Atomic Routing Games on Maximum Congestion

Costas Busch and Malik Magdon-Ismail

Rensselaer Polytechnic Institute, Dept. of Computer Science, Troy, NY 12180, USA.
{buschc,magdon}@cs.rpi.edu

Abstract. We study atomic routing games on networks in which players choose a path with the objective of minimizing the *maximum congestion* along the edges of their path. The social cost is the global maximum congestion over all edges in the network. We show that the *price of stability* is 1. The *price of anarchy*, PoA, is determined by topological properties of the network. In particular, $PoA = O(\ell + \log n)$, where ℓ is the length of the longest path in the player strategy sets, and n is the size of the network. Further, $\kappa - 1 \leq PoA \leq c(\kappa^2 + \log^2 n)$, where κ is the length of the longest cycle in the network, and c is a constant.

1 Introduction

A fundamental issue in the management of large scale communication networks is to route the packet traffic so as to optimize the network performance. Our measure of network performance is the worst bottleneck (most used link) in the system. We model network traffic as finite, unsplittable packets (atomic flow) [22, 26], where each packet's path is controlled independently by a selfish player. The Nash equilibrium (NE) is a natural outcome for a game with selfish players – a stable state in which no player can unilaterally improve her situation. In the recent literature, the *price of anarchy* (PoA) [15, 24] and the *price of stability* (PoS) [1, 2] have become prevalent measures of the quality of the equilibria of uncoordinated selfish behavior relative to coordinated optimal behavior. The former quantifies the worst possible outcome with selfish agents, and the latter measures the minimum penalty in performance required to ensure a stable equilibrium outcome.

We study routing games with N players corresponding to N source-destination pairs of nodes on a network G. The strategy set available to each player is a set of edge-simple paths from the player's source to the destination (typically the strategy set consists of all edge-simple paths in G). A *pure* strategy profile is a selection of a single path (strategy) by each player from her respective strategy set. We study pure Nash equilibria. In our context, a pure strategy profile corresponds to a *routing* $\mathbf{p}$, a collection of paths, one for each player. We refer to Nash equilibra in this context as *Nash-routings*. A routing $\mathbf{p}$ causes *congestion* in the network: the congestion C_e on an edge e is the number of paths in $\mathbf{p}$ that use this edge; the congestion C_{p_i} of a path $p_i \in \mathbf{p}$ is the maximum congestion over all edges on the path; the congestion C of the network is the maximum congestion over all edges in the network. The *dilation* D is the maximum path length in $\mathbf{p}$.

Since a packet is to be delivered along each player's path, a natural choice for social cost is the maximum delay incurred by a packet. The packets can be scheduled along the paths in $\mathbf{p}$ with maximum delay $O(C+D)$ [6, 17, 18, 23, 25]. In heavily congested networks, $C \gg D$, and the maximum delay of a packet is governed by the congestion C. Thus, the network congestion is an appropriate social cost – this choice for the social cost is often referred to as the *maximum social cost* [4, 5, 15, 27].

Consider player i with path $p_i \in \mathbf{p}$. It is shown in [3] that player i's packet can be delivered in time $\widetilde{O}(C_{p_i} + |p_i|)$, where $|p_i|$ is the path length (this holds for all players simultaneously). In congested networks, $C_{p_i} \gg |p_i|$, and so it is appropriate to use C_{p_i} as the player cost, along her chosen path. This choice of player cost is typically referred to as the *maximum player cost.* The maximum player cost is is appropriate since this is what governs the delay experienced by that player in a highly congested network [3]. In the literature it is common to use the sum player cost (instead of the maximum) [7, 9, 14, 15, 21, 27]. However, the sum of congestions does not govern the packet delays, since when a packet waits for a particular congested edge to clear of other packets, the other congested edges in its path can be cleared simultaneously. It is the maximum player and social costs that are appropriate metrics for atomic routing games.

1.1 Contributions

We give the first comprehensive analysis of routing games with maximum player and social cost. We study the quality of pure Nash-routings with respect to the price of stability and anarchy.

In our first result, we establish that there exist *optimal* Nash-routings where the social cost (congestion) is equal to the optimal coordinated cost; in other words, $PoS = 1$ (the price of stability expresses the ratio of the optimal social cost in the Nash-routing with the optimal coordinated cost). We also show that any *best response dynamic*, a sequence of best response moves of players, converges to a Nash-routing in a finite amount of time. Thus, we can easily obtain Nash-routings, starting from arbitrary initial routings.

Theorem 1. *For every routing game:*
(i) There is a pure Nash-routing which is optimal ($PoS = 1$).
(ii) Every best response dynamic converges to a Nash-routing in finite time.

We continue by examining the quality of the worst case Nash-routings. The price of anarchy, PoA, expresses the ratio of the social cost in the worst-case Nash-routing to the optimal coordinated cost. We bound the price of anarchy in terms of topological properties of the network. The next result bounds the price of anarchy for arbitrary instances of routing games in terms of the maximum path-lengths in the strategy sets:

Theorem 2. *For any routing game where the strategy sets of the payers have paths with length at most* ℓ, $PoA < 2(\ell + \log n)$.

Theorem 2 gives good bounds for the price of anarchy for networks where it is natural to use paths with short length. For example in the Hypercube and Butterfly [16], if we choose bit-fixing paths, then $\ell = O(\log n)$, which implies that $PoA \leq c \log n$, for some constant c.

Our next result characterizes the worst case Nash-routing in terms of the longest cycle of the network. For a graph G, the *edge-cycle number* $\kappa_e(G)$ is the length of the longest edge simple cycle in G; we will drop the dependence on G when the context is clear.

Theorem 3. *For any undirected graph* G *with edge-cycle number* κ_e*,*
(i) there exists a routing game for which $PoA \geq \kappa_e - 1$*;*
(ii) for any routing game, $PoA \leq c({\kappa_e}^2 + \log^2 n)$*, for some constant* c*.*

Let m denote the number of edges in the network. Since $\kappa_e \leq m$, we have that $PoA \leq c \cdot m^2$. In graphs with Euler cycles, $\kappa_e = m$. Therefore, Theorem 3 implies that $m - 1 \leq PoA \leq c \cdot m^2$ (we use c to represent a generic constant).

The lower bound of Theorem 3 (part i) is obtained by constructing a game instance where the players have their sources and destination on the largest cycle. To prove the upper bound of Theorem 3 (part ii), we use Theorem 2. For 2-connected graphs, every pair of nodes has two edge-disjoint paths connecting them (Menger's theorem [32]), from which we establish that $\ell \leq c \cdot {\kappa_e}^2$. The cycle upper bound follows immediately by using Theorem 2.

If the graph G is not 2-connected, then the relation $\ell \leq c \cdot {\kappa_e}^2$ may not hold. To obtain the result for a general graph G, we decompose G into a tree of 2-connected components. We show that if in G the Nash-routing has network congestion C, then there is some 2-connected component G' which has congestion $C' \approx C$. At the same time the players in G' are in a *partial* Nash-routing, where many of them are locally optimal. A generalization of Theorem 2 to partial Nash-routings, helps to establish the upper-bound of Theorem 3.

1.2 Related Work

General congestion games were introduced and studied in [22, 26]. The application of game theory in computer science, specifically the introduction of the price of anarchy was introduced in [15]. Since then, many models have been studied, categorized by: the topology of the network; the nature of the player and social costs; the nature of the traffic (atomic or splittable); the nature of the strategy sets; the nature of the equilibria studied (pure or mixed). A brief taxonomy of some relevant existing results, according to the kind of flow (atomic or splittable) and equilibria (mixed or pure), and according to the social cost SC and player cost pc (sum or maximum), are shown in the following two tables.

	Atomic Flow	Splittable Flow
Pure	[4, 19, 26], [31]*, **Our Work**	[27–30]
Mixed	[7–9, 11–15, 20, 21, 24]*	[5], [10]*

	Max SC	Sum SC	Other SC	**
Max pc	**Our Work**	–	–	[19]
Sum pc	[4, 5, 27] [7–11, 14, 15, 21, 24]*	[4, 28–30], [13, 31]*	[12, 20]*	[19, 26]

(∗: A Specific network model is used, eg. parallel links, or specific player strategy sets, eg. singleton sets. ∗∗: Results on existence or convergence to equilibrium, as opposed to quality of equilibria).

Typically, the research in the literature has focused on computing upper and lower bounds on the price of anarchy. The vast majority of the work on maximum social cost has been for parallel link networks, with only a few recent results on general topologies [4, 5, 27]. Essentially, all of the work has focused on the sum player cost, which corresponds to the sum of the edge congestions on a path (as opposed to the maximum edge congestion on the path, which we consider here).

The only result which has a brief discussion of the maximum player cost is [19] where the authors focus on parallel link networks, but also give some results for general topologies. In [19], the main content is to establish the existence of pure Nash-routings. We present a systematic study of pure Nash-routings in atomic routing games. Pure equilibria with atomic players and maximum player cost introduces essentially combinatoric conditions for the equilibria, in contrast to infinitelly splittable flow, or mixed equilibria, which can be characterised by Wardrop-type equilibrium conditions.

Outline of Paper. In Section 2 we give some basic definitions. We prove Theorem 1 in Section 3. We continue with the proof of Theorem 2 in Section 4. The lower bound of Theorem 3 is proven in Section 5. In the same section we prove the upper bound of Theorem 3 for 2-connected graphs. We give the general version of the upper bound in Section 6. We conclude in Section 7. Some of the technical proofs have been placed in an appendix.

2 Definitions

An instance $\mathcal{R}$ of a *routing (congestion) game* is a tuple $(\mathbf{N}, G, \{\mathcal{P}_i\}_{i\in\mathbf{N}})$, where $\mathbf{N} = \{1, 2, \dots, N\}$ are the players, $G = (V, E)$ is an undirected connected graph with $|V| = n$, and $\mathcal{P}_i$ is a collection of *edge-simple* paths. Each path in $\mathcal{P}_i$ is a path in G that has the same source $s_i \in V$ and destination $t_i \in V$; each path in $\mathcal{P}_i$ is a pure strategy available to player i. A pure strategy profile $\mathbf{p} = [p_1, p_2, \cdots, p_N]$ is a collection of pure strategies (paths), one for each player, where $p_i \in \mathcal{P}_i$. We refer to a pure strategy profile as a *routing*. On a finite network, a routing game is necessarily a finite game.

For any routing $\mathbf{p}$ and any edge $e \in E$, the *edge-congestion* $C_e(\mathbf{p})$ is the number of paths in $\mathbf{p}$ that use edge e. For any path p, the *path-congestion* $C_p(\mathbf{p})$ is the maximum edge congestion over all edges in p, $C_p(\mathbf{p}) = \max_{e\in p} C_e(\mathbf{p})$. The *network congestion* is the maximum edge-congestion over all edges in E, $C(\mathbf{p}) = \max_{e\in E} C_e(\mathbf{p})$. The *social or global cost* $SC(\mathbf{p})$ is the network congestion, $SC(\mathbf{p}) = C(\mathbf{p})$. The *player or local cost* $pc_i(\mathbf{p})$ for player i is her path-congestion, $pc_i(\mathbf{p}) = C_{p_i}(\mathbf{p})$. When the context is clear, we will drop the dependence on $\mathbf{p}$ and use C_e, C_p, C, SC, pc_i.

We use the standard notation $\mathbf{p}_{-i}$ to refer to the collection of paths $\{p_1, \cdots, p_{i-1}, p_{i+1}, \cdots, p_N\}$, and $(p_i; \mathbf{p}_{-i})$ as an alternative notation for $\mathbf{p}$ which emphasizes the dependence on p_i. Player i is *locally optimal* in routing $\mathbf{p}$ if $pc_i(\mathbf{p}) \leq pc_i(p'_i; \mathbf{p}_{-i})$ for all paths $p'_i \in \mathcal{P}_i$. A routing $\mathbf{p}$ is in a Nash Equilibrium ($\mathbf{p}$ is a *Nash-routing*) if every player is locally optimal. Nash-routings quantify the notion of a stable selfish outcome. A routing $\mathbf{p}^*$ is an optimal pure strategy profile if it has minimum attainable social cost: for any other pure strategy profile $\mathbf{p}$, $SC(\mathbf{p}^*) \leq SC(\mathbf{p})$.

We quantify the quality and diversity of the Nash-routings by the *price of stability* (PoS) and the *price of anarchy* (PoA) (sometimes referred to as the coordination ratio). Let $\mathbf{P}$ denote the set of distinct Nash-routings, and let SC^* denote the social cost of an optimal routing $\mathbf{p}^*$. Then,

$$PoS = \inf_{\mathbf{p} \in \mathbf{P}} \frac{SC(\mathbf{p})}{SC^*}, \qquad PoA = \sup_{\mathbf{p} \in \mathbf{P}} \frac{SC(\mathbf{p})}{SC^*}.$$

3 Existence of Optimal Nash-routings

The goal in this section is to establish Main Theorem 1. For routing $\mathbf{p}$, the *congestion vector* $\mathbf{C}(\mathbf{p}) = [m_0(\mathbf{p}), m_1(\mathbf{p}), m_2(\mathbf{p}), \ldots]$, where each component $m_k(\mathbf{p})$ is the number of edges with congestion k. Note that $\sum_k m_k(\mathbf{p}) = m$, where m is the number of edges in the network. The social cost (network congestion) $SC(\mathbf{p})$ is the maximum k for which $m_k > 0$. We define a lexicographic total order on routings as follows. Let $\mathbf{p}$ and $\mathbf{p}'$ be two routings, with $\mathbf{C}(\mathbf{p}) = [m_0, m_1, m_2, \ldots]$, and $\mathbf{C}(\mathbf{p}') = [m'_0, m'_1, m'_2 \ldots]$. Two routings are equal, written $\mathbf{p} {=_c} \mathbf{p}'$, if and only if $m_k = m'_k$ for all $k \geq 0$; $\mathbf{p} {<_c} \mathbf{p}'$ if and only if there is some k^* such that $m_{k^*} < m'_{k^*}$ and $\forall k > k^*,\ m_k \leq m'_k$.

Let $(\mathbf{N}, G, \{\mathcal{P}_i\}_{i \in \mathbf{N}})$ be an instance of a routing game. Since there are only finitely many routings (as a player's path may use any edge at most once), there exists at least one minimum routing w.r.t. the total order $<_c$. There may be many distinct routings all of which are minimum (and equal to each other). Let $\mathbf{p}^*$ be a minimum routing (which exists); then, for all routings $\mathbf{p}$, $\mathbf{p}^* {\leq_c} \mathbf{p}$. Every minimum routing is optimal; indeed, if $SC(\mathbf{p}) < SC(\mathbf{p}^*)$ for some other routing $\mathbf{p}$, then the maximum k for which $m_k(\mathbf{p}) > 0$ is smaller than the corresponding k for $\mathbf{p}^*$, contradicting the fact that $\mathbf{p}^* {\leq_c} \mathbf{p}$.

Lemma 1. *Every minimum routing (at least one exists) is optimal.*

A greedy move is available to player i if she can obtain a lower path congestion by changing her current path from p_i to p'_i – the greedy move takes the original routing $(p_i; \mathbf{p}_{-i})$ to $(p'_i; \mathbf{p}'_{-i})$ in which p_i is replaced by p'_i.

Lemma 2. *If a greedy move by any player takes* $\mathbf{p}$ *to* $\mathbf{p}'$*, then* $\mathbf{p}' {<_c} \mathbf{p}$.

Thus, a greedy move decreases the number of high congestion edges, by transferring the congestion to lower congestion edges. Since there are only a finite number of routings, every best response dynamic is finite. By Lemma 2, no player can have an available greedy move at a minimum routing, as this would contradict the minimality of the routing. Hence,

Lemma 3. *Every minimum routing is an optimal Nash-routing.*

Hence, $PoS = 1$. Theorem 1 now follows from Lemmas 2 and 3.

4 Path Length Bound on Price of Anarchy

Here, we prove Theorem 2. In order to do so we will use the *edge-expansion process*, that we introduce here. Before we describe this technique we need to give some necessary definitions.

Let $\mathcal{R} = (\mathbf{N}, G, \{\mathcal{P}_i\}_{i\in\mathbf{N}})$ be an instance of a routing game. Let $\mathcal{P} = \bigcup_{i\in\mathbf{N}} \mathcal{P}_i$. The *path-length* of $\mathcal{R}$ is $\ell = \max_{p\in\mathcal{P}} |p|$. A *path-cut* for player i is a set of edges E_i such that every path in $\mathcal{P}_i$ must use at least one of the edges in E_i. The congestion of a path-cut $C(E_i)$ is the minimum congestion of any edge in E_i, $C(E_i) = \min_{e\in E_i} C_e$. If player i is locally optimal with congestion pc_i, then every alternative path for that player must have congestion at least $pc_i - 1$.

Lemma 4. *Let* $\mathbf{p} = [p_1, p_2, \cdots, p_N]$ *be a routing for which player* i *is locally optimal. Then, there is a path-cut* E_i *for player* i *with congestion* $C(E_i) \geq pc_i - 1$.

4.1 Edge-Expansion Process

If only some players are locally optimal in a routing $\mathbf{p}$, then $\mathbf{p}$ is a *partial* Nash-routing (a Nash-routing is a special case of a partial Nash-routing). The edge expansion process applies to any partial Nash-routing.

Suppose routing $\mathbf{p}$ has network congestion C, and suppose that at least one player is locally optimal with player cost C. Let $\mathcal{E}_0$ be the set of edges with congestion $C_0 = C$ that are used by at least one locally optimal player, and let Π_0 be the set of these locally optimal players that use at least one edge in $\mathcal{E}_0$. By Lemma 4, each player in Π_0 has a path-cut with congestion at least $C_0 - 1$. Let $\mathcal{E}_1$ denote the union of $\mathcal{E}_0$ with all these path-cuts of every player in Π_0. Thus, $\mathcal{E}_0 \subseteq \mathcal{E}_1$ and each edge in $\mathcal{E}_1$ has congestion at least $C_1 = C_0 - 1$. Let Π_1 denote the set of locally optimal players whose paths in $\mathbf{p}$ use at least one edge in $\mathcal{E}_1$. Note that $\Pi_0 \subseteq \Pi_1$. Each player in Π_1 has player cost at least C_1, since every edge in $\mathcal{E}_1$ has congestion at least C_1.

We repeat this process as follows. Suppose that for $i \geq 1$, edge set $\mathcal{E}_i$ has been constructed as the union of $\mathcal{E}_{i-1}$ with path cuts for the players in Π_{i-1}, thus every edge in $\mathcal{E}_i$ has congestion at least $C_i = C_{i-1} - 1 = C - i$. We now construct Π_i, the set of locally optimal players whose paths use at least one edge in $\mathcal{E}_i$; every player in Π_i has player cost at least C_i. By Lemma 4, each player in Π_i has a path-cut with congestion $C_i - 1$, and we construct $\mathcal{E}_{i+1}$ to be the union of $\mathcal{E}_i$ with all these path-cuts of the players in Π_i.

Using this inductive construction, we obtain a sequence of edge sets, $\mathcal{E}_0 \subseteq \mathcal{E}_1 \subseteq \mathcal{E}_2, \cdots$, with $C(\mathcal{E}_j) \geq C_j = C - j$, and corresponding to each edge set, a set of locally optimal players $\Pi_0 \subseteq \Pi_1 \subseteq \Pi_2 \cdots$. We continue this inductive construction up to edge set $\mathcal{E}_s$ which is the first set for which $|\mathcal{E}_s| \leq 2|\mathcal{E}_{s-1}|$. We will refer to this process as the *edge-expansion process*.

4.2 Edge-Expansion Properties

Since $|\mathcal{E}_i| \leq \frac{1}{2}n^2$ and each expansion at least doubles the size of the edge set,

Lemma 5. $|\mathcal{E}_s| \geq 2^{s-1}$ *and* $1 \leq s < 2\log n$.

In routing $\mathbf{p}$, let $F(C') \subseteq \mathbf{N}$ denote the set of non-locally optimal players with player cost at least C'. We now establish a relationship between the congestion of a partial Nash-routing and the optimal routing.

Lemma 6. $C < 2\ell \cdot (C^* + F(C - 2\log n)) + 2\log n$.

Proof. From the edge-expansion process, each edge in $\mathcal{E}_{s-1}$ has congestion at least C_{s-1}. Let M be the number of times edges in $\mathcal{E}_{s-1}$ are used by the paths in $\mathbf{p}$. Then, $M > C_{s-1} \cdot |\mathcal{E}_{s-1}|$. By construction, in $\mathbf{p}$, the congestion in each of the edges of $\mathcal{E}_{s-1}$ is caused only by the players in $A = \Pi_{s-1} \cup B$, where $B \subseteq F(C_{s-1})$ contains the non locally optimal players that use edges in $\mathcal{E}_{s-1}$. Since path lengths are at most ℓ, each player in A can use at most ℓ edges in $\mathcal{E}_{s-1}$. Hence, $C_{s-1} \cdot |\mathcal{E}_{s-1}| < M \leq \ell \cdot |A|$. Since, $|A| \leq |\Pi_{s-1}| + |F(C_{s-1})|$, we obtain, $C_{s-1} < \frac{\ell}{|\mathcal{E}_{s-1}|} \cdot (|\Pi_{s-1}| + |F(C_{s-1})|)$. We now bound $|\Pi_{s-1}|$.

$\mathcal{E}_s$ contains a path-cut for every player in Π_{s-1}, and every such player must use at least one edge in $\mathcal{E}_s$ in any routing, including the optimal routing $\mathbf{p}^*$. Thus, edges in $\mathcal{E}_s$ are used at least $|\Pi_{s-1}|$ times, hence some edge is used at least $|\Pi_{s-1}|/|\mathcal{E}_s|$ times, by the pigeonhole principle. Hence, $C^* \geq |\Pi_{s-1}|/|\mathcal{E}_s|$ (note that $|\mathcal{E}_s| > 0$). By the definition of s, $|\mathcal{E}_s| \leq 2|\mathcal{E}_{s-1}|$. Hence, $|\Pi_{s-1}| \leq 2|\mathcal{E}_{s-1}|C^*$, and $C_{s-1} < 2\ell \cdot \left(C^* + \frac{|F(C_{s-1})|}{2|\mathcal{E}_{s-1}|}\right)$. Since $C_{s-1} = C - (s-1)$ and $2|\mathcal{E}_{s-1}| \geq 2^s$ (Lemma 5), we obtain $C < 2\ell \cdot \left(C^* + \frac{|F(C-s+1)|}{2^s}\right) + s - 1$. To conclude, $2^s \geq 2$, and note that $C'' < C'$ implies $F(C') \subseteq F(C'')$, hence $|F(C')|$ is non-increasing in C'. Thus $|F(C - s + 1)| \leq |F(C - 2\log n)|$.

Since in a Nash-routing, $F(C') = 0$, $\forall C' > 0$, by dividing the result of Lemma 6 with C^*, we obtain Theorem 2.

5 Basic Cycle Bounds on Price of Anarchy

Here, we first give the lower bound (part i) of Theorem 3 for the price of anarchy; we then prove the upper bound (part ii) of Theorem 3, for the special case of 2-connected graphs. The next result establishes the lower bound of Theorem 3.

Lemma 7. *For any graph G, there is a routing game with $PoA \geq \kappa_e(G) - 1$.*

Proof. Let $Q = e_1, \ldots, e_{\kappa_e}$ be an edge simple cycle with length κ_e. We construct a routing game with κ_e players, where player i corresponds to edge $e_i = (u_i, v_i)$ in Q, that is, the source of i is $s_i = u_i$ and the destination $t_i = v_i$. The strategy set of i is the collection of all edge simple paths from s_i to t_i.

There are two special paths in the strategy set of player i, the *forward path* which is composed solely of the edge (u_i, v_i), and the *backward path* which consists of the remaining edges of cycle Q. Since Q is edge simple, if every player uses his forward path $C = 1$. Thus, the optimal social cost is 1. If on the other hand, all the players use their backward paths (backward routing $\bar{\mathbf{p}}$), then player i uses every edge in Q except e_i exactly once. Thus, the congestion on every edge in Q is $N - 1 = \kappa_e - 1$. Hence, if $\bar{\mathbf{p}}$ is a Nash-routing, then $PoA \geq \kappa_e - 1$.

We will show that $\bar{\mathbf{p}}$ is a Nash-routing by contradiction. Suppose that some player k is not locally optimal – so player k has lower congestion for some other path p. Since every edge on Q has congestion $\kappa_e - 1$ in routing $\bar{\mathbf{p}}$, at least $\kappa_e - 2$ players other than player k use every edge on Q. Thus, if p uses any edge on Q, then $pc_k(p; \bar{\mathbf{p}}_{-k}) = \kappa_e - 1$, which does not improve its cost, so we conclude that p does not use any edge on Q. Therefore, p has length at least 2 (since $p \neq e_k$ and G is not a multi-graph). Thus, replacing $e_k \in Q$ by p results in a new edge simple cycle Q' that is strictly longer than Q, a contradiction. Thus, $\bar{\mathbf{p}}$ is a Nash-routing.

We now continue with the upper bound on the price of anarchy. A graph G is *k-connected* if its minimum edge-cut has size at least k. By Menger's theorem [32], G is k-connected if and only if there are at least k edge-disjoint paths between every two nodes. Let L be the longest path length in G.

Lemma 8. *If G is 2-connected, then $\kappa_e(G) \geq \sqrt{2L} - \frac{3}{2}$.*

The proof relies on the observation that the longest path p must have at least $\sqrt{L}$ edges in common with the largest cycle q, since otherwise, we would be able to construct a larger cycle by combing pieces of p and q.

Lemma 8 bounds the longest path length in G with respect to $\kappa_e(G)$. Theorem 2 bounds the price of anarchy in terms of the longest path ℓ in the players' strategy sets. Since $\ell \leq L$, we obtain the following result, with proves the upper bound of Theorem 3 for 2-connected graphs:

Lemma 9. *For any routing game on a 2-connected graph G, $PoA \leq c({\kappa_e}^2(G) + \log n)$, for some constant c.*

6 Cycle Upper Bound for General Graphs

We now prove the upper bound (part ii) of Theorem 3 for general graphs. We will bound the price of anarchy with respect to the square of the longest cycle. The main idea behind the result is that any Nash-routing in G can be mapped to a partial Nash-routing on some 2-connected subgraph of G. In this partial Nash-routing, many players are locally optimal, and we can apply Lemma 6 in combination with Lemma 7 to obtain the result.

6.1 Canonical Subgraphs

Consider an arbitrary connected graph $G = (V, E)$. A *subgraph* $G' = (V', E')$ of G contains a subset of the nodes, $V' \subseteq V$, and a subset of the edges $E' \subseteq E$, where each edge in E' is incident with two nodes in V'. We say that G' is an *induced* subgraph by the node set V' if E' contains all the edges in E that are incident with a pair of vertices in V'. We say that two subgraphs are *adjacent* if the intersection of their node sets is non-empty. The *union* of two subgraphs $G' = (V', E')$ and $G'' = (V'', E'')$ is $\widehat{G} = (V' \cup V'', E' \cup E'')$.

We will focus on 2-connected subgraphs. It is easy to verify that G contains a 2-connected subgraph if and only if it is not a tree. A 2-connected subgraph G' is *maximal* if there is no larger 2-connected subgraph $G'' = (V'', E'')$ that contains G', so if G'' is 2-connected, then $E' \not\subset E''$. Let $A_1, \ldots, A_\alpha$ be all the maximal 2-connected subgraphs of G, where $\alpha \geq 1$, and $A_i = (V_{A_i}, E_{A_i})$. Any two subgraphs A_i and A_j, $i \neq j$, are node-disjoint since otherwise their union would be 2-connected, which contradicts their maximality.

Therefore, we can construct from G two subgraphs A and B, where A consists of $A_1, \ldots, A_\alpha$, while B consists of the remaining edges in G: $A = (V_A, E_A)$ and $B = (V_B, E_B)$, where $E_A = \bigcup_{i=1}^{\alpha} E_{A_i}$, $E_B = E - E_A$, and V_A and V_B are the nodes adjacent to the edges in E_A and E_B, respectively. Note that graphs A and B are edge-disjoint, however, they may have common nodes. Subgraph B consists also of one or more disjoint maximal connected components (each containing at least two nodes), which we will denote $B_1, \ldots, B_\beta$. (Graph A consists of connected components $A_1, \ldots, A_\alpha$.) We refer to the A_i as the *type-a canonical subgraphs* of G and the B_i as the *type-b canonical subgraphs* of G. One can show:

Lemma 10. *Every type-b subgraph is a tree. Any pair of type-a and type-b subgraphs can have at most one common node.*

We now define a simple bipartite graph $H = (V_H, E_H)$ that represents the structure of G. In $V_H = \{a_1, \ldots, a_\alpha, b_1, \ldots, b_\beta\}$, the nodes a_i, b_j correspond to the the type-a canonical subgraph A_i and the type-b canonical subgraph B_j respectively. The edge $(a_i, b_j) \in E_H$ if and only if the canonical subgraphs A_i and B_j are adjacent (have a common node). The bipartition for H is $(\mathcal{A}, \mathcal{B})$, where $\mathcal{A} = \{a_1, \ldots, a_\alpha\}$ and $\mathcal{B} = \{b_1, \ldots, b_\beta\}$. The nodes in H inherit the same type as their corresponding canonical subgraph in G. Since G is connected, it follows immediately that H is connected too. Further, we have:

Lemma 11. *Graph H is a tree.*

6.2 Canonical Subpaths

A node in G can belong to at most one type-a subgraph and one type-b subgraph, since no two canonical subgraphs of the same type are adjacent. If a node is a member of one canonical subgraph, then its type is the type of the subgraph. If the node belongs to two canonical subgraphs then it is of type-a (we assign it to the type-a canonical subgraph). An edge belongs to exactly one canonical subgraph and inherits the type of that subgraph.

Let $p = v_1, v_2, \ldots, v_k$, $k > 1$, be an edge-simple path in G. We can write p as a concatenation of subpaths $p = q_1 q_2 \cdots q_k$, where $|q_i| > 0, \forall i$, with the following properties: (i) the subpaths are edge disjoint; (ii) all the nodes of a subpath q_i are in the same subgraph and have the same type, which will also be the type and subgraph of q_i; (iii) the types of the subpaths alternate, i.e. the types of q_i and q_{i+1} are different; (iv) There is no type-a subpath with one node (any type-a subpath with one node can be merged with two adjacent type-b subpaths in the same type-b subgraph). We refer to the q_i as the *canonical subpaths* of p. Note that there is a unique canonical subpath decomposition for path p.

Since type-b subgraphs are trees and graph H is a tree, an arbitrary path in G can form cycles only inside type-a canonical subgraphs (in the respective type-a canonical subpaths). As a consequence, a path from a source node to a destination node follows a unique sequence of type-b edges (the union of all the edges in the type-b subpaths). Thus, we can obtain the following crucial result on paths that connect the same endpoints in G.

Lemma 12. *Any two edge-simple paths from nodes s to t in G use the same sequence of type-b edges.*

6.3 Subgames in Canonical Subgraphs

Consider a routing game $\mathcal{R} = (\mathbf{N}, G, \{\mathcal{P}_i\}_{i\in\mathbf{N}})$ in G. Let $\mathbf{p}$ be a routing with network congestion C. Let $\mathbf{p}^*$ denote an optimal routing for $\mathcal{R}$ with congestion C^*. An immediate consequence of Lemma 12 is that every path in $\mathbf{p}$ uses the same type-b edges as its corresponding path in $\mathbf{p}^*$, hence

Lemma 13. *Any type-b edge e has the same congestion in $\mathbf{p}$ and $\mathbf{p}^*$, i.e. $C_e(\mathbf{p}) = C_e(\mathbf{p}^*) \le C^*$.*

By Lemma 13, all the edges in $\mathbf{p}$ with congestion higher that C^* must occur in type-a subpaths.

Lemma 14. *For path p, if $C_p(\mathbf{p}) > C^*$, then p must have a type-a subpath q with $C_q(\mathbf{p}) = C_p(\mathbf{p})$.*

Suppose now that $\mathbf{p}$ is an arbitrary Nash-routing which has network congestion C. For a type-a subgraph Λ, let $\mathbf{p}_\Lambda = \{p_1, \ldots, p_\gamma\}$ denote the paths in $\mathbf{p}$ that use edges in Λ, and denote the respective users as $\mathbf{N}_\Lambda$, where $|\mathbf{N}_\Lambda| = \gamma$. Let $Q_\Lambda = \{q_1, \ldots, q_\gamma\}$ denote the type-a canonical subpaths of the paths in $\mathbf{p}_\Lambda$ that are in Λ (q_i is a subpath of p_i).

In subgraph Λ, we define a new routing game $\mathcal{R}_\Lambda = (\mathbf{N}_\Lambda, \Lambda, \{\mathcal{P}_i^\Lambda\}_{i\in N_\Lambda})$, where $\mathcal{P}_i^\Lambda$ contains all the type-a subpaths of $\mathcal{P}_i$ that are in Λ and have the same source and destination as q_i. We refer to $\mathcal{R}_\Lambda$ as the *subgame* of $\mathcal{R}$ for subgraph Λ. Q_Λ is a possible routing for $\mathcal{R}_\Lambda$. If q_i is locally optimal for player i in Λ, we say that its corresponding path p_i in G is *satisfied* in subgame $\mathcal{R}_\Lambda$. In other words, if path p_i is satisfied in $\mathcal{R}_\Lambda$, player i does not wish to change the choice q_i in Λ. Every player with high player cost (higher than C^*) must be satisfied in a type-a subgraph, since otherwise it would violate Lemma 14. Thus:

Lemma 15. *If player i has path p_i and $pc_i > C^*$, then player i is satisfied in some subgame $\mathcal{R}_\Lambda$ in a type-a subgraph Λ, and player i has congestion pc_i in Λ.*

6.4 Main Result

Consider routing game $\mathcal{R} = (\mathbf{N}, G, \{\mathcal{P}_i\}_{i \in \mathbf{N}})$ in G and a Nash-routing $\mathbf{p}$ with congestion $C(\mathbf{p}) = C$. Lemma 15, implies that each user is satisfied in some type-a subgraph (not necessarily the same). In any type-a subgraph, the resulting routing in the subgame may be a partial Nash-routing, since some users may not be satisfied in it. We first show that there is a subgraph with high congestion where the number of unsatisfied players is bounded. For a canonical type-a subgraph Λ, let $F_\Lambda(C')$ denote the set of non-locally optimal players in the subgame $\mathcal{R}_\Lambda$ whose congestion in $\mathcal{R}$ is at least C'. We will use C_Λ to denote the congestion in the canonical subgraph Λ. We have:

Lemma 16. *Suppose that $C(\mathbf{p}) > C^* + x(1 + \log n)$ for some $x > 0$. Then, there is a type-a canonical subgraph Λ with congestion $C_\Lambda \geq C - x \log n$ and $|F_\Lambda(C_\Lambda - x)| \leq 2C^*$.*

By combining Lemma 6 and Lemma 16 we obtain the following result which establishes the upper bound of Theorem 3.

Lemma 17. *$PoA \leq c \cdot ({\kappa_e}^2(G) + \log^2 n)$, for some constant c.*

Proof. Let $x = 2\log n$. If $C \leq C^* + x(1 + \log n)$, then there is nothing to prove because $C/C^* \leq 1 + 2\log n(1 + \log n)/C^* \leq c \log^2 n$, for some generic constant c. So, suppose that $C > C^* + x(1 + \log n)$. By Lemma 16, there exists a type-a subgraph Λ such that $C_\Lambda \geq C - 2\log^2 n$ and $|F_\Lambda(C_\Lambda - 2\log n)| \leq 2C^*$. By applying Theorem 6 to the subgame $\mathcal{R}_\Lambda$ we obtain,

$$C_\Lambda < 2\ell \cdot (C_\Lambda^* + F_\Lambda(C_\Lambda - 2\log n')) + 2\log n',$$

where ℓ is the length of the longest edge-simple path in the player strategy sets in $\mathcal{R}_\Lambda$, n' is the number of nodes in Λ and C_Λ^* is the optimal congestion for the subgame $\mathcal{R}_\Lambda$. Note that $n' \leq n$, and the subgame $\mathcal{R}_\Lambda$ cannot have a higher optimal congestion than the full game $\mathcal{R}$, hence $C^* \geq C_\Lambda^*$. Since $|F_\Lambda|$ is monotonically non-increasing ($F_\Lambda(C') \subseteq F_\Lambda(C'')$ for $C'' < C'$), we have that:

$$C - 2\log^2 n < 2\ell \cdot (C^* + F(C_\Lambda - 2\log n)) + 2\log n \leq 2\ell \cdot (C^* + 2C^*) + 2\log n.$$

From Lemma 8, $\ell \leq c{\kappa_e}^2(\Lambda) \leq c{\kappa_e}^2(G)$, and so $C \leq c \cdot ({\kappa_e}^2(G)C^* + \log^2 n)$. After dividing by C^*, we obtain the desired result.

7 Discussion

We believe that the price of anarchy upper bound can be improved. Specifically, we leave open the following conjecture: *for any routing game, $PoA \leq \kappa_e - 1$.*

An interesting future direction is to obtain similar results when the latency functions at each link are more general and not necessarily identical. We conclude by noting that all our results have been stated for paths that are edge-simple. Analogous results (in terms of the node-cycle number) could be obtained for strategy sets containing node-simple paths, and social and player costs defined using node-congestion.

References

1. E. Anshelevich, A. Dasgupta, J. Kleinberg, , E. Tardos, T. Wexler, and T. Roughgarden. The price of stability for network design with fair cost allocation. In *Proc. FOCS*, 2004.
2. E. Anshelevich, A. Dasgupta, E. Tardos, and T. Wexler. Near optimal network design with selfish agents. In *Proc. STOC*, 2003.
3. P. Berenbrink and C. Scheideler. Locally efficient on-line strategies for routing packets along fixed paths. In *10th ACM-SIAM Symposium on Discrete ALgorithms (SODA)*, pages 112–121, 1999.
4. G. Christodoulou and E. Koutsoupias. The price of anarchy of finite congestion games. In *Proc. STOC*, 2005.
5. J. R. Correa, A. S. Schulz, and N. E. Stier Moses. Computational complexity, fairness, and the price of anarchy of the maximum latency problem. In *Proc. 10th Conf. on Integer Programming and Combinatorial Optimization (IPCO)*, 2004.
6. R. Cypher, F. Meyer auf der Heide, C. Scheideler, and B. Vöcking. Universal algorithms for store-and-forward and wormhole routing. In *In Proc. of the 28th ACM Symp. on Theory of Computing*, pages 356–365, 1996.
7. A. Czumaj, P. Krysta, and B. Vöcking. Selfish traffic allocation for server farms. In *Proc. STOC*, 2002.
8. A. Czumaj and B. Vöcking. Tight bounds for worst-case equilibria. In *Proc. SODA*, 2002.
9. D. Fotakis, S. Kontogiannis, E. Koutsoupias, M. Mavronicolas, and P. Spirakis. The structure and complexity of Nash equilibria for a selfish routing game. In *Proc. ICALP*, 2002.
10. D. Fotakis, S. Kontogiannis, and P. Spirakis. Selfish unsplittable flows. In *Proc. ICALP*, 2004.
11. M. Garing, T. Lücking, M. Mavronicolas, and B. Monien. Computing nash equilibria for scheduling on restricted parallel links. In *Proc. STOC*, 2004.
12. M. Garing, T. Lücking, M. Mavronicolas, and B. Monien. The price of anarchy for polynomial social cost. In *Proc. MFCS*, 2004.
13. M. Garing, T. Lücking, M. Mavronicolas, B. Monien, and M. Rode. Nash equilibria in discrete routing games with convex latency functions. In *Proc. ICALP*, 2004.
14. E. Koutsoupias, M. Mavronicolas, and P. Spirakis. Approximate equilibria and ball fusion. In *Proc. SIROCCO*, 2002.
15. E. Koutsoupias and C. Papadimitriou. Worst-case equilibria. In *Proc. STACS*, 1999.
16. F. T. Leighton. *Introduction to Parallel Algorithms and Architectures: Arrays - Trees - Hypercubes*. Morgan Kaufmann, San Mateo, 1992.
17. F. T. Leighton, B. M. Maggs, and S. B. Rao. Packet routing and job-scheduling in $O(congestion + dilation)$ steps. *Combinatorica*, 14:167–186, 1994.
18. T. Leighton, B. Maggs, and A. W. Richa. Fast algorithms for finding O(congestion + dilation) packet routing schedules. *Combinatorica*, 19:375–401, 1999.
19. L. Libman and A. Orda. Atomic resource sharing in noncooperative networks. *Telecomunication Systems*, 17(4):385–409, 2001.
20. T. Lücking, M. Mavronicolas, B. Monien, and M. Rode. A new model for selfish routing. In *Proc. STACS*, 2004.
21. M. Mavronicolas and P. Spirakis. The price of selfish routing. In *Proc. STOC*, 2001.

22. D. Monderer and L. S. Shapely. Potential games. *Games and Economic Behavior*, 1996.
23. R. Ostrovsky and Y. Rabani. Universal $O(\text{congestion}+\text{dilation}+\log^{1+\varepsilon} N)$ local control packet switching algorithms. In *Proceedings of the 29th Annual ACM Symposium on the Theory of Computing*, pages 644–653, New York, May 1997.
24. C. H. Papadimitriou. Algorithms, games and the internet. In *Proc. STOC*, pages 749–753, 2001.
25. Y. Rabani and É. Tardos. Distributed packet switching in arbitrary networks. In *Proceedings of the Twenty-Eighth Annual ACM Symposium on the Theory of Computing*, pages 366–375, Philadelphia, Pennsylvania, 22–24 May 1996.
26. R. W. Rosenthal. A class of games possesing pure-strategy Nash equilibria. *International Journal of Game Theory*, 1973.
27. T. Roughgarden. The maximum latency of selfish routing. In *Proc. SODA*, 2004.
28. T. Roughgarden. Selfish routing with atomic players. In *Proc. SODA*, 2005.
29. T. Roughgarden and Éva Tardos. How bad is selfish routing. *Journal of the ACM*, 49(2):236–259, March 2002.
30. T. Roughgarden and Éva Tardos. Bounding the inefficiency of equilibria in nonatomic congestion games. *Games and Economic Behavior*, 47(2):389–403, 2004.
31. S. Suri, C. D. Tóth, and Y. Zhou. Selfish load balancing and atomic congestion games. In *Proc. SPAA*, 2004.
32. D. B. West. *Introduction to Graph Theory*. Prentice Hall, Upper Saddle River, NJ, U.S.A., 2001.

A Appendix

A.1 Proofs of Section 3

Proof of Lemma 2. Suppose that a greedy move by player i takes $\mathbf{p}$ to $\mathbf{p}'$, and so $C_{p'_i}(\mathbf{p}') < C_{p_i}(\mathbf{p})$. Let $k = C_{p_i}(\mathbf{p})$. Since only player i has changed his path, the only edges with higher congestion in $\mathbf{p}'$ than in $\mathbf{p}$ are edges on the path p'_i. Some edges on p_i decreased in congestion by 1 as a result of the greedy move. In particular, all edges of congestion k on p_i have decreased in congestion by 1, since all edges on p'_i have final congestion less than k. Thus, $m_k(\mathbf{p}') \leq m_k(\mathbf{p}) - 1$, since at least one edge of congestion k dropped in congestion and no new edges reached congestion k. To conclude that $\mathbf{p}' <_c \mathbf{p}$, we note that no edge with congestion greater than k has been affected by the greedy move, hence $m_j(\mathbf{p}') = m_j(\mathbf{p})$ for all $j > k$. ∎

A.2 Proofs of Section 4

Proof of Lemma 4. Since player i is locally optimal, every path in $\mathcal{P}_i$ must have path-congestion at least $pc_i - 1$. Indeed, if not, then there is a path $p'_i \in \mathcal{P}_i$ with path-congestion at most $pc_i - 2$. If player i switches from p_i to p'_i, his cost is at most $pc_i - 1$, which contradicts p_i being locally optimal for i. For every path $p \in \mathcal{P}_i$, let edge $e(p) \in p$ be an edge with maximum congestion on p ($C(e) \geq pc_i - 1$). Let $E_i = \cup_{p \in \mathcal{P}_i} e(p)$. Since E_i contains at least one edge from every path in $\mathcal{P}_i$, it is a path-cut for player i and every edge in E_i has congestion at least $pc_i - 1$. Thus, $C(E_i) \geq pc_i - 1$. ∎

Proof of Lemma 5. If $s = 1$ there is nothing to prove, so assume that $s > 1$. Since $|\mathcal{E}_k| > 2|\mathcal{E}_{k-1}|$ for $k = 1, \ldots, s-1$, $|\mathcal{E}_k| > 2^k |\mathcal{E}_0|$. Since $|\mathcal{E}_0| \geq 1$, we have $|\mathcal{E}_k| > 2^k$. By construction, $\mathcal{E}_{i-1} \subseteq \mathcal{E}_i$, for $1 \leq i \leq s$; thus, $|\mathcal{E}_s| \geq |\mathcal{E}_{s-1}| \geq 2^{s-1}$. Since $|\mathcal{E}_s| \leq |E| < \frac{1}{2}n^2$, $2^{s-1} < \frac{1}{2}n^2$ implying $s < 2 \log n$. ∎

A.3 Proofs of Section 5

Proof of Lemma 8. Let u and v be the respective starting and ending nodes in the longest path in G. Since the min (u,v)-cut has size at least two, by Menger's theorem [32], there is a pair of edge-disjoint paths p_1, p_2 from u to v; let $l_1 \leq l_2$ be the lengths of these paths respectively. Let p be an edge-simple (u,v)-path with length L. Path p can be decomposed into $2k$ path segments as follows,

$$p = \lambda_0 \kappa_1 \lambda_1 \kappa_2 \lambda_2 \cdots \kappa_k \lambda_k,$$

where each κ_i has length at least one and consists only of edges on p_1, and each "excursion" λ_i does not contain any edges on p_1. Since each excursion λ_i connects two (not necessarily distinct) nodes on p_1, it follows that there is an edge simple cycle composed of λ_i together with the segment of p_1 between these two nodes. The length of this cycle is at least $|\lambda_i|$, so we have that $\kappa_e(G) \geq |\lambda_i|$

for all $i \in [0, k]$. Since path p is edge simple, there can be at most l_1+1 excursions (as each of the κ_i must contain distinct edges), i.e. $k \leq l_1$. We now compute the length of p as follows,

$$\begin{aligned} L &= \sum_{i=0}^{k} |\lambda_i| + \sum_{i=1}^{k} |\kappa_i|, \\ &\leq \sum_{i=0}^{k} \kappa_e(G) + \sum_{i=1}^{k} |\kappa_i|, \\ &\leq \kappa_e(G) \cdot (l_1 + 1) + l_1. \end{aligned}$$

Solving for $\kappa_e(G)$, we have that $\kappa_e(G) \geq (L+1)/(l_1+1) - 1$. Since p_1 and p_2 form an edge-simple cycle, $\kappa_e(G) \geq l_1 + l_2 \geq 2l_1$. Combining these inequalities, we have

$$\kappa_e(G) \geq \max\left\{2l_1, \frac{L+1}{l_1+1} - 1\right\}.$$

Suppose that $\kappa_e(G) < \sqrt{2L} - \frac{3}{2}$. Since $\kappa_e(G) \geq 2l_1$, we have $l_1 < \sqrt{L/2} - \frac{3}{4}$. Therefore,

$$\begin{aligned} \kappa_e(G) &\geq \frac{L+1}{l_1+1} - 1, \\ &> \frac{L+1}{\sqrt{L/2} + \frac{1}{4}} - 1, \\ &= \sqrt{2l} - \tfrac{3}{2} + \epsilon, \end{aligned}$$

where $\epsilon = 9/(\sqrt{32L} + 2) > 0$. This contradiction concludes the proof. ∎

A.4 Proofs of Section 6.1

The following result follows from the definition of a 2-connected graph.

Lemma 18. *G is 2-connected if and only if every pair of nodes are on some edge-simple cycle.*

Note that only canonical subgraphs of different types can be adjacent (since both the type-a and type-b subgraphs have a maximality property). We obtain the following basic result.

Lemma 19. *Any edge-simple path p that leaves a type-a subgraph does not re-enter it.*

Proof. To the contrary, suppose that such a path p exists which leaves a type-a subgraph A at node v and re-enters it for the first time at node u (u may be the same node as v). Let p' be the edge-simple subpath of p from v to u. By construction, none of the edges in p' are in A, and the same is true for all nodes in p' except v, u. We will show that the union of A with p' is 2-connected, contradicting the maximality of A. Let $A' = A \cup p$, and let z, w be any two nodes in A'. If both z and w are in A, then they lie on an edge-simple cycle (from Lemma 18, as A is 2-connected). Let q be an edge-simple (v, u)-path in A, possibly of length zero. The cycle $r = q \cup p'$ is an edge-simple cycle in A' that contains any two nodes on r so if $z, w \in p'$, then they are on an edge-simple cycle. Suppose that $z \in A$ and $w \notin A$. There is edge-simple cycle s containing z and u (since $z, u \in A$). Starting at z, this cycle must therefore enter cycle r, leave r and return to z. We can splice r onto s at the first node at which s enters r and the last node at which s leaves r, choosing the splice of r that contains w. In this way, we construct an edge-simple cycle containing z, w. Thus every pair of nodes in A' lies on an edge simple cycle, so A' is 2-connected (Lemma 18).

Proof of Lemma 10. Suppose some B_i is not a tree. Then it contains a 2-connected subgraph, a contradiction since every 2-connected subgraph is a subgraph of some A_i which must be edge disjoint from B_i by construction.

If some B_i has at least two distinct nodes u, v in common with some A_i, then an edge-simple path exists in B_i from u to v as B_i is connected. This path leaves A_i and re-enters it, which contradicts Lemma 19. ■

Proof of Lemma 11. Since H is connected, it suffices to show that H is acyclic. To the contrary, suppose that H contains a node-simple cycle $c_H = h_0, h_1, \ldots, h_k, h_{k+1}$, where $h_{k+1} = h_0$. Note that since H is bipartite, c_H is an even cycle. Any pair of adjacent canonical subgraphs identifies a unique (by Lemma 10) node of intersection. Thus corresponding to c_H is a unique sequence of nodes $v_0, v_1, \ldots, v_k$ in G, where v_i is the node that corresponds to the adjacent canonical subgraphs h_i, h_{i+1}. The nodes v_i are not necessarily all different, however every consecutive pair is in a single canonical component, including the pair (v_k, v_0). Thus, in G, there is an edge-simple path p_i, $i = 0, \ldots, k$, from v_i to v_{i+1} (where $v_{k+1} = v_0$) which is contained in the canonical subgraph h_{k+1}, since the h's are connected. Since the canonical subgraphs form an edge partition of E, the concatenation of these paths, $q = p_0 p_1 \ldots p_k$ is an edge-simple cycle in G. At least one of the h_i is of type-a, hence q must leave a type-a canonical subgraph and re-enter it, which contradicts Lemma 19. ■

A.5 Proofs of Section 6.2

Here we prove Lemma 12 after first giving some elementary properties of canonical subgraph decompositions. Consider the canonical subpath decomposition of a path $p = q_1 \cdots q_k$. Let $S(p) = G_1, \ldots, G_k$ denote the *subgraph sequence* of the respective canonical subgraphs that contain the canonical subpaths. We have,

Lemma 20. *For an edge-simple path p, no type-a canonical subgraph repeats in $S(p)$.*

Proof. Let A be a type-a repeated canonical subgraph. Some other canonical subgraph separates two occurrences of A in $S(p)$. Thus, p must leave and re-enter A, contradicting Lemma 19.

For any edge-simple path p let $r(p)$ denote the *reduced* node-simple path that we obtain from p after removing any cycles. Note that cycles in p exist only in type-a canonical subpaths. Thus, $r(p)$ is similar to p with the difference that some type-a subpaths are removed (and the adjacent type-b subpaths are merged). Therefore, we have,

Lemma 21. *Let p be an edge-simple path with reduced node-simple path $r(p)$, then: (i) no canonical subgraph (type-a or type-b) repeats in $S(r(p))$, and (ii) paths p and $r(p)$ visit the same sequence of type-b edges.*

Proof. (ii) follows immediately because none of the cycles removed contain type-b edges. For (i), we need only consider repeated type-b canonical subgraphs. let B be the first such subgraph that is repeated for the first time. If there is more than one subgraph between these two occurrences of B, then there is an edge-simple cycle in H, which contradicts H being a tree (Lemma 11). Therefore the only possibility is that a single type-a canonical subgraph A occurs between the two occurrences of B in the decomposition, i.e. $S(r(p)) = \cdots BAB \cdots$. Since A and B have exactly one common node (Lemma 10), this means the subpath in A has the same first and last node, and hence A should have been removed, a contradiction.

For subgraph sequence $S(p) = G_1, \ldots, G_k$, let $h_1, \ldots, h_k$ be the nodes in H corresponding to $G_1, \ldots, G_k$. The *projection* $H(p)$ of p into H is the walk in H given by $H(p) = h_1, \ldots, h_k$. Note that $H(p)$ may not be a path, since p may visit the same type-b subgraph more than once. However, from Lemma 21 the projection $H(r(p))$ is a node-simple path. Using projections on H, and the fact that H is a tree (Lemma 11), we can obtain an equivalence result between reduced paths.

Lemma 22. *For any two edge-simple paths p and q from node s to t in G, $S(r(p)) = S(r(q))$.*

Proof. Suppose that $S(r(p)) \neq S(r(q))$, then $H(r(p)) \neq H(r(q))$. Let $H(r(p)) = u_1, \ldots, u_k$ and $H(r(q)) = w_1, \ldots, w_l$, and assume that $k \leq l$. If $l = 1$, then $u_1 \neq w_1$, and s, t are both in u_1, w_1, which means that two nodes are common to two different canonical subgraphs, contradicting Lemma 10. Therefore $l \geq 2$. Since s is a node common to the subgraphs corresponding to u_1 and w_1, either $u_1 = w_1$ or u_1 and w_1 are adjacent in H. Similarly, $u_k = w_l$ or u_k and w_k are adjacent in H, as t is a common node.

Let j be the first index for which $u_j \neq w_j$. First suppose that $1 < j \leq k$, so $u_1 \cdots u_{j-1} = w_1 \cdots w_{j-1}$, and u_{j-1} is adjacent to both u_j, w_j. Now consider the sequence of nodes $w_{j-1} w_j w_{j+1} \cdots w_l u_k u_{k-1} \cdots u_j u_{j-1}$. This is a valid walk (with possibly repeated nodes), since either $u_k = w_l$ or u_k is adjacent to w_l. Further, this is a cycle, since $w_{j-1} = u_{j-1}$. This cycle contains at least 3 different nodes, because $w_{j-1} \neq w_j$ and $u_{j-1} \neq u_j$ (Lemma 21), and $u_j \neq w_j$. This cycle can be reduced to a node simple cycle with at least 3 different nodes, contradicting the fact that H is a tree (Lemma 11).

If $j = 1$, then we construct a cycle $w_1 \cdots w_l u_k \cdots u_1 w_1$, which we know is valid since u_1 is adjacent to w_1. This cycle contains at least 3 different nodes since $u_1 \neq w_2$. If $j = k + 1$ (it must be that $l > k$), then we construct a cycle $w_k \cdots w_l u_k$, which is valid since $u_k = w_k$. This cycle also contains at least 3 different nodes because since $r(q)$ is node simple, it cannot use t to get into w_{k+1}. Hence, $w_{k+1} \neq w_l$ because this would imply that w_{k+1} and u_k have at least two nodes in common. Therefore either $l > k + 1$ or $w_l \neq u_k$.

We are now ready to give the proof of Lemma 12.

Proof of Lemma 12. Let p and q be two edge-simple paths from s to t. By Lemma 21, we only need to show that $r(p)$ and $r(q)$ use the same set of type-b edges. The result follows essentially because each type-b subgraph is a tree (Lemma 10).

From Lemma 22, $S(r(p)) = S(r(q))$. Type-b edges are used only in the type-b subgraphs which appear in the same order in $S(r(p))$ and $S(r(q))$. Consider a type-b subgraph occurring in this subgraph decomposition. Either it is the first, or the last, or type-a subgraphs occur before and after. In all cases the subpath in this type-b subgraph is from the same node u to the same node v in both of the subgraph decompositions. This is because these nodes are either the unique nodes of intersection between the same type-a and type-b subgraph, or they are the source or destination, which are the same for both paths. Since each type-b subgraph is a tree, there is a unique path from u to v, which must be the same in both subpath decompositions. ■

A.6 Proofs of Section 6.3

Proof of Lemma 15. We show that if the claim is false, then path p_i is not locally optimal for player i in $\mathcal{R}$, contradicting the fact that $\mathbf{p}$ is a Nash-routing. Indeed, we know from Lemma 14 that p_i uses type-a canonical subgraphs. If none of these type-a canonical subpaths are not locally optimal for their respective subgames, then they can all be switched in favor of paths with strictly lower congestion than C. This will give a valid path for player i with strictly lower congestion than C, hence p_i is not locally optimal for player i. ■

A.7 Proofs of Section 6.4

Proof of Lemma 16. Let $f_A = |F_A(C_A - x)|$, and suppose that $C > C^* + x(1 + \log n)$. Assume, that every type-a subgraph A with congestion $C(A) \geq C - x \log n$ has $f_A > 2C^*$. we will obtain a contradiction by showing that H has a cycle. Since $\mathbf{p}$ is a Nash-routing, every player with congestion $C > C^*$ is locally optimal in at least one type-a subgame of $\mathcal{R}$ (Lemma 15). Thus, there is at least one type-a canonical subgraph A_1 with $C_{A_1} = C$. We will now root H at the its type-a node a_1 which corresponds to A_1 and define a type-a tree H_a composed only of the type-a nodes in H. The root of H_a is also A_1. By assumption, $f_{A_1} > 2C^*$. Since $C - x > C^* + x \log n$, these f_{A_1} players which are not locally optimal in subgame $\mathcal{R}_{A_1}$ have congestion at least $C - x$ and are locally optimal in some other subgame. Therefore, their paths leave A_1 and enter some other type-a canonical subgraph.

Claim. If K paths leave a type-a canonical subgraph A, they must use at least $\lceil K/C^* \rceil$ distinct edges out of A

Proof: If not, then one of the exit edges (which is a type-b edge) will have congestion greater than C^*, contradicting Lemma 13. □

We now build the rooted tree H_a inductively as follows. The root node is a_1. Suppose that α is a node in H_a corresponding to type-a subgraph A, with the following two properties:

(i) $C_A - x > C^*$;
(ii) $f_A = |F_A(C_A - x)| > 2C^*$.

Then, we define three potential children for α as follows. Since there are $f_A > 2C^*$ players with congestion at least $C_A - x > C^*$ which are not locally optimal in subgame $\mathcal{R}_A$, these f_A players must be locally optimal in some other subgame. Therefore all these f_A paths leave A and proceed to their respective subgames where they are locally optimal with congestion at least $C_A - x$. By Claim A.7, they use at least three distinct type-b edges e_1, e_2, e_3 in leaving A (note that these three edges may be in the same type-b canonical subgraph, but this will not affect the argument). Let p_1, p_2, p_3 be three paths with congestion at least $C_A - x$ that exit A on the edges e_1, e_2, e_3 respectively and continue on to their respective canonical subgraphs A_1, A_2, A_3 in which they are locally optimal. At least two of these subgraphs correspond to nodes that are not the parent (if it exists) of α in H_a; these two nodes are two children $c_1(\alpha)$ and $c_2(\alpha)$ of α in H_a (if more than two of these children are different from the parent, we arbitrarily pick two). The depth of a child is one greater than the depth of its parent (the depth of the root is 0). The next few lemmas give some properties of H_a that will be needed to complete the proof of the theorem.

Claim. H_a is a tree.

Proof: H_a is connected, by construction. Suppose that H_a contains a node simple cycle. By construction, an edge between nodes α_1, α_2 in H_a implies the existence of an edge simple path leaves one type-a subgraph and enters the second. Hence, there is a path that leaves a type-a node and re-enters it. This path can be made edge simple by removing all cycles, which contradicts Lemma 19. □

The nodes in H_a can be viewed as constructed level by level. Each node in H_a that satisfies the two conditions above has exactly two children. Note that a_1 satisfies these two conditions, initiating the construction of H_a. The nodes in H_a which do not satisfy the conditions (i) and (ii) are leaves. Thus all nodes in H_a are either leaves or have two children.

Claim. A node at depth $d \leq \log n$ cannot be a leaf.

Proof: Let α be a node at depth d, corresponding to type-a canonical subgraph A. We show that $C_A \geq C - d \cdot x$ by induction on d. Certainly when $d = 0$, the claim holds since $C_{A_1} = C$. Consider $d > 0$. The parent of A, Par_A, has depth $d-1$, so $C_{Par_A} \geq C - (d-1)x$, by the induction hypothesis. Since $d - 1 \leq \log n$, by assumption $f_{Par_A} > 2C^*$ and by construction of the children in H_a, A is a subgraph in which some player is locally optimal in the subgame $\mathcal{R}_A$ and has congestion at least $C_{Par_A} - x \geq C - (d-1) \cdot x + x$. Therefore, $C_A \geq C - d \cdot x$.

Since $d \leq \log n$, we conclude that $C_A - x \geq C - x(1 + \log n) > C^*$ by assumption in the statement of the theorem. Thus, condition (i) is satisfied for α to have children. Since $C_A \geq C - x \log n$, by assumption $f_A > 2C^*$, hence condition (ii) is satisfied for α to have children. Since both conditions are satisfied, α cannot be a leaf node. □

We are now ready to conclude the proof of the theorem by obtaining a contradiction. Since H_a must have a leaf node, we conclude that the depth of H_a is at least $1 + \log n$. Since every node at depth at most $\log n$ has 2 children, we conclude that H_a has $\sum_{i=1}^{\log n} 2^i$ nodes. Evaluating this sum, we have that H_a contains $2n - 1$ nodes, and since $n > 1$, we have our contradiction since H_a cannot possibly contain more nodes than G. ■