

Network Formation and Routing by Strategic Agents using Local Contracts

Elliot Anshelevich¹ * and Gordon Wilfong² **

¹ Department of Computer Science, Rensselaer Polytechnic Institute, Troy, NY.

² Bell Labs, Murray Hill, NJ.

Abstract. The Internet consists of various independent entities that range from small residential customers up to large backbone networks. The global network formation of the Internet is a result of the independent actions of these entities trying to achieve their individual connectivity needs balanced against their business goals. For example, customers accept contracts with ISPs to connect them to the Internet while ISPs and larger Autonomous Systems (ASes) make contracts called Service Level Agreements (SLAs) between each other to transit one another's traffic. ASes also try to control the routing of traffic to and from their networks in order to achieve efficient use of their infrastructure and to attempt to meet some level of quality of service globally.

We introduce a game theoretic model in order to gain understanding of this interplay between network formation and routing. Strategies for the players in the game allow them to make contracts with one another to forward traffic, and to re-route traffic that is currently routed through them. This model extends earlier work of [3] that only considered the network formation aspect of the problem. We study the structure and quality of Nash equilibria and quantify the prices of anarchy and stability, that is, the relative quality of a centralized optimal solution versus that of the Nash equilibria. For example, we show that when only the sender benefits (i.e., gains utility) for routed traffic demands then there is a Nash equilibrium as good as the centralized optimal solution and so the price of stability is 1. However when both the sender and receiver benefit from routed demands, then the price of stability is at most 2.

1 Introduction

The current Internet consists of tens of thousands of sub-networks known as *Autonomous Systems* (ASes), with each AS under the control of a single administrative authority. The task of this authority is to balance ensuring connectivity with the rest of the network against its own particular economic goals for managing the traffic entering and leaving the AS. These sometimes conflicting goals of connectivity versus traffic engineering are typically reflected in the local contracts called *Service Level Agreements* (SLAs) formed with neighboring ASes, which are essentially agreements to forward traffic. Without an appropriate SLA, no traffic would get from one AS to another, and so this system of business relationships is at the heart of Internet connectivity. The economic goals of an AS are also encoded into its routing policies, which are expressed in its local configuration of the Border Gateway Protocol (BGP) where BGP is the interdomain routing protocol in common use today. The local configuration of BGP determines the traffic into and out of the AS. These resulting traffic patterns may cause an AS to modify existing SLAs or create new ones. Thus there is an interplay between the contracts formed and the traffic patterns resulting from the encoding of the implied policies into BGP. The major goal of this paper is to develop an understanding of this interplay between network formation and traffic routing in a network where contracts are formed only between immediate neighbors.

Game theory is a natural approach for studying the behavior of such a collection of independent entities that have motivation to cooperate as well as conflicting self-interested economic goals. A number of different game theoretic approaches have appeared, most of which can be classified as studying network formation (e.g., [1, 10, 24]) or as interested in routing issues (e.g., [11, 32]). Our approach straddles these categories since we are interested in both aspects of a network.

* This research supported in part by the NSF Postdoctoral Fellowship in Mathematical Sciences and by NSERC Discovery grant 342457-07

** This research supported in part by NSERC Discovery grant 342457-07.

This paper builds on the earlier work of [3] in which a network formation game called the *Local Contract Formation Game* (LCFG) was defined. LCFG ignored the routing aspect of the resulting network, assuming that each traffic demand could only be sent along a given fixed route. In our paper, we allow routing choices as part of a player’s strategy in addition to the choice of which contracts to form. We also incorporate edge (or node) capacity constraints into our model. While these new features add complexity to the model, they are an important step towards modeling real world constraints and degrees of freedom faced by entities in today’s Internet. Even with these added features, we obtain meaningful results concerning the structure and quality of the stable business relationships in the resulting network.

Some aspects of LCFG as defined in [3] are retained in our model. For instance, both models have players representing typical Internet entities such as ASes, enterprise networks, and residential customers. Also, relationships between neighboring players are designed to capture three major properties of real-world business relationships seen in the Internet. The first of these properties is that relationships are strictly locally negotiated [20, 21]. That is, players negotiate contracts only with their neighbors. While these contracts take into account global issues such as the topology of the rest of the network and traffic demands throughout the network, they are formed strictly by negotiations between two neighboring players. Another property is that two possible types of contracts can be formed: customer-provider or peer-peer [8, 9, 12, 15]. Lastly, both models incorporate penalties such as those found in typical real world SLAs for failure to meet the conditions of a contract [19]. In particular, these penalties are designed to penalize a provider that accepts payment from a customer, but fails to satisfy the transit service for which the customer formed the contract.

Local Contract and Routing Formation Game (LCRFG) LCRFG is the game we study in this paper. An instance of LCRFG consists of a network topology of nodes and edges where nodes are the players in the game, and the edges represent potential contracts (and hence potential connections over which traffic can flow). A contract is an agreement to allow traffic headed for a given destination to flow on the edge in a specified direction. These contracts are subject to possible node and edge capacity constraints. We are also given a set of demands, where a demand d is a request for some amount of traffic to be sent from a source node $s(d)$ to a destination node $t(d)$. A demand is *active* (satisfied) if it is routed along a path that has the appropriate contracts formed along it. The source node and destination node of a demand each place a value on having the demand satisfied, and so their utility increases (say by some value $\lambda(d)$) if d is active. For each unit of traffic that gets routed through a node v , however, that node is “charged” a unit of utility for transiting this traffic. This *transit cost* represents the extra infrastructure cost that a node would incur due to its increased traffic load. Given this conflict between utility of active demands and transit cost, the major goal of a node v is to make sure that a lot of demands having v as an endpoint become active, while having as few demands as possible routed through v .

If node u agrees to pay some amount to neighboring node v , and v agrees to forward a unit of traffic from u towards some node t for this amount, then a contract between u and v is formed and the contract is said to be *active*. Node u is said to be the *customer* and v is said to be the *provider* for this contract. If the provider v fails to ensure that the traffic from u gets to t then v pays a *penalty* to u . Notice that more realistic contracts would involve a provider agreeing to forward traffic from its customer as well as *to* its customer. We show that our results hold with more realistic customer-provider contracts in Section 4, as well as with the introduction of peering contracts. For the majority of this paper, however, we consider the types of contracts defined above, which imply that an active demand must be routed on a directed path of contracts, each directed from a customer to a provider.

A node’s utility then is a function of the fees it pays (or receives) to form contracts, the cost for transited traffic, the payment (or receipt) of penalties, and the value it receives for each successfully routed demand for which it is an endpoint. A node’s strategy involves its choice of which contracts to form and how to route the demands that go through it. If no node can increase its utility by changing its current strategy, then the configuration of strategies is said to be a *Nash Equilibrium* (NE). We are interested in understanding and computing such stable configurations.

Our Results We focus on understanding the structure and quality of Nash equilibria, and on quantifying the prices of anarchy and stability³, which compare the quality of a centrally-defined optimal solution with the quality of the worst and best Nash equilibrium, respectively. We do this with respect to two objective functions defined as follows.

- The *social welfare*⁴ of a given set of active contracts equals the total value of the demands that are active minus the total transit cost incurred. If only the source of a demand is rewarded for this demand getting to its destination, then we show how to form a Nash equilibrium that is as good as the centralized optimal solution. In the case where both the source and destination of a demand are rewarded, however, both the prices of anarchy and stability in LCFG can be unbounded. In this case, we show that if every demand value $\lambda_{s(d)}(d)$ is increased by a factor of 2, then there is a NE as good as the original centrally defined social optimum.
- Instead of the social welfare objective, we also consider a non-mixed sign objective function we call the *cut-loss* objective function: the total transit cost incurred plus the total value of the demands that are not connected. This objective function is also studied in [3, 24]. Notice that the minimum of this objective is also the solution with maximum social welfare, so the change of objective only matters for approximation ratios such as the prices of anarchy and stability. For this objective, we show that the price of stability of LCRFG is at most 2, as well as how to efficiently compute a good Nash equilibrium.
- In Section 4, we show that the above results also hold for a variety of extensions to our model, including more realistic customer-provider contracts and the addition of peering contracts. We also prove that, with a slightly more constrained routing protocol, our results apply to contract systems where the contracts do not specify the destination. In other words, in this type of contract the provider would be obligated to forward all traffic from its customer, no matter where this traffic was destined.

Related Work The tradeoff between the benefits to a player (such as an AS) from routing its demands versus the loss of utility from transiting other traffic, was explored in several papers (e.g. [11, 24, 29, 28, 30]). Typically, however, the concentration has been on short-term routing and pricing schemes, e.g., [16, 18, 23, 26, 31]. While short-term pricing and admission models are of significant interest, they rely on an underlying set of business (longer-term economic) relationships. Moreover, agreements between entities (ISPs, enterprise or residential customers) tend to be based on more rigid contracts such as fixed bandwidth, or peak bandwidth contracts [20, 21]. This is largely due to the complexity (and expense) of monitoring IP traffic at a packet or flow level.

Motivated by this, several game theoretical models have addressed the strongly related notion of network formation. Some of these do not look at contract formation, but instead assume that edges have intrinsic costs [1, 2, 4, 5, 10, 17, 27]. On the other hand, contract formation models of networks have been heavily addressed as well, mostly in the economics literature (e.g., [13], for a survey see [22]). This body of work mostly addresses questions distinct from those studied here. In particular, none of these combine routing with contract formation, consider customer-provider and peering contracts, or measure the impact of provider penalties. In [6] a very general model of network formation is considered, and the model of [24] focuses on solutions where all demands are satisfied, and on so-called pairwise-stable equilibria. In addition, [7, 23] specifically address intra-domain concerns when forming peering contracts.

Our model is a direct extension of [3], which dealt with a similar network formation game, but without the ability to re-route traffic. Unlike most previous work, our goal is to study the interplay between routing and network formation, and so we must study both at the same time.

2 The Model and Basic Results

An instance of the Local Contract and Routing Formation Game (LCRFG) consists of a directed capacitated graph $G = (V, E)$ and a set of demands D , both of which we describe below.

³ See [25] for a definition of price of anarchy (known there as the coordination ratio), and [2] for a definition of price of stability.

⁴ Recall that the social welfare of a solution is the sum of the players' utilities.

We represent the network under consideration as a *capacitated* directed graph $G = (V, E)$ where each (directed) edge $e \in E$ has a capacity on the amount of traffic that can be transited along e . Nodes in V represent entities that might wish to establish contracts with one another to forward their traffic. For example, a node might represent an AS or a residential customer. An edge represents that there is a physical connection between the nodes (e.g., a fiber connection) and hence the capacity of the edge represents the actual capacity of this connection. We can also include node capacities without changing our results. The physical connection between nodes implies that there is the potential for the nodes on that edge to form a contract to forward their traffic directly rather than through intermediate nodes.

A *demand* d represents the desire of a source node $s(d)$ to be able have a unit of traffic routed to a destination node $t(d)$. A demand d also includes *values* $\lambda_{s(d)}(d)$ and $\lambda_{t(d)}(d)$ that represent the value that the source and destination nodes respectively place on having the demand traffic successfully routed. We describe below what it means for a demand to be successfully routed. Note that we chose to limit demands to be unit sized for notational simplicity, however in most cases extending our results to variable sized demands is straightforward. For technical reasons, we henceforth assume that each $\lambda_v(d) \geq 2$.

The nodes of V are the players in the game. As described above, the edges represent physical connections and potential contracts. Contracts are only allowed to be formed between entities that are physically connected in order to model systems of *local* contracts that occur in the real world [20, 21]. We next describe how contracts are formed in our game.

For each edge $e = (v, u)$ and a destination node t , u can *request* some amount, say c , from v . If v “accepts” this request, then v and u form a contract having the meaning that v pays u the amount c , and u is obligated to forward a unit of traffic flow from v to t . The amount c is called the *cost* of the contract. If v accepts the request then the contract is said to be an *active contract*. Let $X_e(t)$ be the set of all such active contracts for edge e and destination t . The capacity of an edge e is *violated* if $\sum_t |X_e(t)|$ is greater than the capacity of e . We use the notation $\chi_e(t)$ to denote an element of the set $X_e(t)$ and $c(\chi_e(t))$ to indicate the cost of the contract $\chi_e(t)$. Then we define $c(X_e(t)) = \sum_{\chi_e(t) \in X_e(t)} c(\chi_e(t))$. In the context of such a contract $\chi_e(t)$, node u is said to be the *provider* and v is the *customer*. We say that the contract $\chi_e(t)$ is a *customer contract* of node u and a *provider contract* of node v .

Given a collection of active contracts, $\bigcup_{e \in E, t \in V} X_e(t)$, one can define a directed multi-graph having the same node set V as G but with $\sum_t |X_e(t)|$ copies of directed edge e . Then in what follows when we speak of a path of contracts we will mean a directed path in this multi-graph.

One aspect of a node’s strategy in an instance of LCRFG is then to decide which requests to make for contract payments, and which requests to accept. Another aspect of a node’s strategy is to determine how to route demands over active contracts. We do not limit our model to a single routing protocol, but instead leave open what exact routing protocol is being used, in order to be most general. The only assumption we make about routing decisions is that they satisfy the following constraints.

- (R1) *The routing does not violate the contracts.* That is, if a demand d is routed from s to t , then the edges $e = (u, v)$ that are included in its path must have a contract $\chi_e(t)$ between u and v . Moreover, if x demands with destination t are forwarded from u to v , then there must be at least x contracts in $X_e(t)$ between u and v (since each contract $\chi_e(t)$ only obligates v to get a single demand to its destination t).
- (R2) *If a valid route exists between a source and destination of a demand d , then demand d is successfully routed.* By a *valid* route, we mean a path of contracts $\chi_e(t(d))$ between $s(d)$ and $t(d)$, with the capacity provided by these contracts still available (not taken up by other demands). If a contract is in the route of an active demand then the contract is said to *carry* the demand.

The basic goal of a player v is to establish contracts with its neighbors so as to manage the trade-off between getting its demands (i.e., the demands having v as an endpoint) successfully routed versus allowing the demands of others to pass through it. The strategy of a player v consists of making (and accepting) requests for contract payments, and of choosing the routes for demands passing through v , as long as the routing constraints (R1) and (R2) are satisfied. The details of player strategies are addressed in Section 2.1. We now describe the utilities of the players, and the possible equilibria.

Utilities The utility for a player v will include the payments v makes (or receives) for contracts and the values of active demands for which v is an endpoint. The utility will also involve a *transit cost* for active demands that include v in their path as well as certain *penalties*. These latter two components of the utilities are defined after we first introduce some required notation.

A *state* (solution) of this game is a set of active contracts (along with resulting payments) together with a routing of the demands along paths of active contracts. For a particular state of the network, let $P(d)$ be the path on which demand d is routed. We will call a demand *active* if it is successfully routed in the current state. Let $D^{end}(v)$ be the active demands having endpoint v , and $D(v)$ be the set of active demands for which $v \in P(d)$. Similarly, for any $S \subseteq V$, $D(S)$ is the set of active demands that include nodes of S .

Transit Costs: For any state of the network, define the *traffic transited* by a node v to be $D(v)$. For $x \in V$, let $t(x)$ be the total amount of traffic that x is transiting, i.e. $t(x) = |D(x)|$. We define the *cost of transiting* for a node x to be $t(x)$. That is, there is a normalized cost of 1 for each active demand transited.

Penalties: A provider must pay penalties to its customers if it fails to meet its obligations. There are many penalty systems that we could consider in this context. Just as with the routing protocol, instead of defining exactly what the penalty system is, we just give required conditions for it. Our results work for any penalty system meeting these conditions. Specifically, a penalty system must be such that:

- If a demand d is active (i.e., successfully reaches its destination), then no one pays penalties because of it.
- If a demand d that was previously active and routed through node v becomes inactive, but there is still a valid route from $s(d)$ to v , then v must pay a penalty to its customer on this valid route.

Notice that the second condition above assumes some notion of time passing. This assumption is unnecessary, as we can define many static penalty systems that obey the above conditions. For example, we can say that all providers with valid routes from $s(d)$ pay penalties for an inactive demand. Or we could make a more complicated system and say that inactive demands still have routes assigned to them, and the last provider in such a route that fails to forward this demand to the next hop must pay the penalty. In either case, the conditions above are satisfied, since if a provider decided to prevent a demand from getting to its destination (without re-routing or changing contracts), it would have to pay a penalty to its customer for failing in its provider duties.

Intuitively, these conditions say that if d becomes inactive, then the person to pay the penalty for this is the last in the chain of providers to which there still exists a valid route. It is the duty of this provider to get d to its destination, and since the valid route ends at it, this provider has failed in its duty. Accordingly, it pays a penalty to its customer. We set the size of the penalty for failing to form a route for demand d to be $\lambda_{s(d)}(d) - 1$. The intuition behind our definition of penalties is that a node $s(d)$ should be compensated for loss of any λ -value from one of its demands d routed through one of its (paid for) connections to a provider. The benefit to $s(d)$ is $\lambda_{s(d)}(d) - 1$ (since $s(d)$ receives $\lambda_{s(d)}(d)$ value if d is active but has a cost of 1 for transit of d). If penalties were any smaller, there would be instability in the system, since providers would prefer to pay a penalty instead of forwarding a customer's traffic. On the other hand, our results still hold if the penalties were allowed to be higher.

The Utility Function: Given the endpoint values for active demands, the transiting costs, payments, and the penalties, the utility of node v (for a solution \mathcal{S}) is written as $utility(v)$ (or $utility_{\mathcal{S}}(v)$), and is defined as follows.

$$utility(v) = \sum_{d \in D^{end}(v)} \lambda_v(d) - t(v) + \sum_{e=(w,v),t} c(X_e(t)) - \sum_{e=(v,w),t} c(X_e(t)) - \sum_d p_v(d) \quad (1)$$

In other words, a node v gains the value of $\lambda_v(d)$ for each active demand that it originates, loses 1 for every demand it transits, gets payment $c(\chi_e(t))$ according to the contract it makes with its neighbor on e (either positive if v is paid or negative if it pays) and loses $p_v(d)$ for penalties (either positive or negative depending on whether it pays or receives the penalty).

The utility function defined above applies to the case when for each edge e incident to a node v , the number of active contracts $\sum_t |X_e(t)|$ does not violate the capacity constraint of e . The second case is when the number of active contracts $\sum_t |X_e(t)|$ for some e incident to v violates the capacity of e . In this case, v is assigned a negative utility. It is easy to see that in a Nash equilibrium no node is involved with contracts that violate capacities, since they could always do better by doing nothing and receiving 0 utility.

2.1 Player Strategies and Nash Equilibria

Player Strategies As mentioned above, a strategy of a player v consists of making (and accepting) requests for contract payments, and of possibly affecting the routes for demands passing through v , as long as the routing constraints **(R1)** and **(R2)** are satisfied. Many concrete games can be defined that fit into this framework (see the Appendix for some examples). Instead of focusing on only one such game, however, we choose to leave our framework as general as possible. In fact, our results hold for *any* game where a player v only has power over its contracts and over the demand routes passing through v , as long as the routing constraints **(R1)** and **(R2)** are guaranteed to hold. As long as a player v cannot affect anything else by changing its strategy, we will show the existence of good Nash equilibria. To carefully define an equilibrium, we do not need to define precisely the strategies of the game, but must define the set of possible *player deviations*, with a solution being an equilibrium exactly when no player would want to take one of their possible deviations.

Nash Equilibria Given a solution \mathcal{S} , a *deviation* for a node v is a solution where v changes its actions while all others remain as in \mathcal{S} . We will see later on that the only deviations of interest will not result in the ability to activate currently inactive demands. Specifically, a node v can deviate by performing the following actions, possibly both at once.

- Changing the costs of its requests for customer contracts, and changing its decisions on the acceptance/ non-acceptance of provider contracts.
- Re-routing any of the active demands that pass through v .

This means that v controls all of the contracts it makes, and all of the routes that pass through it. Just as our other assumptions about the routing protocol, this is a somewhat general assumption, and many routing protocols obey this condition (see some examples in the Appendix). In the most powerful deviation that we allow, a node would be able to re-route by canceling all routes that pass through it (and thereby freeing their capacity), and then forming new routes for these demands instead. Our results will hold for any weaker deviations as well, such as ones where v can only change some of the routes that pass through it, instead of all of them. In essence, our results will hold for any game where by deviating, a player v is only able to affect its contracts and the routes passing through v , but nothing else in the network.

A *best deviation* for a node v is a deviation for v that results in the highest utility for v over all possible deviations for v . In the case that a best deviation for v is to stay with its original strategy, we say that v is *stable*. If all nodes are stable in a solution \mathcal{S} , then \mathcal{S} is said to be a *Nash equilibrium* (or NE). We call a NE *nontrivial* if it has at least one active demand (and trivial otherwise). We say that a set of contracts C and a set of demands D *induces* a NE when there is a NE whose set of active contracts and demands are C and D . We sometimes say that a set of contracts alone induces a NE when the set of active demands is obvious from context.

3 Good Equilibria and Price of Stability

Definite Nash Equilibria and Edge-Cutting Deviations In our model a trivial Nash equilibrium always exists. A solution where all players v set all $c(\chi_e(t))$ to a number M so large that accepting such a contract guarantees the accepting node would have negative utility, results in no active edges, and is a Nash equilibrium with all nodes having a utility of 0. Moreover, for every Nash equilibrium there is an equivalent one where the payments demanded on inactive edges are M . We can also assume that there

are no active contracts without demands being routed on these contracts, as such contracts would have no reason to exist. Call such a Nash equilibrium *definite*. Since our primary interest is to determine how well the players (and society) can do via stable solutions, we may restrict attention without loss of generality to such definite NEs. In a definite NE, no new contracts can be formed since all requests for inactive contracts are too high to be accepted. Thus we can now simply think of deviations as a node “cutting” some of the active contracts incident to it, as well as re-routing the demands that pass through it. Cutting is achieved either by a customer refusing a request it is currently accepting or a provider requesting M for that contract. Moreover, in a definite NE, the capacity of each active contract is fully saturated, and so no new demands can be routed without active ones being re-routed. Notice that since there are no valid routes with excess capacity in such a solution, no penalties are paid by anyone.

The Centralized Optimum An *optimal centralized solution*, that we denote by OPT, is a set of active edges together with a routing that maximizes the social welfare function $\sum_{v \in V} utility(v)$. Notice that all the contract payments and penalties cancel out, so this objective function is just $\sum_{d \in D(V)} (\lambda_{s(d)}(d) + \lambda_{t(d)}(d) - \|P(d)\|)$ where $\|P(d)\|$ is the number of nodes in $P(d)$. That is, the objective function is just the total value for each endpoint of active demands minus the transit costs of active demands.

Nash equilibria as good as OPT We now compare the quality of the centralized optimal solution with the quality of Nash equilibria. First, we consider the case where only the source $s(d)$ of a demand d is rewarded for d becoming active. In other words, we consider the case where $\lambda_{t(d)}(d) = 0$ for all demands d . For this special case, we show that there exists a Nash equilibrium that is as good as OPT, and so the price of stability is 1. The full proofs of all our results appear in the Appendix.

Theorem 1. *If $\lambda_{t(d)}(d) = 0$ for all demands d , then there exists a Nash equilibrium that is as good as the optimal centralized solution.*

Price of Stability is at most 2 In this section, we consider the more general version of our model, where both the source and the destination of a demand may benefit from this demand being successfully routed. Indeed, in many applications it is not clear who benefits more from a packet getting to its destination: the person who sent it, or the person who received it. This is why we leave this question open, and say that the source of a demand benefits by $\lambda_{s(d)}(d)$, while the destination benefits by $\lambda_{t(d)}(d)$.

For this model, we consider the quality of Nash equilibria using two different objective functions. The first one is the social welfare, as described above. The second objective is a non-mixed sign objective function we call the *cut-loss* objective function. In this objective, we seek to minimize the total transit cost incurred plus the total value of the demands that are not connected. This objective function is also studied in [3, 24]. Notice that the minimum of this objective is also the solution with maximum social welfare, so the change of objective only matters for approximation ratios such as the prices of anarchy and stability. The cut-loss objective also makes sense intuitively, since it simply corresponds to minimizing the overall cost in the network, which includes the transit cost, as well as the cost incurred from failing to route demands.

While in Theorem 1 we showed that there always exists an equilibrium as good as the optimal solution, this is no longer true if destinations also benefit from active demands. This is true even for examples without routing, and [3] gives examples where for the social welfare objective, all equilibria may be arbitrarily far away from OPT, since every Nash equilibrium has social welfare of 0, while the quality of OPT is positive. These examples are still valid for the model with routing presented here.

For the cut-loss objective, however, there always exist good Nash equilibria. In the proof of the following theorem we not only show the existence of such equilibria, but also how to find them efficiently starting with an approximation to OPT. For this result, we assume that $\lambda_{s(d)}(d) = \lambda_{t(d)}(d)$ for all d . If this were not the case, then instead of a factor of 2, we obtain a price of stability that depends on the ratio between $\lambda_{s(d)}(d)$ and $\lambda_{t(d)}(d)$.

Theorem 2. *With respect to the cut-loss objective function, the price of stability is at most 2 in the case where for each demand d , $\lambda_{s(d)}(d) = \lambda_{t(d)}(d)$.*

Creating good equilibria While for the social welfare objective, all Nash equilibria may be bad compared to OPT, we can still *create* good equilibria using similar methods as in [3]. For example, if we give incentives to certain nodes, we can form an equilibrium that is as good as the centralized optimal solution (see [3] for discussion about different types of incentives). Specifically, if we increase the $\lambda_{s(d)}(d)$ for every demand by a factor of 2, then in the resulting game instance there is a Nash equilibrium whose active contracts are exactly the active contracts of OPT, with the same demands being active. As in Theorem 2, we assume $\lambda_{s(d)}(d) = \lambda_{t(d)}(d)$ for all d . If this were not the case, then instead of increasing $\lambda_{s(d)}(d)$ by a factor of 2, the results hold if we set $\lambda_{s(d)}(d)$ to $\lambda_{s(d)}(d) + \lambda_{t(d)}(d)$, which is still a factor of 2 increase in total.

Theorem 3. *Let E^* be the set of active contracts in OPT, and D_{OPT} be the active demands in OPT. If we increase $\lambda_{s(d)}(d)$ by a factor of 2 for every d , then E^* together with D_{OPT} induce a Nash equilibrium.*

4 Model Variations

What if contracts are bi-directional? What if we have peering contracts? By “bi-directional” contracts, we mean that there are two kinds of customer-provider contracts between a customer u and provider v : ones that require v to forward u ’s traffic to t , and ones that require v to forward traffic destined for t to u . The first kind of contract is the same as we had in most of this paper, and the second kind of contract represents the desire for nodes to receive traffic, and the payments they give to their providers so that these providers forward traffic to them.

In addition to these two types of customer-provider contracts, suppose we also have peering contracts. Peering contracts usually represent the case when two neighboring nodes agree to forward each other traffic, without exchanging any payments. In our model, peering contracts are not labeled with a destination, and consist of two nodes agreeing to form a peering contract by not demanding any payment from each other. Peering contracts are allowed to have a capacity, but are not required to. We can also allow peering contracts where neighboring nodes u and v agree to exchange traffic without any payments, but require that the same amount of traffic is sent from u to v as from v to u .

By a simple argument given in the Appendix, all of our results from Section 3 hold even if we allow the existence of the above types of contracts.

What if contracts do not have capacities? Suppose that an active contract $\chi_e(t)$ means that the provider has the obligation to forward *all* traffic from its customer to the destination t , instead of just a single unit. As we show in the Appendix, if the edge capacities are large enough (or non-existent), then all our results still hold. Unfortunately, if the edge capacities are too small, we can get instability in the system, ending up with the only stable solutions being much worse than the centralized optimum. Specifically, there are examples where two routes are possible for a demand with a very high $\lambda_{s(d)}(d)$ -value, and nodes on these two routes try to make sure that the demand does not proceed through them, leading to a stable solution only if d becomes inactive.

What if contracts do not specify the destination? In the case where the contracts do not specify a particular destination nor have a capacity associated with them, there are examples where the only Nash equilibria are extremely bad compared to the centralized optimum. Specifically, the prices of anarchy and stability can be arbitrarily large for both the social welfare and the cut-loss objective. Notice that this was not true for the model in [3], and occurs because nodes are able to re-route demands, which results in fewer Nash equilibria.

In the case where a contract means that the provider has the obligation to forward a single unit of traffic from its customer, but without a specified destination, things are a bit better. Specifically, we can prove the following theorem. This theorem says that if we constrain the routing powers of all nodes to only controlling the route that traffic takes *after* it leaves them, instead of before and after, then the results of Section 3 still hold.

Theorem 4. *If every node v can only affect the route portion that comes after v (for demands that currently pass through v), then the results of Theorems 1, 2, and 3 still hold, even if the contracts do not specify particular destinations.*

Acknowledgments

We deeply thank Bruce Shepherd for illuminating discussions and for his many contributions.

References

1. E. Anshelevich, A. Dasgupta, É. Tardos, T. Wexler. Near-optimal network design with selfish agents. *STOC*, 2003.
2. E. Anshelevich, A. Dasgupta, J. Kleinberg, É. Tardos, T. Wexler, T. Roughgarden. The Price of Stability for Network Design with Fair Cost Allocation. *FOCS*, 2004.
3. E. Anshelevich, B. Shepherd, G. Wilfong. Strategic Network Formation through Peering and Service Agreements. In *FOCS 2006*.
4. V. Bala, S. Goyal. Self-organization in communication networks. McGill University Technical Report, 1996.
5. V. Bala, S. Goyal. A noncooperative model of network formation. *Econometrica*, 68, pp. 1181–1229, 2000.
6. Francis Bloch, M.O. Jackson. The Formation of Networks with Transfers among Players. *J. Economic Theory*, 133 (1), 2007, pp. 83–110.
7. Jacomo Corbo, Thomas Petermann. Selfish peering and routing in the Internet. CoRR cs.GT/0410069, 2004.
8. T. Erlebach, A. Hall, A. Panconesi, D. Vukadinović. Cuts and Disjoint Paths in the Valley-Free Path Model of Internet BGP Routing. *Proceedings of CAAN*, pp. 49–62, 2004.
9. T. Erlebach, A. Hall, T. Schank. Classifying Customer-Provider Relationships in the Internet. *Proceedings in Informatics*, pp. 52–64, September 2002.
10. A. Fabrikant, A. Luthra, E. Maneva, C.H. Papadimitriou, S. Shenkar. On a network creation game. *PODC*, 2003.
11. J. Feigenbaum, C. Papadimitriou, R. Sami, and S. Shenkar. A BGP-based Mechanism for Lowest-Cost Routing. *Distributed Computing* 18, pp. 61-72, 2005. (Special issue of selected papers from Proc. of ACM PODC'02.)
12. L. Gao, J. Rexford. Stable internet routing without global coordination. *Proceedings of ACM SIGMETRICS '00*, Santa Clara, CA, June 2000.
13. A. Gomes. Multilateral contracting with externalities. *Econometrica*, 73(4), pp. 1329–1350, 2005.
14. T. Griffin, F. B. Shepherd and G. Wilfong. The Stable Paths Problem and Interdomain Routing. *IEEE/ACM Transactions on Networking*, Vol. 10, No. 2, pp. 232-243, 2002.
15. T. Griffin, G. Wilfong. On the Correctness of IBGP Configuration. *Proceedings of SIGCOMM*, pp. 17–29, 2002.
16. L. He, J. Walrand. Dynamic provisioning of service level agreements between interconnected networks. *Conference on Stochastic Networks*, June 19-24, 2002.
17. H. Heller, S. Sarangi. Nash networks with heterogeneous agents. *Virginia Tech. Working Paper Series*. E-2001-1, 2001.
18. B.E. Hermalin, M.L. Katz. Network interconnection with two-sided user benefits. *Working paper available: http://faculty.haas.berkeley.edu/hermalin/Interconnection_v14.pdf*
19. http://hosting.bellsouth.net/bellsouthhosting/s/s.dll?spage=cg/news/service_level.htm
20. <http://www.verio.com/access/pricing.cfm>
21. <http://www.xo.com/products/smallgrowing/internet/dia/oc3.html>
22. M. Jackson, A survey of models of network formation: stability and efficiency. *Group Formation in Economics: Networks, Clubs and Coalitions*, eds. G. Demange and M. Wooders, Cambridge Univ. Press. <http://www.hss.caltech.edu/~jacksonm/netsurv.pdf>
23. R. Johari, J. Tsitsiklis. Routing and Peering in a Competitive Internet. *Proceedings of the IEEE Conference on Decision and Control*, 2004.
24. R. Johari, S. Mannor, and J. Tsitsiklis. A contract-based model for directed network formation. *Games and Economic Behavior*, 56 (2), 2006, pp. 201–224.
25. E. Koutsoupias, C. Papadimitriou. Worst-case equilibria. In *Proceedings of the 16th Annual Symposium on Theoretical Aspects of Computer Science*, pp. 404–413, 1999.
26. J.J. Laffont, J.S. Marcus, P. Rey, J. Tirole. Internet connection and the off-net pricing principle, *The RAND Journal of Economics*, vol. 34, n. 2, pp. 370–390, 2003. Earlier version in *Conference on the Economics of the Internet and Software Industries* Jan. 18-20, 2001.
27. J. Larrosa, F. Tohmé. Network formation with heterogeneous agents. Economics Working Paper Archive EconWPA, Microeconomics series, No. 0301002, 2003.

28. E. Markakis and A. Saberi. On the core of the multicommodity flow game. *Proc. of EC*, pp 93–97, 2003.
29. C. Papadimitriou, Algorithms, Games, and the Internet. *Proc. of ACM STOC*, 749–753, 2001.
30. B. Shepherd, G. Wilfong. Multilateral Transport Games. *Proceedings of INOC*, pp. B2-378–B2-385, 2005.
31. S. Shakkottai, R. Srikant. Economics of network pricing with multiple ISPs. *INFOCOM*, 2005.
32. H. Wang, H. Xie, Y.R. Yang. On the stability of interdomain inbound traffic engineering. *SIGCOMM Poster Session*, 2005.

Appendix

A Some examples of concrete games that fit into our framework

Consider the following routing protocol. For every incoming active demand d on contract $\chi_e(t)$ into v , v decides where to send the traffic of d by choosing a *next-hop* contract. That is, v chooses some node w that it has a contract with, and forwards d 's traffic to w . w then forwards d to its next-hop, and this continues until d reaches its destination. With this routing protocol, v 's strategy consists of the choice of the next-hop node for every incoming active demand (together with the payment offers/requests for neighboring contracts, as described in Section 2). This game fits into our framework as long as a node is not allowed to send traffic into “dead ends”. By this we mean that if node v has a route with free capacity from itself to t , and has an incoming demand with destination t , then it must forward this demand on its route to t , instead of purposefully sending this demand into a dead end where it will never reach its destination. Such practices would not go unnoticed for long, and so we assume that all nodes actually forward their traffic to the appropriate next-hop as long as there exist viable routes. This property guarantees that the routing constraints **(R1)** and **(R2)** defined in Section 2 are satisfied. Moreover, the assumptions of Section 2.1 are satisfied as well, since by changing the next-hop nodes, a node v can only affect the demands that are traveling through it, and none other. Therefore, all our results hold for this game.

Other games of this type also fit into our framework. For example, versions of BGP as described in [14] satisfy all of our assumptions. In such games, players have a preference ordering on all the paths from themselves to every destination. Using this preference ordering, they choose the next-hop nodes in a systematic way, and propagate routing information to their neighbors. With the extra assumption that a demand is always successfully routed as long as a viable route exists, all our results hold for such routing protocols as well.

In the above examples, each player is able to control the routes of the demands going through it to a large degree. Now suppose that a player is not able to control the routing at all, and instead it is decided by some centralized protocol. All our results certainly hold in this case, since the players only have less power in such games, and so their possible deviations are more limited.

Finally, consider the routing protocol where the source of the demand (or the destination) chooses the route, and no node that transits this demand has any choice. This important game also fits into our framework, as long as a source s routes its demand d on a viable route (i.e., with appropriate contracts and with free capacity) if one exists, and does not purposefully choose a route that does not let d reach its destination. In this game, a strategy of a player v would be a route for every demand originating at v , together with the payment offers/requests for neighboring contracts, as described in Section 2. Since a player is only able to control the routes of demands that pass through it (in fact, it is only able to control the routes of demands that originate at it), all our assumptions are satisfied.

B Model Variations

What if contracts are bi-directional? What if we have peering contracts? These types of contracts result in demands being routed only on valley-free paths [8], also called Type-1 and Type-2 paths in [12], valid signaling paths in [15], and valid paths in [3]. A valley-free path for demand d consists of some customer-provider contracts directed away from $s(d)$, possibly followed by one peering contract, and followed by more customer-provider contracts directed towards $s(d)$. These are the only paths on which traffic can be routed, since any other path would contain a node v forwarding traffic between two providers, or

between a peer and a provider. In our game (as well as in reality [9, 12]), node v would have an incentive to not forward this traffic, since it would save on transit cost, and would not have to pay any penalties.

All of our results hold for this extension. Consider an optimal solution OPT. Because OPT is the solution that maximizes the total value of the active demands while minimizing the total transit cost, it does not matter what types of contracts we assign to different edges in OPT, as long as all the demands specified by OPT are active. For every node t , consider the active demands of OPT with destination t , and make the contracts along which they are routed be the first type of customer-provider contracts, as we defined in our original model. This is still an optimal solution, but this solution does not use any peering contracts or the second type of customer-provider contracts. By the results of Section 3, we can form Nash equilibria that are either as good, or almost as good, as this optimal solution. Therefore, all the results of Section 3 still hold in the presence of peering contracts and the second type of customer-provider contract, simply because the optimal solution remains the same, and the Nash equilibria using only the old types of contracts are still equilibria.

What if contracts do not have capacities? Suppose that an active contract $\chi_e(t)$ means that the provider has the obligation to forward *all* traffic from its customer to the destination t , instead of just a single unit. If the edge capacities are large enough (or non-existent), then all our results still hold. As in the proof of Theorem 5, the sets of contracts labeled with different destinations do not affect each other, so we can consider the graph OPT_t consisting of contracts labeled with t separately. In this case, OPT_t is simply a tree directed towards t . This tree can be found efficiently using dynamic programming, and since OPT_t is a tree, then no re-routing is possible, and so the results from [3] apply. This means that all of the results of Section 3 can be extended to this case.

C Proofs

The Structure of Equilibria Below we show the existence of good Nash equilibria, as well as how to compute them in many cases. To do this, we first develop the tools needed for such arguments, by giving a sufficient condition for the existence of a Nash equilibrium.

Consider a set of active contracts C together with a routing R . First, notice that all contracts are with respect to a particular destination, and so demands heading to different destinations do not affect each other, because they cannot use the same contracts. This means that we can look at all the contracts and demands labeled with some destination t separately from other destinations. Let G_t be the directed multi-graph of active contracts in C labeled with destination t , together with a routing R_t of demands destined for t . For any node u , let P_u be a longest path in G_t from u to t , and let $\|P_u\|$ be the number of nodes in P_u . Below we establish sufficient conditions for G_t to induce a Nash equilibrium, i.e., conditions that imply that there is a set of payments for the contracts in G_t that results in a stable solution with all the same demands being active.

Theorem 5. *If G_t is acyclic, every contract in G_t carries an active demand, and for each active demand d , $\lambda_{s(d)}(d) \geq \|P_{s(d)}\|$, then G_t induces a Nash equilibrium.*

To prove this theorem, we will first need several helpful lemmas. Below, we often refer to a capacity of an edge (u, v) in G_t , by which we mean the number of contracts in $X_e(t)$ from u to v in G_t . The assumption in the theorem above says that these capacities are fully saturated, so the number of contracts in $X_e(t)$ made from u to v is exactly the number of demands with destination t that are routed from u to v . When we talk about the direction of a contract, we will mean the direction from the customer to the provider.

Lemma 1. *Let G_t be acyclic and let A be the set of active demands with destination t in G_t that are routed in R_t through node v . Consider a deviation of v where the only changes v makes is to re-route active demands. Then no matter how v re-routes these demands, it must be that all $d \in A$ are still active, and their routes still pass through v . In addition, after v re-routes the demands, all the capacities of G_t are still saturated.*

Proof. By our assumption about the routing protocol and the nature of deviations, demands in A are exactly the demands whose routes v can change. Then the routes of demands not in A do not change. The contracts and capacities of the edges do not change, since v is only re-routing. Let $\kappa(e)$ be the number of demands in A that include e in their route in R_t . Then, from v 's point of view, it is routing the demands of A in any way it likes, on a graph with residual capacity $\kappa(e)$ (call this directed graph G_κ , and delete from it the edges with capacity $\kappa(e) = 0$). This is because all the demands that are not in A remain on the same routes, and so v cannot use the capacity that is taken up by these demands. Node v cannot route more than $\kappa(e)$ demands on edge e , since we assumed that the capacities of the edges before re-routing are tight in G_t . Notice that G_κ has no cycles, since it is a subgraph of G_t .

We will first show that every demand in A is active after re-routing. Suppose to the contrary, that some demand d does not reach its destination after re-routing. By our assumption about the routing protocol, it is enough to show that there exists a valid path from $s(d)$ to t with free capacity. Consider the edges of G_κ leaving $s(d)$ after re-routing. Since the capacity out of $s(d)$ is greater than the capacity entering $s(d)$ (because d was successfully routed starting with $s(d)$ before, and took up one capacity in the edges leaving $s(d)$), there must be an edge $(s(d), u_1)$ with free capacity. Now consider u_1 . The capacity of edges leaving u_1 in G_κ equals exactly the capacity of edges entering it plus the number of demands in A starting at u_1 . Therefore, since there was an edge with free capacity entering u_1 , there must also be one leaving it. We can continue this argument until we get to t , and we cannot cycle since G_κ has no cycles. Therefore, all demands of A are still active after re-routing.

To prove the rest of the lemma, suppose that after re-routing, a demand $d \in A$ still gets through to its destination t , but is not routed through v . This means that there is a path P in G_κ from $s(d)$ to t that does not go through v . Every edge of P must be in some route of a demand from A before v 's re-routing, since otherwise it would not have any residual capacity, and would not be in G_κ . For each $e \in P$, let P^e be such a route that contains edge e . In P^e , e appears either before v , or after v , since all routes of A before re-routing pass through v . For the last edge e of P , e appears after v in P^e , since e leads directly to t . Suppose that there exist two consecutive edges $e_1 = (u_1, w)$, $e_2 = (w, u_2)$ in P , such that e_1 appears before v in P^{e_1} , and e_2 appears after v in P^{e_2} . This means that if we take the union of the portion of P^{e_1} from w to v with the portion of P^{e_2} from v to w , then it contains a cycle. Thus we can obtain a cycle in G_κ , which we know to be acyclic.

Now suppose that there is no node w as described above. This means that for all edges e of P , e appears after v in P^e , and in particular this is true for the first edge e_1 of P . Consider the path before re-routing of demand d that is now taking path P , however. Its route started at $s(d)$, and passed through v . By concatenating part of this route with a subpath of P^{e_1} , we can create a cycle, using the same argument as above.

Finally, to show that all capacity on G_t is still saturated after re-routing, suppose that there is an edge $e = (u, v)$ in G_t with free capacity. Since the capacity of edges leaving v in G_t is exactly the amount entering v plus the number of active demands in G_t originating at v , this means that there must be an edge with free capacity leaving v as well. We can continue this argument until we either create a cycle (which is impossible since G_t is acyclic), or reach t . But if t has an edge with free capacity, this means that not all the demands that were active before are now reaching t , which contradicts with the fact that all demands still reach their destinations after re-routing. ■

We can now show a solution such that for any node v , after re-routing and changing some of its contracts, v 's utility does not improve. A deviating node v might re-route some demands that pass through it, and cut some contracts. Although node v is allowed to do both at the same time, we will instead think of node v as first re-routing demands, and then destroying contracts. We will show that v 's utility will not increase after either action, which shows that our solution is an equilibrium.

Lemma 2. *After v re-routes some demands, its utility stays the same.*

Proof. By Lemma 1, all the demands that got through in G_t still are active after re-routing, and so the λ benefit that v receives does not change. All of the re-routed demands still pass through v , so its transit cost is the same as before, and there are no new penalties added or removed. Since the contract

payments do not change because of re-routing, v 's utility after re-routing is exactly the same as before.

■

We now form the payment scheme on G_t that forms a Nash equilibrium. Specifically, we set the payments on contracts $\chi_e(t)$ entering node u to have the value of $\|P_u\|$ (that is, u asks for a payment of $\|P_u\|$, and its customer offers a payment of $\|P_u\|$).

We are now ready to prove the main theorem of this section, showing that the payments above form a Nash equilibrium.

Proof of Theorem 5. Let the contract payments on G_t be as described above. We must now show that if any node v deviates from its current strategy by re-routing some demands and changing some contracts, then its utility will only decrease. Fix a node v . By Lemma 2, we already know that v 's utility does not change from re-routing. We must now show that after cutting some adjacent contracts, v 's utility can only decrease.

Suppose v cuts a set of customer contracts S_1 , and a set of provider contracts S_2 . We will show that for every improvement in v 's utility resulting from this action, there is a corresponding decrease. First, consider why v 's utility might improve. For every provider contract $\chi_e(t) \in S_2$, v gains $c(\chi_e(t))$ utility, since it no longer has to pay money to its provider, and for every demand d whose route contains either an edge from S_1 or S_2 , v gains 1 unit of utility, since it no longer has to transit this demand. v cannot gain any utility from no longer having to pay penalties, since it is not paying any penalties in G_t . This is because for a penalty to be paid, there have to be edges with free capacity, and we assumed that capacities are tight in G_t .

Consider the increase of $c(\chi_e(t)) + 1$ in v 's utility resulting from cutting the contract $\chi_e(t) \in S_2$. By Lemma 1, there must be some demand d that is routed on $\chi_e(t)$, since all capacities are saturated after re-routing. Consider the previous contract $\chi_f(t)$ in this demand's path. There are 3 cases: d originated at v ; $\chi_f(t) \in S_1$; and $\chi_f(t) \notin S_1$. We will deal with each of these cases separately.

If d originated at v , then by destroying the contract $\chi_e(t)$, v makes sure that d is no longer active. This means that v loses $\lambda_{s(d)}(d)$ utility that was received from d being active. We will now show that $\lambda_{s(d)}(d) \geq c(\chi_e(t)) + 1$. Let w be the provider of v that v makes the contract $\chi_e(t)$ with. By construction, $c(\chi_e(t))$ is the length of the longest path P_w in G_t from w to t . By the assumption in the theorem statement, $\lambda_{s(d)}(d) \geq \|P_v\|$, and $\|P_v\| \geq \|P_w\| + 1$, since G_t is acyclic. This means that $\lambda_{s(d)}(d) \geq c(\chi_e(t)) + 1$, and so v loses at least as much as it gains by cutting $\chi_e(t)$.

Now consider the case where $\chi_f(t) \notin S_1$. This means that there is a valid path from $s(d)$ to v with free capacity, and so v must pay a penalty to its customer on $\chi_f(t)$. This penalty is of size $\lambda_{s(d)}(d) - 1$. By the assumption in the theorem statement, $\lambda_{s(d)}(d)$ is greater than the longest path from $s(d)$ to t . Let w be the provider of v that v makes the contract $\chi_e(t)$ with. Since $c(\chi_e(t))$ is the length of the longest path from w to t , which is at least 2 nodes shorter than the one from $s(d)$ to t (because d 's route is a path that goes from $s(d)$ to v to w), this means that $\lambda_{s(d)}(d) - 1 \geq c(\chi_e(t)) + 1$, and so v loses at least as much as it gains by cutting $\chi_e(t)$.

Finally, consider the case where $\chi_f(t) \in S_1$. That means that $\chi_f(t)$ is also cut by v . By construction, $c(\chi_f(t))$ is the length of the longest path from v to t , so $c(\chi_f(t)) \geq c(\chi_e(t)) + 1$, and so v loses at least as much as it gains by cutting $\chi_e(t)$.

We have now argued that v does not gain anything from cutting provider contracts. If v cuts a customer contract $\chi_f(t)$, it can save a unit of transit cost because it no longer has to transit the demand that was routed through $\chi_f(t)$. By our construction of payments, however, $c(\chi_f(t)) \geq 1$, so v does not gain anything from cutting a customer contract either, as it loses the payment on this contract.

■

Theorem 1 *If $\lambda_{t(d)}(d) = 0$ for all demands d , then there exists a Nash equilibrium that is as good as the optimal centralized solution.*

Proof. We will show that for the optimal solution, we can form a set of contract payments so that no one will want to deviate. Consider a subgraph (and a sub-routing) of OPT, where we include only contracts of OPT labeled with destination t , and only the demands with destination t . Call this subgraph

(or sub-solution) OPT_t . Since demands with destination t are only routed along contracts in OPT_t , we have that any demands with different destinations do not affect each other. The utility function of each node consists of separate components for each destination t , so if we can prove that OPT_t is an equilibrium, then so is the entire solution OPT.

To show that OPT_t induces a Nash equilibrium, we need only invoke Theorem 5. Therefore, we need to show that OPT_t meets the required conditions for Theorem 5. We can assume that all active contracts have a demand routed through them in OPT_t , since otherwise we could remove those contracts without altering the quality of the solution. We must now show that OPT_t is acyclic, and that for all active demands d , $\lambda_{s(d)}(d) \geq \|P_{s(d)}\|$.

To see that OPT_t is acyclic, consider a cycle in OPT_t , and eliminate this cycle in OPT (by reducing the capacity of this cycle by 1). All of the active demands in OPT can still be successfully routed. To see this, consider that OPT_t is a single-commodity flow of demands from many sources to a single destination. By eliminating a cycle from this flow, we result in a valid flow of the same size, since flow conservation is still satisfied. The same amount of flow reaches t , so all the demands can still be routed after elimination of this cycle. Therefore, the λ benefit from active demands does not change by our elimination of the above cycle. By eliminating this cycle, however, we reduced the total transit cost by the length of this cycle, since we assumed that every active contract was carrying an active demand in OPT_t . Therefore we have created a solution that is strictly better than OPT, a contradiction.

To prove the final condition needed for Theorem 5, consider a demand d that is active in OPT_t , and decrease the capacities along $P_{s(d)}$ by 1. By Lemma 3, every demand except d can still be successfully routed in this new solution. Thus, by deleting $P_{s(d)}$, the quality of this solution decreases by $\lambda_{s(d)}(d)$, since d is no longer routed. This is the only part of the proof where we use the assumption that only the source benefits from an active demand, since otherwise the quality of the solution would decrease by $\lambda_{s(d)}(d) + \lambda_t(d)$. There is no more transit cost that is paid along $P_{s(d)}$, however, so the transit cost decreases by $\|P_{s(d)}\|$. Therefore, if $\lambda_{s(d)}(d) < \|P_{s(d)}\|$, this would mean that we have created a solution that is better than OPT, a contradiction. This implies that $\lambda_{s(d)}(d) \geq \|P_{s(d)}\|$, as desired. ■

To complete the proof of the above theorem, we need the following lemma.

Lemma 3. *Let P be any path in OPT_t from the source of a demand d to t . Form OPT'_t from OPT_t by reducing the capacity of each edge in P by 1. Then, all the demands except d that are active in OPT_t can still be successfully routed in OPT'_t .*

Proof. The number of contracts $|X_e(t)|$ in OPT_t is exactly the number of active demands in OPT_t passing through edge e , since we can assume that capacities are fully saturated in OPT_t (otherwise we could remove the unneeded contracts). Therefore, the capacity out of node v is exactly the capacity into node v plus the number of active demands leaving v . This is still true in OPT'_t , since we reduce both in-capacity and out-capacity by 1 for all nodes in P , except for $s(d)$, where we reduce the out-capacity by 1, but also remove demand d .

Consider a graph H_t , where we add a new super-source node s to OPT'_t , with edges $(s, s(d'))$ leading from s to each source of an active demand d' with destination t , and the capacity of these edges being the number of such demands originating at $s(d')$. By the above argument, the capacity of any s - t cut exactly equals the number of active demands, and so we are able to route all the demands except d successfully. ■

Theorem 2 *With respect to the cut-loss objective function, the price of stability is at most 2 in the case where for each demand d , $\lambda_{s(d)}(d) = \lambda_t(d)$.*

Proof. Just as with Theorem 1, we can consider each subgraph OPT_t separately, and attempt to form stable solutions with respect to a particular destination t . By the same arguments as in the proof of Theorem 1, OPT_t is acyclic and we can assume that every active contract carries an active demand. In order to invoke Theorem 5, all we need is that for all d , $\lambda_{s(d)}(d)$ is larger than the length of $P_{s(d)}$, the longest path from $s(d)$ to t . Consider a demand in OPT_t such that $\lambda_{s(d)}(d) < \|P_{s(d)}\|$. Now create a new solution by reducing the capacity of edges along $P_{s(d)}$ by one. By Lemma 3 all the demands

except d can still be successfully routed in this new solution. This new solution is still acyclic, and so if $\lambda_{s(d')}(d') \geq \|P_{s(d')}\|$ for all active d' , then we can form an equilibrium on this solution. Otherwise, there must be a demand that violates this inequality, and we get rid of this demand by reducing the capacity of edges along $P_{s(d')}$. This process eventually results in a Nash equilibrium, since in the worst case, all capacities are reduced to 0, and we end up with the trivial Nash equilibrium in which no demands or contracts are active.

We now consider the quality of the Nash equilibrium that this process produces. Every time we make a demand d inactive by reducing the capacity of $P_{s(d)}$ and re-routing, the cut-loss objective is increased by $2\lambda_{s(d)}(d)$ (since demand d becomes inactive), and is decreased by $\|P_{s(d)}\|$. The latter is because at every step of the above process, all the capacities are fully saturated, and so decreasing the capacities by one on edges of $P_{s(d)}$ means decreasing the total transit cost by $\|P_{s(d)}\|$. Therefore, since $\lambda_{s(d)}(d) < \|P_{s(d)}\|$ for all demands we made inactive, then at the end of this process we can end up with a solution that is at most twice as expensive as the one we started with. This means exactly that the equilibrium we find is at most twice as expensive as OPT. ■

Theorem 3 *Let E^* be the set of active contracts in OPT, and D_{OPT} be the active demands in OPT. If we increase $\lambda_{s(d)}(d)$ by a factor of 2 for every d , then E^* together with D_{OPT} induce a Nash equilibrium.*

Proof. Just as with Theorem 1, we can consider each subgraph OPT_t separately, and attempt to form stable solutions with respect to a particular destination t . By the same arguments as in the proof of Theorem 1, OPT_t is acyclic and we can assume that every active contract carries an active demand. In order to invoke Theorem 5, all we need is that for all d , $\lambda_{s(d)}(d)$ is larger than the length of $P_{s(d)}$, the longest path from $s(d)$ to t . If this were not the case, then we can reduce the capacity of $P_{s(d)}$ by one to create a new solution. By Lemma 3, all demands except d can be successfully routed in this new solution. This means that the quality of this solution is reduced by $2\lambda_{s(d)}(d)$ (since d is no longer routed), but is improved by $\|P_{s(d)}\|$ because of the reduction in transit cost. We know that in OPT, it must be the case that $2\lambda_{s(d)}(d) \geq \|P_{s(d)}\|$, since otherwise this would create a solution better than OPT. But since $\lambda_{s(d)}(d)$ was increased by a factor of 2, it must be that now $\lambda_{s(d)}(d) \geq \|P_{s(d)}\|$, as desired. ■

D Contracts Without Destination Labels (proof of Theorem 4)

Theorem 4 *If every node v can only affect the route portion that comes after v (for demands that currently pass through v), then the results of Theorems 1, 2, and 3 still hold, even if the contracts do not specify particular destinations.*

We begin by giving an extension to Lemma 1.

Lemma 4. *Let H be a set of contracts together with a routing R , with the property that we cannot reduce the transit cost by simply re-routing, and that all active contracts carry an active demand. Let A be the set of active demands (to all destinations) in H that are routed through node v . No matter how v re-routes these demands, it must be that all $d \in A$ are still active, and their routes still pass through v . In addition, after v re-routes the demands, all the capacities of OPT are still saturated.*

Proof. By our assumption about the routing protocol, demands in A are exactly the demands for which v can change the routes, so the routes of all demands not in A do not change. The contracts and capacities of the edges do not change, since v is only re-routing. Let $\kappa(e)$ be the number of demands in A that include e in the portion of their route in H that comes after v . Then, from v 's point of view, it is routing the demands of A in any way it likes, on a graph with residual capacity $\kappa(e)$ (call this directed graph G_κ , and delete from it the edges with capacity $\kappa(e) = 0$). Since v can only control the portion of the routes that comes after the traffic reaches it, this is the same as v trying to route the demands of A to their destinations, with all of them starting at v . v cannot route more than $\kappa(e)$ demands on edge e , since we assumed that the capacities of the edges before re-routing are tight in H . Moreover, while H may not be acyclic, it must be that G_κ is acyclic, by the same argument as in the proof of Theorem 1

for OPT_t being acyclic. This is true because of our assumption that we cannot reduce the transit cost by simply re-routing, and we would be able to eliminate the transit cost on a cycle in G_κ while still routing all the demands in A .

We will first show that every demand in A is active after re-routing. Suppose to the contrary, that some demand d does not reach its destination after re-routing. Suppose that we have routed all the demands in A from v to their destination that we could. By our assumption about the routing protocol, it is enough to show that there exists a valid path from $s(d)$ to $t(d)$ with free capacity. We know that the portion of the route from $s(d)$ to v was not changed, so there is a subpath to v with free capacity. Now consider the edges of G_κ leaving v after re-routing. Since the capacity in G_κ out of v is greater than the capacity entering v (because d was successfully routed in G_κ starting with v before, and took up one capacity on the edges leaving v), there must be an edge (v, u_1) with free capacity. Now consider u_1 . The capacity of edges leaving u_1 equals exactly the capacity of edges entering it minus the number of demands in A ending at u_1 . Therefore, since there was an edge with free capacity entering u_1 , there must also be one leaving it. We can continue this argument until we get to some $t(d')$, with $d' \in A$ not active after re-routing. This means that we have a route with free capacity from $s(d')$ to $t(d')$, implying that d' must be active, a contradiction. Notice that the above process cannot cycle since G_κ is acyclic. Therefore, all demands of A are still active after re-routing.

To prove the rest of the lemma, notice that all the demands in A must still go through v since v only controls the part of their paths after they reach it. Moreover, all the capacity of H is still saturated after re-routing. Suppose to the contrary that there is an edge $e = (u, w)$ in G_κ with free capacity. Since the capacity of edges leaving w in G_κ is exactly the amount entering w , minus the number of demands of A ending at w , and since the capacities of G_κ are tight, this means that there must be an edge with free capacity leaving w as well. We can continue this argument until we create a cycle in G_κ , which we know is acyclic. But if after v re-routes, there is an edge with free capacity in H , then there must also be an edge with free capacity in G_κ , which we just showed is impossible. ■

Just as with Lemma 2, this lets us easily conclude that the utility of a node v does not change if it only re-routes, without changing any contracts.

We now form the payment scheme on H that forms a Nash equilibrium. Let G_κ^u be the graph defined as in the above proof for node u . In other words, G_κ^u contains the edges on which u can re-route demands. For any node u , let P_u^t be the longest path in G_κ^u from u to t , and let $\|P_u^t\|$ be the number of nodes in it. This is the longest possible path on which u could re-route a demand with destination t .

Now consider a demand path $P(d)$ of demand d in H . Suppose this path consists of contracts e_1, e_2, \dots, e_k , with e_i being a contract between a customer u_{i-1} and provider u_i , so the nodes of $P(d)$ in order are $s(d) = u_0, u_1, u_2, \dots, u_k = t(d)$. We want to make sure that the money paid for contract e_i is enough to pay for any route that u_i could re-route d on, and so we want it to be at least $\|P_{u_i}^{t(d)}\|$. Unfortunately, it is quite possible that $\|P_{u_i}^{t(d)}\|$ is much larger than $\|P_{u_{i-1}}^{t(d)}\|$, since $G_\kappa^{u_i}$ and $G_\kappa^{u_{i-1}}$ can look very different. Therefore, we set the payments on contract e_i to be $\max_{j \geq i} (\|P_{u_j}^{t(d)}\| + j - i)$.

We are now ready to prove the main structural theorem of this section, which is similar to Theorem 5.

Theorem 6. *If H is as in Lemma 4, and for all active d ,*

$$\lambda_{s(d)}(d) \geq \max_{0 \leq j \leq \|P(d)\|} (\|P_{u_j}^{t(d)}\| + j),$$

then H induces a Nash equilibrium.

Proof. Let the contract payments on H be as described above. We must now show that if any node v deviates from its current strategy by re-routing some demands and changing some contracts, then its utility will only decrease. Fix a node v . By the above argument, we already know that v 's utility does not change from re-routing. We must now show that after cutting some adjacent contracts, v 's utility can only decrease.

Suppose v cuts a set of customer contracts S_1 , and a set of provider contracts S_2 . We will show that for every improvement in v 's utility resulting from this action, there is a corresponding decrease. First,

consider why v 's utility might improve. For every provider contract $\chi_e \in S_2$, v gains $c(\chi_e)$ utility, since it no longer has to pay money to its provider, and for every demand d whose route contains either an edge from S_1 or S_2 , v gains 1 unit of utility, since it no longer has to transit this demand. v cannot gain any utility from no longer having to pay penalties, since it is not paying any penalties in H . This is because for a penalty to be paid, there have to be edges with free capacity, and we assumed that capacities are tight in H .

Consider the increase of $c(\chi_e)$ in v 's utility resulting from cutting the contract χ_e . By Lemma 4, there must be some demand d that is routed on χ_e , since all capacities are saturated after re-routing. Let $P(d)$ be the path of this demand. Consider the previous edge χ_f in $P(d)$. There are 3 cases: d originated at v ; $\chi_f \in S_1$; and $\chi_f \notin S_1$. We will deal with each of these cases separately.

If d originated at v , then by destroying the contract χ_e , v makes sure that d is no longer active. This means that v loses $\lambda_{s(d)}(d)$ utility that was received from d being active. We will now show that $\lambda_{s(d)}(d) \geq c(\chi_e) + 1$. By construction, $c(\chi_e)$ is equal to some $\|P_w^{t(d)}\| + i - 1$, where node w is the $(i+1)$ 'st node in $P(d)$. By our assumption, $\lambda_{s(d)}(d) \geq \|P_w^{t(d)}\| + i$, and so $\lambda_{s(d)}(d) \geq c(\chi_e) + 1$. This means that v loses at least as much as it gains by cutting χ_e .

Now consider the case where $\chi_f \notin S_1$. This means that there is a valid path from $s(d)$ to v with free capacity, and so v must pay a penalty to its customer on χ_f . This penalty is of size $\lambda_{s(d)}(d) - 1$. By construction, $c(\chi_e)$ is equal to some $\|P_w^{t(d)}\| + i - j$, where w is the $(i+1)$ 'st node of $P(d)$, and v is the j 'th node of $P(d)$. By our assumption, $\lambda_{s(d)}(d) \geq \|P_w^{t(d)}\| + i$. Since v is at least the second node in $P(d)$, this means that $\lambda_{s(d)}(d) - 1 \geq c(\chi_e) + 1$, and so v loses at least as much as it gains by cutting χ_e .

Finally, consider the case where $\chi_f \in S_1$. That means that χ_f is also cut by v . By construction, $c(\chi_f) \geq c(\chi_e) + 1$, and so v loses at least as much as it gains by cutting χ_e .

We have now argued that v does not gain anything from cutting provider contracts. If v cuts a customer contract χ_f , it can save a unit of transit cost because it no longer has to transit the demand that was routed through χ_f . By our construction of payments, however, $c(\chi_f) \geq 1$, so v does not gain anything from cutting a customer contract either, as it loses the payment on this contract. ■

Now that we have the above structural result, we can use it to show that the results in Section 3 still hold.

Proof of Theorem 4. First we must show that if $\lambda_{t(d)}(d) = 0$ for all demands d , then there exists a Nash equilibrium that is as good as the optimal centralized solution. Since we can assume that in OPT, all contracts carry active demands, and we cannot re-route to decrease transit cost, then we only need to show that the condition in Theorem 6 is satisfied. Consider a demand d that is active in OPT, and suppose that $\lambda_{s(d)}(d) < \|P_w^{t(d)}\| + j$ with w being the $(j+1)$ 'st node of $P(d)$, the path of d in OPT. Now take OPT, and decrease the capacities along $P_w^{t(d)}$ by 1, as well as removing the subpath of $P(d)$ from $s(d)$ to w . By Lemma 5, every demand except d can still be successfully routed in this new solution. Then, by deleting $P_w^{t(d)}$ and the subpath of $P(d)$, the quality of this solution decreases by $\lambda_{s(d)}(d)$, since d is no longer routed. There is no more transit cost that is paid along $P_w^{t(d)}$, however, so the transit cost decreases by $\|P_w^{t(d)}\| + j$. Therefore, if $\|P_w^{t(d)}\| + j > \lambda_{s(d)}(d)$, this would mean that we have created a solution that is better than OPT, a contradiction.

We will now prove that with respect to the cut-loss objective function, the price of stability is at most 2 if destinations are not specified by the contracts. Let $P(d)$ be the path of d in OPT. As before, we only need to prove that the condition in Theorem 6 holds. Suppose that for some active demand d , $\lambda_{s(d)}(d) < \|P_w^{t(d)}\| + j$ with w being the $(j+1)$ 'st node of $P(d)$, the path of d in OPT. Now create a new solution by reducing the capacity of edges along $P_w^{t(d)}$ and the first j edges of $P(d)$ by one. By Lemma 5 all the demands except d can still be successfully routed in this new solution, and so if $\lambda_{s(d')}(\mathcal{d}') \geq \|P_w^{t(d')}\| + j$ for all active d' , then we can form an equilibrium on this solution. Otherwise, there must be a demand that violates this inequality, and we get rid of this demand by reducing the capacity of edges along $P_w^{t(d')}$ as before. This process eventually results in a Nash equilibrium, since in

the worst case, all capacities are reduced to 0, and we end up with the trivial Nash equilibrium in which no demands or contracts are active.

We now consider the quality of the Nash equilibrium that this process produces. Every time we make a demand d inactive by reducing the capacity of a path and re-routing, the cut-loss objective is increased by $2\lambda_{s(d)}(d)$ (since demand d becomes inactive), and is decreased by $\|P_w^{t(d)}\| + j$. The latter is because at every step of the above process, all the capacities are fully saturated, and so decreasing the capacities by one on edges of a path means decreasing the transit cost by the length of this path. Therefore, since $\lambda_{s(d)}(d) < \|P_w^{t(d)}\| + j$ for all the demands we made inactive, then at the end of this process we can end up with a solution that is at most twice as expensive as the one we started with. This means exactly that the equilibrium we find is at most twice as expensive as OPT.

Finally, the proof of Theorem 3 can be extended to the case where the contracts do not specify destinations in the same way. ■

To complete the proofs above, we need the following lemma.

Lemma 5. *Let P be any path in G_κ^u from u to a destination of a demand d (that passes through u). Form a new solution from OPT by reducing the capacity of each edge in P by 1. Then, all the demands except d that are active in OPT can still be successfully routed in this new solution.*

Proof. By a similar argument as in the proof of Lemma 3, all the demands can still be successfully routed in G_κ^u , and so they can be routed in the new OPT solution as well. The only difference in the argument is that while in Lemma 3, all the demands are heading to the same destination, now all the demand paths are instead starting at the same source node u . ■