Routing-proofness in Congestion-prone Networks

Ruben Juarez and Michael Wu

Department of Economics, University of Hawaii at Manoa

May 2014

Abstract

We consider the problem of sharing the cost of connecting a large number of agents in a network. The centralized agency elicits the target nodes that agents want to connect and charges agents based on their demands. We look for a cost-sharing mechanism that satisfies three desirable properties: *Efficiency* which charges agents based on the minimum total cost of connecting them in a network, *Stand-alone core stability* which requires charging agents not more than the cost of connecting by themselves directly, and *Limit Routing-proofness* which prevents agents from profitable by reporting as several agents connecting from A to B to C instead of A to B. We show that these three properties are not always compatible for any set of cost functions and demands. However, when these properties are compatible, we propose the *Egalitarian mechanism (EG)* and show that it satisfies them. When the properties are not compatible, we find a rule that meets Stand-alone core stability, Limit routing-proofness and minimizes the budget deficit.

1 Introduction

Network congestion and cost-sharing problem

Congestion is a form of negative externality to the networks of any size. These networks can include but are not limited to road networks, internet networks, telecommunication networks as well as utility power grid networks ([4], [8], [11], [10]). The emergence of congestion can be attributed to the instance in which a large number of selfish agents who each wishes to route traffic via one's preferred links (or edges), without taking into account the actions of other agents in the network ([1]). Such selfish routing behavior generates discrepancy between the socially optimal traffic pattern (minimizing overall delay or cost) and the selfish-routing behavioral outcome (also known as non-cooperative Nash equilibrium). ([2], [17]) suggest that the non-cooperative Nash equilibrium need not in general optimize the social welfare (minimizing overall delay or cost in this context). Indeed, the non-cooperative Nash equilibrium could generate a loss substantially higher than the social optimum. This implies that the discrepancy could cause some unwanted direct or indirect consequences such as network congestion, traffic paralysis as well as social welfare loss.

An earlier work by Roughgarden in [2] investigates and quantifies the efficiency loss of the network performance due to the self-interested agents. The degradation is calculated by the worst-possible ratio between the Nash equilibrium and the social optimal traffic and is called the *Price of anarchy* (POA) ([2]). The closer the POA is to 1, the less efficiency loss it generates between the Nash equilibrium and social optimum. Roughgarden shows that if the cost (latency) functions on the links in the network are linear, then the POA is equal to $\frac{4}{3} \approx 1.333$; if the cost functions are cubic, then the POA is approximately 1.896; and if the cost functions are polynomials of degree $\leq p$ where $p > 0$, then POA is a asymptotically equal to $\theta\left(\frac{p}{\ln p}\right)$ as $p \to \infty$[1]. Finally, if the cost functions are unrestricted to any allowable class of functions (e.g. linear, quadratic, cubic, polynomials, etc), then the POA is unbounded. These results indicate that the efficiency loss can be immensely large and the consequences generated by the selfish routing behavior can be extremely severe in the network.

In order to avoid such consequences, it is therefore significantly important to manage the congestion of a network. In this paper, we consider a network with a set of nodes in which every node interconnects with at least one other node by a link. There is a variable congestion cost associated with each link that depends on the units of demands sent by agents. The cost may represent any maintenance cost or accessing fee of the links. Different links may have different congestion costs. There are a large number of self-interested agents who wish to route their traffic from an arbitrary node to another node using the links in the network. We assume the self-interested agents are interested in routing only via links that will incur the minimal cost or delay. In addition, we also include the possibility for a single or a coalition of agents performing a special type of strategic maneuver (or similarly, routing maneuver). The concept of strategic maneuver was first introduced by Moulin in [11]. In the more relevant context of [3], it says that a coalition of agents in a large network can furtively create several alias and misreport themselves as several agents. Formally, that means an agent may report and route her traffic to the target node as two or more agents in hope of lowering her connection cost. To illustrate such scenario, we suppose there is an agent with initial node i and target node j. Then, she may report herself as two agents first connecting from node $i \leftrightarrow k$, then from node $k \leftrightarrow j$ if she finds such strategic maneuver is less costly than connecting directly from node $i \leftrightarrow j$.

Unlike the *decentralized* networks such as road networks or internet networks, the networks being considered in this paper are operated in a *centralized* manner. We imagine there is a central authority that possesses the power to elicit the demand characteristics from all agents in the networks and is capable of computing and implementing the optimal

[1]See [2] for more POAs of different classes of functions such as the quadratic, M/M/1 Delay functions and M/G/1 Delay functions

traffic pattern at any time. Meanwhile, we imagine there is a centralized agency that will help satisfying the needs demanded by the agents at minimal total delay or cost. Since agents are self-interested, they need not have the incentive to reveal their true routing schemes. The centralized agency is therefore vulnerable to any misreports by individual units, or by a coalition of subgroup of units. Such strategic maneuver behavior will potentially distort the socially optimal traffic pattern (minimizing overall cost or delay) in the sense that links with relatively lower cost are more likely to be congested, which may in turns incurring higher total cost or delay.

Therefore, in order to prevent effectively such strategic maneuvers behavior thereby improving the performance of a network, there is a need to design or propose cost-sharing mechanisms to charge on each agent in the network so that the action of strategic maneuver will make no difference in terms of lowering connection costs. Moulin calls any cost-sharing mechanisms or cost-sharing rules *Routing-proof* if they are invulnerable to such strategic maneuvers ([3]). Technically, it is not as difficult to find a routing-proof mechanism, if routing-proofness is the only constraint that we are aiming to satisfy. For instance, the simple *average cost mechanism (AC)* will do so. By splitting the total cost equally among all agents in the network, each agent is to pay a share of the average cost. As a result, any agent who intends to report as two or more agents will simply have to pay double or more of the average cost. Nevertheless, this mechanism may not be charging every agent or coalition fairly if the average cost exceeds the cost of a subnetwork meeting all connection needs of the group (*stand-alone objections*). This fairness-related constraint is known as *stand-alone core stability* and is the primary design constraint in the *minimum cost spanning tree* problem ([3]). Precisely, we wish to design or propose a cost-sharing mechanism or rule satisfying the following properties ([3]):

- Efficiency: the minimum total cost of connecting all agents to their target nodes;
- Stand Alone core stability: no groups of agents pay more than the cost of a subnetwork meeting all connection needs of the group;
- Routing-proofness: no agents can lower its cost by reporting as several agents along an alternative path connecting her target nodes.

Essentially, this paper may be considered as an extension of the earlier work by Moulin in [3]. The prominent difference between this research paper and that of [3] is the nature of the congestion costs pertained in the networks. In [3], Moulin assumes the cost associated with each link is fixed. That is, the costs are independent of the units of traffic sent by agents, whereas we consider variable congestion costs that depend on the units of traffic being connected in our network. Hence, the former network is *congestion-free* whereas we have a *congestion-prone* network. The adoption of variable congestion costs substantially increases the difficulty for computing the efficiency of a general network. Typically, computing the social optimum (efficiency) of a general network is NP-hard ([5], [6]), unless the cost functions satisfy certain conditions (e.g. linearity or all cost functions are identical in the network ([5])). This is sharply different from [3] where the minimum cost (efficiency) can easily be computed by applying a simple *greedy* algorithm ([7]). In there, Moulin successfully constructed two cost-sharing rules that satisfy the stand-alone core stability and routing-proofness properties. Therefore, the major objective of this paper is to see whether we can also design or propose a cost-sharing rule that is as robust against strategic maneuvers and stand alone objections in our congestion-prone network. This research is of particular significance as networks involving variable congestion costs tend to be more realistic in reflecting the real world situation.

Another difference is drawn from the fact that there are a large number of self-interested agents in the networks and each of them controls only a negligible fraction of the overall traffic. This implies that even if an agent reroutes in the network, then because the amount of traffic each agent controls is infinitesimally small relative to the overall traffic, the costs charged to agents should not be too sensitive and changing significantly. Even if there are more than one agent intended to reroute, a substantially large amount of population is needed to take effect and it is relatively hard to coordinate such large group of agents in reality. In that sense, the literature [3] by Moulin does not seem to fully capture this aspect very well. And that is where our new observation of *Limit Routing-proofness* starts to make sense when we consider the strategic maneuvers are extremely small relative to the overall traffic. Our paper will mainly emphasize the investigation of routing-proofness in a limiting perspective. In particular, the necessary and sufficient conditions for the existence of limit routing-proofness along with the mechanism that we are proposing. Finally towards the end of the paper, we will make a conjecture that the genuine routing-proofness studied in [3] may also be satisfied.

1.1 Examples

In this section, we will introduce some common cost-sharing rules such as the average cost-sharing rule (AC) and the proportional to stand alone cost-sharing rule (η^{pr}). We will then provide a numerical example showing how the stand-alone core stability and routing-proofness are incompatible with each other under AC and η^{pr}. Consider the simple triangular network in Figure 1 with three nodes 1, 2, 3. Assume all cost functions associated with each link are different, strictly convex and strictly increasing. Let $C_1(\theta_1) = 2(\theta_1)^2$; $C_2(\theta_2) = 3(\theta_2)^2$ and $C_3(\theta_3) = 6(\theta_3)^2$. Further suppose that there are $n = 4$ agents in the network with 1 agent wants to send her traffic from node $2 \leftrightarrow 3$; 2 agents want to send from node $1 \leftrightarrow 2$ and 1 agent wants to send from node $1 \leftrightarrow 3$. Assume each agent send one unit of traffic via the links.

The first step is to compute the efficiency of this network[2]. Thus, our objective is to minimize the total cost, in particular w.r.t θ_3 since $C_3(\cdot)$ will likely to incur relatively

[2]Note that we are able to compute the efficiency for very limited amount of simple networks, but it still requires us to check all the feasible combinations of demands.

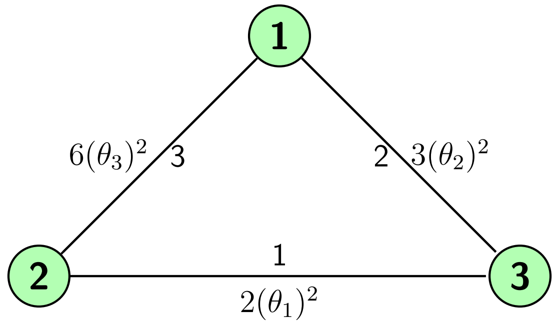


Figure 1: A simple Network with 3 nodes and 3 links

higher cost than $C_1(\cdot)$ and $C_2(\cdot)$. This will lead to the corner solution. Denote by z_3 the units of traffic on link 3 being rerouted to achieve efficiency.

$$\min_{z_3 \geq 0} C(z_3) = \min_{z_3 \geq 0} 6(2 - z_3)^2 + 3(1 + z_3)^2 + 2(1 + z_3)^2$$
$$\frac{\partial C}{\partial z_3} : -12(2 - z_3) + 6(1 + z_3) + 4(1 + z_3) = 0$$
$$10 + 10z_3 = 24 - 12z_3$$
$$z_3^* = \frac{14}{22} = 0.\overline{63}$$

Hence, rerouting $\frac{14}{22} = 0.\overline{63}$ units of z_3 through node $1 \to 2 \to 3$ will minimize the total cost of connecting all agents in the network. Simply substituting $z_3 = \frac{14}{22} = 0.\overline{63}$ into the objective function gives us the efficiency equals to $\frac{270}{11} = \$24.\overline{54}$.

We will now introduce two common cost-sharing rules:

- *Average cost-sharing rule (AC)*: AC divides the minimum cost of connecting all agents in the network equally among all agents ([4]). That is,

$$AC_i = \frac{EFF(\theta_1, ..., \theta_s)}{\sum_{i=1}^{s} \theta_i} \cdot \theta_i$$

In this example, $AC_i = \frac{270}{11(4)} = \$6.1\overline{36}$ for each agent i, that is, agents demanding link 1,2 will be charged at $\$6.1\overline{36}$ and agents demanding link 3 will together be charged $6.1\overline{36}(2) = \$12.26$ under AC. Clearly, AC is routing-proof since if any agent on link 1 poses herself as two agents intended to connect from node $2 \leftrightarrow 1 \leftrightarrow 3$ instead of node $2 \leftrightarrow 3$ directly, then her charge will increase to $2\left(\frac{270}{11(4)}\right) = \$12.\overline{27} > \$6.1\overline{36} = \frac{270}{11(4)}$. This implies that agent i will have to pay a higher cost if she misreports.

However, AC does not satisfy the stand-alone core stability. To see this, note that the stand-alone cost of link 1 is $2(1)^2 = \$2 < \$6.1\overline{36}$. Thus, agent on link 1 is being charged more than she needs to pay to connect herself under AC, which raises the stand-alone objections. Hence, AC satisfies routing-proofness but it fails to meet the stand-alone core stability.

- *Proportional to stand-alone cost-sharing rule (η^{pr})*: η^{pr} divides the minimum cost of connecting all agents in the network in proportion to the stand-alone cost of each link. ([4]).

$$\eta_i^{pr} = \frac{C_i(\theta_i)}{\sum_{i=1}^{s} C_i(\theta_i)} EFF(\theta_1, ..., \theta_s)$$

where $C_i(\theta_i)$ is the stand-alone cost of link i in the network.

η^{pr} divides the minimum cost in proportion to the stand-alone cost of each link and therefore satisfies the stand-alone core stability. The rule charges $\frac{6(2)^2}{(6(2)^2)+3(1)^2+2(1)^2}\left(\frac{270}{11}\right) \approx$ \$20.31 to agents on link 3, \$2.54 to link 2 and \$1.69 to link 1. Nevertheless, η^{pr} is vulnerable to any misreports by agents. Suppose an agent demanding link 3 intends to reroute her traffic from node $1 \leftrightarrow 2$ to node $1 \leftrightarrow 3 \leftrightarrow 2$ because she foresees profitability, then her charge will go from $\frac{6(2)^2}{6(2)^2+3(1)^2+2(1)^2}\left(\frac{270}{11}\right) \approx \$20.31 > \$18.88 = \frac{3(2)^2+2(2)^2}{6(1)^2+3(2)^2+2(2)^2}\left(\frac{270}{11}\right)$. Thus, the agent is able to lower her cost by rerouting her traffic while maintaining her objective of connecting from node $1 \leftrightarrow 2$. Hence, η^{pr} satisfies the stand-alone core stability but fails to meet routing-proofness property.

From the above examples, we can see how the stand-alone core stability and routing-proofness are incompatible with each other under *AC* and η^{pr}. Indeed, more of the traditional cost-sharing rules such as *Shapley-shubik (SS)*, *Serial (SL)* and *Aumann-Shapley (AS)*[3] also do not seem to satisfy simultaneously both stand-alone core stability and routing-proofness. Therefore, we would like to propose a cost-sharing rule that will meet genuinely all three properties in this paper. And the cost-sharing rule that we will be proposing is the *Egalitarian mechanism (EG)*.

1.2 Preliminaries about the Egalitarian mechanism

The Egalitarian mechanism (EG) divides the total cost of connecting every agent in the network equally across all agents, subject to no agent pays more than her stand-alone cost ([4], [16]). EG assigns a vector of cost shares to every agent in a systematic way such that, the sum of the cost charged to all agents, when multiplied by the units of demand is exactly equal to the total cost of the network. Thus, whenever the total cost coincides with the minimum cost of connecting all agents (efficiency), the egalitarian solution, if exists, is said to be efficient. This observation helps to understand the main result of Theorem 1. Indeed, EG is fundamentally similar to that of AC in the sense that both mechanisms divide the

[3] See [16] for more detail information about these traditional cost-sharing rules

cost across agents regardless of their target nodes. However, the cost division of EG is in addition subject to agents' stand-alone costs. This is the primary key feature that AC lacks and therefore fails to satisfy the stand-alone core stability. The inclusion of this constraint in EG ensures the satisfaction of stand-alone core stability since the cost shares charged to each agent is now upper bounded by the stand-alone costs. Moreover, EG satisfies limit routing-proofness under certain conditions. These conditions will be discussed in detail in the later section. Given the pessimistic results that most of the traditional cost-sharing rules: AC, η^{pr}, SL, SS and AS fail to satisfy the three properties simultaneously, EG does shed some light on this particular type of network cost-sharing problem.

Unfortunately, we must also submit to the fact that sometimes we might not have a solution to this type of network cost-sharing problem. Therefore, this paper is just to propose one solution and the necessary and sufficient conditions for which the egalitarian solution exists. We will also be considering the possible budget deficit as well as the maximal amount that we can achieve (collect) if no such solution exists to the problem[4].

2 Related Literature

The stand-alone core stability and efficiency are two traditional properties in the network synthesis problems ([3], [10], [14], [15]); while routing-proofness is a relatively novel idea in the network cost-sharing literature ([3], [12], [11]). The predecessor of routing-proofness was first coined the axiom of no transit originated by Hanriet and Moulin in [11] ([12]). In [11], Hanriet and Moulin suggest two cost-allocation methods: private-cost and external cost methods that satisfy the axiom of additivity, sustainability and no transit.

In the network cost-sharing problem described in [3], Moulin formally renames the no transit axiom as routing-proofness. He then constructs two core stable and routing-proof cost-sharing rules: weighted Shapley value (wsh) and weighted spanning rule (wsp)[5] together satisfying the stand-alone core stability and routing-proofness. The first family of the rules requires only that the connecting costs to be 0 or 1, whereas the second family allows for arbitrary costs but requires a spanning traffic. In terms of computational tractability, wsh is not computable in polynomial time whereas wsp is of polynomial complexity of the size of the set of agents.

Another related literature is a real world application of routing-proofness by Dong, Guo and Wang. In [12], a highway toll pricing method is proposed to assign the toll charges to different types of agents. The proposed method satisfies the stand-alone test for vehicles, cost recovery (for road sections with no cross-subsidizing) and routing-proofness in the sense that no agents can reduce the toll charges by exiting and re-entering the highway

[4]Note that sometimes we will use "prices" interchangeably with "charges" as well as "revenue targets" and "costs" throughout the paper depending on the context of the sections, but they are literarily meaning the same thing

[5]See [3] for more detail information about *wsp* and *wsh*.

consecutively ([12]). The model presented in [12] is in contrast a congestion-prone network rather than congestion-free.

The literature [4] by Juarez and Kumar discusses the appropriateness on the implementation of various efficient cost-sharing mechanisms. Due to the nature of a particular mechanism, agents are choosing their paths strategically as if they are in a game ([4]). From the perspective of a social planner, which mechanism to implement to achieve the desirable goals is not as obvious. On the other hand, the nature of those efficient mechanisms could provide a misleading incentive to the agents who wish to choose an efficient graph at the equilibrium ([4]). Juarez and Kumar compare different class of mechanisms such as the average cost mechanism (AC), proportional to stand alone mechanism (η^{pr}), (weighted) egalitarian mechanism (EG) and Shapley mechanism (Sh), etc. They also characterize the mechanisms that use the minimal information and introduce two new properties of implementations: Pareto Nash implementation (PNI) and weakly Pareto Nash implementation (WPNI).

3 The model

A network cost-sharing problem is a triple $\boldsymbol{P} = \langle N, C, \theta \rangle$, where $N = (V, L)$ is a network with finite set of nodes $V = \{n_1, ..., n_r\}$ such that n_j represents node j and a finite set of links $L = \{l_1, ..., l_s | s \geq r\}$ with generic element i representing an undirected link. Let $C = (C_1(\cdot), ..., C_s(\cdot))$ be the vector of strictly convex and strictly increasing cost functions associated with each link i. Without loss of generality, we assume $C_1(\theta_1) < C_2(\theta_2) \cdots < C_s(\theta_s)$. In addition, let $\overline{AC} = (AC_1(\cdot), ..., AC_s(\cdot))$ be the vector of average costs. Denote by $\theta = (\theta_1, ..., \theta_s)$ the vector of traffic demand profiles with each θ_i representing the units of traffic routed via link i. Let $\mathcal{P}_i = \{p_1, ..., p_t\}$ be the set of all feasible paths (without cycles)[6] that alternatively connect link i in the network. We also denote the set of all problems by $\mathscr{P}$.

Definition 1 *(a) Given any link $i \in L$, a connection for demand θ_i is feasible if agents can split their traffic demands to the alternative paths $p_k \in \mathcal{P}_i$ connecting to their target nodes.*
(b) Given demands $(\theta_1, ..., \theta_s)$ with θ_i on each link i, a connection vector $(w_1, .., w_s)$ is ***feasible*** *if it is the sum of the feasible connections for each link. That is, there exists a feasible connection vector $(w_1^i, ..., w_s^i)$ for every link $i \in \{1, ..., s\}$ such that $(w_1, ..., w_s) = \sum_{i=1}^{s}(w_1^i, ..., w_s^i)$.*

The demands of traffic θ_i on link i can be split into multiple alternative paths[7] where each particular path indirectly connects link i. A connection vector $(w_1, ..., w_s)$ is said to

[6] A cycle consists of a sequence of consecutive and adjacent nodes and links where a path in a cycle starts and ends at the same node

[7] A path consists of a set of consecutive links that connect from node n_i to n_j

be feasible if each of the element w_i sums up the traffic of each feasible connection of link i.

Definition 2 *The total cost of a feasible connection vector* $(w_1, ..., w_s)$ *is* $C(w_1, ..., w_s) = \sum_{i=1}^{s} C_i(w_i)$.

With each element w_i represents all possible amount of traffic on link i, the total cost of connecting all agents with connection vector $(w_1, ..., w_s)$ is the sum of each cost function $C_i(\cdot)$ evaluated at w_i for all $i = \{1, ..., s\}$.

Definition 3 *Given a set of demands* $(\theta_1, ..., \theta_s)$*, a feasible connection vector* $(w_1^*, ..., w_s^*)$ *is* ***efficient*** *if for any other feasible connection vector* $(\widetilde{w_1}, ..., \widetilde{w_s})$*, we have* $C(w_1^*, ..., w_s^*) \leq C(\widetilde{w_1}, ..., \widetilde{w_s})$.

We say that a feasible connection vector of a given set of demands is efficient if there exists a feasible connection vector $(w_1^*, ..., w_s^*)$ such that the total cost evaluated at $(w_1^*, ..., w_s^*)$ is always less than or equal to the total cost evaluated any other connection vectors $(\widetilde{w_1}, ..., \widetilde{w_s})$. Such connection vector will generate the least cost of serving all the agents in the network.

Definition 4 *(a) Given a problem* $\boldsymbol{P} \in \mathscr{P}$*,* ***a cost-sharing mechanism****, or a* ***cost-sharing rule*** *is a continuous function* $\xi : \mathscr{P} \rightarrow \mathbb{R}_+^s$ *allocates a set of cost shares to every agent* i.

(b) An ***efficient cost-sharing mechanism*** *allocates the efficient cost such that*

$$\sum_{i \in L} \xi_i(\boldsymbol{P}) \cdot \theta_i = C^*(\boldsymbol{P})$$

where $C^*(\boldsymbol{P})$ *is the efficient cost at a problem* $\boldsymbol{P}$.

Given an efficient connecting allocation and an efficient cost, an efficient cost-sharing mechanism ξ allocates a set of cost shares to every agent in the network such that, the sum of the charges assigned by the mechanism, when multiplied by the demands of each link is equal to the the efficient cost at a problem $\boldsymbol{P}$.

Definition 5 *A cost-charing mechanism* ξ *satisfies the* ***stand-alone core stability*** *if for any link* i*, we have that* $\xi_i(\boldsymbol{P})\theta_i \leq C_i(\theta_i)$.

Definition 5 ensures the absence of stand alone objections to agents caused by over-charging from the implemented mechanism (e.g. the AC mechanism). This constraint plays an important role in terms of fairness to prevent agents from being charged more than the cost of the links they actually demanded. Moreover, it also helps to prevent agents who demand less units from subsidizing those who demand more units in a network.

Prior to defining routing-proofness, consider two problems: $\boldsymbol{P} = \langle N, C, \theta \rangle$, given path p_k and $0 \leq z \leq \theta_i$, define $\widetilde{\boldsymbol{P}}(p_k, z) = \langle N, C, \widetilde{\theta} \rangle$ where the traffic profiles θ and $\widetilde{\theta}$ are such that, for any $0 \leq z \leq \theta_i$ units of traffic, $p_k \in \mathcal{P}_i$ for all k, let

$$\widetilde{\theta}_l = \begin{cases} \theta_l & \text{if } l \notin \{p_k \cup i\}, \\ \theta_l + z & \text{if } l \in \{p_k\}, \\ \theta_l - z & \text{if } l = i. \end{cases}$$

then θ and $\widetilde{\theta}$ are the vectors of truthful and misreported demands, respectively. Hence, if $0 \leq z \leq \theta_i$ units of traffic are rerouted from link i to any feasible alternative paths p_k by agent i in seek of lowering connecting cost, we call such $\widetilde{\theta}$ *routing* or *strategic maneuvers* at θ. Now we can define routing-proofness formally as in the following:

Definition 6 *(a) For any traffic profile $(\theta_1, ..., \theta_s)$, a cost-sharing mechanism ξ is* ***routing-proof (RP)***[8] *if for any $z \leq \theta_i$, we have*

$$\xi_i(\boldsymbol{P})z \leq \sum_{j \in p_k} z\xi_j(\widetilde{\boldsymbol{P}}(p_k, z)) \tag{1}$$

(b) A cost-sharing mechanism is ***limit routing-proof (LRP)*** *if we have*

$$\xi_i(\boldsymbol{P}) \leq \sum_{j \in p_k} \xi_j(\boldsymbol{P}) \tag{2}$$

Routing-proofness prevents agents with positive mass to route their demands via an alternative path. Indeed, the left hand side of (1) is the total charge to an agent for owning z units of traffic on link i under ξ. If this agent reroutes her demand of traffic via a feasible alternative path p_k, then her charge is exactly equal to the sum of the charges on the alternative path multiplied by z units of traffic with the changing prices. A routing-proof cost-sharing mechanism ξ ensures that the charge allocated to this agent with routing-maneuver will cost no less than if she does not reroute.

On the other hand, limit routing-proofness only focuses in the case where agents have zero mass (i.e. as $z \to 0$), therefore any rerouting by agents will not change the price or have close to no influence on the prices that charged to the agents.

Note that any mechanism that is routing-proof is also limit routing-proof. However, the converse is not true. For instance, if a mechanism is routing-proof, then as $z \to 0$, $\widetilde{\boldsymbol{P}}(p_k, z) \to \boldsymbol{P}$ and $\sum_{j \in p_k} \xi_j(\widetilde{\boldsymbol{P}}(p_k, z)) \to \sum_{j \in p_k} \xi_j(\boldsymbol{P}) \Rightarrow \xi_i(\boldsymbol{P}) \leq \sum_{j \in p_k} \xi_j(\boldsymbol{P})$. Otherwise,

[8] Note that a more general version of routing-proofness requires that $\xi_i(\boldsymbol{P})\gamma_i \leq \xi_i(\widetilde{\boldsymbol{P}})(\gamma_i - z) + \sum_{j \in p_k} z\xi_j(\widetilde{\boldsymbol{P}})$ for any $\gamma_i \leq \theta_i$ and $z \leq \gamma_i$. Under this interpretation, we allow agents to route only a portion of their traffic, whereas the Cournot version considers routing the entire traffic of an agent. Unfortunately, our main results **do not** hold in this general version of routing-proofness, so we leave this as an open question

if $\sum_{j \in p_k} \xi_j(\boldsymbol{P}) > \sum_{j \in p_k} \xi_j(\widetilde{\boldsymbol{P}}(p_k, z))$, then by continuity of ξ, there exists z very small such that $\xi_i(\boldsymbol{P}) > \sum_{j \in p_k} \xi_j(\widetilde{\boldsymbol{P}}(p_k, z)) \Rightarrow \xi_i(\boldsymbol{P})z > \sum_{j \in p_k} z\xi_j(\widetilde{\boldsymbol{P}}(p_k, z))$, which clearly violates the definition of routing-proofness given by (1). In this paper, we will mainly be considering limit routing-proofness (2).

4 Egalitarian mechanism

4.1 Comparisons of EG and AC mechanism

The examples in section 1.1 allows us to see the unique characteristics of AC and η^{pr}. Recall that AC stands against routing maneuvers but fails to satisfy the stand-alone core stability whereas η^{pr} is in an exact opposite situation. Therefore, neither of them serves as an appropriate mechanism to be implemented here. In addressing to the incompatibility between stand-alone core stability, efficiency and limit routing-proofness, EG captures the spirit of AC and η^{pr} in the sense that EG divides the total cost equally and subject to the stand-alone cost of each link. The two features combined make EG a fairly decent cost-sharing mechanism.

Initially, EG and AC share an identical feature in allocating charges to agents in the network. That is, both EG and AC divide the total cost of the network across all agents[9]. The similarity of the two mechanisms only begins to diverge when the charges (independently) reach the threshold of the stand-alone cost of any link (presumably the smallest stand-alone cost first). In that case, AC will disregard the stand-alone cost and continue increasing charges to agents on each link i equally without bound. In contrast, EG will stop assigning charges to agents whose links have reached the stand-alone costs. The remaining cost will then be covered by the links in which the stand-alone costs are yet to reach. Note that this characteristic will definitely preserve the property of stand-alone core stability. However, at the same time, it weakens the compatibility of the three properties to some extent. We will provide and discuss the conditions under which circumstances an egalitarian solution is guaranteed to exist (Theorem 1). Prior to that, we will pose and set up the network cost-sharing problem.

4.2 Egalitarian mechanism and solution

Given a network cost-sharing problem $\boldsymbol{P} = \langle N, C, \theta \rangle$, our ultimate objective is to find a cost-sharing mechanism such that the set of cost shares allocated by the mechanism will be efficient, limit routing-proof and core stable. We formulate the network cost-sharing problem along with the constraints into a set of system of equations.

[9] Since the total number of agents in the network is unknown, we are really dividing the total cost by the total number of units. So each agent is paying a share of cost proportional to the units of traffic she sends. But technically, the cost per unit of traffic is the same for every agent. In that sense, every agent is paying a unit of traffic at the same cost, which is similar to the average cost.

$$\sum_{i=1}^{s} \lambda_i \theta_i = EFF(\theta_1, ..., \theta_s) \tag{3}$$

$$\lambda_i \theta_i \leq C_i(\theta_i) \qquad \text{for all } i = 1, ..., s \tag{4}$$

$$\lambda_i \leq \sum_{l \in p_k} \lambda_l \qquad \text{for all } p_k \in \mathcal{P}_i \tag{5}$$

$$\lambda_i \geq 0 \qquad \text{for all } i = 1, ..., s. \tag{6}$$

We denote λ_i[10] the payment charged to each agent on link i. Other variables are defined exactly the same way as the model in section 3. $EFF(\theta_1, ..., \theta_s)$ represents the efficiency of the network computed as a function of $(\theta_1, ..., \theta_s)$. We express efficiency in this form because computing the efficiency (social optimality) of a network is generally a NP-hard problem[11] ([5]).

Equation (3) requires that the sum of the charges, when multiplied by the demands over all links $i = 1, ..., s$ is exactly equal to the efficiency of the network. In particular, this constraint represents the efficiency property.

Equation (4) describes the stand-alone core stability for all link i. Essentially, this constraint says that the amount of assigned charges to each agent on link i should never exceed the cost of connecting the target pairs by themselves via link i.

Equation (5) represents the limit routing-proofness constraint for a general network. For any link i, $\mathcal{P}_i$ is the set of all feasible alternative paths $\{p_1, ..., p_t\}$ that connect link i. The limit routing-proofness constraint then implies that the charge λ_i on link i must be less than or equal to the sum of λ_l for $l \in p_k$ on all possible paths $\{p_1, ..., p_t\}$[12].

Equation (6) is an obvious one, which says that the charges assigned must be greater than or equal to zero.

Now, we formally define the egalitarian solution for parameter λ^*, where $\lambda^* \in \mathbb{R}_+$. For a general network, the *egalitarian solution* ($EG^{T_{\lambda^*}}$) takes the following form:

$$\lambda_i^* = \min \left\{ \lambda^*, AC_i(\theta_i), \left[\sum_{l \in p_k} \lambda_l^* \right] \right\} \qquad \forall i = 1, ..., s, \forall p_k \in \mathcal{P}_i \tag{7}$$

[10] Note that this λ_i needs not be the same as λ_i^*. The former λ_i can be any solution to the system, whereas λ_i^* is the solution of EG, that is, λ_i^* is just one of the solutions of λ_i

[11] In [10], Moulin suggests that if no simple algorithm can characterize the cost minimization outcomes, then we will have no choice but to drop the cost minimization requirement and turn to the exogenously given suboptimalities

[12] Note that if a simple triangular network is analyzed here, then the routing-proofness constraint is just a set of triangle inequalities

Let

$$T_{\lambda^*} = \sum_{i=1}^{s} \lambda_i^* \theta_i \tag{8}$$

For any given λ^*, we are able to find a solution $\text{EG}^{T_{\lambda^*}}$ to the system of equations (7). This $\text{EG}^{T_{\lambda^*}}$ will generate revenue T_{λ^*} where $T_{\lambda^*} = \sum_{i=1}^{s} \lambda_i^* \theta_i$ where λ_i^* is a function of λ^* (for simplicity, we will just omit it). Note that for any given $\underline{\lambda}^*, \overline{\lambda}^*$ such that $\underline{\lambda}^* \neq \overline{\lambda}^*$, the same level of revenue might be generated. To see this, suppose $\underline{\lambda}^*$ generates a set of cost shares such that λ_i^* reaches the average cost for all i, then such allocation $\text{EG}^{T_{\underline{\lambda}^*}}$ generates revenue $T_{\underline{\lambda}^*}$. Suppose now we have $\overline{\lambda}^* > \underline{\lambda}^*$. Since all λ_i^* have reached its average cost for all i, the revenue $T_{\overline{\lambda}^*} = T_{\underline{\lambda}^*}$. Hence, T_{λ^*} is non-decreasing in λ^*. Note that the converse is not true, as we will see later.

We can see immediately that the maximal revenue generated under $\text{EG}^{T_{\lambda^*}}$ is when $\lambda^* = \max\{AC_1(\theta_1), ..., AC_s(\theta_s)\}$. That is, for any $\widetilde{\lambda^*} > \lambda^*_{max}$, the revenue $T_{\widetilde{\lambda^*}} = T^{max}_{\lambda^*}$. We denote such allocation $\text{EG}^{T^{max}_{\lambda^*}}$. If the revenue $T^{max}_{\lambda^*} > EFF(\theta_1, ..., \theta_s)$, then by the continuity of $T_{\lambda^*}(\lambda^*)$, we can always find one λ^{**} such that $T_{\lambda^{**}} = EFF(\theta_1, ..., \theta_s)$. This λ^{**} will generate an allocation $\text{EG}^{T_{\lambda^{**}}}$[13] and it will be called the ***egalitarian solution*** ($\boldsymbol{EG^{T_{\lambda^*}}}$). Note also that when this egalitarian solution exists, it will be efficient, core stable and limit routing-proof.

We denote λ_i^* the egalitarian solution for each agent on link i. Hence, if an egalitarian solution exists, it will be a vector of charges $(\lambda_1^*, ..., \lambda_s^*)$. It is noteworthy to mention that the last term within the min operator in equation (6) is the set of limit routing-proofness constraints. Depending on the configuration for the networks, these limit routing-proofness constraints will appear differently for every different λ_i^*. Precisely, the formulation of EG will be different for each particular problem $\boldsymbol{P}$.

We will now explain how EG allocates the cost shares to every agent in the network. Note that from above, we see that the EG proposed in this paper is a small variation of the traditional EG[14]. The main difference lies on the inclusion of the limit routing-proofness constraints. Such variation changes how EG would normally assigns cost shares to agents. In particular, the cost-sharing assignment can be divided into a number of steps:

STEP 1: Initially, EG starts assigning an identical charge $\lambda^* > 0$ to all agents on all links in the network.

STEP 2a: If λ^* does not reach any of the stand-alone constraint for all links such that $\sum_{i=1}^{s} \lambda_i^* \theta_i = T_{\lambda^{**}}$, then we have found an egalitarian solution that is efficient, core stable and limit routing-proof. To see this, note that none of the stand-alone constraint is

[13] If it happens that $T_{\lambda^{**}}$ is the maximal revenue target that EG can achieve, then $T_{\lambda^{**}} = T^{max}_{\lambda^{**}}$ and the allocation $\text{EG}^{T_{\lambda^{**}}}$ coincides with $\text{EG}^{T^{max}_{\lambda^{**}}}$

[14] See [4] and [16] for the traditional EG; In [16], EG is named under constrained equal gains

reached implies:

$$\lambda^* < AC_1(\theta_1) \leq \cdots \leq AC_s(\theta_s) \Rightarrow \lambda_i^* < AC_1(\theta_1) \leq \cdots \leq AC_s(\theta_s) \qquad \forall i$$

Clearly, the stand-alone core stability is satisfied. Moreover, limit routing-proofness implies

$$\sum_{l \in p_k} \lambda_l^* = a\lambda^* > \lambda^* = \lambda_j^* \qquad \forall p_k \in \mathcal{P}_j \text{ and } a \in \mathbb{N}_+,$$

where a represents the number of links contained in an alternative path $p_k \in \mathcal{P}_j$ with the assumption that link j is demanded. Obviously, efficiency is also achieved since we have $T_{\lambda^{**}}$. Hence, $\text{EG}^{T_{\lambda^{**}}} = (\lambda_1^*, ..., \lambda_s^*) = (\lambda^*, ..., \lambda^*)$[15].

STEP 2b: If the assigned charge λ^* reaches one (or more) of the stand-alone constraints, for instance, $\lambda^*\theta_i = C_i(\theta_i) \Leftrightarrow \lambda^* = AC_i(\theta_i)$. Then by (6), EG is just

$$\min\left\{AC_i(\theta_i), \left[\sum_{l \in p_k} \lambda_l\right]\right\} = \lambda_i^* \qquad \text{for all } p_k \in \mathcal{P}_i$$

In this case, $\lambda_i^* = AC_i(\theta_i)$. This is because as λ^* increases, the stand-alone constraints will always be reached prior to any limit routing-proofness constraints before an *n-cycle*[16] is formed.

STEP 3: Once λ^* hits the stand-alone cost[17], we assign the reached average cost multiplied by θ_i on every link i and sum them up to check if the revenue target is reached. If it is reached, then the solution will be core stable, efficient and limit routing-proof and the reason is similar to step 2a. If it not reached, we continue increasing λ^* while leaving the first stand-alone as it is until λ^* reaches the second stand-alone cost. Repeat step 3 until any n-cycle is formed.

STEP 4: Once the reached stand-alone costs form an $(n\backslash\{i\})$-cycle, equation (7) initiates and it takes the minimum of AC_j or $\sum_{l \in p_k} \lambda_l$ of the last link that forms the n-cycle. We continue increasing λ^* equally on the rest of the links until the next stand-alone cost is reached or if the target revenue is reached. When every link has a cost but the last link j, we use the efficiency to minus the sum of the stand-alone costs and divided it by θ_j. Repeat step 2 and 3 if the target revenue is not reached.

There is one noticeable point we should emphasize with caution about this variation of EG. That is, given a problem $\boldsymbol{P}$, a solution of EG satisfying simultaneously all three properties might or might not exist. The existence of EG depends on whether the last

[15] Note that in this case, the allocation of cost shares under EG coincides with AC

[16] A n-cycle is a closed path that consists of n nodes and n links with every two adjacent nodes connected by one link, where $n \in \mathbb{N}$. Thus, an alternative path p_k is an $(n\backslash\{i\})$-cycle that has exactly $n-1$ links

[17] Literarily, it is really $\lambda^*\theta_i$ that reaches $C_i(\theta_i)$, so $\lambda^* = AC_i(\theta_i)$

stand-alone cost being reached can cover the loss incurred by the replacement of the limit routing-proofness constraints from the average costs of the links. To see this, note that, if $EFF(\theta_1, ..., \theta_s) < T_{\lambda^*}^{max}$, then *no* egalitarian solution exists if the following holds:

$$T_{\lambda^*}^{max} - EFF(\theta_1, ..., \theta_s) < \sum_{i=1}^{s} \left(\max \left\{ 0, AC_i(\theta_i)\theta_i - \min_{p_k \in \mathcal{P}_i} \left[\sum_{l \in p_k} \lambda_l^* \theta_l \right] \right\} \right) \tag{9}$$

Equation (9) implies that if the difference between the maximal revenue target $T_{\lambda^*}^{max}$ and the efficiency cannot cover the total loss between the average costs and the λ_l^* for all l on the alternative paths, then efficiency will not be achieved. And hence, no egalitarian solution will be found to satisfy all three properties simultaneously. On the other hand, if $EFF(\theta_1, ..., \theta_s) > T_{\lambda^*}^{max}$, then clearly efficiency is not satisfied. We provide an example in the next subsection.

4.2.1 An example for problem with no egalitarian solution

Consider the network below with each agent sending one unit of traffic via a particular link. There are 4 agents who wish to connect from node 1 $\leftrightarrow$ 2; 1 agent wants to connect from node 1 $\leftrightarrow$ 3 and 1 agent from node 2 $\leftrightarrow$ 3. As an usual routine, we first compute the efficiency of this network. Denote by z_3 the units of traffic on link 3 being rerouted.

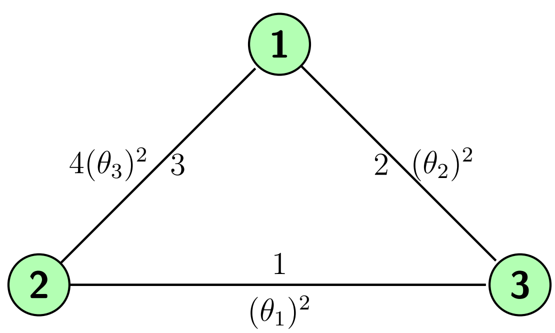


Figure 2: A simple Network with 3 nodes and 3 links

$$\min_{z_3 \geq 0} C(z_3) = \min_{z_3 \geq 0} 4(4 - z_3)^2 + (1 + z_3)^2 + (1 + z_3)^2$$
$$\frac{\partial C}{\partial z_3} : -8(4 - z_3) + 2(1 + z_3) + 2(1 + z_3) = 0$$
$$12z_3 = 28$$
$$z_3^* = \frac{7}{3}$$

Hence, efficiency is equal to $4(4-\frac{7}{3})^2+(1+\frac{7}{3})^2+(1+\frac{7}{3})^2 = \frac{100}{3} = 33.\bar{3}$. Now, the stand-alone cost for each link is evaluated at the number of units initially sent by agents. That is,

$$C_1(\theta_1) = (1)^2 = 1, C_2(\theta_2) = (1)^2 = 1, C_3(\theta_3) = 4(4)^2 = 64.$$

Thus, if the central authority decides to impose a personalized charge to every agent, then it can impose at most a total amount of \$1 to each agent on link 1 and 2; \$16 to each agent on link 3. Under EG, as we increase λ^* , the stand-alone constraints of link 1 and 2 are reached immediately followed by link 3. That is, $\lambda_i^* = AC_i(\theta_i) = 1$ and thus $C_i(\theta_i) = AC_i(\theta_i)\theta_i = 1$ for all $i = 1, 2$.

By efficiency, we have

$$\begin{aligned}\lambda_3^* &= \frac{EFF(\theta_1,\theta_2,\theta_3) - \lambda_1^*\theta_1 - \lambda_2^*\theta_2}{\theta_3} \\ &= \frac{EFF(\theta_1,\theta_2,\theta_3) - AC_1(\theta_1)\theta_1 - AC_2(\theta_2)\theta_2}{\theta_3} \\ &= \frac{EFF(\theta_1,\theta_2,\theta_3) - C_1(\theta_1) - C_2(\theta_2)}{\theta_3} \\ &= \frac{\frac{100}{3} - 1 - 1}{4} \\ &= \frac{94}{12} = 7.8\bar{3} < \frac{64}{4} = 16\end{aligned}$$

Hence, the charges assigned to each agent under EG are $(\lambda_1^*, \lambda_2^*, \lambda_3^*) = (1, 1, 7.8\bar{3})$. Note that this solution satisfies stand-alone core stability and efficiency but it fails to meet the limit routing-proofness constraint. We show it graphically in Figure 3.

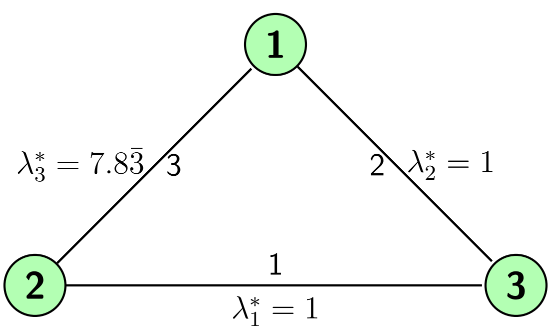


Figure 3: A simple network with charges allocated to agents

For agents who wish to send their traffic from node $1 \leftrightarrow 2$ via link 1, they will be profitable sending their traffic indirectly via link $1 \leftrightarrow 3 \leftrightarrow 2$ rather than directly from link $1 \leftrightarrow 2$. This is because $\lambda_1^* + \lambda_2^* = 1 + 1 = 2 < 7.8\bar{3} = \lambda_3^*$. It is easy to check that equation

(9) holds. Hence, no egalitarian solution satisfying simultaneously all three properties is found for this particular problem. This implies the assigned cost shares will not be able to prevent agents from rerouting or incentivize agents to connect to their target nodes via the original links, especially for agents who are demanding link 1.

4.3 A numerical example of EG

In this section, we will provide a step-by-step numerical example illustrating how EG allocates a personalized charge to each agent. This example has a concrete egalitarian solution in which the three main properties are simultaneously satisfied.

Example: Consider the following network. There are a total of 11 agents with each agent sending one unit of traffic. 2 agents want to send traffic via link 1; 6 agents want to send traffic via link 2 and 3 agents via link 3.

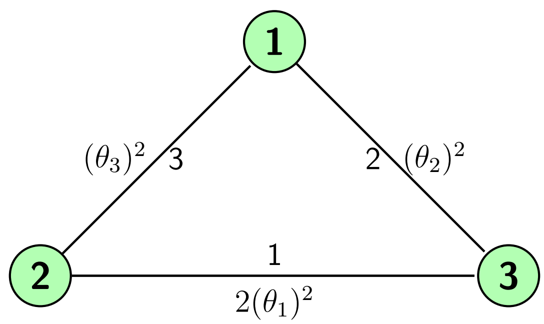


Figure 4: A simple network with 3 nodes

Notice that computing the efficiency of this network is not as obvious as in the previous examples. Indeed, solving for this combinatorial optimization problem involves the evaluations of all feasible solutions and selecting the solutions that truly minimize our objective function ([13]). Denote by x_1, y_2, z_3 the units of traffic being rerouted for θ_1, θ_2 and θ_3 respectively. We check the following seven expressions:

$$
\begin{aligned}
\min(3 - z_3 + y_2 + x_1)^2 + (6 - y_2 + x_1 + z_3)^2 + 2(2 - x_1 + y_2 + z_3)^2 \\
\min(3 - z_3)^2 + (6 + z_3)^2 + 2(2 + z_3)^2 \\
\min(3 + y_2)^2 + (6 - y_2)^2 + 2(2 + y_2)^2 \\
\min(3 + x_1)^2 + (6 + x_1)^2 + 2(2 - x_1)^2 \\
\min(3 - z_3 + x_1)^2 + (6 + x_1 + z_3)^2 + 2(2 - x_1 + z_3)^2 \\
\min(3 + x_1 + y_2)^2 + (6 - y_2 + x_1)^2 + 2(2 - x_1 + y_2)^2 \\
\min(3 - z_3 + y_2)^2 + (6 - y_2 + z_3)^2 + 2(2 + y_2 + z_3)^2 \\
s.t. \qquad x_1, y_2, z_3 \geq 0
\end{aligned}
$$

We check that the optimal solution $(x_1^*, y_2^*, z_3^*) = (0,0,0)$ for all expressions and thus the efficiency is just $3^2 + 6^2 + 2(2)^2 = 53$. The stand-alone cost for each link is:

$$C_1(\theta_1) = 2(2)^2 = 8, C_2(\theta_2) = 6^2 = 36 \text{ and } C_3(\theta_3) = 3^2 = 9$$

Now, we can formulate the problem and apply EG as in equations (3) - (6):

$$2\lambda_1 + 6\lambda_2 + 3\lambda_3 = 53 \tag{3'}$$

$$2\lambda_1 \leq 2(2)^2, 6\lambda_2 \leq (6)^2, 3\lambda_3 \leq (3)^2 \tag{4'}$$

$$\lambda_1 + \lambda_2 \geq \lambda_3, \lambda_2 + \lambda_3 \geq \lambda_1, \lambda_1 + \lambda_3 \geq \lambda_2 \tag{5'}$$

$$\lambda_1 \geq 0, \lambda_2 \geq 0, \lambda_3 \geq 0 \tag{6'}$$

Under EG, we have:

$$\underbrace{2\min\{\lambda^*, 2(2), (\lambda_2^* + \lambda_3^*)\}}_{\lambda_1^*\theta_1} + \underbrace{6\min\{\lambda^*, (6), (\lambda_1^* + \lambda_3^*)\}}_{\lambda_2^*\theta_2} + \underbrace{3\min\{\lambda^*, (3), (\lambda_1^* + \lambda_2^*)\}}_{\lambda_3^*\theta_3} = 53 \tag{6.1'}$$

By (7), we have

$$\lambda_1^* = \min\{\lambda^*, 2(2), (\lambda_2^* + \lambda_3^*)\} \tag{7.1'}$$

$$\lambda_2^* = \min\{\lambda^*, (6), (\lambda_1^* + \lambda_3^*)\} \tag{7.2'}$$

$$\lambda_3^* = \min\{\lambda^*, (3), (\lambda_1^* + \lambda_2^*)\} \tag{7.3'}$$

Solving the above system simultaneously will give us an egalitarian solution, if there exists one. We will attack this problem by gradually increasing λ^* until the first stand-alone constraint $AC_i(\theta_i)$ for $i = 1, 2, 3$ is reached. This happens when $\lambda^* = 3$.

Note the fact that the stand-alone constraints of λ_1^* and λ_2^* are still non-binding implies $\lambda_1^* + \lambda_2^* > \lambda_3^*$. So by (6.1′), suppose $\lambda^* = 3$ for all λ_i^*, we check if (6.1′) is satisfied:

$$2\lambda^* + 6\lambda^* + 3\lambda^* = 2(3) + 6(3) + 3(3) = 33 < 53$$

Obviously, $\lambda^* = 3$ does not achieve efficiency. Hence, we must continue increasing λ^* on λ_1^* and λ_2^* while leaving $\lambda_3^* = 3$ as its stand-alone constraint is reached.

Next, we apply the same logic and continue increasing λ^*. It is easy to see that the next stand-alone constraint being reached will be $AC_1(\theta_1)$. That is, when $\lambda^* = 4$.

We know that $\lambda_3^* = 3$ and the stand-alone constraint of λ_2^* is still non-binding when $\lambda^* = 4$. This implies $\lambda_2^* + \lambda_3^* = \lambda_2^* + 3 > 4 = \lambda_1^*$. Thus, the minimum value in (7.1′) will never be the limit routing-proofness constraint. We check (6.1′),

$$2\lambda^* + 6\lambda^* + 3\lambda^* = 2(4) + 6(4) + 3(3) = 41 < 53$$

Again, efficiency is still not attained with $\lambda^* = \lambda_1^* = \lambda_2^* = 4$ and $\lambda_3^* = 3$. Leaving $\lambda_1^* = 4$ and $\lambda_3^* = 3$, we have by efficiency,

$$\begin{aligned}\lambda_2^* &= \frac{EFF(\theta_1, \theta_2, \theta_3) - \lambda_1^* \theta_1 - \lambda_2^* \theta_2}{\theta_3} \\ &= \frac{53 - 2(4) - 3(3)}{6} \\ &= 6\end{aligned}$$

Recall that $(7.2')$ implies

$$\lambda_2^* = \min\{\lambda^*, 6, 6(4+3))\}$$

In this case, we need to be cautious about the limit routing-proofness constraint. If it happens that the minimum value is obtained from the limit routing-proofness constraint, then there will be no egalitarian solution satisfying $(6.1')$.

We proceed to check if $(\lambda_1^*, \lambda_2^*, \lambda_3^*) = (4, 6, 3)$ satisfies all the constraints in $(3')$ - $(6.1')$.

$$\begin{array}{lcl} (3') & \Rightarrow & 2(4) + 6(6) + 3(3) = 53 \\ (4') & \Rightarrow & 2(4) = 2(2)^2, 6(6) = 36, 3(3) = 9 \\ (5') & \Rightarrow & \lambda_1^* + \lambda_2^* = 4 + 6 = 10 > 3 = \lambda_3^* \\ & \Rightarrow & \lambda_2^* + \lambda_3^* = 6 + 3 = 9 > 4 = \lambda_1^* \\ & \Rightarrow & \lambda_1^* + \lambda_3^* = 4 + 3 = 7 > 6 = \lambda_2^* \\ (6') & \Rightarrow & \lambda_1^* = 4 > 0, \lambda_2^* = 6 > 0, \lambda_3^* = 3 > 0 \end{array}$$

Hence, we have found an egalitarian solution $\text{EG}^{T^{max}_{\lambda^{**}}} = (\lambda_1^*, \lambda_2^*, \lambda_3^*) = (4, 6, 3)$ such that efficiency, stand-alone core stability and limit routing-proofness are simultaneously satisfied. Interestingly, since the demands θ_1, θ_2 and θ_3 are precisely the optimal allocations that generates the total minimum cost, the stand-alone core stability and efficiency force the egalitarian solution $\lambda_i^* = AC_i(\theta_i)$ for all $i = 1, 2, 3$.

We now turn our focus on investigating the necessary and sufficient conditions for the existence of an efficient, core stable and limit routing-proof egalitarian solution. To derive those conditions, we introduce an algorithm called the *RP algorithm*.

The RP algorithm:

STEP 1: For any general network N, we start assigning some ***temporary prices*** $T_i = AC_i(\theta_i)$ to agents on link i in the network.

STEP 2: Define a ***semi-cycle*** such that all but one link have ***permanent prices*** P_i. We continue to assign temporary prices. At any point in time, the smallest temporary price becomes the permanent price until any semi-cycle is formed.

STEP 3: Once a semi-cycle is formed, we replace the last temporary price on the link that readily forms a cycle from the semi-cycle with a new temporary price $T_i' = \min\left\{AC_i, \sum_{l\in p_k} P_l\right\}$[18].

STEP 4: Repeat step 2 and 3 of the algorithm and extract all possible semi-cycles until all prices become permanent.

By assigning temporary prices at the average cost in step 1, the RP algorithm ensures for stand-alone core stability. On the other hand, replacing a new temporary price either with the minimum of the average cost, or the sum of the prices on the alternative paths of a semi-cycle respects limit routing-proofness[19].

Note that the construction of the RP algorithm and EG are similar, but their objectives and operations are slightly different. To be precise, the RP algorithm does not allocate charges to agents as EG does, but it will always find the feasible and maximal prices that it can achieve for any achievable revenue target. Note that the sum of these maximal prices might even exceed the revenue target. This is in contrast different from EG where it always allocates the minimal prices to agents subject to their stand-alone costs in achieving the revenue target. Such discrepancy is due to the nature of the operation of EG and the RP algorithm. The RP algorithm always starts from the maximal revenue as well as average costs and solves the problem backwardly, whereas EG attacks the problem from the other direction and charges to agents with the lowest cost every time while gradually increases it and stops upon efficiency is reached.

We conjecture that the backward approach will be more convenient and relatively easier for deriving the necessary and sufficient conditions for the existence of egalitarian solution that is efficient, core stable and limit routing-proof. Therefore, our strategy would be to utilize the RP algorithm and show that egalitarian solutions exist whenever the RP algorithm can assign a vector of prices that meets its revenue target.

5 Main Results

We now turn to the main results of this paper. As mentioned in the previous section, our results will rely heavily on the application of RP algorithm.

Proposition 1 *The prices given by the RP algorithm coincide with $EG^{T^{max}_{\lambda^*}}$.*

The RP algorithm takes the permanent price $P_i = \min_{p_k\in\mathcal{P}_i}\left\{AC_i, \sum_{l\in p_k} P_l\right\}$ for all i. If the minimum temporary price is $\sum_{l\in p_k} P_l$ for some p_k, then the RP algorithm will never replace any permanent prices less than that. Therefore, the revenue generated under these prices and demands is going to be maximal (See section 5.1 below). Note that this implies

[18]When there is no confusion, we write $AC_i(\theta_i)$ as AC_i for short

[19]Similar to EG, the RP algorithm might not satisfy the efficiency property, but for any revenue target (regardless of whether efficiency is achievable), it always generates prices that maximize the possible revenue. See section 2 and proposition 2

the RP algorithm will be irrespective to any revenue target, including the efficiency. That is, it merely searches for the maximal prices that it can achieve while satisfying together the stand-alone core stability and limit routing-proofness.

On the other hand, EG also allocates charges to agents with respect to the stand-alone core stability and limit routing-proofness. It chooses, or controls the parameter λ^* where the revenue target (this might include efficiency) is just achieved and stops. Hence, if the revenue target is precisely the maximal revenue that is achievable by the RP algorithm, then the allocation $\text{EG}^{T^{max}}_{\lambda^*} = (\lambda^*_1, ..., \lambda^*_s)$ coincides with the price vector $\overline{P} = (P_1, ..., P_s)$ simply because EG and RP algorithm are constructed in an identical way. Indeed, since the RP algorithm always searches for the maximal revenue that it can achieve in a network, the total charge (resp. total revenue) allocated (resp. collected) by EG can do no better than that maximal revenue. Thus, the maximal revenue provided by the RP algorithm is an upper bound for which EG can achieve in a network.

Theorem 1 ***(a)*** *Let agent j^* be the last agent in the RP algorithm. A stand-alone core stable, efficient and limit routing-proof price vector $\overline{P} = (P_1, ..., P_s)$ exists if and only if one of following conditions*

i. $\frac{EFF(\theta_1,...,\theta_s)-\sum_{i\neq j^*} P_i\theta_i}{\theta_{j^*}} \leq \min_{p_k\in\mathcal{P}_{j^*}} \left\{AC_{j^*}, \sum_{l\in p_k} P_l\right\}$, *where $\mathcal{P}_{j^*}$ is the set of all feasible alternative paths of j^*;*

ii. $EFF(\theta_1, ..., \theta_s) \leq \sum_{i=1}^s P_i\theta_i$

is satisfied.
(b) *The egalitarian solution exists if and only if either condition (i) or (ii) is satisfied.*

Part (a) of Theorem 1 states the necessary and sufficient conditions for which the price vector $\overline{P} = (P_1, ..., P_s)$ is core stable, efficient and limit routing-proof. Note that condition (ii) is implied by condition (i). Part (b) tells us that the egalitarian solution is guaranteed to exist if either condition (i) or (ii) hold simultaneously. The reason is discussed above. Note that if condition (ii) is relaxed, then EG will typically satisfy the stand-alone core stability and limit routing-proofness without any issues.

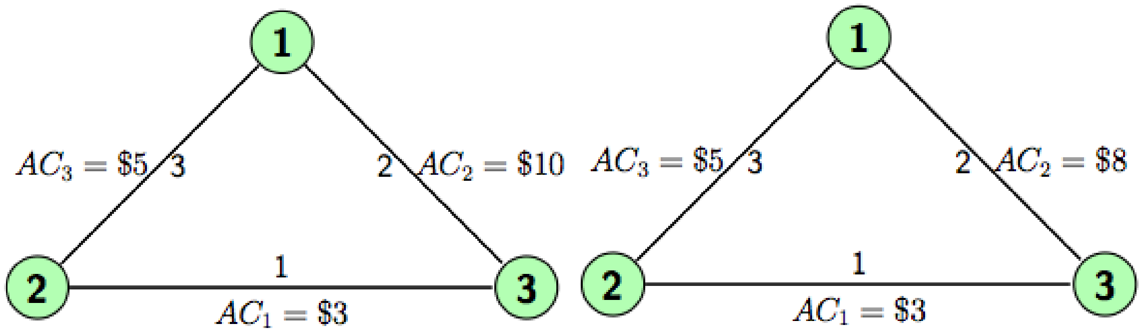
Corollary 1 *Given a general network, suppose that the average cost satisfies the (generalized)*[20] *triangle inequality, that is,*

$$AC_i(\theta_i) \leq \min_{p_k\in\mathcal{P}_i} \left\{ \sum_{l\in p_k} AC_l(\theta_l) \right\} \tag{10}$$

for any demand of link i, then the egalitarian solution exists.

[20] Note that this does not necessary have to be a triangle, it can be a "n-cycle" inequality where $n \in \mathbb{N}_+$ is the number of links

Note that this condition is sufficient but not necessary for the existence of egalitarian solution. To see this, consider the following network:



(a) A network with assigned average costs

(b) The prices under RP algorithm

Figure 5: An example illustrating the existence of egalitarian solution

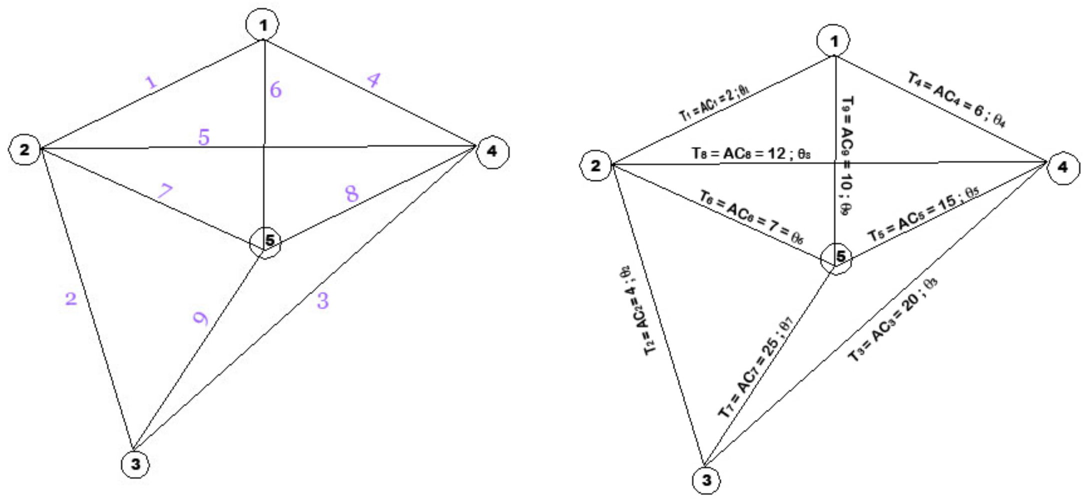
In figure 5(a), the average costs are given by \$3, \$10 and \$5 for link 1, 2, 3 respectively. Clearly, the triangle inequality is not satisfied since \$5 + \$3 = \$8 < \$10. Under the RP algorithm, AC_2 is replaced with a new permanent price taken from the sum of \$5 + \$3 = \$8. We see that there exists an egalitarian solution whenever $EFF(\theta_1, \theta_2, \theta_3) \leq 3\theta_1 + 5\theta_2 + 8\theta_3$. In other words, no solution will be found if $EFF(\theta_1, \theta_2, \theta_3) > 3\theta_1 + 5\theta_2 + 8\theta_3$. Hence, (10) is a sufficient condition rather than a necessary condition.

5.1 The maximal revenue under RP algorithm

Theorem 1 provides the necessary and sufficient conditions for the existence of an egalitarian solution. If any of the conditions is not met, then immediately we know there is no egalitarian solution that satisfies the three properties simultaneously. Therefore, it seems natural to ask what is the maximal revenue that we can collect from the agents such that limit routing-proofness is compatible with the stand-alone core stability under the RP algorithm. In other words, what could be the minimum level of efficiency needed to sacrifice in exchange for the compatibility of limit routing-proofness and stand-alone core stability?

We denote this maximum revenue by a real number $K^* \leq EFF(\theta_1, ..., \theta_s)$ such that limit routing-proofness and stand-alone core stability are compatible. We provide a numerical example showing how K is computed under the RP algorithm. Consider an arbitrary network with the links numbered as in figure 4(a):

Without loss of generality, suppose the average costs are given as in figure 4(b). By the RP algorithm, we start assigning a temporary price — average cost to each link until every link has a temporary price. For instance, we assign $T_1 = AC_1 = 2$ to link 1 and $T_2 = AC_2 = 4$ to link 2, etc. Starting from the smallest temporary price, $T_1 = AC_1 = 2$,



(a) A general network with 5 nodes and 9 links numbered in order

(b) A network with assigned temporary prices

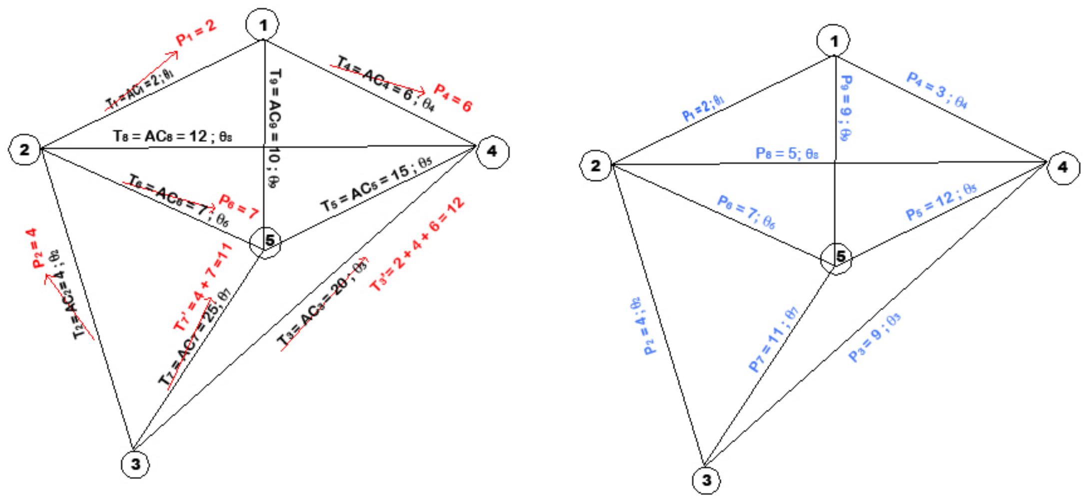
Figure 6: Step 1 of RP algorithm

T_1 will be replaced with a permanent price $P_1 = 2$ followed by the second, third smallest temporary prices and so forth at every point in time until the first semi-cycle is formed. Note that the first semi-cycle is formed by links 1, 2 and 4. Thus, the temporary price T_3 on link 3 is replaced by a new temporary price $T_3' = \min\{\underbrace{20}_{AC_3}, \underbrace{12}_{2+4+6}\} = 12$. This new temporary price T_3' will become a permanent price only if all the ascending smaller prices are replaced with permanent prices. Once the price is replaced, we continue to replace the next smallest price, which is price of link 7 in this case. In there a second semi-cycle is formed by links 2 and 7 and we replace the temporary price T_9 with $T_9' = \min\{\underbrace{25}_{AC_9}, \underbrace{9}_{2+7}\} = 9$.[21]

Continue replacing in this fashion and ensuring all the semi-cycles are extracted until each temporary price becomes permanent price as shown in figure 5(b). Now, computing the maximal revenue is relatively simple. Merely summing the permanent prices over all links multiplied by its associated demands will give us the maximal revenue K:

$$2\theta_1 + 4\theta_2 + 9\theta_3 + 3\theta_4 + 12\theta_5 + 7\theta_6 + 11\theta_7 + 5\theta_8 + 9\theta_9 = K^* \leq EFF(\theta_1, ..., \theta_s).$$

[21] Depending on the configuration of the networks, a temporary price may be replaced more than once before it finally becomes a permanent price



(a) In the process of replacing temporary prices to permanent prices

(b) All links have permanent prices

Figure 7: Step 2 - 4 of RP algorithm

Relaxing efficiency, K can be interpreted as the maximal revenue (resp. maximal charge) that the central authorities can collect (resp. charge) from the agents while maintaining the properties of limit routing-proofness and stand-alone core stability.

Proposition 2 *Let $P_1^*, ..., P_s^*$ be the permanent prices generated by the RP algorithm. Let $K^* = P_1^*\theta_1 + \cdots P_s^*\theta_s$. Then,*

i if there exists a limit routing-proof and stand-alone core stable allocation, then it generates revenue $\leq K^* < EFF(\theta_1, ..., \theta_s)$. Moreover, K^* is exactly achieved by prices $(P_1^*, ..., P_s^*)$.

ii there exists a limit routing-proof, stand-alone core stable and efficient allocation if and only if $EFF(\theta_1, ..., \theta_s) \leq K^*$.

Proof.
The proof is the same as the proof from theorem 1. ■

6 Limit Routing-proofness to Routing-proofness

When the mass of the traffic rerouted by agents tends to zero, limit routing-proofness outweighs routing-proofness in the sense that tiny amounts of rerouting will not affect the charges. However, when the size of a group of agents becomes substantially large enough to influence the charges, satisfying limit routing-proofness might not be sufficient to justify the satisfaction of routing-proofness. As we have mentioned earlier, any mechanism that is routing-proof always implies limit routing-proof but the converse is not necessarily true. Therefore, we need another condition to be able to say that limit routing-proofness also implies routing-proofness so that the two properties are bilateral to each others.

Proposition 3 *Suppose that* $\sum_{j\in p_k}\xi_j(\widetilde{\boldsymbol{P}}(p_k,z)) \geq \sum_{j\in p_k}\xi_j(\boldsymbol{P})$ *for all path* p_k*, then limit routing-proofness is equivalent to routing-proofness.*

Proof.
⇒) Suppose that $\sum_{j\in p_k}\xi_j(\widetilde{\mathbf{P}}(p_k,z)) \geq \sum_{j\in p_k}\xi_j(\mathbf{P})$, then by the definition of limit routing-proofness, $\xi_i(\boldsymbol{P}) \leq \sum_{j\in p_k}\xi_j(\boldsymbol{P})$. Together we have $\sum_{j\in p_k}\xi_j(\widetilde{\mathbf{P}}(p_k,z)) \geq \sum_{j\in p_k}\xi_j(\mathbf{P}) \geq \sum_i \xi_i(\boldsymbol{P})$. Hence, limit routing-proofness implies routing-proofness.

⇐) Clearly, routing-proofness implies limit routing-proofness as $z \to 0$. ■

In addition, we conjecture the following:

Conjecture: Suppose that the set of cost functions $C_i(\cdot)$ for all i and demands θ satisfy

$$AC_i(\theta_i - z) \leq \sum_{j\in p_k} AC_j(\theta_j + z) \qquad \forall \quad 0 \leq z \leq \theta_i, \quad p_k \in \mathcal{P}_i \tag{11}$$

then a routing-proof, efficient and core stable mechanism exists. In addition, we conjecture that EG also satisfies this property.

Since the average cost functions are arbitrary, we do not know in general the shapes and how they behave. For instance, how convex the average cost functions are. If it happens that $AC_i(\theta_i)$ is relatively more convex than other average cost functions $AC_j(\theta_j)$ for all $j \neq i$, then decreasing a small amount of z units from θ_i can lead to a substantially large drop in the average cost $AC_i(\theta_i - z)$. That means the sum of the less convex functions evaluated at $\theta_j + z$ might not necessarily cover the loss from the function that is relatively more convex. As discussed earlier, in that case there exists z very small such that $\xi_i(\boldsymbol{P}) > \sum_{j\in p_k} z\xi_j(\widetilde{\boldsymbol{P}}(p_k,z))$. Thus, by requiring condition (11) to hold, we eliminate this critical obstacle and hence, we conjecture that a routing-proof, core stable and efficient mechanism exists. Moreover, we conjecture that EG also satisfies this property by corollary 1.

7 Future Research and Some Open Questions

There are some remarkable open questions remain to be investigated that may help to improve this paper, or to facilitate the future research of this topic. The first open question that comes to mind is the natural extension of demanding multiple links by agents. In this paper, we have assumed agents are demanding only a single link between any pairs of nodes for simplicity purpose. However, this is an extraordinarily strict assumption that is not likely the case in reality. By extending the model into demanding multiple links should potentially enhance the applicability of EG in practice.

The second open question is the accommodation of the generalized routing-proofness with EG. The generalized version weakens the strong assumption of rerouting all traffic owned by agents. By allowing agents to reroute only a portion of their traffic, the model will be able to capture a better aspect from the reality. Moreover, EG will become a more powerful mechanism in allocating charges as it covers a wider range of preventing strategic maneuvers, if it also satisfies the generalized version of routing-proofness. Unfortunately, our main results show that EG does not seem to satisfy it. In that case, we might have to either improve EG by adding some other conditions. Or continue discovering more cost-sharing mechanisms that might satisfy it while keeping the other two properties. Nonetheless, such task can be hard to achieve as we know the three properties are almost simultaneously incompatible to each others.

The third open question is related to the conjecture in the previous section. In order for the conjecture to become a valid property, it is therefore significantly important for us to figure out under which conditions EG satisfies the proposed property. If we can derive such conditions, then EG will truly become a core stable, efficient and routing-proof mechanism that allocates charges (egalitarian solution) satisfying these three properties. One possibility to be considered is by having an identical average cost function on each link in the network. Then, rerouting any $z > 0$ units from θ_i will certainly incur a higher cost. This is because an identical average cost function will ensure the loss from the discrepancies between functions with different shapes and convexities is completely covered. In that sense, agents will simply have to pay a higher cost for the extra links that they use for rerouting with the ex-post efficiency being no less than before.

8 Appendix: Proofs

8.1 Proof of Proposition 1

Proof.
Let $(P_1, ..., P_s)$ be the prices given by the RP algorithm and let $(\lambda_1^*, ..., \lambda_s^*)$ be the EG$^{T_{\lambda^*}^{max}}$ solution. Fix a problem $\boldsymbol{P}$ and let $T_{\lambda^*}^{max}$ be the maximal revenue target achieved by EG and the RP algorithm, we will show that $(P_1, ..., P_s) = (\lambda_1^*, ..., \lambda_s^*)$. Without loss of generality, suppose that the average cost functions satisfy $AC_1(\theta_1) < \cdots < AC_s(\theta_s)$.

Solving by RP algorithm: We assign temporary prices $T_i = AC_i$ on each link i. At each point in time, the smallest temporary price is replaced by a permanent price, i.e. $T_i = AC_i(\theta_i) = P_i$ until any semi-cycle is formed. Let j be the last link that completes the semi-cycle. We replace T_j by $T_j' = \min\{\underbrace{AC_j}_{T_j}, \sum_{l \in p_k} P_l\}$.

Case 1: If $T_j' = \min\left\{AC_j, \sum_{l \in p_k} P_l\right\} = AC_j$, then we have:
Subcase 1: If $T_j' = AC_j$ is the next smallest AC_j, then AC_j becomes P_j.
Subcase 2: If $T_j' = AC_j$ is not the next smallest AC_j, then we continue replacing other temporary prices $T_m, m \notin \{i \cup j\}$ with $P_m, m \notin \{i \cup j\}$ until either T_j' becomes the smallest AC_j or another semi-cycle forms.

Note that if $T_j' = AC_j$ is the smallest price, then $T_j' = AC_j$ becomes P_j.

Case 2: If $T_j' = \min\left\{AC_j, \sum_{l \in p_k} P_l\right\} = \sum_{l \in p_k} P_l$, then we have:
Subcase 1: If $T_j' = \sum_{l \in p_k} P_l$ is the next smallest price, then $T_j' = \sum_{l \in p_k} P_l$ becomes P_j.
Subcase 2: If $T_j' = \sum_{l \in p_k} P_l$ is not the next smallest price, then we continue replacing other temporary prices $T_m, m \notin \{i \cup j\}$ with $P_m, m \notin \{i \cup j\}$ until $T_j' = \sum_{l \in p_k} P_l$ becomes the smallest price or another semi-cycle forms.

Together, this implies that $T_i = AC_i = P_i$ and $T_j' = AC_j = P_j$ or $T_j' = \sum_{l \in p_k} P_l$.

(**Remark**: If another semi-cycle forms before T_j' becomes the smallest price at some point in time, then we can follow the same proof as described above.)

Now, we solve $\boldsymbol{P}$ by EG. From (7), we know that $\lambda_i^* = \min\left\{\lambda^*, AC_i(\theta_i), \left[\sum_{l \in p_k} \lambda_l^*\right]\right\}$. As $\lambda^* \to \infty$, $AC_1(\theta_1)$ is reached, followed by $AC_2(\theta_2)$, ... until a semi-cycle is formed. Then, $\lambda_i^* = AC_i(\theta_i)$ for all i before a semi-cycle is reached. Suppose link j is the last link that completes the semi-cycle. Then we have,

Case 1: If $\min\left\{\lambda^*, AC_j(\theta_j), \sum_{l \in p_k} \lambda_l^*\right\} = AC_j(\theta_j)$, then $\lambda_j^* = AC_j(\theta_j)$.

Case 2: If $\min\left\{\lambda^*, AC_j(\theta_j), \sum_{l \in p_k} \lambda_l^*\right\} = \sum_{l \in p_k} \lambda_l^*$, then $\lambda_j^* = \sum_{l \in p_k} \lambda_l^*$.

Hence, we have that $P_i = AC_i(\theta_i) = \lambda_i^*$ before any semi-cycle forms. Since EG is constructed as the same way as the RP algorithm, i.e. $T_j' = \min\left\{AC_j, \sum_{l \in p_k} P_l\right\} \Leftrightarrow$

$\min\left\{\lambda^*, AC_j(\theta_j), \sum_{l\in p_k}\lambda_l^*\right\} = \lambda_j^*$. Hence, $T_j' = AC_j(\theta_j) = P_j \Leftrightarrow \lambda_j^* = AC_j(\theta_j)$; on the other hand, $T_j' = \sum_{l\in p_k} P_l = P_j \Leftrightarrow \lambda_j^* = \sum_{l\in p_k}\lambda_l^*$. Thus, $T_{\lambda^*}^{max} = P_1\theta_1+\cdots+P_s\theta_s = \lambda_1^*\theta_1+\cdots+\lambda_s^*\theta_s$ since $P_i = \lambda_i^*$ for all $i = 1, ..., s$. ■

8.2 Proof of Theorem 1

Proof.
Part (a): $\Rightarrow$) We first prove the sufficiency part, i.e., suppose a stand-alone core stable, efficient and limit routing-proof solution exists, then we have $\frac{EFF(\theta_1,...,\theta_s)-\sum_{i\neq j^*} P_i\theta_i}{\theta_j^*} \leq \min_{p_k\in\mathcal{P}_{j^*}}\left\{AC_{j^*}, \sum_{l\in p_k} P_l\right\}$.

STEP 1: To prove the sufficient part, we need to show the following first. Let $(\lambda_1^*, ..., \lambda_s^*)$ be some prices on a limit routing-proof and core stable allocation. Then for any $i \neq j^*$, we have $\lambda_i^* \leq P_i$.

Note that the temporary price $T_i = AC_i \geq \lambda_i^*$ by the stand-alone property. We see that at any point in time, the RP algorithm will replace a permanent price on link i. First, note that prior to the formation of a semi-cycle, $T_i = AC_i$ and then it becomes a permanent price $\geq \lambda_i^*$. Once a semi-cycle is formed, we replace T_i with a new temporary price $T_i' = \min\{\underbrace{AC_i}_{T_i}, \sum_{l\in p_k} P_l\}$ and $P_l \geq \lambda_l^*$ for all l in the path p_k. Now, we need to show that $T_i' \geq \lambda_i^*$ for all i.

Note that $\min\left\{AC_i, \sum_{l\in p_k} P_l\right\} \geq \min\left\{\lambda_i^*, \sum_{l\in p_k}\lambda_l^*\right\}$ since $T_i = AC_i \geq \lambda_i^*$ and $P_l \geq \lambda_l^*$. Since $T_i' = \min\left\{AC_i, \sum_{l\in p_k} P_l\right\}$, we have:

Case 1: If $T_i' = \min\left\{AC_i, \sum_{l\in p_k} P_l\right\} = AC_i \Rightarrow T_i' = AC_i \geq \lambda_i^*$.

Case 2: If $T_i' = \min\left\{AC_i, \sum_{l\in p_k} P_l\right\} = \sum_{l\in p_k} P_l \Rightarrow T_i' = \sum_{l\in p_k} P_l \geq \sum_{l\in p_k}\lambda_l^* \geq \lambda_i^*$. where the last inequality holds by limit routing-proofness. Together case 1 and case 2 imply $T_i' \geq \lambda_i^*$. Continue checking in this fashion for all $i \neq j^*$, we have that $P_i \geq T_i' \geq \lambda_i^* \Rightarrow P_i \geq \lambda_i^*$, as required.

STEP 2: Suppose a solution $(\lambda_1^*, ..., \lambda_s^*)$ satisfies limit routing-proofness, stand-alone core stability and efficiency, then by step 1, we have $\sum_{i\neq j^*}\lambda_i^*\theta_i \leq \sum_{i\neq j^*} P_i\theta_i$.
$\Rightarrow \lambda_{j^*}^* = \frac{EFF(\theta_1,...,\theta_s)-\sum_{i\neq j^*}\lambda_i^*\theta_i}{\theta_{j^*}} \geq \frac{EFF(\theta_1,...,\theta_s)-\sum_{i\neq j^*} P_i\theta_i}{\theta_{j^*}}$.

Since $P_i \geq \lambda_i^*$ for all $i \neq j^* \Rightarrow \min\left\{\sum_{l\in p_k} P_l\right\} \geq \min\left\{\sum_{l\in p_k}\lambda_l^*\right\} \geq \lambda_{j^*}^*$ where the last inequality holds by limit routing-proofness. By the stand-alone core stability, $AC_{j^*} \geq \lambda_{j^*}^*$. Together, this implies $\min_{p_k\in\mathcal{P}_{j^*}}\left\{AC_{j^*}, \sum_{l\in p_k} P_l\right\} \geq \frac{EFF(\theta_1,...,\theta_s)-\sum_{i\neq j^*} P_i\theta_i}{\theta_{j^*}} = \lambda_{j^*}^*$, as required.
$\Leftarrow$) Now, we proceed to prove the necessity part. Note that by the construction of the

RP algorithm, the allocation $(P_1, ..., P_s)$ is core stable and limit routing-proof. By proposition 1, we know that the prices given by the RP algorithm will coincide with $\text{EG}^{T^{max}_{\lambda^*}}$, that is, $\sum_{i=1}^{s} \lambda_i^*(\lambda^*)\theta_i = T^{max}_{\lambda^*} = \sum_{i=1}^{s} P_i\theta_i$. Thus, the only thing left to show is the efficiency. If we can show that $EFF(\theta_1, .., \theta_s) \leq T^{max}_{\lambda^*}$, then egalitarian solution will be a solution that satisfies efficiency, stand-alone core stability and limit routing proofness. By assumption, $\frac{EFF(\theta_1,...,\theta_s)-\sum_{i\neq j^*} P_i\theta_i}{\theta_{j^*}} \leq \min_{p_k\in\mathcal{P}_j^*}\left\{AC_{j^*}, \sum_{l\in p_k} P_l\right\}$. By definition, $T'_{j^*} = \min_{p_k\in\mathcal{P}_j^*}\{\underbrace{AC_{j^*}}_{T_{j^*}}, \sum_{l\in p_k} P_l\} = P_{j^*}$. Hence, we have,

$$
\begin{aligned}
& \frac{EFF(\theta_1, ..., \theta_s) - \sum_{i\neq j^*} P_i\theta_i}{\theta_{j^*}} \leq \min_{p_k\in\mathcal{P}_j^*}\left\{AC_{j^*}, \sum_{l\in p_k} P_l\right\} = P_{j^*} \\
\Leftrightarrow \quad & EFF(\theta_1, ..., \theta_s) - \sum_{i\neq j^*} P_i\theta_i \leq P_{j^*}\theta_{j^*} \\
\Leftrightarrow \quad & EFF(\theta_1, ..., \theta_s) \leq P_{j^*}\theta_{j^*} + \sum_{i\neq j^*} P_i\theta_i \\
& = \sum_{i=1}^{s} P_i\theta_i \\
& = T^{max}_{\lambda^*}
\end{aligned}
$$

as desired.

Part (b): $\Rightarrow$) Suppose the egalitarian solution $\text{EG}^{T_{\lambda^{**}}} = (\lambda_1^*, ..., \lambda_s^*)$exists, then it is efficient, core stable and limit routing-proof. By proposition 1, we know that the maximal revenue given by the RP algorithm is always an upper bound for which EG can achieve. Moreover, the RP algorithm respects stand-alone core stability and limit routing-proofness. Thus, if the egalitarian exists, then there must exist a price vector $\overline{P} = (P_1, ..., P_s)$ that is also efficient, core stable and limit routing-proof. Hence, by Theorem 1(a), condition (i) or condition (ii) is satisfied.

$\Leftarrow$) Firstly, note that condition (i) clearly implies condition (ii). Secondly, if condition (ii) is satisfied, then we have that $EFF(\theta_1, ..., \theta_s) \leq \sum_{i=1}^{s} P_i\theta_i = T^{max}_{\lambda^*}$ by Theorem 1. Hence, by proposition 1 and continuity, there exists λ^{**} such that $T_{\lambda^{**}} = EFF(\theta_1, ..., \theta_s)$ when we slowly decrease the revenue target from $T^{max}_{\lambda^*}$ to $EFF(\theta_1, ..., \theta_s)$. This λ^{**} will generate exactly the egalitarian solution $\text{EG}^{T_{\lambda^{**}}} = (\lambda_1^*, ..., \lambda_s^*)$. ∎

8.3 Proof of Corollary 1 of Theorem 1 and Proposition 1

Proof. Suppose without loss of generality that agents demand link i, EG starts charging $\lambda^* > 0$ to all agents. At any point in time, the stand-alone constraint of every link will be reached.

Without loss of generality, suppose the stand-alone constraint of link i is reached first, i.e., $\lambda^* = AC_i(\theta_i) = \lambda_i^*$. By assumption,

$$\lambda_i^* = AC_i(\theta_i) \leq \min_{p_k \in \mathcal{P}_i} \left\{ \sum_{l \in p_k} AC_l(\theta_l) \right\} \Rightarrow \frac{C_i(\theta_i)}{\theta_i} \leq \sum_{l \in p_k} AC_l(\theta_l) \qquad \forall p_k \in \mathcal{P}_i$$

$$\Rightarrow \lambda_i^* \leq \sum_{l \in p_1} AC_l(\theta_l), ..., \lambda_i^* \leq \sum_{l \in p_t} AC_l(\theta_l)$$

This implies that agents on link i are charged at the lowest cost among all feasible alternative paths p_k that connect link i, so agents will not have an incentive to reroute. Limit routing-proofness is trivially satisfied by the assumption. Note that λ_i^* satisfies the stand-alone core stability since $\lambda_i^* = AC_i(\theta_i)$. It also satisfies efficiency since agents on link i are being charged at the lowest cost.

Now, it remains to check the cases $\lambda_1^*, ...\lambda_{i-1}^*, \lambda_{i+1}^*, ...\lambda_{s-1}^*$. Note that the assumption

$$AC_i(\theta_i) \leq \min_{p_k \in \mathcal{P}_i} \left\{ \sum_{l \in p_k} AC_l(\theta_l) \right\}$$

holds for all $j = 1, ..., s$, hence by the same reasoning, we can deduce that $\lambda_1^*, ..., \lambda_{i-1}^*, \lambda_{i+1}^*, ...,$ λ_{s-1}^* are also core stable, limit routing-proof and efficient. Now, without loss of generality, suppose the stand-alone cost of link s is reached last, then by efficiency,

$$\lambda_s^* = \frac{EFF(\theta_1, ..., \theta_s) - \sum_{i \neq s} C_i(\theta_i)}{\theta_s}$$

Assumption (9) then implies that,

$$\lambda_s^* = \frac{EFF(\theta_1, ..., \theta_s) - \sum_{i \neq s} C_i(\theta_i)}{\theta_s} \leq AC_s(\theta_s) \leq \min_{p_k \in \mathcal{P}_i} \left\{ \sum_{l \in p_k} AC_l(\theta_l) \right\}$$

Thus, each agent who demands link s is paying at λ_s^*, which essentially is the lowest cost among all other feasible paths. Hence, λ_s^* is also routing-proof, core stable and efficient. Therefore, EG has a solution $EG_{\lambda^{**}}^T = (\lambda_1^*, ..., \lambda_s^*)$ such that every λ_i^* for all $i = 1, ..., s$ is limit routing-proof, core stable and efficient. ■

References

[1] Melo E., 2011. *Congestion pricing and learning in traffic network games.* Journal of Public Economic Theory, Association for Public Economic Theory, vol. 13(3), pages 351-367, 06.

[2] Roughgarden T., 2003. *The Price of anarchy is independent of the network topology.* Journal of Computer and System Sciences, vol. 67(2), pages 341-364.

[3] Moulin H., 2009. *Pricing traffic in a spanning network.* Proceedings of the 10th ACM Conference on Electronic Commerce, EC09, Stanford.

[4] Juarez R., Kumar R., 2013. *Implementing efficient graphs in connection networks.* Economic Theory, Springer, vol. 54(2), pages 359-403.

[5] Chakrabarty D., Mehta A., Nagarajan V., 2005. *Fairness and optimality in congestion games.* Proceedings of the 6th ACM conference on Electronic commerce, EC '05, Pages 52-57.

[6] Carol M., Andreas S., 2008. *The complexity of congestion games.*

[7] Joel S., 2007. *Linear programming note X Integer programming.*

[8] Frederick H., Gerald L., 2009. *Introduction to operations research.*

[9] Thomas C., Charles L., Ronald R., Clifford S., 2009. *Introduction to algorithms.*

[10] Moulin H., 2013. *Cost sharing in networks: some open questions* International Game Theory Review (IGTR), World Scientific Publishing Co. Pte. Ltd., vol. 15(02), pages 1340001-1-1.

[11] Dominique H., Moulin H., 1996. *Traffic-based cost allocation in a network.* RAND Journal of Economics vol. 27, no. 2, pp. 332-345.

[12] Dong B., Guo G., Wang Y., 2012. *Highway toll pricing.* European Journal of Operation Research, Elsevier, vol. 220(3), pp. 744-751.

[13] Alazzam A., Lewis III H., 2013. *A new optimization algorithm for combinatorial problems.* (IJARAI) International Journal of Advanced Research in Artificial Intelligence, vol. 2, no. 5.

[14] Bogomolnaia A., Holzman Ro., Moulin H., 2010. *Sharing the cost of a capacity network.* Mathematics of Operations Research, vol. 35, No.1, pp.173-192.

[15] Dutta B., Mishra D., 2011. *Minimum cost arborescences.* Games and Economic Behavior, Elsevier, vol. 74(1), pp. 120-143.

[16] Hougaard JL., 2009. *An introduction to allocation rules*. Springer.

[17] Fotakis D., Spirakis P., 2007. *Cost-balancing tolls for atomic network congestion games*. Proceeding WINE'07 Proceedings of the 3rd international conference on Internet and network economics, pp. 179-190.