

The RIMO Gateway Selection Approach for Mesh Networks: Towards a Global Internet Access for All [†]

Khulan Batbayar ¹, Emmanouil Dimogerontakis ¹ , Roc Meseguer ¹ , Leandro Navarro ¹ , Esunly Medina ² and Rodrigo M. Santos ³ 

¹ Department of Computer Architecture, Universitat Politècnica de Catalunya, Barcelona 08034, Spain; batbayar@ac.upc.edu (K.B.); edimoger@ac.upc.edu (E.D.); meseguer@ac.upc.edu (R.M.); leandro@ac.upc.edu (L.N.)

² BAO Systems, Washington, DC 20007, USA; emedina@baosystems.com

³ Department of Electrical and Computers, Universidad Nacional del Sur, CONICET, Bahía Blanca 8000, Argentina; ierms@uns.edu.ar

* Correspondence: ierms@uns.edu.ar; Tel.: +54-291-4595101 (ext. 3304)

† Presented at the 12th International Conference on Ubiquitous Computing and Ambient Intelligence (UCAmI 2018), Punta Cana, Dominican Republic, 4–7 December 2018.

Received: date; Accepted: date; Published: date

Abstract: Community wireless mesh networks have emerged as cooperative initiatives to provide Internet Access in areas where traditional ISP costs are not affordable for the population. It is common in wireless mesh networks sharing several capacity limited Internet gateways to provide Internet access. As routing does not handle capacity planning, end-users have to select gateways in such a way that the overall capacity of all gateways could be used effectively. An efficient gateway selection should minimize the processing logic and measurements over the mesh network. Selecting a high performance gateway can also ensure that the overall network load is balanced. This paper presents *RIMO*, a standalone best-effort algorithm for client nodes to select their preferred gateway without interacting with other client nodes. *RIMO*-based selection matches the gateway performance of the reference *brute-force* and *omniscient* algorithms for 60% of the test duration while reducing the gateway performance measurement cost from a factor of n to 2. With a reduced overhead and high efficiency, the *RIMO* algorithm automates the aggregation of multiple Internet gateways in wireless mesh networks, which results in robust last mile Internet connectivity to people in vulnerable situation.

Keywords: gateway selection; access to Internet; randomized sampling; load balancing

1. Introduction

Internet access has become a de facto requirement to participate in today's social activities; for instance, to access public services, education material, social media and also to support the everyday work of millions of organizations. However, the majority of the world's population is not yet online [1], far from the vision of a "universal service". Among many others, the major obstacle in providing Internet access to everyone is the high cost involved, as implied by the relation between the GDP per capita and Internet access, shown in Figure 1. In fact, the ITU's Measuring Internet report [2] or the Deloitte's Value of connectivity report [3] highlight a strong link between the Internet uptake and high GDP growth rate, number of jobs and income for both, the developed and developing world. Global access to the Internet requires a dramatic reduction in the connectivity costs especially in geographical areas and population sectors with low infrastructural penetration [4]. Network infrastructures are, in most cases, controlled by former monopolies, now telecom incumbencies. These entities control the offer and have the strength to influence regulation and discourage competitors. Except in developed urban areas, there is usually a lack of competitive offers, defined as "market failure" [5]. This affects

negatively people’s lives by reducing their capability of improving their digital literacy and increasing the digital divide. Rural poor communities and particular age brackets (such as children and elders) are especially vulnerable to this situation. This widens the digital divide between some communities, regions or countries and the rest of the world.



Figure 1. Internet/Web Access Not For All (Source: World Bank, 2016).

Bottom-up initiatives in local communities follow the path of cooperation instead of competition, based on the observation that sharing resources can contribute to reduce connectivity costs. As a way to mitigate the challenge of the high Internet access cost, in many regions worldwide the citizens self-organize to explore alternative models for getting local connectivity and Internet access under accessible conditions, creating collaborative network infrastructures built by citizens and organizations, who pool their resources and coordinate their efforts. However, this requires technology and services that support cooperative organizations, commodity devices, and decentralized infrastructures, which is quite opposite to the main direction of industrial developments.

IEEE 802.11-based wireless mesh networks (WMNs) have been used as an economic and convenient technology to provide connectivity to rural areas, especially in developing regions. Even in developed countries, WMNs have provided Internet connectivity to a large geographic area. There are many examples of networks that can fit in this scheme. The paper [6] outlines 18 cases, with 9 described in detail, and 267 potential cases in 41 countries. Wireless routers are placed on the roof or at poles to form a mesh network, as a backbone. Some of these routers are connected to the Internet and function as gateway nodes for the network that can be shared by participants in the mesh network. Contrary to classic operators with a centrally planned network in terms of capacity and coverage, WMNs grow organically, so we cannot assume capacity planning. Therefore bottlenecks can appear as the mesh grows in coverage, number of participants and number of Internet gateways or their equivalent aggregated capacity.

As [7] shows, the capacity bottleneck is not only on the wireless mesh network but also on the gateways (a overloaded node) and on the Internet access connections (a limited, congested and/or slow connection). This poses an important research question: how should the mapping of clients to gateways be performed in order to use successfully the aggregated capacity of these limited Internet access connections reached over the mesh network?

To answer such a question, the main goal of this paper is to propose a gateway selection mechanism that uses informed decision-making with minimal gateway performance monitoring overhead. This mechanism uses randomized sampling to introduce unbiased gateway selection at different client nodes and to reduce the gateway probing request overhead at the gateway nodes.

The main contributions of this research work are the following: (1) The use of *latency to content*, as the most representative metric that would allow a meaningful ranking of gateway choices.

(2) The design of the *Randomized Informed Minimized Overhead* (RIMO) algorithm, where the client node selects independently and efficiently (with as less monitoring and processing overhead as possible) a gateway node to connect, while considering the global network-wide load balancing.

The paper is structured as follows. The next section presents the proposed RIMO algorithm for gateway selection. Section 3 describes the results of the experimental evaluation of the algorithm performed on a real-world Community Network. A discussion of the results and possibilities of improvements is presented in Section 4. A comparison of the proposed approach with some related works is presented in Section 5. Finally, conclusions and future research directions are outlined in Section 6.

2. The Proposed RIMO Algorithm

Fixed and mesh wireless network consist of different subsets of topologies and equipment (heterogeneous in nature) that are part of an access network. Therefore, centralized algorithms, such as [8], or homogeneous algorithms, such as [9], are not suitable for this network setting. The centralized algorithms often involve periodic performance measurement, proactive gossiping and information processing in order to select one gateway node. However, in a limited resource network, it is often prohibitive to introduce an additional processing overhead in central or in gateway nodes. On the other hand, decentralized gateway selection algorithms require strict cooperation, scheduling and organization between network nodes, often associated to a large amount of communication interchanges between the nodes involved [10].

The described network environment imposes the following design requirements to a gateway selection algorithm: (1) Decentralized, (2) Scalable to a medium-sized Wireless Mesh Network, (3) Incremental implementation (4) No changes in infrastructure. Compatible with existing communication technologies, transport protocols, and (5) Compatible with the heterogeneous environment. Based on these requirements, we form our main design considerations, which are: (1) Informed decision-making (2) Low monitoring overhead (3) Client-side decision-making.

A brute-force approach, where each client node measures the performance of all gateway nodes, would impose large amount of network traffic and eventually overload the nodes with the measurements required to provide the best gateway selection, which involves a significant waste of the scarce connectivity. At the other extreme is a simple non-informed gateway selection leads to load imbalance over different gateway nodes by over-popularizing some gateways. The idea behind the RIMO approach proposed in this paper is finding a balance between a simple non-informed gateway selection method (random), which has zero communication overhead but can lead to poor performance, and a comprehensive informed gateway selection (brute-force), which has a significant communication overhead but can achieve the best overall performance.

We argue that a partial knowledge of the gateways' performance can be used to select a "good-enough" gateway and to avoid the worst choices. The idea of exploiting partial knowledge while providing some options to the client node led us to explore approaches based on the *power of two choices* [11]. This algorithm has proved to be effective on distributing the computing resources over a vast number of client services and has been implemented in many state-of-the-art data-center wide congestion control algorithms [12,13].

The design of the gateway selection framework broken down into two-layered architecture. The bottom layer, the Probing layer, is responsible for: (1) to filter and assign a weight to the all available gateways and (2) to measure the performance of a random sample of gateways. The candidate list is passed to the top layer, the Selection layer, where the final gateway selection is made.

In our Wireless mesh network scenario, $\mathbf{C} = \{C_1, C_2, \dots, C_i\}$ denotes the set of clients, and $\mathbf{\Gamma} = \{G_1, G_2, \dots, G_j\}$ the set of available gateways.

The idea of Algorithm 1 is to weight and filter all the available gateways to improve the result of the randomized selection by eliminating bad choices. The **user's preferences** (Filter procedure, line 1) are some user-based reasons (e.g. the economic cost and the trust) to filter the available gateway list $\mathbf{\Gamma}$

and obtain a subset of gateways Γ_{filtered} . The **impact to the network** (HOPStoGateways procedure, line 9) is minimized by a weighted mechanism that reduces the network usage. In this mechanism the weight of each gateway has an inversely linear relationship with the number of hops between the client and the gateway: $MaxHOPS + 1 - hops$ (line 12). This can be explained due to the fact that communication with a close gateway uses less network resources (traffic on links) than communication with a far one. As a secondary impact of this weighting mechanism, we expect a slightly better network performance, as shown in [7]. The output of the Algorithm 1 is the weighted gateway list Γ_{weighted} . We use network hops as a quasi-static client-gateway distance so the result of the algorithm is also quasi-static.

Algorithm 1 Weighting gateways

Input: $\Gamma = \{G_1, G_2, \dots, G_j\}$ ▷ Available gateway list
Input: *preferences* ▷ User preferences

```

1: procedure FILTER( $\Gamma$ , preferences)
2:   for all gateway  $\in \Gamma$  do
3:     if gateway  $\in$  preferences then
4:        $\Gamma_{\text{filtered}} \leftarrow \text{ADD}(\textit{gateway})$ 
5:     end if
6:   end for
7:   return  $\Gamma_{\text{filtered}}$ 

8: procedure HOPSTOGATEWAYS( $\Gamma_{\text{filtered}}$ )
9:   for all gateway  $\in \Gamma_{\text{filtered}}$  do
10:    hops  $\leftarrow$  HOPS(gateway)
11:     $\Gamma_{\text{weighted}} \leftarrow \text{ADD}(< \textit{gateway}, MaxHOPS + 1 - hops >)$ 
12:   end for
13:   return  $\Gamma_{\text{weighted}}$ 

```

Output: Γ_{weighted} ▷ Weighted gateway list

The randomized sampling mechanism in Algorithm 2 is used to reduce the probing overhead in the network while still providing a minimal set of choices for the client. The candidate list contains a maximum of three gateway nodes, including two random gateways from the weighted list Γ_{weighted} (line 2) and one gateway to which the client was previously connected. The measured candidate list $\Gamma_{\text{candidates}}$ is an input of the selection algorithm presented in Algorithm 3, which selects the best performing gateway from the candidate list.

Algorithm 2 Probing gateways

Input: Γ_{weighted} ▷ Weighted gateway list
Input: *previouslyUsedGateway*

```

1: procedure PROBEGATEWAYS( $\Gamma_{\text{weighted}}$ )
2:    $\{C_1, C_2\} \leftarrow \text{WEIGHTEDRANDOMSET}(\Gamma_{\text{weighted}}, 2)$  ▷ Two candidates
3:   for all candidate  $\in \{C_1, C_2, \textit{previouslyUsedGateway}\}$  do
4:     CONNECT(candidate)
5:     sample  $\leftarrow$  PROBE(candidate)
6:      $\Gamma_{\text{candidates}} \leftarrow \text{ADD}(< \textit{candidate}, \textit{sample} >)$ 
7:   end for
8:   return  $\Gamma_{\text{candidates}}$ 

```

Output: $\Gamma_{\text{candidates}}$ ▷ Candidate list

The overall the gateway selection mechanism is performed every time a user needs a new gateway, which can be either when the node has recently connected to the network or when the node is not satisfied with the current gateway performance. The previously selected gateway in the candidate list is used for consistent gateway selection. Changing the gateway could introduce short term performance degradation at the client node, therefore we try to be consistent with the choice of the gateway.

Algorithm 3 Selecting the gateway to connect

Input: $\Gamma_{candidates}$ ▷ Candidate list

1: **procedure** SELECTGATEWAY($\Gamma_{candidates}$)
 2: $selectedGateway = BEST(\Gamma_{candidates})$
 3: **return** $selectedGateway$

Output: $selectedGateway$ ▷ The selected gateway

3. Evaluation Results*3.1. Experimental Environment*

We performed experiments in the guifi.net community network [14], a real heterogeneous wireless mesh network environment. The metrics and the selection mechanism were instantiated in the Community-Lab.net experimental testbed where nodes acting as clients interacted with a set of gateways. For the experiments, we were given access to 5 end-nodes (clients) and 12 gateway nodes, also being used by real guifi.net users, distributed in the Barcelona city area. We collected a dataset to analyze and select a suitable metric for client's quality of experience (QoE) estimation. A total of 307217 rounds, every 5 s over a period of 15 days of requests were collected.

To validate the proposal, the Omniscience metric was defined, a synthetic and post-experiment metric, to show the real performance of the gateway. This metric is defined as the average HTTP content latency of a window of 5 min from the selection moment.

3.2. Selection of the Probing Metric

Different types of metrics were collected. Metrics that measure the Client-Gateway relationship: topology hops, Routing hops and ICMP round-trip time (Ping RTT). We also measured end-to-end metrics (Client-Gateway-Server): TCP handshake time, Time-To-First-Byte (TTFB) of a HTTP request, Content Latency of an HTTP request and Content Size of a HTTP request. In addition, all these metrics were correlated against the Omniscience metric in order to select the closest one in terms of the estimated gateway performance perceived by the client. Two different correlation methods were used: Pearson correlation, which measures the linear association; and Spearman correlation, which measures monotonic association (not only linear).

Figure 2 shows the ECDF of the correlations of Content Latency and TTFB with the Omniscience metric for all client-gateway pairs. The Content Latency metric is defined as the time to download a small probing content using the gateway. The TTFB is defined as the time duration from the client request to the reception of the first byte of content. The best correlation line is the closest to the right side: the Content Latency using Spearman's method. Spearman's correlation achieve better results than Pearson correlation for the majority of the correlations computed for all client-side metrics. It means that the relationship is closer to a monotonic association than to a linear association.

Figure 2 shows a moderate positive relationship for Content Latency using Spearman's correlation that ranges from 0.20 to 0.78 for different clients and gateways, with a mean of 0.45. The 40% (the 0.6 value of the y axis) of client-gateway pair correlations have a higher correlation value than 0.5. The rest of the client-side metrics show weak or no relationship by achieving an average of 0.05 in a range from -0.39 to 0.19 . As expected, the metrics based on the transport layer and the end-to-end metrics are the best suitable ones for the gateway selection algorithm since they are able to detect three different types of bottlenecks: congestion at the gateway node, saturated Internet access connection and congested inter-network path.

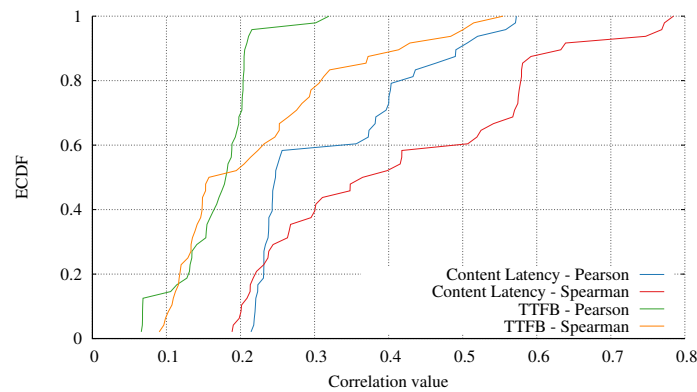


Figure 2. Correlation for Content Latency and TTFB.

Figure 3 shows an example the relation between the Omniscience metric and the Content Latency as a cloud of points for a specific client-gateway pair. The figure reveals a moderate correlation between them, and is representative for the rest of the client-gateway pairs, whose figures we excluded for brevity. Figure 3 also contains the performed Principal Component Analysis (PCA), whose vectors represent the principal components of the data. The first principal component is [0.424 0.905]. The explained variance shows how much information (variance) can be attributed to each one of the principal components. In this example, the first principal component contains 84.19% of the variance, hence representing the majority of the information. As a result, it contains enough information to prove the moderate relationship between the Content Latency and the Omniscience metric.

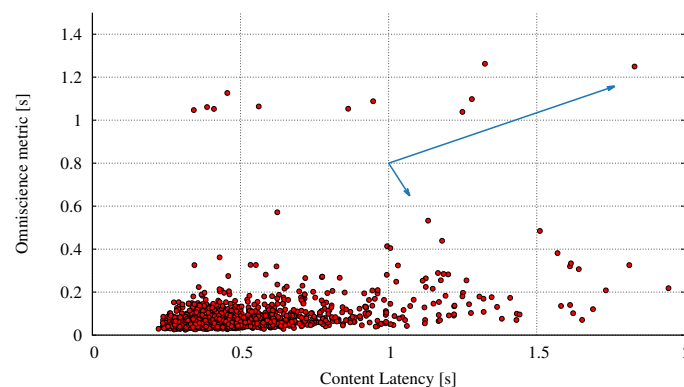


Figure 3. Scatter plot of the client 3 using the gateway 10.138.85.130.

For the gateway selection process, we select the Content Latency as client-side metric and active probing of the gateways. In a production environment, the metric could be passively collected (using the HTTP traffic of the real users). However, the experiment shows that the Content Latency metric is dependent to the data size. Therefore, to be able to collect passively, it is recommended to reduce the range of possible data sizes.

3.3. Results of the Selection Process

Each client performs a gateway selection process every round for validation purposes. The RIMO proposal is compared to: **Random-** Not informed selection, **Brute-force-** Best informed selection based on complete knowledge of the gateways' performance. Probing all gateways, **Power of Two-** Randomized informed selection based on partial knowledge of the gateways' performance. Probing only two gateways, **Omniscient-based-** Post-experiment selection based on the best selection using the Omniscience metric.

The Figure 4 shows an empirical cumulative distribution function (ECDF) of the performance of the different selection gateway approaches. The performance of RIMO is better than the Random and the Power of Two approaches, but also very close to the Brute-force method. For the 80% of the requests (0.8 of the y axis) the Content Latency is lower than 0.03, 0.09, 0.18, 0.21 and 0.54 s for the Omniscient-based, Brute-force, RIMO, Power of Two and Random approaches respectively.

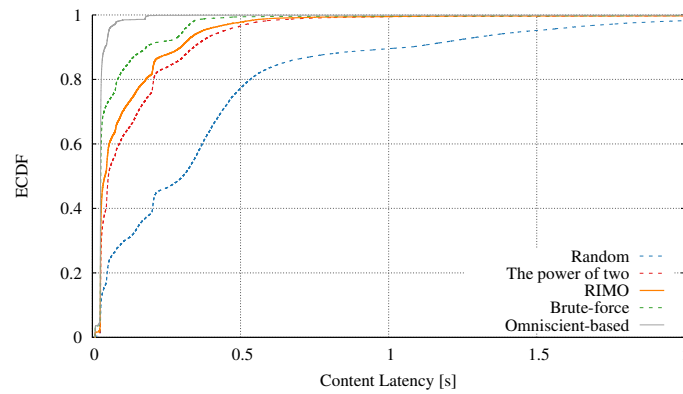


Figure 4. Client perception of the selected gateway's performance.

Analyzing deeper the results, we observe that a user perceives an additional delay when the best gateway is not selected. This cost in time can be defined as the difference between the Content Latency of the approach and the Omniscient-based as a best one. For the 80% of the requests this difference is lower than 0.06, 0.14, 0.18 and 0.5 s for Brute-force, RIMO, Power of Two and Random approaches respectively. These additional delays are small enough to not have any impact on the users QoE.

From the overhead perspective, Table 1 shows the network cost of the different selection approaches in terms of messages per selection round, where Γ is the set of gateways and C is the set of clients. The RIMO algorithm is significantly better than the Brute-force approach, being close to zero communication overhead approaches (e.g. Random). It is important to notice that the cost of the Brute-force approach depends linearly on the number of available gateways and the number of clients involved in the selection process at the current round. The experimental results consider 5 clients and 12 gateways.

Table 1. Message overhead per selection round.

Strategy	Analytical	Experimental
Brute force	$O(\text{size}(\Gamma) * \text{size}(\mathbf{C}))$	59.39
RIMO	$O(\text{size}(\mathbf{C}))$	9.95
Power of Two	$O(\text{size}(\mathbf{C}))$	9.94
Random	$O(0)$	0

As far as load balancing is concerned, Figure 5 shows the distribution of the clients by gateway of the different selection approaches. The proposal (RIMO) balances the load among the gateways, while the Brute-force approach overloads the best gateways. This load balancing behaviour is specially good in a scenario with gateways with limited resources, where only the best gateways cannot handle the clients' load. A possible problem of the Brute-force approach is that all clients performing the selection process may select the same gateway, and it can be overloaded, similarly to a flash crowd event.

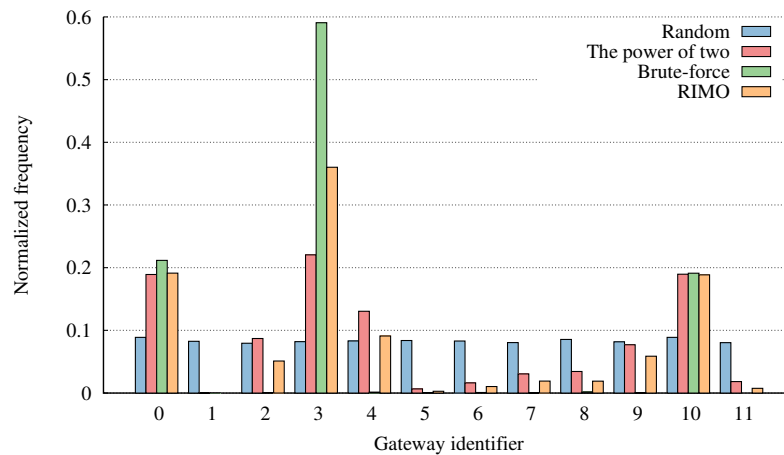


Figure 5. Load Balancing

4. Discussion

The evaluation results show that the proposed RIMO algorithm is effective. However, there are some details to be discussed.

Concerning **Less-Than-Best-Effort gateway selection**, RIMO does not find the best possible gateway, but it avoids the selection of a poor performing gateway as much as possible while offering an affordable gateway choice. Considering an environment where Internet access is a scarce recourse, users that have Internet connectivity can share their exceeding bandwidth but not necessarily provide the expected QoE. Despite the fact that RIMO can set the basis for a broader access to the Internet, it comes without performance guarantees, following a Less-Than-Best-Effort (LBTE) access model. Under this perspective, a client of a LBTE service cannot expect an optimal QoE. Nevertheless, through the RIMO approach, the selection of an overloaded gateway can be avoided, which would deteriorate the QoE, and at the same time would affect the contributor’s QoE due to the scarcity of resources.

Concerning **Scalability**, the proposed gateway selection algorithm is applicable to medium sized wireless networks. The proposal is good for a real-world WMNs (e.g., 84 nodes on QMP-Sants mesh network and 222 nodes on Funkfeuer Wien mesh network.) It could also be applied to a large scale network (e.g., the guifi.net is close to 35,000 nodes and 400 gateways) because the message overhead is constant regardless of the network size. However, in a large scale network, the number of gateways would be too big to be able to select randomly a good proxy or at least to not select a bad one, since currently only two gateways are probed to perform the selection.

Concerning **Performance**, the overall performance of RIMO depends on the distribution of the gateways’ performance, and more specifically on the poor performing ones. A possible solution to deal with this variability in performance is to introduce a collaborative probing of all the available gateways or a collaborative gateway selection. Another option is to adapt the current proposal so that instead of considering the entire network, it could only use the proximity of the client. For this, the filter of the Algorithm 1 could be adapted. It could filter and weight a set of close proxies (from the client’s proximity). The *MaxHOPS* value of the network could be changed for a proxy selection horizon *Horizon*. Moreover, an analysis to define client’s proximity and change the *Horizon* accordingly must be performed.

5. Related Work

There are some interesting works on informed gateway selection for mesh networks. To our understanding, most these works have some important limitations. Proposals like [15,16] fail in heterogeneous environments since they are based in solutions that operate on the mesh routing layer and require modifications to the infrastructure routers. A similar work presented in [17], also operates on the mesh routing layer and requires additional software on the gateways’ side. In [18] the authors

present a flexible and not only network-based criteria (e.g., economical features or user preferences) for gateway selection. However, it cannot be easily implemented and deployed on a heterogeneous and distributed community network because it is a conceptual proposal based on game theory. The proposal in [19] fails on the low monitoring overhead requirement, since it uses a kind of brute-force approach periodically probing all the available gateways.

Some of the works previously mentioned, despite proposing interesting solutions, lack practical implementations or testing in a real-world environment. In terms of experimentally tested solutions, the main limitation is the size of these environments, too small and simple for large scale validation and to model a community network (e.g., a outdoor wireless mesh network with only 10 routers or only two gateways), that do not allow for generalization of their results.

To achieve a low monitoring overhead, some works use collaborative probing [8–10]. In [8], the monitoring modules collect statistics about network usage. This approach fails in relying on centralization. All the collected information must be sent and stored near the controller. In [9], the clients collaborate to probe the gateways and use the results to select one. It is based on a brute-force approach. The work presented in [10] also uses collaboration among clients to probe the gateways without using brute-force. Each client only probes one gateway and shares the information on this overlay network in a gossiping way. Our work builds on this. The works presented in [8–10] proposes solutions but at the cost of a high monitoring overhead, because all of them rely on periodic probing. The RIMO proposal only probes the gateways once, when the client needs a gateway or the user has changed their preferences.

Another important difference between RIMO and other approaches is the types of metrics used to probe the gateway performance. These metrics should be sensitive to three different bottlenecks: an overloaded gateway, an overloaded Internet connection, and an overloaded internal path. Most of the gateway selection methods fail on detecting all these bottlenecks because they use the IP layer and internal mesh network metrics (e.g., RTT from client to gateway). To detect gateway or Internet connection overload an end-to-end metric in the application layer should be used as RIMO does.

Finally, there is a clear understanding that availability and cost, together with local regulations are the main barriers to accessible Internet connectivity [2,3,20]. Furthermore, Community Networks, relying on wireless mesh networks and the sharing and pooling of Internet gateways, provide an effective alternative to develop shared networking infrastructures [21,22], expand connectivity at a affordable cost [23,24] and contribute to sustainable socio-economic local development [25–27].

6. Conclusions

This paper presents the proposal and evaluation of RIMO, a gateway selection algorithm for Community Networks. This algorithm allows to easily share several Internet gateways in a mesh access network. The result is a simple, robust and inexpensive way to deliver Internet connectivity, particularly in populations exposed to vulnerable conditions, that cannot afford proper network capacity planning and traffic management. RIMO improves the network utilization, achieving good performance with a low traffic overhead. RIMO achieves a balance across client nodes using gateways which increases the perceived Internet performance and enhances the resilience of the access network.

As future work, it would be interesting to explore a collaborative approach to extend current algorithm. In the present version of RIMO the informed gateway selection is limited to the client node. Therefore, a light-weight collaborative sensing and selection method would allow to share gateway performance information across client nodes without having to do the measurements themselves.

Funding: This research was funded by the EU Horizon 2020 Framework Program project netCommons (H2020-688768), by the EMJD-DC program, by the Spanish Government under contract TIN2016-77836-C2-2-R, and by the Generalitat de Catalunya as Consolidated Research Group 2017-SGR-990.

Conflicts of Interest: The authors declare no conflict of interest.

References

1. Internet Society. Global Internet Report. 2015. Available online: <http://www.internetsociety.org/globalinternetreport/> (accessed on 1 June 2018).
2. ICT Facts and Figures. 2017. Available online: <https://www.itu.int/en/ITU-D/Statistics/Pages/facts/default.aspx> (accessed on 1 June 2018).
3. Consulting, D. Value of Connectivity: Economic and Social Benefits of Expanding Internet Access. Available online: <https://www.bl.uk/collection-items/value-of-connectivity-economic-and-social-benefits-of-expanding-internet-access> (accessed on 1 June 2018).
4. GAIA WG. Global Access to the Internet for All Research Group. 2016. Available online: <https://irtf.org/gaia> (accessed on 1 June 2018).
5. Navarro, L.; Freitag, F.; Baig, R.; Roca, R. A Commons-oriented Framework for Community Networks. In *Community Connectivity: Building the Internet from Scratch*; Chapter A Commons-Oriented Framework for Community Networks; Internet Governance Forum: Paris, France, 2016; pp. 25–71.
6. netCommons. *Deliverable D1.2, Report on the Existing CNs and their Organization (v2)*; EU: Brussels, Belgium, 2016.
7. Dimogerontakis, E.; Meseguer, R.; Navarro, L. Internet Access for All: Assessing a Crowdsourced Web Proxy Service in a Community Network. In Proceedings of the 19th International Conference on Passive and Active Measurement, PAM, 2017, Berlin, Germany, 30–31 March 2017, pp. 72–84.
8. Abujoda, A.; Dietrich, D.; Papadimitriou, P.; Sathiaselan, A. Software-defined wireless mesh networks for internet access sharing. *Comput. Netw.* **2015**, *93*, 359–372. doi:10.1016/j.comnet.2015.09.008.
9. Ko, B.J.; Liu, S.; Zafer, M.; Wong, H.Y.S.; Lee, K.W. Gateway selection in hybrid wireless networks through cooperative probing. In Proceedings of the Integrated Network Management (IM 2013), Ghent, Belgium, 27–31 May 2013; pp. 352–360.
10. Dimogerontakis, E.; Neto, J.; Meseguer, R.; Navarro, L.; Veiga, L. Client-side routing-agnostic gateway selection for heterogeneous Wireless Mesh Networks. In Proceedings of the Integrated Network and Service Management (IM), Lisbon, Portugal, 8–12 May 2017; pp. 377–385.
11. Mitzenmacher, M. The power of two choices in randomized load balancing. *IEEE Trans. Parallel Distrib. Syst.* **2001**, *12*, 1094–1104.
12. Zhang, H.; Zhang, J.; Bai, W.; Chen, K.; Chowdhury, M. Resilient Datacenter Load Balancing in the Wild. In Proceedings of the Special Interest Group on Data Communication (SIGCOMM), Los Angeles, CA, USA, 21–25 August 2017; pp. 253–266.
13. Ghorbani, S.; Godfrey, B.; Ganjali, Y.; Firoozshahian, A. Micro Load Balancing in Data Centers with DRILL. In *ACM Workshop on Hot Topics in Networks*; ACM: New York, NY, USA, 2015; pp. 17:1–17:7, doi:10.1145/2834050.2834107.
14. Baig, R.; Roca, R.; Freitag, F.; Navarro, L. guifi.net, a crowdsourced network infrastructure held in common. *Comput. Netw.* **2015**, *90*, 150–165. doi:10.1016/j.comnet.2015.07.009.
15. Boushaba, M.; Hafid, A. Best path to best gateway scheme for multichannel multi-interface wireless mesh networks. In Proceedings of the Wireless Communications and Networking Conference (WCNC), Cancun, Mexico, 28–31 March 2011; pp. 689–694.
16. Acharya, P.A.; Johnson, D.L.; Belding, E.M. Gateway-aware routing for wireless mesh networks. In Proceedings of the Mobile Adhoc and Sensor Systems (MASS), IEEE, San Francisco, CA, USA, 8–12 November 2010; pp. 564–569.
17. Ashraf, U.; Abdellatif, S.; Juanole, G. Gateway selection in backbone wireless mesh networks. In Proceedings of the Wireless Communications and Networking Conference, (WCNC), IEEE, Budapest, Hungary, 5–8 April 2009, doi:10.1109/WCNC.2009.4917735.
18. Wang, X.; Qu, D.; Li, K.; Cheng, H.; Das, S.K.; Huang, M.; Wang, R.; Chen, S. A flexible and generalized framework for access network selection in heterogeneous wireless networks. *Pervasive Mob. Comput.* **2017**, *40*, 556–576.
19. Salsano, S.; Patriarca, F.; Lo Presti, F.; Ventre, P.L.; Gentile, V. Accurate and efficient measurements of IP level performance to drive interface selection in heterogeneous wireless networks. *IEEE Trans. Mob. Comput.* **2018**, doi:10.1109/TMC.2018.2807842.

20. International Telecommunications Union. *Measuring the Information Society Report 2017*; International Telecommunications Union: Geneva, Switzerland, 2017.
21. ITU. *Broadband Situations in Rural and Remote Areas*; ITU-D; International Telecommunication Union: Geneva, Switzerland, 2014.
22. ITU. *Telecommunication for Rural and Remote Areas*; ITU-D; International Telecommunication Union: Geneva, Switzerland, 2010.
23. Rey-Moreno, C.; Roro, Z.; Tucker, W.D.; Siya, M.J.; Bidwell, N.J.; Simo-Reigadas, J. Experiences, challenges and lessons from rolling out a rural WiFi mesh network. In Proceedings of the ACM Symposium on Computing for Development (ACM-DEV), Bangalore, India, 11–12 January 2013; p. 11.
24. Song, S. *Unlocking Affordable Access in Sub-Saharan Africa*; 2016. GCIG Paper No. 43; Centre for International Governance Innovation: Waterloo, ON, Canada, 2016.
25. Baig, R.; Dalmau, L.; Roca, R.; Navarro, L.; Freitag, F.; Sathiaseelan, A. Making Community Networks Economically Sustainable, the Guifi.Net Experience. In Proceedings of the GAIA '16 Proceedings of the 2016 workshop on Global Access to the Internet for All Global Access to the Internet for All (GAIA), Florianopolis, Brazil, 22–26 August 2016; pp. 31–36, doi:2940157.2940163.
26. Rey-Moreno, C. *Supporting the Creation and Scalability of Affordable Access Solutions: Understanding Community Networks in Africa*; Internet Society: Reston, VA, USA, 2017.
27. Micholia, P.; Karaliopoulos, M.; Koutsopoulos, I.; Navarro, L.; Baig, R.; Boucas, D.; Michalis, M.; Antoniadis, P. Community Networks and Sustainability: A Survey of Perceptions, Practices, and Proposed Solutions. *IEEE Commun. Surv. Tutor.* **2018**, in press.



© 2018 by the authors. Licensee MDPI, Basel, Switzerland. This article is an open access article distributed under the terms and conditions of the Creative Commons Attribution (CC BY) license (<http://creativecommons.org/licenses/by/4.0/>).