Situation-aware routing for wireless mesh networks with mobile nodes

Hlabishi I. Kobo, William D. Tucker
Department of Computer Science, University of the Western Cape, Private Bag X17, Bellville 7535 South Africa.
*Tel: +27 21 9592461, Fax: +27 21 9591274, Email: {**2654952**, btucker}@uwc.ac.za*

Abstract: This paper describes a situation-aware algorithm based on the current situation of a mesh network with mobile nodes that improves quality of service. After running laboratory performance tests, we concluded that a situation-aware mesh routing protocol offers potential to address issues pertaining to mobility, congestion and scalability in dynamic mesh networks with mobile nodes. Such networks appear promising to provide connectivity to people living in rural areas in developing regions of Africa, and can be easily interconnected to telco-styled networks through gateways for voice and Internet services. Such services can remain free in the mesh, yet can also be billed for interconnection. Our vision offers an attractive business model for up scaling a rural customer base for telcos, while at the same time offering increased quality of service for mobile users on rural mesh networks.

Keywords: Open Source Software, ICT for eInclusion and eAccessibility, Mobile Applications

1. Introduction

This paper presents a situation-aware routing metric calculation prioritizing the most recent link quality data to inform routing decisions on static wireless mesh networks (WMN) with mobile nodes. This type of network can be referred to as dynamic mesh network. We believe these types of mesh networks will become prevalent as mesh network protocols improve and mobile devices become more powerful and able to run such protocols. Fig. 1 illustrates the dynamic mesh network concept. Our goal is to improve the effective usage of link quality using situation-aware routing in BATMAN (described below) in order to optimize quality of service (QoS) and throughput on such networks. We chose BATMAN because it has high stability level and high packet delivery ratio [1]. We tested the protocol enhancement on two experimental test bed setups: small scale and large scale. The core factors of this research include mobility, congestion and scalability. The results from the small test bed are encouraging, and not too far off the mark from the original BATMAN. They are however better than the original BATMAN in the larger scaled network.

A Better approach to mobile ad-hoc network (BATMAN) is a proactive mesh network routing protocol. BATMAN's control messages, called originator messages (OGMs), are relatively small packets of about 52 bytes. BATMAN's nodes do not maintain the routing information of the entire network [1]. Rather, each node only maintains information about the best next-hop towards the destination [1] [2]. This reduces the signal overhead and avoids unnecessary knowledge about the whole network. The objective of this protocol is to enhance the probability of delivering a packet. The protocol maintains information about the existence of a node and thus does not check the quality of a link [3].

All BATMAN nodes periodically send/broadcast OGMs. Each OGM contains the original sender's address, address of the node rebroadcasting the OGM, TTL (time to live) and a sequence number. The sequence number is incremented for each OGM, i.e. the first OGM gets 1 and so on. Thus, BATMAN also keeps track of the freshness of an OGM. Any sequence number received with a value lower than the previous one gets dropped [2]. The TTL is used to limit the number of hops on which the packets must pass through before it expires (gets dropped). Upon receiving the OGM, each node then rebroadcasts it to its neighbours. However, each node only rebroadcasts OGMs coming through the current best next-hop. The number and the reliability of the OGMs determine the route discovery as well as neighbour selection.

Fig. 1: A dynamic mesh network, applicable to a rural area, with static nodes inside homes, and mobile nodes on cell phones. Note that link quality in such networks is continually changing as phones move around.

The rest of the paper is organized as follows: Section 2 reviews related work. Section 3 outlines the objectives of this paper. Section 4 presents methods and the experimental setting. Section 5 presents preliminary results and final results, and discusses them. Section 6 presents business benefits. Section 7 concludes the paper and recommends some future work.

2. Related work

Routing is a process of delivering data packets from a source (sender) node to destination (receiver) node on a network. Routing protocols deal with the maintenance, creation, establishment and discovery of such routes [2]. Routing protocols are based on three protocol classification categories: reactive, proactive and hybrid.

Reactive protocols are on-demand; they create a route from source to destination only when needed, i.e. when there is actual data to be sent. This scheme uses network flooding to find the routes [2] [5]. It is suited for mobile ad-hoc networks where there are frequent topological changes due to the mobility of routers [1] [5]. According to [5], flood based route discovery provides high network connectivity and low message overhead. More importantly, the method does not waste bandwidth by propagating control packets when it is not necessary [1]. This scheme, however, leads to higher latency on the network because of route discovery. [2] Argues that reactive protocols are more suitable for a network with static traffic patterns whilst proactive protocols suit dense networks with bursty traffic patterns.

Ad-hoc on demand distance vector (AODV) is one of the popular reactive protocols and hence creates routes on demand. AODV has single path routing and is based on hop-by-hop routing [2]. Single path routing means that a node can only have one path towards a destination [2]. The AODV routing table only stores information about the best next-hop towards a destination [2] [6]. Sequence numbers are used to ensure loop-free routes and to ensure the freshness of the routing information [7]. The AODV protocol uses unicast, broadcast, as well as multicast for communication on the network. It uses broadcast to flood route requests, then the intermediate nodes and the destination nodes send a unicast route reply [7]. There are multicast groups where a multicast of sequence numbers takes place [6] [7]. Other reactive protocols include Ad On-demand multipath distance vector routing (AOMDV), Dynamic Source Routing (DSR), Link Reversal Routing (LRR) and Associativity Based Routing (ABR).

In **proactive protocols**, each node in the network maintains a table containing routing information of the entire network. Each node then periodically broadcasts control packets to the whole network to let other nodes know about its existence. The routing information is periodically updated to maintain the adequacy of the routing information. The biggest advantage of this scheme is the minimization of route discovery delay and consequently lower latency in delivering a packet. However because of the periodic updates of control messages that get propagated through the entire network, the overhead increases. Thus, bandwidth consumption also rises and also requires a lot of memory for the tables.

Optimized link state routing (OLSR) protocol is a proactive protocol based on a link state (LS) algorithm. OLSR's objective is to reduce the size of the control packets as well as the overhead cost by broadcasting control packets [4]. This protocol is an optimization of the link state protocol for mobile ad-hoc networks [4]. It uses a hop by hop routing metric. Multipoint relays (MPR) are the key concepts in OLSR. MPRs are the subsets of the neighbours of which a node uses to forward broadcast messages. MPRs reduce duplicate retransmission in the same region and thus minimize flooding overhead [4].

Fisheye State routing (FSR) is also a LS routing protocol inspired by the fish-eye technique created to reduce the size of information required for graphical data representation [8]. Other proactive protocols include Destination-Sequenced Distance Vector (DSDV), Wireless Routing Protocol (WRP).

The BATMAN algorithm (described in Section I) is also a proactive protocol. However, it experiences serious flaws in dealing with asymmetric links. BATMAN advanced, referred to as BATMAN-adv, and is a Layer 2 protocol introduced to overcome this setback by using a Transmit Quality (TQ) algorithm. BATMAN-adv consists of two fundamental functions: receiving link quality (RQ) and transmit link quality. Receiving link quality deals with the probability of transmitting a packet successfully towards a node [9]. The transmitting link illustrates the probability of transmitting a packet successfully towards a neighbor [9]. TQ is the most important because RQ does not influence the routing decision. TQ is determined by the by the number of received OGMs. Echo link quality (EQ) is the number of the rebroadcasted OGMs from neighbors. TQ is calculated by dividing the EQ by the RQ i.e. TQ = EQ/RQ [9].

Hybrid protocols exhibit the behavioural design of the two above mentioned protocols. Hybrid protocols are very challenging because the switch from one protocol to another needs to be very sharp. However, this is still a major concern and thus hybrid protocols are still theoretical rather than practical due to their complex implementation [1].

Zone Routing Protocol (ZRP) is a zone based hybrid protocol. ZRP proactively maintains routing information for the local neighbourhood, referred to as the routing zone [11]. It reactively acquires routes to destinations that are outside the routing zone. Zone-based Hierarchical Link State (ZHLS) routing protocol is another zone based hybrid routing protocol, and is based on global positioning system (GPS) [12]. Other hybrid protocols includes SHARP (Hybrid Adaptive Routing Protocol) and HARP (Hybrid ad hoc routing).

Routing protocols use **metrics** to select the best routing path. Several situation-aware routing metrics have been proposed, as well as applied, in many routing protocols.

Expected transmission count (ETX) is a situation-aware metric which considers the number of MAC layer transmissions needed to successfully deliver a packet through a link [5] [13]. The ETX metric captures the effects of packet loss and path length. Each node broadcasts probe packets to its neighbors and they send a back a reply/report [13]. The metric is calculated by the number of probe packets received by its neighbor in both directions. ETX is isotonic, thus ensures easy calculations of minimum weight paths [5]. The ETX metric does not consider bandwidth, interference, or the link transmission variance [5]. Other metrics includes expected transmission time (ETT), weighted cumulative ETT (WCETT) and the metric of interference and channel-switching (MIC).

3. Objectives

The former United Nations secretary-general Kofi Annan observed, "*Wireless technologies have a key role to play everywhere, but specially in developing countries and countries with economies in transition. With considerable speed and without enormous investments, Wi-Fi can facilitate access to knowledge and information, for example by making use of unlicensed radio spectrum to deliver cheap and fast Internet access. Indeed, it is precisely in places where no infrastructure exists that Wi-Fi can be particularly effective, helping countries to leapfrog generations of telecommunications technology and infrastructure and empower their people*." [10]

In an effort to address information access related concerns in the rural areas, this paper describes our efforts to optimize BATMAN for a dynamic mesh network. The crux of the problem is to optimize the routing protocol so it can adapt and react quickly to rapid and dynamic topological changes. We propose situation-aware methods to improve the routing decisions based on link quality to achieve better QoS and throughput. BATMAN's routing algorithm checks for the existence of a link and increases the probability of delivering a packet through that link [4]. Our method adapts the routing protocol to use a simple weighting mechanism. We believe recent packets provide a clearer indication of link quality at a particular moment in time so we give recently received packets more weight in the routing decision.

4. Methodology

This section describes how we optimized the BATMAN protocol to make situation-aware routing decisions based on link quality. Given the mobility of mobile nodes, rapid topological changes in a dynamic mesh network are inevitable. Thus, the ideal approach is to take the current network situation into consideration when making routing decisions. In BATMAN, the best link is measured by the highest number of OGMs received from the destination over a current sliding window. Much can happen within a second in an ad hoc wireless network. Any link with a sliding window that records a lot of OGMs at the beginning and fewer at the end due to superior link strength at the beginning stands a chance of being the best as opposed to the one that records a lot towards the end but fewer in total. For example, suppose one has a sliding window of 10, link L1 records [1111100000] with 5 OGMs at the front, and link L2 [0000001111] with 4 OGMs seen at the end. BATMAN will chose L1 as the best next hop because of the higher number of OGMs, but actually, the current best option would be L2 because the most OGMs have arrived there more recently.

Our method prioritizes the recently received OGMs in the sliding window, and would therefore correctly choose L2 over L1. We sum the indices on which OGMs were recorded in a given window. From the example above, we would have link L1: 1+2+3+4+5 = 15 and link L2: 7+8+9+10 = 34. This is a more accurate numeric representation describing the current situation of the two links.

This section explains the experimental design and procedure to evaluate our BATMAN modification. We created a preliminary mesh network test bed composed of four nodes as shown in Fig. 2. Subsequently the mesh test bed was extended to 12 nodes (see Fig. 3).

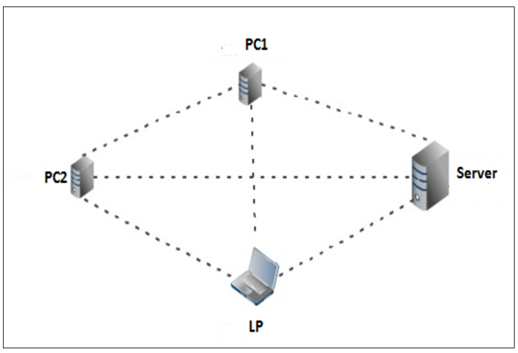

Fig. 2: The small scale experimental test bed, consist of 2 static and 1 mobile nodes. It is controllable and easy to debug. Adopted from [14] where they also tested BATMAN.

Eight mesh potato nodes were added to the initial test bed. A mesh potato is a wireless access point combined with an Asynchronous Telephone

Adapter (ADT). Two laptops were used in this case to take the mobility evaluation to length. All nodes ran Linux version 2.6.32-31 with 802.11bg network cards. BATMAN advanced version 2011.2.0 was used.

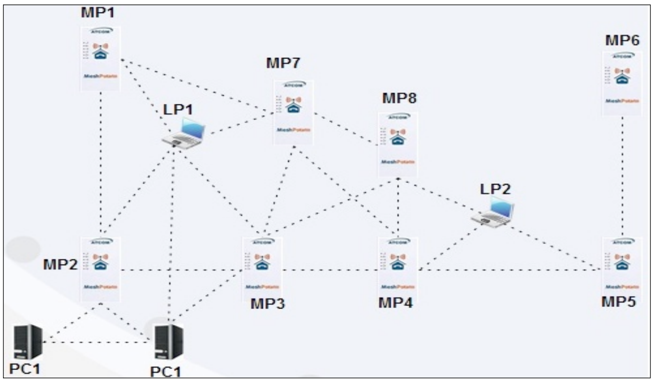

Fig. 3: The larger scale test bed with 12 nodes used to evaluate the impact of network growth on BATMAN oriented mesh network.

The experiment was designed to compare the performance of unmodified BATMAN-adv with our modified version on a dynamic mesh network. Our main objective is to show that situation-aware routing is viable and effective in a dynamic WMN. The test parameters examined were jitter, packet loss and throughput. We assume that jitter also covers latency. These performance metrics were used to investigate the impact of mobility, congestion and scalability on dynamic networks. The mobility was realized by moving the laptops around in hand during the experimentation. The congestion procedure is explained below while the scalability was evaluated by comparing the overall performance between the small scale and the large scale test bed.

Iperf was used to generate packets and monitor performance. We set a transfer interval of 60 seconds with a report back of 10 seconds. This was run 10 times for each parameter (herein referred as 10 flows). During the transfer interval, Iperf sent about 4000 UDP (User Datagram protocol) packets, with a maximum size of 1500 bytes. The parameters were tested over 1MB, 100MB and 150MB speeds whilst buffer size was varied over 41KB, 31KB and 11KB. The first comparison combination consisted of all the transfer speeds with the default buffer size of 41KB. The second comparison combination applied the buffer size variations to the default transfer rate of 1MB/s. lastly, the 150MB/s rate was applied to the 11KB buffer size to achieve maximum congestion of the compared rates and buffer sizes.

5. Results

The results presented in Table 1, Table 2 and Table 3 of jitter, packet loss and throughput respectively are from the larger scale test bed. The small scale results are summarized. We measured average jitter, packet loss and throughput with the rate/buffer size combinations mentioned above. Each cell is an average of 10 flows e.g. the value 28.08ms in Table 1is an average of 10 flows fromLP1 to the server under 1MB-41KB settings.

Table 1: Jitter comparison from larger scale test bed.

Jitter							
	1M 41K	100M 1K	150M-41K	150M-1K	1M-1K	1M-1K	Average
BATMAN-adv							
LP1	28.08	51.10	49.35	302.5	27.94	21.97	**80.165**
LP2	40.29	54.70	55.22	31.85	39.67	22.19	**40.657**
PC2	96.19	46.29	47.01	27.72	19.07	16.51	**42.164**
BATMAN-adv modified							
LP1	35.12	57.49	52.93	28.87	26.34	21.46	**37.025**
LP2	26.30	48.67	52.72	31.88	20.17	21.66	**33.561**
PC2	2.067	48.67	52.72	31.88	20.17	21.66	**29.522**

Table 2: Packet loss comparison from larger scale test bed

Packet Loss							
	1M-41K	100M 1K	150M-1K	150M-11K	1M-31K	1M-11K	Average
BATMAN-ADV							
LP1	0.17	0.20	0.12	0.18	0.22	0.14	**0.173**
LP2	0.77	1.93	1.39	3.3	3.47	0.82	**1.946**
PC2	0.006	0.45	0.25	0.20	0.10	0.13	**0.189**
BATMAN-ADV modified							
LP1	0.02	0.03	0.03	0.018	0.042	0.040	**0.029**
LP2	0.14	0.03	0.05	0.11	0.04	0.15	**0.085**
PC2	0.002	0.21	0.08	0.16	0.28	0.22	**0.158**

Table 3: Throughput comparison from larger scale test bed

Throughput							
	1M-41K	100M-1K	150M-1K	150M-11K	1M-31K	1M-11K	Average
BATMAN-adv							
LP1	0.08	0.08	0.02	0.09	0.08	0.08	**0.074**
LP2	0.08	0.08	0.08	0.006	0.07	0.08	**0.066**
PC2	0.04	0.08	0.09	0.09	0.09	0.09	**0.081**
BATMAN-adv modified							
LP1	0.08	0.08	0.09	0.09	0.08	0.09	**0.084**
LP2	0.08	0.08	0.09	0.08	0.09	0.08	**0.083**
PC2	0.12	0.09	0.09	0.08	0.083	0.08	**0.090**

The results show that our metric is well suited for unstable and dynamic networks under strenuous circumstances. Jitter which is a variation of packet delay, showed a huge difference between the protocols in the small scale test bed. BATMAN-adv original shows the best (low) jitter of less than 55ms across all variation settings. The jitter is consistent irrespective of the transfer rate or the buffer size. Node LP, which had mobility throughout the tests, achieved the best jitter compared to PC1 and PC2. On the other hand, the modified BATMAN-adv lacks consistency as some points rise abruptly, reaching 336.2ms. The results from the larger test bed are however different, the modified protocol recorded the best and consistent jitter compared to the original and thus improving significantly from small scale test bed (see Table 1).

The average packet loss results for BATMAN-adv original appear inconsistent in the preliminary tests. The most distinctive and significant factor in this case is the consistency of packet loss for BATMAN-adv across all settings while the modification shows reduction as per variation settings. At default, the average packet loss on the three links is about 8%. The loss rate then reduces proportionally to the transfer rate and buffer size. This shows that situation-aware routing metrics perform well on large and inconsistent networks with congested links. In the larger scale tests, the packet loss reduced significantly was show inTable 2. The lowest recorded packet loss can be seen on the mobile nodes, LP1 and LP2 which proves the effectiveness of situation-aware routing technique in dynamic situations. The results show no practical relation between jitter and packet loss.

Unlike packet loss, the consistency in jitter correlates well with the consistency in throughput. The average throughput is also independent of the variation settings. The average throughput for BATMAN-adv in the small scale is consistently at 0.08 MB/s. On the other hand our modified version tends to fluctuates a bit. The maximum recorded throughput in flow for BATMAN-adv is 0.09MB/s while our modified version could reach

3MB/sec in a particular flow but due to its fluctuation tendency, the overall average amounts to 0.76MB/sec. However in the larger scale test bed, the maximum recorded in a flow was 1.12Mbyte per second in both protocols. The original BATMAN-adv recorded an averagely lower throughput than the modified version (see Table 3). The overall average between the three averages is 0.074 and 0.086 for BATMAN-adv original and BATMAN-adv modified respectively. Thus the modified protocol proved to be superior.

6. Business Benefits

Mesh networks enable 'bottom up' networks to be built quickly and inexpensively, and they can be independent of telecom providers like Telkom or Vodacom. A mesh potato network provides a seamless model of communication and information access in communities [15]. This model works such that local calls can be made for free and people only pay for breakout calls, which is how telcos can benefit with interconnection. A village telco also needs a gateway for Internet access. A typical operator's business model tends to cater for wealthy customers thereby marginalizing the poor. It is highly recommended that these big communication giants seize the opportunity of being part of this revolutionary development. This presents an opportunity for them to connect their existing infrastructure to bottom-up small enterprise networks in an 'over-the-top fashion'. This will mutually benefit both parties as the companies can charge for Internet provision and interconnect voice calls.

7. Conclusions

This paper demonstrated that the situation-aware method improves QoS on dynamic WMNs by making routing decisions based on the current situation of the network. BATMAN-adv counts OGMs received as a link quality measurement. We apply a prioritization technique to calculate the link quality metric. We give the more recently received OGMs more weight in deciding the link quality by summing their indices in a given window rather than counting their quantity. Therefore, more recently received OGMs contribute more to the metric in order to give a more precise indication of the current state of a link.

The results show little relation between jitter/PDV and packet loss. Jitter is, however, proportional to throughput. The original BATMAN-adv achieved the best jitter in the small scale experiment. The average throughput achieved on both protocols was almost the same. Our protocol modification suffered from packet loss at low bandwidth rates but this reduces as the transfer rate increases and buffer size shrinks, i.e. it performs well with congestion. However in the larger scale experiment, the modified version outperformed the original version in all performance metrics used. This was in the presence of an increased mobility and scalability. The congestion level was the same in both experimental setups. We have demonstrated that situation-aware routing offers a great potential solution to address issues pertaining mobility, congestion and scalability in dynamic mesh networks. We can also deduce that the number of nodes in any mesh network is directly proportional to its performance.

There were some **limitations** in both experimental design setups. The preliminary (small scale) test bed was in a single computer laboratory room with only four nodes. The distance between the nodes was small. Also, there are several other wireless networks accessible in the same room. Despite having a larger scale network test bed ideally perceived close to a real rural network it was still not enough. The challenge in this case being was the walls and the glasses dividing the building inside, the walls have built-in steals. These elements dissipate the network signal negating the performance. The persistent packet loss rate encountered in the larger scale test bed could possibly result from network interference.

In **future**, we would like to see how the protocol performs under a wider range of traffic patterns, and also in a more geographically spread mesh network and therefore a rural area is ideal environment to take this forward. It ought to be quite straightforward to a) recode the decimal-based index sliding window with binary bit-shifting to get it to consume fewer clock cycles and b) port it to BATMAN running on mobile phones such as Batphone.

References

[1] M. Abolhasan, B. Hagelstein, and J. C. P. Wang, "Real-world performance of current proactive multi-hop mesh protocols," *15th Asia-Pacific Conference on Communications, 2009. APCC 2009*, pp. 44–47.

[2] R. Pinto, *WMM-Wireless Mesh Monitoring*, Technical report, INESC-ID, 2009.

[3] D. Johnson, N. Ntlatlapa, and C. Aichele, "Simple pragmatic approach to mesh routing using BATMAN," *2nd IFIP International Symposium on Wireless Communications and Information Technology in Developing Countries, CSIR, Pretoria, South Africa*, 2008, pp. 6–7.

[4] A. Munaretto, H. Badis, K. Al Agha, and G. Pujolle, "QoS-enhanced OLSR protocol for mobile ad hoc networks," *Proceedings of ANWIRE 1st International Workshop, Glasgow Scotland*, 2003, pp. 171–183.

[5] Y. Yang, J. Wang, and R. Kravets, "Designing routing metrics for mesh networks," *IEEE Workshop on Wireless Mesh Networks (WiMesh)*, 2005.

[6] E. M. Royer and C. E. Perkins, "An implementation study of the AODV routing protocol," in *Wireless Communications and Networking Conference, 2000. WCNC. IEEE*, vol. 3, 200, pp. 1003–1008.

[7] M. K. Marina and S. R. Das, "Ad hoc on-demand multipath distance vector routing," *ACM SIGMOBILE Mobile Computing and Communications Review*, vol. 6, no. 3, 2000, pp. 92–93.

[8] G. Pei, M. Gerla, and T. W. Chen, "Fisheye state routing in mobile ad hoc networks," *ICDCS Workshop on Wireless Networks and Mobile Computing*, 2000, pp. D71–D78.

[9] S. Wunderlich, M. Lindner and W. Tsai "B.A.T.M.A.N advance" http://www.open-mesh.org/wiki/batman-adv.

[10] M. Khalil, "The wireless internet opportunity for developing countries," *I-WAYS-The Journal of E-Government Policy and Regulation*, vol. 27, no. 1, pp. 17-20, 2004.

[11] Z. J. Haas and M. R. Pearlman, "The performance of query control schemes for the zone routing protocol," *IEEE/ACM Transactions on Networking (TON)*, vol. 9, no. 4, 2001, p. 427–438.

[12] M. Joa-Ng and I. T. Lu, "A peer-to-peer zone-based two-level link state routing for mobile ad hoc networks," *Selected Areas in Communications, IEEE Journal on*, vol. 17, no. 8, 1999, pp. 1415–1425.

[13] M. Caleffi and L. Paura, "Bio-inspired link quality estimation for wireless mesh networks," *IEEE International Symposium on a World of Wireless, Mobile and Multimedia Networks & Workshops, 2009. WoWMoM 2009*, 2009, pp. 1–6.

[14] R. G. Garroppo, S. Giordano, and L. Tavanti, "Experimental evaluation of two open source solutions for wireless mesh routing at layer two," *5th IEEE International Symposium on Wireless Pervasive Computing (ISWPC)* , 2010, pp. 232–237.

[15] http://www.villagetelco.org/.