Interdomain routing and games

Hagay Levin and Michael Schapira and Aviv Zohar

The Hebrew University of Jerusalem

Interdomain Routing and Games

Hagay Levin∗ Michael Schapira† Aviv Zohar‡

Abstract

We present a game-theoretic model that captures many of the intricacies of interdomain routing in today’s Internet. In this model, the strategic agents are source nodes located on a network, who aim to send traffic to a unique destination node. The interaction between the agents is dynamic and complex – asynchronous, sequential, and based on partial information. Best-reply dynamics in this model capture crucial aspects of the only interdomain routing protocol de facto, namely the Border Gateway Protocol (BGP).

We study complexity and incentive-related issues in this model. Our main results are showing that in realistic and well-studied settings, BGP is incentive-compatible. I.e., not only does myopic behaviour of all players converge to a “stable” routing outcome, but no player has motivation to unilaterally deviate from the protocol. Moreover, we show that even coalitions of players of any size cannot improve their routing outcomes by collaborating. Unlike the vast majority of works in mechanism design, our results do not require any monetary transfers (to or by the agents).

∗The Department of Economics, The Hebrew University of Jerusalem, Israel. hagayl@mscc.huji.ac.il.
†The School of Computer Science and Engineering, The Hebrew University of Jerusalem, Israel. mikesch@cs.huji.ac.il. Supported by grants from the Israel Science Foundation and the USA-Israel Bi-national Science Foundation.
‡The School of Computer Science and Engineering, The Hebrew University of Jerusalem, Israel. avivz@cs.huji.ac.il. Supported by a grant from the Israel Science Foundation.
1 Introduction

The Internet is composed of smaller networks called Autonomous Systems (ASes). The task of ensuring connectivity between ASes is called interdomain routing. Since not all ASes are directly connected, packets often have to traverse several ASes. The packets' routes are established via complex interactions between ASes that enable them to express preferences over routes, and are affected by the nature of the network (message delays, malfunctions, etc.). The only interdomain routing protocol de facto is the Border Gateway Protocol (BGP). ASes are owned by selfish, often competing, economic entities (Microsoft, AT&T, etc.), and cannot be expected to adhere to BGP without proper incentives. It is therefore natural to consider interdomain routing from a game-theoretic point of view.

Routing Games. The first contribution of this paper is the presentation of a game-theoretic model of interdomain routing that captures many of its intricacies (e.g., the asynchronous nature of the network). In our model (as in [17, 16]), the network is defined by an undirected graph $G = (N, L)$. The set of nodes $N$ represents the ASes, and consists of $n$ source-nodes $1, ..., n$ (the players), and a unique destination-node $d$. The set of edges $L$ represents physical communication links between the nodes. Each source node $i$ has a valuation function $v_i$ that expresses a full-order of strict preferences over simple routes from $i$ to $d$.

The model consists of two games: At the heart of the model is the sequential, asynchronous, and private-information Convergence Game, which is meant to model interdomain routing dynamics. Best-reply dynamics in the Convergence Game crucial features of BGP dynamics, in which each AS is instructed to continuously execute the following actions:

- Receive update messages from neighbouring nodes announcing their routes to the destination.
- Choose a single neighbouring node, whose route you prefer most (given $v_i$), to send traffic to.
- Announce your new route to all neighbouring nodes.

We also define a One-Round Game, which will function as an analytic tool. The One-Round Game can be regarded as the full-information non-sequential game underlying the Convergence Game. Pure Nash equilibria in the One-Round Game correspond to “stable solutions” in networking literature [17, 16], and are the “sinks” to which best-reply dynamics (BGP) can “converge”.

We study several complexity and strategic problems in this model. Most importantly, we address the issue of incentive-compatibility of best-reply dynamics in the Convergence Game. We provide realistic settings in which the execution of best-reply dynamics (BGP) is in the best-interest of the players (ASes). We also address the following questions: How hard it is to establish whether a pure Nash exists in the One-Round Game (the nonexistence of a pure Nash implies that best-reply dynamics will go on indefinitely)? How hard is it to get good approximations to the optimal social welfare?

Existence of Pure Nash Equilibria. Griffin and Wilfong have shown that determining whether a pure Nash equilibrium in the One-Round Game (stable solution) exists is NP-hard [17]. We prove that this result extends to the communication model.

**Theorem:** Determining whether a pure Nash equilibrium in the One-Round Game exists requires exponential communication (in $n$) between the source-nodes.

BGP Convergence and Incentives. Networking researchers, and others, invested a lot of effort into identifying sufficient conditions for the existence of a stable solution to which BGP always
converges (see, e.g., [16, 27, 14, 13, 15, 5, 26, 4]). The most general condition known to guarantee this is “No Dispute Wheel”, proposed by Griffin Shepherd and Wilfong [16]. No Dispute Wheel guarantees a unique pure Nash in the One-Round Game, and convergence of best-reply dynamics to it in the Convergence Game. No Dispute Wheel allows nodes to have significantly more expressive and realistic preferences than than always preferring shorter routes to longer ones. In particular, a special case of No Dispute Wheel is the celebrated Gao-Rexford setting [14, 13] that is said to depict the commercial structure that underlies the Internet [21] (see Section 2 for an explanation about No Dispute Wheel and interesting special cases).

Feigenbaum, Papadimitriou, Sami, and Shenker [7] initiated an economic, or mechanism design, approach to interdomain routing. While BGP was designed to guarantee connectivity between trusted and obedient parties, in the age of commercial Internet these are no longer valid assumptions (ASes are owned by different economic entities with very different, and often contradicting, commercial interests). Identifying realistic settings in which BGP is incentive-compatible has become the paradigmatic problem in Distributed Algorithmic Mechanism Design [8, 12, 24], and the subject of many works [25, 6, 10, 9, 3, 11, 23, 18].

Recently, a step in this direction was taken in [9, 11]. It was shown that if No Dispute Wheel and an additional condition named Policy Consistency hold then BGP is incentive-compatible in ex-post Nash. Informally, policy consistency means that no two neighbouring nodes disagree over which of any two routes is preferable. This is obviously a very severe restriction that does not necessarily hold in practice. We take a significant step forward by removing it (in particular, we allow the Gao-Rexford setting for which Policy Consistency does not hold).

Unfortunately, we prove that best-reply dynamics are not incentive-compatible if Policy Consistency does not hold. This is true even if No Dispute Wheel holds, and can be shown to hold even in the natural Gao-Rexford commercial setting.

**Theorem:** Best-reply dynamics are not incentive-compatible in ex-post Nash even if the No Dispute Wheel condition holds.

However, there is still hope for BGP. We consider a property called “Route Verification”. Route Verification means that a node can verify whether a route announced by a neighbouring node is indeed available to that neighbouring node (and if not simply ignore that route announcement). Unlike Policy Consistency, Route Verification does not restrict the preferences of ASes, but is achieved by modifying BGP (e.g., this can be achieved via cryptographic signatures). Achieving Route Verification in the Internet is an important agenda in security research\(^1\). Security researchers seek ways to implement Route Verification that are not only theoretically sound, but also reasonable to deploy in the Internet (see [2]).

We note, that even if announcements of non-available routes are prevented by Route Verification, nodes still have many other forms of manipulation available to them: Pretending to have different preferences (“lying”), conveying inconsistent information (e.g., displaying inconsistent preferences over routes), denying routes from neighbours, and more. Hence, it still needs to be shown that Route Verification guarantees immunity of best-reply dynamics (BGP) to all forms of manipulation.

Our main result is the following:

**Theorem:** Best-reply dynamics are incentive-compatible in (subgame-perfect) ex-post Nash if No Dispute Wheel and Route Verification hold.

We stress that this result is achieved without any monetary transfers between nodes (as in [11],

\(^1\)“The US government cites BGP security as part of the national strategy for securing the Internet [Department of Homeland Security 2003]” [2]
and unlike most prior works on interdomain routing, and in mechanism design in general).

Our result highlights an interesting connection between the two current research agendas that
address the problem of disobedience and lack of trust in interdomain routing – security research
and Distributed Algorithmic Mechanism Design. One of the implications of this result is that one
can achieve incentive-compatibility in realistic settings (e.g., networks for which the Gao-Rexford
constraints and Route Verification hold). This should further motivate security research, as it
provides a strong strategic justification for modifications of BGP that guarantee Route Verification
via cryptographic and other means (e.g., Secure BGP [2]).

In [11] the notion of collusion-proofness in ex-post Nash is defined. Informally, collusion-
proofness in ex-post Nash means that a group of agents cannot collaborate to improve the outcome
of any player in the group without strictly harming another player in the group. This means that
the group as a whole has no interest to deviate from a strategy profile (at least one member will
be harmed by doing so). The previous theorem can actually be strengthened to the following one:

**Theorem:** Best-reply dynamics are collusion-proof in (subgame perfect) ex-post Nash if No Dispute
Wheel and Route Verification hold.

In particular, this holds even for the coalition that contains all nodes. This implies that, if No
Dispute Wheel holds, BGP is actually socially just in the following sense (also observed in [3]):

**Corollary:** If No Dispute Wheel holds, the unique Nash equilibrium in the One-Round Game
(to which best-reply dynamics always converge) is Pareto optimal.

**Maximizing Social Welfare.** Finally, we turn our attention to the objective of maximizing
the social-welfare, that has also been studied in the context of interdomain routing (see [10]).
Maximizing the social welfare means finding a tree rooted in \(d\), \(T = R_1, ... , R_n\), in which node \(i\) is
assigned route \(R_i\), such that \(\sum v_i(R_i)\) is maximized. In [9] it is shown that if No Dispute Wheel
and Policy Consistency hold then BGP converges to a stable solution that also maximizes the
social welfare. In contrast, we show that the removal of Policy Consistency can be disastrous in
terms of welfare maximization. We do so by presenting two complementary bounds, one in the
computational complexity model, and one in the communication complexity model.

**Theorem:** Obtaining an approximation of \(O(n^{1-\epsilon})\) to the social welfare is impossible unless \(P = NP\). Obtaining an approximation of \(O(n^{1-\epsilon})\) to the social welfare is impossible unless \(P = ZPP\). This holds for any \(\epsilon > 0\) and even if No Dispute Wheel holds.

**Theorem:** Obtaining an approximation of \(O(n^{1-\epsilon})\) to the social welfare requires exponential com-
munication (in \(n\)). This holds for any \(\epsilon > 0\) and even if No Dispute Wheel holds.

These two bounds actually hold even in the Gao-Rexford setting. These results should be
compared with the previously known lower bound of \(\Omega(n^{1-\epsilon})\) [10] (dependent on \(P \neq ZPP\))
for the case of general valuation functions. They show that even narrow conditions that ensure
existence of pure Nash in the One-Round Game, and convergence of best-reply dynamics in the
CONVERGENCE GAME, might be very far from guaranteeing a good social welfare. A trivial
matching upper bound of \(n\) exists even for general valuation functions.

### 1.1 Organization of the Paper

In Section 2 we present the model and the communication result about pure Nash equilibria. In
Section 3 we present the results regarding incentives and best-reply dynamics in the CONVERGENCE
GAME. In Section 4 we present lower bounds on social-welfare maximization. In Section 5 we discuss the open questions.

2 Routing Games

Here we define the game-theoretic model, and begin exploring the two games it contains.

2.1 Two Routing Games

In our model, the network is defined by an undirected graph $G = (N, L)$. $N$ consists of $n$ source-nodes $1, ..., n$ (the players), and a unique destination-node $d$. Each source node $i$ has a valuation function $v_i$ that assigns a non-negative value to every possible simple route from $i$ to $d$ (i.e., to every simple route in the complete graph over the nodes of $G$). We make the standard assumption [16] that players have strict preferences: For any node $i$, and every two routes $P, Q$ from $i$ to $d$ that do not have the same first link, it holds that $v_i(P) \neq v_i(Q)$. The model consists of two routing games:

2.1.1 A benchmark - the One-Round Game:

The One-Round Game is a full-information game in which a strategy of a node $i$ is a choice of an outgoing edge ($i$’s choice of an AS to forward traffic to). The payoff of node $i$ for a strategy profile is $v_i(R)$ if the strategies induce a route $R$ from $i$ to $d$, and 0 otherwise.

2.1.2 The Convergence Game:

The Convergence Game is a multi-round game with an infinite number of rounds. In each round one or more players (nodes) are chosen to participate by a scheduler. The scheduler models the asynchronous nature of the Internet, and decides which players participate in each round of the game. The schedule chosen must allow every player to play in an infinite number of rounds (the scheduler cannot deny a node from playing indefinitely). In each round of the game, a player $i$ chosen to play can perform the following actions:

- Receive update messages from neighbouring nodes. Each update message announces a simple route from the sending neighbouring node to the destination.
- Choose a single outgoing edge $(i, j) \in L$ (representing a choice of a neighbouring node to forward traffic to), or $\emptyset$ (not to forward traffic at all).
- Announce simple routes (from $i$ to $d$) to $i$’s neighbouring nodes.

The scheduler decides in which round sent route announcements reach their destinations or if they will be dropped. It can arbitrarily delay update messages, but cannot indefinitely prevent update messages of a node from reaching its neighbour (see [16] for a formal model).

A strategy of a player in the Convergence Game specifies his actions in every round in which that player is chosen to participate. Best-reply dynamics is the strategy-profile in which every player continuously performs the following actions: Receive the most recent route announcements from all neighbours. Choose the neighbour with the most preferred simple route to $d$ (according to your $v_i$). Announce this route to all neighbours.

---

2 The reason for this is that in today’s Internet, routes for each destination AS are computed independently.
3 The reason for this is that nodes are not assumed to be familiar with the topology of the network and must be able to express preferences over all routes announced by other nodes.
If from some round onwards $i$’s assigned route is constant then $i$’s payoff is its value for that route. Otherwise, $i$’s payoff is 0. More formally, the payoff of player $i$ is $v_i(R)$ if $R$ is a simple route from $i$ to $d$ and from some round onwards, for every link $(r, s)$ on $R$, $r$ always chooses $s$. Otherwise, $i$’s payoff is 0. (All of our results hold even if when $i$’s route “oscillates” $i$’s payoff is set to be $i$’s value for the highest valued route it is assigned an infinite number of times.)

2.2 Stable Solutions and Pure Nash Equilibria

Pure Nash equilibria in the ONE-ROUND GAME are known in networking literature as stable solutions. It is not hard to verify that each such stable solution forms a tree rooted in $d$. An important requirement from BGP is that it always converge to such a stable solution. However, this is not guaranteed in general, and definitely will not happen if a pure Nash does not exist. Griffin and Wilfong have shown that determining whether a pure Nash equilibrium in the ONE-ROUND GAME exists is NP-hard [17]. We strengthen this result by extending it to the communication model. The use of communication complexity for analyzing uncoupled Nash equilibrium procedures was recently presented in [19]. Our result can be seen as continuing this line of research.

Theorem 2.1 Determining whether a pure Nash equilibrium in the ONE-ROUND GAME exists requires exponential communication (in $n$).

Proof: We shall prove a reduction from the communication SET DISJOINTENESS problem (studied in [1]). In this problem, there are $n$ communication parties. Each party $i$ holds a subset $A_i$ of \{1, ..., $K$\}. The goal is to distinguish between the two following extreme subcases:

- $\bigcap_i A_i \neq \emptyset$
- For every $i \neq j$ $A_i \cap A_j = \emptyset$

It is known [1] that in order to distinguish between these two subcases the parties must exchange $\Omega(K)$ bits. We set $K = 2^n$. The reduction to the problem of determining whether a pure Nash in the ONE-ROUND GAME exists is as follows: Consider a network with $2n + 1$ source nodes and a unique destination node $d$. The set of nodes $N$ consists of 3 disjoint subsets: $n$ sending nodes, a connecting node $c$, and $n$ transit nodes. Each party $i \in [n]$ in the SET DISJOINTENESS problem is associated with a sending node $s_i$.

The transit nodes are divided into $n$ pairs $T_1, ..., T_{n/2}$. Each such pair of nodes $T_r$ contains a specific node we shall call a 0-node and another node we shall call a 1-node. All sending nodes are connected to the connecting node, which, in turn, is connected to both nodes in $T_1$. For every $r = 1, ..., n/2 - 1$, each 0-node in $T_r$ is connected to the 1-node in $T_r$, and to the 0-node in $T_{r+1}$. Similarly, each 1-node in $T_r$ is connected to the 0-node in $T_r$, and to the 1-node in $T_{r+1}$. Both nodes in $T_{n/2}$ are connected to each other and directly to $d$. We divide the sending nodes into groups of size 3, $S_1, ..., S_{n/2}$. For every $S_r$, the 3 nodes in $S_r$ are connected to each other and to $d$. See a description of the network in Figure 1a.

We must now define the valuation functions of the different nodes. Let us start with the transit nodes: A 0-node in $T_r$ $u$ has a value of $\frac{1}{2}$ for any route to $d$ in which the next-hop node after $u$ is the 1-node in $T_r$, and a value of $\frac{1}{4}$ for routes in which the next-hop node after $u$ is the 0-node in $T_{r+1}$ (and very low values for all other routes to $d$, without ties). Similarly, a 1-node in $T_r$ $u$ has a value of $\frac{3}{4}$ for any route to $d$ in which the next-hop node after $u$ is the 0-node in $T_r$, and a value of $\frac{1}{4}$ for routes in which the next-hop node after $u$ is the 1-node in $T_{r+1}$ (and very low values for all others to $d$).
other routes to \( d \), without ties). Both nodes in \( T_{\frac{n}{2}} \) prefer going through each other (a value of \( \frac{1}{2} \)) than directly to \( d \) (a value of \( \frac{1}{4} \)).

Fix a specific triplet of nodes \( S_r \) and let 0, 1, 2 be those nodes. Each node \( i = 0, 1, 2 \) will assign a value of \( \frac{1}{2} \) to the route \((i, i + 1 \pmod{3}, d)\) and \( \frac{1}{4} \) to all other simple routes to \( d \) that only include nodes in \( S_r \). This construction of \( S_r \) is known as BAD GADGET \cite{17}, and appears in Figure 1b. BAD GADGET is an example of a small network in which no pure Nash equilibrium exists.

There are \( 2^{\frac{3n}{2}} \) possible routes from \( c \) to \( d \) that go through the transit nodes and correspond to strings in \( \{0, 1\}^n \). In each such route, for every pair \( T_r \) either the 0-node forwards traffic to the 1-node or vice versa. Fix an arbitrary order on these routes \( R_1, \ldots, R_{2^{\frac{3n}{2}}} \). Each source node \( s_i \) assigns a value close to 1 (no ties) to a route \((s_i, c)R_a\) iff \( a \in A_i \). \( s_i \) assigns values very close 0 to all unmentioned routes (without ties).

The reader can verify that if there is some \( a \in \cap_i A_i \) then assigning every node \( i \) the route \((s_i, c)R_a\) is a pure Nash equilibrium (and, in fact, a unique pure Nash equilibrium). If, on the other hand, for every \( i \neq j \) \( A_i \cap A_j = \emptyset \), then there is no pure Nash (because of the BAD GADGET construction for any triple \( S_r \)).

Hence, we have a network with \( O(n) \) nodes, in which determining whether a Nash equilibrium exists is equivalent to solving the SET DISJOINTNESS problem with \( n \) players (each player \( i \) simulates node \( s_i \) ), each holding a subset of \( 1, \ldots, 2^{\frac{3n}{2}} \). It therefore requires at least \( \Omega(2^{\frac{3n}{2}}) \) bits of communication. This concludes the proof of the theorem.

Remark 2.2 The reader might be bothered by the fact that nodes are required to have preferences over very long routes (linear in \( n \), that is, the number of all ASes in the entire Internet). We note that similar constructions in which the lengths of routes are asymptotically bigger than \( \log n \) also lead to non-polynomial communication lower bounds.

2.3 BGP Convergence and Best-Reply Dynamics

No Dispute Wheel is the widest condition known, to date, to guarantee BGP convergence to a stable solution. In our terms, this translates to convergence of best-reply dynamics in the CONVERGENCE GAME. A Dispute Wheel, defined by Griffin et al. \cite{16}, is an abstract mathematical structure that can be induced by the network topology and the valuation functions. Formally, a dispute wheel is defined as the 3-tuple \((U, R, Q)\) where \( U = (u_0, u_1, \ldots, u_{k-1}) \) is a sequence of \( k \) nodes in the network and \( R = (R_0, R_1, \ldots, R_{k-1}) \), \( Q = (Q_0, Q_1, \ldots, Q_{k-1}) \) are sequences of routes that exist in
Each node \( u_i \) would rather route clockwise through node \( u_{i+1} \) than through the path \( Q_i \).
preferred than the lower route). It is easy to show that the valuation functions and topology of
the network do not induce a Dispute Wheel (because route 2md does not actually exist). Hence,
there is a unique pure Nash in the ONE-ROUND GAME, and best-reply dynamics always converges
to this Nash in the CONVERGENCE GAME. The unique Nash is assigning 12d, 2d, m12d to 1, 2, m,
respectively. However, if m announces to 2 repeatedly that its route is md then it is easy to see
that best-reply dynamics will always assign 1 the direct route to d, 1d, thus enabling m to get its
most preferred route m1d. Hence, deviating from best-reply dynamics is beneficial.

![Figure 3: Best-Reply Dynamics is Not Incentive-Compatible](image)

We note, that it is possible to define business relationships between the nodes in this example
that are consistent with the Gao-Rexford constraints.

However, as we have said before, this situation can be rectified if Route Verification holds.

**Theorem 3.2** If No Dispute Wheel and Route Verification hold, then best-reply dynamics is incentive-
compatible in (subgame perfect) ex-post Nash

**Proof:** Consider a network graph $G = (N, L)$ for which No Dispute Wheel holds. There is a
unique stable solution $T$ to which best-reply dynamics is bound to converge in the CONVERGENCE
GAME. We denote the route of every source node $r$ in $T$ by $T_r$.

Assume, by contradiction, that some manipulating node $r_m$ manages to reach a different out-
come $M$ by unilaterally deviating from the best-reply dynamics, and gains by doing so. We shall
show that this implies the existence of a Dispute Wheel. The proof shall proceed in steps, pointing
out a sequence of routes in the graph that will eventually form a Dispute Wheel.

We define the route $M_r$ to be the route node $r$ believes it is assigned in $M$. That is, it could
be that the manipulator tricked nodes that send traffic through it in $M$ to believe that their traffic
is forwarded along a route not used in practice. We note, that it could be the case that node $r_m$
intentionally causes a protocol divergence that does not affect it, in order to improve its routing
outcome (the choices of other nodes cause remote persistent route oscillations). If this is the case
then the route $M_r$ of a node $r$ that is affected by this divergence, will simply be $r$’s most preferred
route, out of the routes assigned to $r$ in the oscillation.

Since we assumed that $r_m$ gained from its manipulation, we deduce:

\[ v_{r_m}(T) < v_{r_m}(M_r) \]  \hspace{1cm} (1)

Because $r_m$ strictly prefers $M_r$ to $T$, but did not choose it in the routing tree $T$, we must
conclude that the route $M_r$ is not available to $r_m$ in $T$. This means that there must exist some
node $r$ (other than $r_m$) that is on the route $M_r$ that does not have the same route in $M$ as it has
in $T$. Let $r_1$ be the node on the path $M_r$ that is closest to $d$ on $M$, such that $M_{r_1} \neq T_{r_1}$.

By definition, all nodes that follow $r_1$ on the route $M_r$ have exactly the same routes in $T$ and
in $M$. This means that the node $r_1$ could choose route $M_{r_1}$ in $T$. Since it did not choose that route
we must conclude that:

\[ v_{r_1}(M_{r_1}) < v_{r_1}(T_{r_1})^4 \]  

We can now proceed to the next step in the proof. Since \( T_{r_1} \) is preferred by \( r_1 \), and was not chosen by \( r_1 \) in the routing tree \( M \), it must be that it was not an available option. Therefore, there is some node \( r \) on the route \( T_{r_1} \), that is not \( r_1 \), such that \( T_r \neq M_r \). We select \( r_2 \) to be the node \( r \) closest to \( d \) on the path \( T_{r_1} \) for which \( T_r \neq M_r \). As before, all nodes closer to \( d \) than \( r_2 \) on the route \( T_{r_1} \) send traffic along identical routes in both \( T \) and \( M \). Hence, the route \( T_{r_2} \) must be available to \( r_2 \) even in \( M \). The fact that it was not chosen in \( M \) implies that \( r_2 \) prefers \( M_{r_2} \) over it. Thus, we have that:

\[ v_{r_2}(T_{r_2}) < v_{r_2}(M_{r_2}) \]  

We can continue these steps, alternating between the routing trees \( T \) and \( M \) and creating a sequence of nodes as follows:

- \( r_0 = r_m \)

for \( n = 0, 1, 2, \ldots \) we perform the following steps:

- **M step**: Let \( r_{2n+1} \) be the node \( r \) on the route \( M_{r_{2n}} \) such that \( M_r \neq T_r \), and \( r \) is closest among all such nodes to \( d \) on \( M_{r_{2n}} \).

- **T step**: Let \( r_{2n+2} \) be the node \( r \) on the route \( T_{r_{2n+1}} \) such that \( M_r \neq T_r \), and \( r \) is closest among all such nodes to \( d \) on \( T_{r_{2n+1}} \).

Note, that the destination node \( d \) cannot appear in this sequence because the route \( L_d = T_d \) is the empty set. Due to our construction, and to arguments similar to the ones presented before, the preferences over routes are as follows:

\[
\begin{align*}
\text{for } i = 0, 2, 4, \ldots & \quad v_{r_1}(T_{r_i}) < v_{r_1}(M_{r_i}) \quad (4) \\
\text{for } i = 1, 3, 5, \ldots & \quad v_{r_1}(M_{r_i}) < v_{r_1}(T_{r_i}) \quad (5)
\end{align*}
\]

Since there is only a finite number of nodes, at some point a node will appear in this sequence for the second time. We denote the first node that appears two times in the sequence by \( u_0 \). Let \( u_0, \ldots, u_{k-1}, u_0 \) be the subsequence of \( r_0, r_1, \ldots \) that begins in the first appearance of \( u_0 \) and ends in its second appearance. We shall examine two distinct cases.

**CASE I**: The manipulator \( r_m \) does not appear in the subsequence \( u_0, \ldots, u_{k-1}, u_0 \).

**Proposition 3.3** If for all \( i \in \{0, \ldots, k - 1\} \) \( r_m \neq u_i \) (the manipulator is not one of the nodes in the subsequence) then \( k \) must be even.

**Proof**: If \( k \) is odd, then it must be that \( u_{k-1} \) and \( u_0 \) (in its second appearance in the subsequence) were both selected in \( M \) steps, or were both selected in \( T \) steps. However, if this is the case we reach a contradiction as both nodes were supposed to be the node \( r \) closest to \( d \) on a certain route, such that \( T_r \neq M_r \). Since \( u_{k-1} \neq u_0 \) this cannot be.

If \( k \) is even then the subsequence of nodes \( u_0, \ldots, u_{k-1}, u_0 \), along with the \( T_{u_k} \) and \( M_{u_k} \), route, and the preferences over these routes (expressed before) form a dispute wheel (as in Figure 4a).

\[ ^4 \text{The reason that the inequality is strict is that, as defined in the problem definition, equality exists only if the two routes go through the same neighbouring node. This cannot be the case as } M_{r_1} \neq T_{r_1}. \]
CASE II: The manipulator \( r_m \) appears in the subsequence (that is, \( u_0 = r_m \)). We now need to handle two subcases: The subcase in which \( k \) is even and the subcase in which \( k \) is odd. If \( k \) is even then the second appearance of the manipulator \((u_0)\) in the subsequence is due to a \( T \) step. If so, a dispute wheel is formed, as in the example in Figure 4\(^5\).

We are left with the subcase in which \( k \) is odd. In this case the second appearance of \( r_m \) was chosen in an \( M \) step. If so, it must be that \( M_{u_{k-1}} \) (that goes through \( r_m \)) is not used in practice (otherwise, both \( u_{k-1} \) and that the second appearance of \( r_m = u_0 \) was chosen in \( M \) steps, and arguments similar to those of Proposition 3.3 would result in a contradiction). This must be the result of a manipulation by \( r_m \). Let \( L_{r_m} \) be the false route reported by the manipulator to the node that comes before it on \( M_{u_{k-1}} \). Recall that the manipulator can only announce a route \( L_{r_m} \) that exists and is available to it in \( M \). Recall, that the second appearance of the manipulator was chosen due to an \( M \) step. Therefore, all nodes that follow it on \( M_{u_{k-1}} \) (which are the same nodes as in \( L_{r_m} \) are assigned the same routes in \( T \) and \( M \). Hence, \( L_{r_m} \) was available to \( r_m \) in \( T \). It must be that \( v_{r_m}(T_{r_m}) \geq v_{r_m}(M_{r_m}) \) because we assumed that the manipulation performed by \( r_m \) was beneficial to it. We get:

\[
v_{r_m}(L_{r_m}) \leq v_{r_m}(T_{r_m}) \leq v_{r_m}(M_{r_m})
\]  
(6)

Thus, we form a dispute wheel with \( L_{r_m} \) as shown in Figure 4b.

\[\]

(a) An even number of nodes  
(b) An odd number of nodes

Figure 4: The Dispute wheels constructed during the proof of Theorem 3.2.

With a similar (more complex) construction, Theorem 3.2 can be strengthened as follows:

**Theorem 3.4** If No Dispute Wheel and Route Verification hold, then the best-reply dynamics is collusion-proof in (subgame perfect) ex-post Nash.

**Proof:** [Sketch] We shall assume, by contradiction, that a group of manipulators colludes in an interdomain routing instance with no dispute wheel in order to improve their routing outcomes. We define \( T_r \) and \( M_r \) as in the proof of Theorem 3.2. We assume, by contradiction, that all manipulators are not harmed by this manipulation:

\[
\forall v \in \text{Manipulators} \quad v_r(T_r) \leq v_r(M_r)
\]  
(7)

\(^5\)The route \( R \backslash S \) (where \( S \) is a sub-route of \( R \)) is route \( R \) truncated before the beginning of \( S \)
We shall arrive at a contradiction by showing the existence of a dispute-wheel in a similar manner to that demonstrated in the proof of Theorem 3.2.

We begin the construction by selecting one of the manipulators that strictly gained from the collusion. We shall denote this manipulator by \( r_m \) (it must be that \( T_{r_m} \neq M_{r_m} \)). We then construct a sequence of nodes in a way similar to that explained in the proof of Theorem 3.2:

- \( r_0 = r_m \)
- **M step**: For a node \( r_n \) which is a manipulator we define \( r_{n+1} \) to be the node \( r \) on the route \( M_r \), such that \( M_r \neq T_r \), and \( r \) is the closest to \( d \) on \( M_{r_n} \) among all such nodes.
- **T step**: For a node \( r_n \) that is not a manipulator, and was chosen in an \( M \) step, we define \( r_{n+1} \) to be the node \( r \) on the route \( T_r \), such that \( M_r \neq T_r \), and \( r \) is the closest to \( d \) on \( T_{r_n} \) among all such nodes.
- **M step**: For a node \( r_n \) that is not a manipulator, and was chosen in a \( T \) step, we define \( r_{n+1} \) to be the node \( r \) on \( M_{r_n} \), such that \( M_r \neq T_r \), and \( r \) is the closest to \( d \) on \( M_{r_n} \) among all such nodes.

We define a subsequence of nodes \( u_0, ..., u_{k-1}, u_0 \) as in the proof of Theorem 3.2. We now handle two cases. The first case is that no manipulator appears in \( u_0, ..., u_{k-1}, u_0 \). The handling of this case is precisely the same as in the proof of Theorem 3.2 (Case I).

The other case, is that at least one of the nodes in \( u_0, ..., u_{k-1}, u_0 \) is a manipulator. First, we prove the following proposition:

**Proposition 3.5** There is no \( i \in \{0, ..., k-1\} \) such that both \( u_i \) and \( u_{i+1} \) (modulo \( k \)) are manipulators (no two manipulators come one after the other in the subsequence \( u_0, ..., u_{k-1}, u_0 \)).

**Proof:** By contradiction, let \( u_i \) and \( u_{i+1} \) be two consecutive manipulators. \( u_{i+1} \) was chosen in an \( M \) step. \( u_{i+1} \) is therefore the node \( r \) closest to \( d \) on \( M_{u_i} \) such that \( M_r \neq T_r \). Hence, \( M_{u_{i+1}} \) must be available to \( u_{i+1} \) in both \( M \) and \( T \). We know that \( v_{u_{i+1}}(T_{u_{i+1}}) \leq v_{u_{i+1}}(M_{u_{i+1}}) \), as \( u_{i+1} \) is a manipulator. Since \( u_{i+1} \) chose \( T_{u_{i+1}} \) over \( M_{u_{i+1}} \) in \( T \) it must also be that \( v_{u_{i+1}}(T_{u_{i+1}}) \geq v_{u_{i+1}}(M_{u_{i+1}}) \). We conclude that \( v_{u_{i+1}}(T_{u_{i+1}}) = v_{u_{i+1}}(M_{u_{i+1}}) \). However, equality of the values of routes assigned by \( u_{i+1} \) is only possible if \( u_{i+1} \) forwards traffic to the same node in both routes. Since both routes are available in \( T \), this means that \( T_{u_{i+1}} = M_{u_{i+1}} \). This contradicts the reason for which \( u_{i+1} \) was selected (\( M_{u_{i+1}} \neq T_{u_{i+1}} \)).

The handling of the case in which at least one of the nodes in \( u_0, ..., u_{k-1}, u_0 \) is a manipulator, is very similar to CASE II in the proof of Theorem 3.2. The tricky part of the proof arises when a manipulator is selected in an \( M \) step. Due to Proposition 3.5, it must be that the node that precedes this appearance of the manipulator in the subsequence is not a manipulator. Such an event can be handled as the subcase in which \( k \) is odd in CASE II (in the proof of Theorem 3.2). \( \square \)

### 4 Maximizing Social Welfare

We prove that obtaining an approximation ratio better than \( n \) is hard even if No Dispute Wheel holds. In fact, this can be shown even for the Gao-Rexford setting. We present two lower bounds, one in the computational complexity model, and one in the communication complexity model.
**Theorem 4.1** Obtaining an approximation of $O(n^{1-\epsilon})$ to the social welfare is impossible unless $P = NP$. Obtaining an approximation of $O(n^{1-\epsilon})$ to the social welfare is impossible unless $P = ZPP$. This holds for any $\epsilon > 0$ and even in the Gao-Rexford setting.

**Proof:** Our proof will be by reduction from CLIQUE. Assume a graph $G = < V, E >$, we construct a network with $N$ nodes and $L$ links. In this network, $N$ consists of $2|V| + 1$ source-nodes and a unique destination node $d$. The source nodes are divided into 3 disjoint sets: Two sets $N_1,N_2$, such that $|N_1| = |N_2| = |V|$ and a connection node $c$. We associate a node $v(N_1) \in N_1$ and a node $v(N_2) \in N_2$ with every node $v \in V$. All nodes in $N_1$ are connected to the connection node. All nodes in $N_2$ are connected to each other, to the connection node, and to $d$. See Figure 5.

![Figure 5: The network used in the proof of Theorem 4.1](image)

All nodes in $N_2$, and $c$, have valuation functions that assigns a value close to 0 to all routes (no ties). Fix some order $O$ on the nodes in $N_2$. A node $v(N_1) \in N_1$ assigns a value close to 1 (no ties) to a route $R$ iff: $(v(N_1), c)$ is the first link in $R$. The order of appearance of the nodes from $N_2$ in $R$ is consistent with $O$. $V(N_2)$ is on $R$. For every node $u(N_2) \neq v(N_2) \in N_2$ on $R$ there is an edge $(v, u)$ in $G$.

The reader can verify that every clique in $G$ corresponds to a routing tree with a social welfare that equals the size of the clique (assign every node in $N_1$ that is in the clique the route that goes through $c$ and then the all the nodes in $N_2$ that are associated with nodes in the clique). In addition, the social welfare of every routing tree corresponds to a clique in the original graph $G$ (the route from $c$ to $d$ through $N_2$ determines the identity of the clique). The theorem follows from the known inapproximability results for CLIQUE [20].

We note, that this result can be made to hold in the Gao-Rexford setting by defining business relationships as follows: $c$ is a customer of all nodes in $N_1$. For every two nodes $r, s \in N_2$, such that $s$ comes after $r$ in $O$, $s$ is $r$’s customer. $d$ is a customer of all nodes in $N_2$.

**Theorem 4.2** Obtaining an approximation of $O(n^{1-\epsilon})$ to the social welfare requires exponential communication (in $n$). This holds for any $\epsilon > 0$ and even in the Gao-Rexford setting.

**Proof:** The proof is similar to the proof of Theorem 2.1, and is by reduction from SET DISJOINTENESS. There are $n$ parties, and each party $i$ holds a subset $A_i$ of $1, ..., K$. The goal is to distinguish between the two following extreme subcases:

- $\bigcap_i A_i \neq \emptyset$
• For every $i \neq j$ $A_i \cap A_j = \emptyset$

In order to distinguish between these two subcases the parties must exchange $\Omega(K)$ bits. We set $K = 2^n$.

Now, consider a network with $2n+1$ source nodes and a unique destination node $d$. The set of nodes $N$ consists of 3 disjoint subsets: $n$ sending nodes, a connecting node $c$, and $n$ transit nodes. Each party $i \in [n]$ in the Set Disjointeness problem is associated with a sending node $s_i$.

The transit nodes are divided into $\frac{2n}{2}$ pairs $T_1, ..., T_{\frac{2n}{2}}$. All sending nodes are connected to the connecting node, which, in turn, is connected to both nodes in $T_1$. For every $r = 1, ..., \frac{2n}{2} - 1$, each node in $T_r$ is connected to both nodes in $T_{r+1}$. Both nodes in $T_{\frac{2n}{2}}$ are connected directly to $d$. See a description of the network in Figure 6.

![Diagram of the network](image)

**Figure 6:** The network used in the proof of Theorem 4.2

All transit nodes and the connecting node have a value close to 0 (no ties) for all routes. There are $2^{\frac{2n}{2}}$ possible routes from $c$ to $d$ that go through the transit nodes. Fix an arbitrary order on these routes $R_1, ..., R_{2^{\frac{2n}{2}}}$. The valuation function of each $s_i$ assigns a value close to 1 (no ties) to a route $(s_i, c)R_a$ iff $a \in A_i$. It assigns a value close to 0 to all other routes (no ties). The reader can verify that there is a route assignment with a social welfare-value close to $n$ if there is some $a \in \bigcap_i A_i \neq \emptyset$ (assign every node $i$ the route $(s_i, c)R_j$). If, on the other hand, for every $i \neq j$ $A_i \cap A_j = \emptyset$, then any route assignment cannot have a social-welfare value better than $1 + \epsilon$.

Hence, we have a network with $O(n)$ nodes, in which determining whether the social-welfare is $n$ or 1 is equivalent to solving the Set Disjointeness problem with $n$ players (each player $i$ simulates node $s_i$), each holding a subset of $1, ..., 2^{\frac{2n}{2}}$. It therefore requires at least $\Omega(2^{\frac{2n}{2}})$ bits of communication. This concludes the proof of the theorem.

This result too can be made to hold for the Gao-Rexford setting if we define business relationships as follows: For every $i = 2, ..., \frac{n}{2}$, the nodes in $T_i$ are customers of the nodes in $T_{i-1}$. $d$ is a customer of the nodes in $T_{\frac{2n}{2}}$, the nodes in $T_1$ are customers of $c$, and $c$ is a customer of all nodes in $N_1$.

A trivial upper bound of $n$ can be achieved by finding the node with the highest value for some route, assigning that route to that node (thus getting an $n$-approximation), and then assigning routes to all other nodes in a way that forms a tree rooted in $d$.

## 5 Open Questions

- No Dispute Wheel is sufficient, but not necessary, for guaranteeing BGP convergence. Find non-trivial characterizations of conditions that guarantee BGP convergence.
• What more general conditions than No Dispute Wheel and Route Verification guarantee incentive-compatible convergence of best-reply dynamics?

• Can we enforce No Dispute Wheel in an incentive-compatible manner? (We present a special case in which this is achieved via filtering in [22]).

• Identify realistic interdomain routing settings in which one can obtain a good approximation to the social-welfare.

Acknowledgements

We thank Sharon Goldberg and Noam Nisan for significant contributions to this paper. We also thank Shahar Dobzinski, Joan Feigenbaum, Felix Fischer, Aaron Jaggard, Ehud Kalai, Ahuva Mu’alem, Motty Perry, Vijay Ramachandran, Jeff Rosenschein, Rahul Sami, and Scott Shenker for helpful discussions.

References


A The Gao-Rexford Framework

Studies of the commercial Internet [21] suggest two types of business relationships that characterize AS inter-connections: Pairs of neighbouring ASes have either a customer-provider or a peering relationship. Customer ASes pay their provider ASes for connectivity – access to Internet destinations through the provider’s links and advertisement of customer destinations to the rest of the Internet. Peers are ASes that find it mutually advantageous to exchange traffic for free among their respective customers, e.g., to shortcut routes through providers. An AS can be in many different relationships simultaneously: It can be a customer of one or more ASes, a provider to others, and a peer to yet other ASes. These agreements are assumed to be longer-term contracts that are formed because of various factors, e.g., the traffic pattern between two nodes.

In a seminal paper Gao and Rexford [14] suggest constraints on routing policies that are naturally induced by the business relationships between ASes.

No customer-provider cycles: Let $G_{CP}$ be the directed graph with the same set of nodes as $G$ and with a directed edge from every customer to its direct provider. We require that there be no directed cycles in this graph. This requirement has a natural economic justification as it means that no AS is indirectly its own provider.

Prefer customers to peers and peers to providers: customer route is a route in which the next-hop AS (the first AS to which packets are forwarded on that route) is a customer. Provider and peer routes are defined similarly. We require that nodes always prefer customer routes over peer routes, which are in turn preferred to provider routes. This constraint is on the valuation functions of the nodes – it demands every node assign customer routes higher values than peer routes, which should be valued higher than provider routes.

Provide transit services only to customers: Nodes do not always carry transit traffic—traffic that originates and terminates at hosts outside the node. ASes are obligated (by financial agreements) to carry transit traffic to and from their customers. However, ASes do not carry transit traffic between their providers and peers. Therefore, ASes should share only customer routes with their providers and peers but should share all of their routes with their customers. This constraint is on the filtering policy of the nodes – it requires that nodes only export peer and provider routes to their customers (customer routes are exported to all neighbouring nodes).