in regards to game theory, how can we define a mechanism that maintains good behavior?

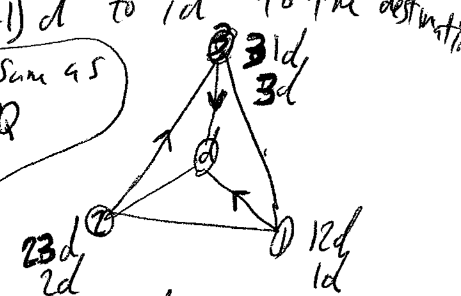
The process by which routing policies are set in BGP can be considered as follows

1) d advertises itself.

2) $\forall v \in V$, $v \neq d$
- iteratively receive updates about path to d.
- receive status updates
- choose best path and update the forwarding table (according to some policy)
- announce best path to neighbors similar to IGRP.

A simple example is when node i prefers to route $i(i+1)d$ to id to the destination

If P and Q are non-empty paths such that the first node in Q is the same as the last node in P, then PQ is the concatenation also $\varepsilon P = P\varepsilon = P$



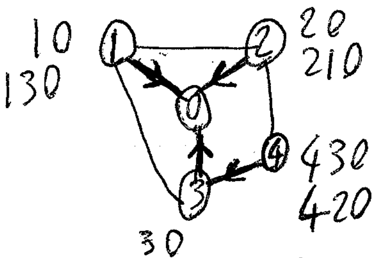
The stable path problem (SPP)

undirected graph $G=(V,E)$
- for each $v \in V$, P^v is the set of permitted (simple) paths from v to destination vertex d.
- For each $v \in V$, there is a ranking function 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, $\varepsilon \in P^v$ is permitted and is ranked lowest $\lambda^v(\varepsilon)=0$, $\lambda(P) > 0$ for $P \neq \varepsilon$
- $P_1, P_2 \in P^v \Rightarrow \lambda^v(P_1) \neq \lambda^v(P_2)$.

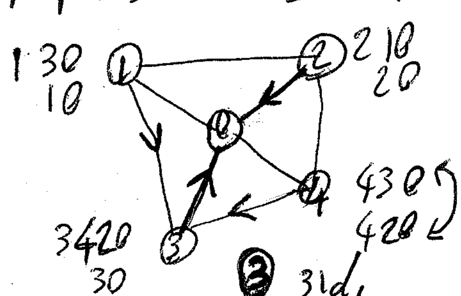
A path assignment is a function π that maps each node $u \in V$ to a path $\pi(u) \in P^u$ (note that this means $\pi(d) = (d)$). We interpret $\pi(u) = \varepsilon$ to mean that u is not assigned a path to the origin. The set of paths choices(π, u) for $u=d$ is $\{(d)\}$ and otherwise is $\{(u,v)\pi(v) \mid \{u,v\} \in E\} \cap P^u$. This set represents all possible permitted paths at u that can be formed by extending the paths assigned to the peers of u. Given a node u, suppose that 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 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) = $ best(choices$(\pi, u), u)$. Note that if π is stable at node u and $\pi(u) = \varepsilon$, then the set of choices at u must be empty.

The path assignment R is <u>stable</u> if it is stable at each node u. We often write a path assignment as a vector $(P_1, P_2, \ldots, P_n)$ where $R(u) = P_u$ and we omit P_d since it is always (d). It is easy to check that by definition if R is stable and $R(u) = (u,w)P$, then $R(w) = P$. Therefore any stable path assignment implicitly defines a tree rooted at the origin however it is not always a shortest path tree (as we see)

The stable paths problem $S = (G, P = \bigcup_v P_v, \lambda)$ is <u>solvable</u> if there is a stable path assignment for S. A stable path assignment is also called a <u>solution</u> for S and if there is no such assignment, S is <u>unsolvable</u>.



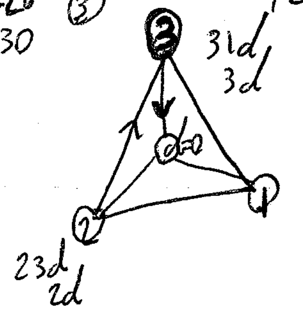
Note that ranking of paths is not required to prefer shorter paths to longer paths. For example



$\boxed{0 = d}$

if we reverse the order for 4 then 4 has edge to 2.

if we exchange no solution

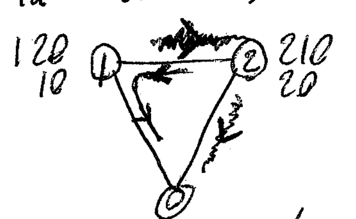


There is no solution here as well.

Note that the problem is in NP since we can check the consistencies.

However
Then SPP is NP-complete by reducing it to 3-SAT [Griffin, Shepherd and Wilfong, ToN 2002]

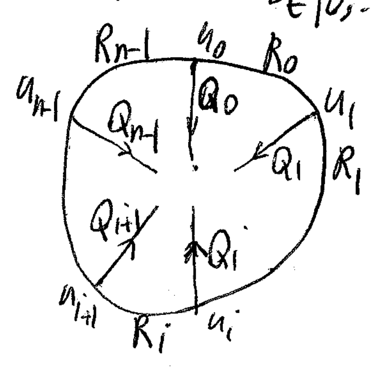
If we chose any edge id we get a contradiction



Disagree gadget and its two solutions

<u>Dispute wheel</u> is an instance of SPP in that a collection on nodes $0, \ldots, n-1$ of spoke paths $Q_{i \in \{0,\ldots,n-1\}}$ and of rim paths $R_{i \in \{0,\ldots,n-1\}}$ such that

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



In general, an instance of stable paths problems may have more than one solutions (see DISAGREE gadget)

Thm: If the stable path problem S has no dispute wheel, then it has a unique solution

(Also note that thm says if S has no dispute wheel, then it is solvable)

Also having no dispute wheels implies safety and robustness with some definitions of safety and Robustness.

let's prove the following weaker Thm:

Thm: let S be an instance of the stable paths problem. If S has no dispute wheel, then S is solvable.

Pf: Consider the following Greedy heuritic which attempts to grow a stable path assignment (a routing tree) in a Greedy manner. Suppose $V' \subseteq V$, such that $0 \in V'$. A partial path assignment π for V' is a path assignment such that for every $u \in V'$, every node in $\pi(u)$ is in V'. The heuristic procedure constructs a sequence of subsets V_i $\{0\} = V_0 \subset V_1 \subset V_2 \subset \cdots$ together with a sequence of partial path assignment for V_i. For each π_i, define π_i' to be the path assignment for V where $\pi_i'(u) = \pi(u)$ for $u \in V_i$, and $\pi_i'(u) = \varepsilon$ for $u \notin V_i$. The partial path assignment π_i is stable for V_i if π_i' is stable for each $u \in V_i$.

If $u \in V - V_i$ and $P \in P^u$, then P is said to be consistent with π_i if it can be written as $P = P_1 (u_1, u_2) P_2$, where P_1 is a path in the digraph induced by $V - V_i$, $u_2 \in V_i$, and $P_2 = \pi(u_2)$, and $\{u_1, u_2\} \in E$. such a P is called a direct path to V_i if P_1 is empty. let D_i be the set of nodes $u \in V - V_i$ that have a direct path to V_i, i.e. $P = (u, u_2) P_2$. without loss of generality, each node has a non-empty permitted path to the origin, and hence if $V - V_i$ is not empty, then D_i is not empty

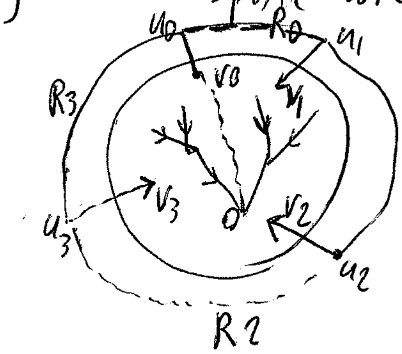
let H_i be the set of nodes $u \in D_i$ whose highest ranked path consistent with π_i is a direct path. Denote this path as B_i^u. If H_i is not empty, let $V_{i+1} = V_i \cup H_i$

Define the partial path assignment π_{i+1} on V_{i+1} as $\pi_{i+1}(u) = \begin{cases} B_i^u & u \in H_i \\ \pi_i(u) & u \in V_i \end{cases}$

This process continues until for some k either 1) $V_k = V$ or 2) $V_k \neq V$ and $H_k = \emptyset$.

In the first case, π_k is clearly a stable path assignment. In the second case we are stuck and the procedure fails to find a solution.

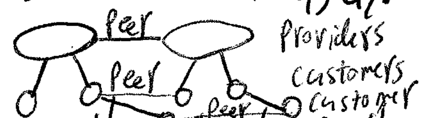
we prove that in this case we have a dispute wheel. let u_0 be any node in D_i and let $Q_0 \in P^{u_0}$ be a direct path. Note that there must be a path P_0, permitted at u_0 and consistent with V_i, which has higher rank than Q_0. Since P_0 is consistent with V_i it has the form $P_0 = R_0(u_1 v_1)Q$ where R_0 is a path from u_0 to u_1 in $V - V_i$, $v_1 \in V_i$, Q is $R_i(v)$ and $\{u_1, u_2\} \in E$. Note that $u_1 \in D_i$, and ~~$v_1 \in$~~ since H_i is empty we can repeat this process with u_1. If we continue in this manner, it is clear that we will eventually form a dispute wheel. ∎



Note that the set of permitted path should be suffixes of each other, otherwise there is no use of the path: this fact has been used in the proof above

~~Note that there is no disjointness in definitions of R_i, Q_i.~~

Gao_Rexford Conditions



are some reasonable conditions in the context of the connectivity of the internet which can guarantee no dispute wheel and thus a stable assignment. E.g.

2 type of edges:
- customer → provider (cost is positive) cost is not really part of the condition
- Peer ↔ Peer (cost is zero)

Global Constraint - There are no customer → provider cycles, which is intuitive in that providers will not purchase bandwidth from customer. (for e.g. 2-levels having directed customer-provider resolves the issue)

Additionally there are forbidden configurations, which can be thought of as a filter on the notifications between nodes such as do not announce paths from provider to provider.

If all these conditions hold there is no dispute wheel.

Finally we can define a game such that Nash equilibria of the game are precisely the stable solutions in the equivalent SPP formulation. More precisely let P be the set of paths from i to d. For every $i \in 1 \dots n$, there exists a valuation function $v_i : P_i \to \mathbb{Z}$ such that $P_1, P_2 \in P_i$, $P_1 \neq P_2 \Rightarrow v_i(P_1) \neq v_i(P_2)$. We can define one-round routing game:

- for a player i, the set of strategies $S_i = N(i)$, set of neighbors of i and the utility of player i is $v_i(P)$ if P is induced by the routing choices $1, \dots, n$ and zero otherwise.

We can also consider multiround games and best respond dynamics that you can read more in the literature.