

Energy Efficiency Analysis by Fine Grained Modification in Link State Routing Protocol

Md Masduzzaman

Dept. of Computer Science, American International University-Bangladesh, Dhaka-1229, Bangladesh
E-mail: masud.prince@aiub.edu

Istiak Ahmed, Allin Arzoo, Tanuka Sharmin

Dept. of Computer Science, American International University-Bangladesh, Dhaka-1229, Bangladesh
E-mail: {kanak.ahmed65, allinarzoo,tanuka1992}@gmail.com

Received: 29 October 2018; Accepted: 18 November 2018; Published: 08 March 2019

Abstract—A major concern in modern explosive development of information and communication technology, especially in wired networking is to reduce of unnecessary energy consumption. It is needed because of expected environmental impact and potential economic benefits. These issues are usually referred as “Green networking”. It is related to increase the awareness of energy consuming in routing protocols, in the devices, including the structure of the whole networking system. Here different issues will be described which is topically related and the possible impact in green networking field. Also, special attention will be monitored in the routing protocol and algorithm especially link state routing protocol which is generally used for calculating the shortest path. This research paper tries to find out a new technique or modified routing technique that will be more effective for every router in terms of energy efficiency.

Index Terms—Energy efficiency, Link state advertisement, Link state database, Neighbor table, Topology Table, Routing Table.

I. INTRODUCTION

Core networks and Access networks consumes the highest amount of power which is almost 30% and 70% respectively in a classical ISP/Telco network. Lenge et al. [1] tries to anticipate the future energy consumption in the field of whole networking process. The largest part of energy is consumed by home networks. It can be clearly predicted that leading role of advancement will be played by IP/MPLS core networks in future [5].

It is necessary to analyze of those part where a handsome amount of improvements would be achieved before attempting to diminish energy consumption or energy reduction. The whole internet can be segmented into a core network and several types of access networks. Energy consumption levels are varied from one another in these different segments. Energy consumption has been analyzed by Roth et al [2] in 2002, considering of several types of global internet accessories. Figure-1 illustrates the information about the local area network devices which

are generally used like hubs, switches, routers etc.

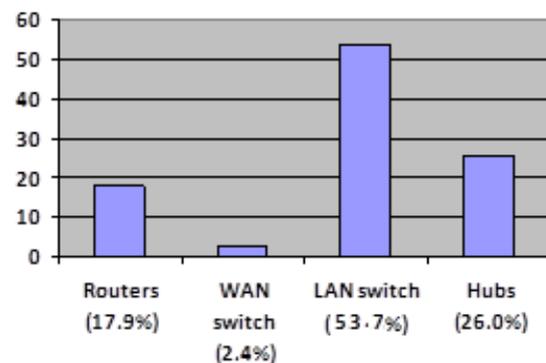


Fig.1. Contribution of different device types to the network energy consumption [5]

It is clearly observed from Fig 1 that graph that only hubs and switches were responsible for consuming almost 80% of total energy at that time. The authors of [3] estimated in 2005, that Network Interface Cards (NICs) and all other networks equipment's are responsible for near about half of the total power consumption in this universe. Deutsche et al. [4] estimated in 2009 that the power consumption of the core network will be identical to the power consumption of access network by the year of 2017.

It is quite complicated task to measure the energy efficiency in a device or in a network. Traditional way for saving energy in a core network is to route the traffic around some other routers during low traffic periods and respective routers or line cards can be put in a sleep mode. That can be also applicable for the IOS based switch [5]. Routers and switches are used in our daily communication channels that are consuming a huge amount of energy. These devices are also responsible for serious environment pollution. If we can reduce a little bit amount of energy loss, that can be remarked as a new step in networking world. Our approach will be the reduction of energy loss as much as possible in our traditional network.

II. RELATED WORK

Consciousness about Green House Gases (GHG) and its effect is increasing all over the world in recent years. Now, to decrease this GHG into a small amount of percentage, numerous studies and statistics have already been started throughout the world [15]. According to the report of European Union (EU), to keep the global temperature rising below 2% before 2020, it needs to decline the emission volume rate up to 15%-30% [6].

Considering the access of Information and Communication Technology (ICT) in everyday life, these branches are involved in GHG reduction objectives. Alone, ICT sector has been responsible to produce an approximately 2% of CO₂ in total man-made emission [7]. These evaluations are likely neither totally correct nor up-to-date because calculation of these numbers is quite challenging task. Nevertheless, all the studies in this sector agree on the fact that ICT is performed as an important source of energy consumption and GHG emissions. It has been an excellent opportunity to build network devices and protocols aware of energy consumption so that they can be more efficient [1]. So, our goal is to build up such a network system where energy will be not only efficient but also to help reducing the GHG from ICT sector if possible.

Traditional way for saving energy in a core network is to route the traffic around some other routers during low traffic periods [8]. But if the router is dead then the problem is that some data can be missed or then the nearest router can't realize that it is dead. Chabarek et al. [9] argues that energy awareness should be considered which has already been deployed in the routers over a set of point-of-presences as a design phase. They recommended that router which allows energy efficient operation, has some limitation to a subset of routers about power-consuming packet processing operation. Vasic and Kostic [12] demonstrate engineering techniques of traffic in a network. As a result, sleep state can be used in both the link and routers. The author has addressed 21% and 16% sleep ratios in the links and routers.

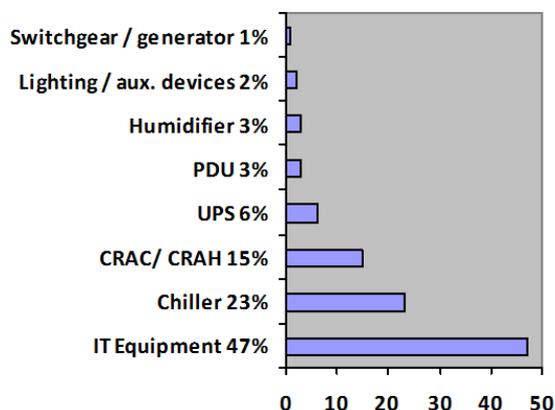


Fig.2. Total Distribution of power flow in a typical data Center [11]

In data center, the process of cooling system and server equipment's consumes 70% of the energy budget. The

network of the center consumes 10%-20% of its total power. It is also needed to be considered for consuming the energy in networking area [11].

Most of the article provides energy consumption solution for improving devices like switches, hubs or ports only. We want to work with devices as well as algorithms which are used now-a-days in modern networking area. But till now most of these algorithms are not considered for maintaining energy efficiency. Link state routing protocol is a well-known routing protocol in the devices to use for finding the shortest path. M. Sivabalan et al. [16] worked on the link state routing protocol based on network connectivity and the state information of all links and designed update mechanism and routing algorithm. But they didn't consider energy efficiency in terms of shortest path calculations in their design so that it can be more effective.

Here we don't want to change the whole algorithm or the whole arrangement of the network devices. We just want to develop a modified algorithm which will work considering energy efficiency and work accordingly.

III. PROPOSED METHOD

To maximize energy savings, the network-management system solves a problem regarding Link state routing protocol. It generally maintains three tables named neighbor table, topology table, and routing table. Firstly, list of all neighboring routers in an area is stored in a neighbor table. Then list of all possible routes to all known networks within an area is contained by the topology table. And finally, a routing table contains the best route to reach for listed network. There is a possibility of many neighbor relationships on the same segment in multi access network. This leads to an unnecessary amount of Link State Advertisement (LSA) traffic. To solve this problem in each multiaccess networks, OSPF elects a designated router (DR). Even it is also elected a Backup Designated Router (BDR) for backup purpose. These are the steps which are followed in link state routing algorithm: -

1. Each node creates the state of the link which is called Link state advertisement packet.
2. Link state advertisement packet is disseminated to all other router. It is known as flooding.
3. Shortest path tree is calculated for each node.
4. Routing table is calculated based on the tree.

In this section, we will try to explain why the Link-state advertisement has a quite harmful effect on CPU memory and power.

A. Neighbor table and Traffic Flow:

Two routers normally have a common network interface. Neighbor relationships are maintained and dynamically discovered by "Hello Protocol". By default, OSPF enabled interface sent Hello packets after every 10 seconds for broadcast and point-to-point interface and 30 seconds for non-broadcast and point-to-multipoint

interface. A relationship has been formed between selected neighbor routers for exchanging routing information. Not every pair of neighboring router becomes adjacent to exchange the routing table information. The process of creating this relationship is:

* Hello packets is sends and receives by the router from its neighboring routers. The destination address is typically a multicast address.

* Based on subject to protocol-specific parameters like same area neighbor, same hello interval and so on, the routers exchange hello packets. When the exchange is complete, router declares the neighbor as active or up.

* They synchronize their Link State Databases (LSDB) after completing neighbor adjacency form creation by using Hello packets. Then they exchange link state advertisement with adjacent router by confirming the receipt of Link State Acknowledgements. Routers LSDBs are now declared as synchronized as full adjacency state with each other.

* To ensure the full synchronization of link-state information, the routers can exchange any new LSAs to its neighboring routers inside an area if necessary.

*OSPF also has a dead interval which is by default 40 seconds. That means the dead interval timer is four times longer than the hello interval.

These are the steps to construct a Neighbor table: -

1. Down: No sign of any active router as a neighbor.
2. Init: Receive the hello packet.
3. Two-Way: In a collected hello packet, Router observes its own router ID to form DR and BDR.
4. ExStart: Master/slave roles determined.
5. Exchange: indicates that routers are exchanging Database Descriptors (DBDs).
6. Loading: Exchange of Link State Requirements and Link State Updates to populate LSDBs. That means routers are finally exchanging Link State.
7. Full: Neighbors are now fully synchronized.

B. Topology Table:

Routing table is formed by routing calculation on the database. This database is called as topology table. All routers within an area will have an identical topology database. Link-state database in an autonomous system indicates a directed graph where is formed of routers and networks. When two routers attached with one another via a point-to-point network, graph edge connects them. Router has an interface on the network is revealed when an edge connecting a router to a network. Those networks are known as transit who can carry data traffic that is neither locally originated nor locally destined. OSPF routers forward Link-state Advertisement to ensure the topology database is consistent on each router within an area. But there are several types of Link state advertisement like Router LSA(Type-1), Network LSA(Type-2), Network Summary LSA (Type-3) etc. Network's type is a vital factor (like point-to-point, broadcast or Point-to-Multi Point) for creating the

neighborhood in each network from a graph. Three cases are described in Figure 3, Routers are represented by rectangles. Networks are illustrated by circles and oblongs. point-to-point networks are depicted by Lines between routers.

C. Routing or Forwarding table

Link-state algorithms are the substructure of link-state protocols, which are also known as shortest path first (SPF) algorithms. After creating the topology table, these algorithms are applied on the topology table and creating a tree to find the best path, where this router acts as a root of this tree. After completing this tree, it builds up a new routing table or forwarding table to find the best path. This table is maintained on the base of the tree. Dijkstra Algorithm is followed in Open Shortest Path First (OSPF) routing protocol to calculate the shortest path.

Periodical updates of routing table are being sent regularly by the routers in Link-state routing protocols which is usually active in the intra-domain Network (Type-1). However, link-state routing protocols required more CPU power and system memory rather than Distance Vector Routing protocols because Distance Vector Routing protocol rely on routing-by-rumor, but link state doesn't. Another reason, Distributed map concept is used in link-state protocols. That reveals that each router contains a copy of network map and it is regularly updated. In addition, with the size of the routing table, the number of routers in an area and the number of adjacencies is also responsible to use more router memory and CPU usage in link state protocols. So, we want to reduce the process of Link state advertisement so that we can create energy efficiency technique in OSPF. Otherwise this traditional technique which are using now consumes more CPU power and system memory.

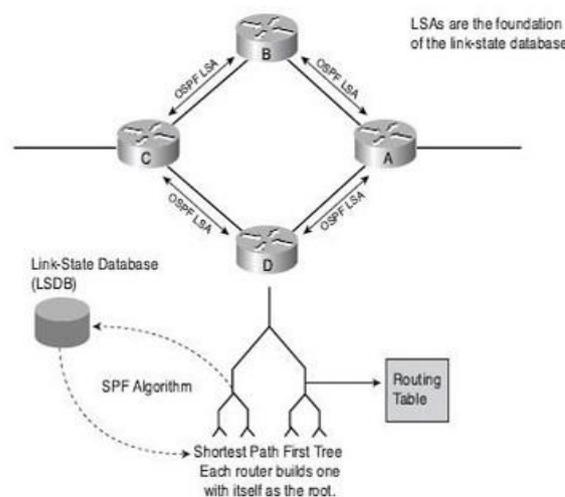


Fig.3. Link State Procedure

D. An optimal Solution to reduce the wastage of CPU power and memory:

A large amount of information is carried by a link state packet (LSP) like minimum amount of data, the node

identity, the list of links, sequence number, age etc. A node must be disseminated this LSP not only to the neighbor nodes but also to all other nodes. This process is known as flooding. This flooding process is normally known as an efficient and reliable way to exchange information. Every other node which has already been received this LSP then compare it with the copy it may already have. After comparing the old LSP with the newer one, it generally discards the old LSP and keeps the newer one. However, this topology is not sufficient to find out the shortest path. So, Dijkstra algorithm is used to create a shortest path tree from a given graph. Here we can see that for receiving LSP packet and make it change in the database table as well as in the routing table, a single node needs a lot of CPU power and memory. Now our approach is to establish a new technique which can reduce the pressure on router and additional CPU power with memory. If we can establish a technique so solve this problem, then a lot of CPU power and memory usage can be decreased. So, we are proposing below techniques to solve this issue.

a) *Reduce the number of LSA updates:*

Every Router sends their LSA updates continuously with their neighbor. We want to send LSA updates on Neighbor router if there are any changes in topology table which is effective. If we don't have any impact on routing table after receiving this LSA updates, then it is unnecessary to process this LSA updates. For this reason, we need to focus on a filter algorithm that can check it whether coming LSA has any impact on routing table or not.

b) *Minimize the database Overwhelm:*

After receiving every LSA packet, the link state neighbor table is compared with its previous data. For that every router must process it by using its CPU power and memory. So, if we can reduce the number of LSA updates or filter the updates packet to make any vital change in the topology table, we can minimize the usage of database overwhelm as well as CPU power and memory usages.

c) *Make the topology table more reliable and accurate:*

In General, traditionally the topology table is formulated from the neighbor table. Then Dijkstra algorithm is used to create shortest path for the destination. Without filtering these LSA update packets, it is not too much accurate and a lots of CPU power and memory is used every time. If we can create the shortest path after comparing the every LSA updates in the topology table, then it will be more accurate and unnecessarily we don't need to use CPU power and Memory as well.

E. *Proposed routing Algorithm for Our New table:*

NHRA(N0,T,PA)

NHRA=New Heuristic Routing algorithm

N0=Neighbor Topology table.

T= Traffic matrix tree.

PA=Performance Accuracy.

start:

```

Set G=N0; //New route Generate
Set Rn=RG(G,T) // Distance Computation
Set D1=(Rn,G,T)
do,
NHRA 2(T0,Rn,D1)
T0=Topology table
Rn=New route
D1=Computed Distance1
Start:
set G1=compare(D1,T0) // Calculate the new Distance
D2=(Rn,G1) //finalize the new route
Set
Rf=TC(Rn,D2)
Set
p: Rn|| Rf && D1||D2
END
while(p>=PA)
return (N0,G1)
END

```

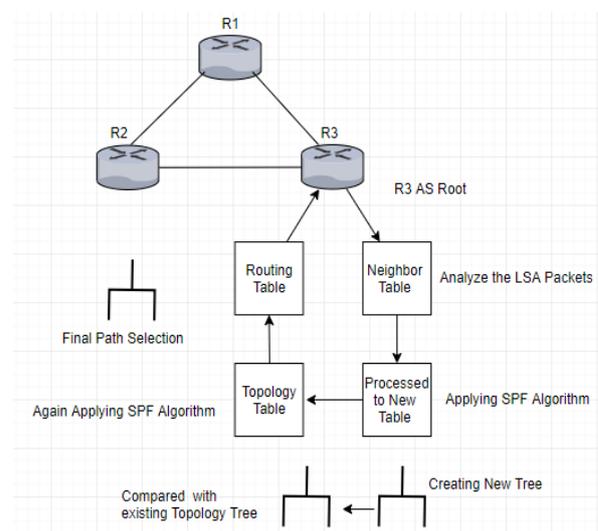


Fig.4. System Design

In this section, we propose a modified routing algorithm for an arbitrary neighbor table. To avoid packet disorder, we use connection-oriented routing, i.e., the packets from one flow take Multi path [13]. We proposed the algorithm which is divided in some few steps. Firstly, we introduce a traffic matrix tree, New Route generate process, distance computation process, and distance compare process, Finalize the new route. Here is our proposed modified algorithm: -

New route generation process: -

At first, we consider the neighbor table as a graph and hold it at a temporary graph G for our calculation. Then from that graph we only calculate distance of each neighbor node so that we can formulate a new route.

Route Distance Calculation Process: -

After that, we calculate distance for every node so that we can compare it with our existing topology table distance which is made earlier. If LSA update does not have any effect on topology table, we'll discard the LSA packets. As a result we don't need to update our topology

table every time. A partial amount of CPU power and memory will be saved every time.

Compare the Node Distance: -

After completing the new route and distance, compare this two with existing topology table distance to find the minimal distance.

Finalize the route for Routing or Forwarding table: -

After comparing, we are getting the new result to finalize new route and distance for travel the destination node with low cost via the best path.

IV. RESULT AND ANALYSIS

We conducted several experiments to understand the time and energy efficiency impact of link state routing protocol [14]. We have analyzed the time complexity of our proposed algorithm and currently used Dijkstra algorithm for finding shortest path.

Table 1. Proposed Algorithm time complexity Analysis

Algorithm	# of Times	Cost
NHRA(N0,T,PA):	-	-
// NHRA=New heuristic Routing algorithm	1	0
// T= Traffic matrix tree.	1	0
// PA=Performance Accuracy.	1	0
// N0=Neighbor Topology table.	1	0
START:	-	-
Set G=N0; //New route Generate	1	c1
Set Rn=RG(G,T) // Distance Computation	1	c2
Set D1=(Rn,G,T)	1	c3
NHRA 2(T0,Rn,D1)	1	c4
// T0=Topology table	1	0
// Rn=New route	1	0
// D1=Computed Distance1	1	0
Do:	-	-
START:	-	-
set G1=compare(D1,T0) // Calculate the new Distance	n-1	c5
D2=(Rn,G1) //finalize the new route	n-1	c6
Set Rf=TC(Rn,D2)	n-1	c7
Set p: Rn Rf && D1 D2	n-1	c8
END	-	-
while(p>=PA)	n	c9
return (N0,G1)	1	c10
END	-	-

$$\text{Total Cost, } T(n) = 0+0+0+0+c1+c2+c3+c4+0+0+0+(c5*(n-1)) + (c6*(n-1)) + (c7*(n-1)) + (c8*(n-1)) + (c9*n) + c10.$$

We compute the time complexity of our proposed algorithm and have compared it with existing Dijkstra's algorithm. From the analysis of table 1, we can see that the time complexity of our proposed approach is $O(|V|)$, where V is the number of adjacent nodes. There are two versions of Dijkstra's algorithm. The original Dijkstra's algorithm does not use a min-priority queue and its time complexity is $O(|V|^2)$ (where |V| is the number of nodes). On the other hand, the implementation of Dijkstra's algorithm based on a min-priority queue has the time complexity $O(|E| + |V| \log |V|)$ (where |E| is the number of edges).

Fig 5 displays the comparison of run times graphically between traditional way of Dijkstra's algorithm and our proposed algorithm in link state routing. In this graph, X-axis represents the number of nodes or routers and Y-axis represents the time. We can observe from that graph that when the number of nodes is one then the complexity is quite same for both algorithms. When the number of nodes is more than one and less than eight, then the original Dijkstra's algorithm which does not use a min-priority queue has higher time complexity but Dijkstra's algorithm which is used a min-priority queue has lower time complexity than ours. But when the number of nodes is more than eight, the time complexity of our proposed algorithm is quite better than the others. So, in the long run with multiple nodes or routers more than eight, our proposed methodology can be more effective in terms of energy efficiency.

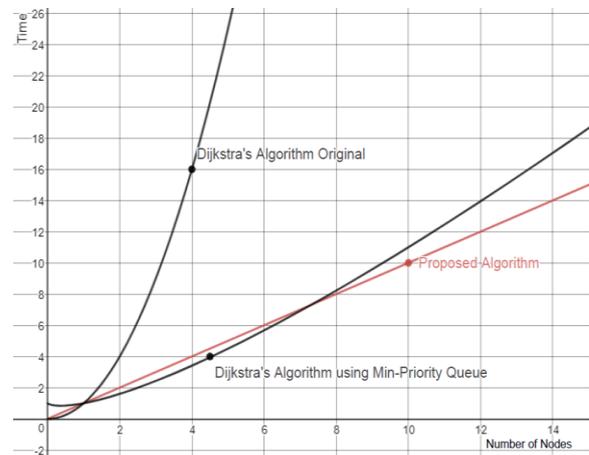


Fig.5. Time Complexity Comparison of Our Proposed Algorithm and Dijkstra Algorithm for both Original and Min-Priority Queue

V. CONCLUSION

Unfortunately, for last few years, no one is interested to working in this domain so that we can reduce the power consumption in current scenario of this world. We believe this technique we have presented must have a significant importance in practical application especially in link state routing protocol. After applying our proposed technique, router overwhelm will be reduced a little bit as well as using CPU power and memory. Routing path will be more accurate as we filtered the LSA packets. In a short brief we can say that earlier in link state routing protocol, three tables perform their act to detect to shortest destination path. Firstly, neighbor table, secondly topology table and thirdly routing table. In this process, firstly a router sends its neighbor router LSA packets with some metric (describe above) to verify the existing table. After getting the information, the router generates its neighbor table. Then it generates the topology table and applying here the Dijkstra algorithm to generate a tree. This router acts as a root and generates a tree or routing table to find the best shortest path. But in this traditional process, every time

router needs to modify its topology table. For solving this issue, we apply a technique that constructs a new table. It gets information from neighbor table and reconstructs a topology table. After that a new tree has been generated and compares it from previous tree of topology table. If any changes occurred in the tree, then it will modify the main topology table. Using this process, the topology table doesn't need to change every time that's why it is the best to reduce router overwhelm with CPU power and memory. This is our approach to reduce power consumption and partially reduce the wastage of CPU power and memory. On the other hand, if CPU power and memory usage can be decreased, then router will generate less heat. Now if we can find out some other techniques to reduce the usage of CPU power and memory in other devices like switches, hubs, servers etc. in our networking world, then greenhouse effect also might be reduced a bit in our living world.

REFERENCES

- [1] C. Lange, D. Kosiankowski, R. Weidmann, and A. Gladisch "Energy consumption of telecommunication networks and related improvement options" *Selected Topics in Quantum Electronics, IEEE Journal of, PP(99):1-11*, 2010.
- [2] K.W. Roth, F.Goldstein, and J.Kleinman "Energy Consumption by Office and Telecommunications Equipment in Commercial Buildings Volume I: Energy Consumption Baseline," *Tech. Rep.National Technical Information Service(NTIS), US department of Commerce,Jan,2002*.
- [3] B. Nordman and K. Christensen, "Reducing the Energy Consumption of Network Devices." *IEEE 802.3 Tutorial*, July 2005.
- [4] C. Lange, "Energy-related aspects in backbone networks," in *Proceedings of 35th European Conference on Optical Communication (ECOC 2009)*, (Wien, AU), September 2009.
- [5] Aruna Prem Bianzino, Claude Chaudet, Dario Rossi, Jean-Louis Rougier "A Survey of Green Networking Research" *Tech. Rep, Institut TELECOM, TELECOM ParisTech, CNRS LTCI UMR 5141, Paris, France*.
- [6] Bianzino, Aruna Prem, Claude Chaudet, Dario Rossi, and Jean-Louis Rougier. "A Survey of Green Networking Research", *IEEE Communications Surveys & Tutorials*, 2012.
- [7] Global Action Plan, "An Inefficient Truth." *Global Action Plan Report*, <http://globalactionplan.org.uk>, Dec, 2007.
- [8] Maruti Gupta and Suresh Singh "Greening of the internet" *In Proceedings of the 2003 conference on Applications, technologies, architectures, and protocols for computer communications, SIGCOMM '03, pages 19-26, New York, NY, USA, 2003. ACM*.
- [9] J. Chabarek, J. Sommers, P. Barford, C. Estan, D. Tsiang, and S. Wright "Power awareness in network design and routing " *In INFOCOM 2008. The 27th Conference on Computer Communications. IEEE, pages 457-465*, April 2008.
- [10] Aleksii Penttinen "Green Networking - A Literature Survey" *World Wildlife Fund and European Telecommunications Network Operators' Association*, April 2007.
- [11] Neil Rasmussen "Electrical efficiency measurement for data centers" APC, 2010. White Paper P:-154 v.2.
- [12] Nedeljko Vasic and Dejan Kostic "Energy-aware traffic engineering" *In Proceedings of the 1st International Conference on Energy-Efficient Computing and Networking, e-Energy '10, pages 169-178, New York, NY, USA,2010,ACM*.
- [13] Yunfei Shang, Dan Li, Mingwei Xu. "Energyaware routing in data center network", *Proceedings of the first ACM SIGCOMM workshop on Green networking - Green Networking '10*, 2010
- [14] Yanpei Chen, Archana Ganapathi, Randy H.Katz. "To compress or not to compress -compute vs. IO tradeoffs for mapreduce energy efficiency", *Proceedings of the first ACM SIGCOMM workshop on Green networking - Green Networking '10*, 2010
- [15] D. Pamlin and K. Szomolanyi, "Saving the Climate @ the Speed of Light-First Roadmap for Reduced CO2 Emissions in the EU and Beyond." *World Wildlife Fund and European Telecommunications Network Operators' Association*, April 2007.
- [16] M. Sivabalan and H. T. Mouftah "Design Considerations for Link-state Routing Protocols" *IEEE Symposium on Computers and Communications. ISCC'98*.

Authors' Profiles



Md. Masuduzzaman is a citizen of Bangladesh and currently working as a lecturer in the dept. of Computer Science at American International University-Bangladesh (AIUB). He has received his Masters degree specialized in Computer Networks and Architecture in 2015 and

received his Bachelor degree in Computer Science and Engineering in 2013 from American International University-Bangladesh (AIUB). His areas of interest includes Computer networks and architecture, Information and Network Security, Cloud Computing and Internet of Things (IoT). He is also involved in some different sports sector and voluntary work in his country.

LinkedIn Profile: <https://www.linkedin.com/in/masud-prince/>



Istiaq Ahmed did his graduation from American International University-Bangladesh on Computer Science & Engineering in 2014. Currently he is working for the IT Development Division of Dutch Bangla Bank LTD. He has worked in the Network development & Security part.

He has an enhanced experience to work with Cisco & Juniper router switch & firewalls. He is interested to research in the field of network security, protocols & Internet of Things (IoT).



Allin Arzoo did his graduation on Computer Science & Engineering from American International University-Bangladesh in 2014. Currently He is working for the IT Operation Department of Dutch Bangla Bank Limited. He uses to work in the network & system operations. He has a sound experience to work with IBM & HP servers and Cisco network products. He is interested to research in the field of network security, protocols & information technology.



Tanuka Sharmin has completed her graduation from American International University-Bangladesh on Computer Science & Engineering in 2014. She has worked for Accenture Bangladesh as Software Engineering Associates. She has an extensive experience in Database

Management. Beside this she has a great interest in Networking. She is interested to research in the field of database management system & energy efficient networking.

How to cite this paper: Masuduzzaman, Istiak Ahmed, Allin Arzoo, Tanuka Sharmin, "Energy Efficiency Analysis by Fine Grained Modification in Link State Routing Protocol", International Journal of Information Technology and Computer Science(IJITCS), Vol.11, No.3, pp.53-59, 2019. DOI: 10.5815/ijitcs.2019.03.06