

Algorithmic Implementation of Load Balancing –in Wireless LAN

Ms. Ashwini Raju Khawaskar

M.Tech Student 2st year Dept. of Computer Science and Engg, ,
Tulsiram Gaikwad Patil College of Engg and Technology
Nagpur.

Contact No:7028292802

E-mail : khawaskarashwini@gmail.com

Ms. Roshani Talmale

Guided By :

Mr. Rajesh Babu

Co-Guided By :

Abstract: Intra domain traffic engineering (TE) has become an indispensable tool for Internet Service Providers (ISPs) to optimize network performance and utilize network resources efficiently. Various explicit routing TE methods were recently proposed and have been able to achieve high network performance. However, explicit routing has high complexity and requires Large Ternary Content Addressable Memories (TCAMs) in the routers. Moreover, it is costly to deploy explicit routing in IP networks. In this project, we present an approach, called Generalized Destination-Based Multipath Routing (GDMR), to achieve the high performance as explicit routing. The main contribution of this project is to enhance an arbitrary explicit routing can be converted to a loop-free destination-based routing without any performance penalty for a given traffic matrix. We present a systematic approach including a heuristic algorithm to realize GDMR. Extensive evaluation demonstrates the effectiveness and robustness of GDMR.

Keywords : Destination-based routing, load balancing, multipath.

1 INTRODUCTION

In today's Internet Service Provider (ISP) network Traffic Engineering (TE) has been widely configures the parameters of the routing system to control traffic distribution across the network to optimize network performance and resource utilization. Given the highly competitive nature of the ISP market and the high cost of network resources [1], TE has become an indispensable tool for ISPs. Quite a few explicit routing TE methods were proposed in the last few years[2]. Explicit routing allows traffic flows of each source–destination pair to be distributed along predetermined paths (a flow can be flexibly de fined, e.g., 5-tuple header fields). Due to the fine-grained traffic distribution control that explicit routing offers, explicit routing TE methods can be used to achieve high network performance.

Explicit routing is supported by several routing mechanisms, such as Multi Protocol Label Switching (MPLS) and Software Defined Networking (SDN).However, with explicit routing, each router has to maintain a complex forwarding table. For instance, to distinguish source and destination addresses of packets, an explicit routing

forwarding table has to store, at worst, entries for a network with hosts. Due to the high cost-to-density ratio (US \$350 for a 1-Mb chip) and high power consumption (about 15 W/1 Mb) of Ternary Content Addressable Memory (TCAM) [10], routers have limited TCAM resources (e.g., the HP 5406 switch supports about 1500 288-bit TCAM entries). Explicit routing relies on TCAM to maintain line rate lookup and thus suffers from scalability issues. Explicit routing can also be deployed in IP networks by attaching to each packet the IP address of each node along the explicit path and forwarding packets hop by hop. However, this approach makes the overhead in the packet prohibitively expensive. Another category of TE is based on destination-based routing, where routers make forwarding decisions solely based on the destination addresses specified in packet headers. Thus, each router forwards packets targeted for the same destination in the same way regardless of the source addresses. Due to this hop-by-hop forwarding property, this type of routing has low forwarding complexity. Each router is only required to maintain a simple forwarding table with, at worst, entries for a network with hosts. Moreover, destination-based routing can save 100% TCAM consumption by storing destination-based forwarding entries

in Random-Access Memory (RAM). Most destination-based routing TE methods aim to optimize Interior Gateway Protocol (IGP) link costs to achieve good network performance. For instance, with IGPs such as Open Shortest Path First (OSPF) and Intermediate System to Intermediate System (IS-IS), routers exchange link state information to learn about a topology map of a network. The contributions of this project are summarized as follows.

1) We design an efficient routing conversion method and theoretically prove the correctness of the conversion from explicit routing to loop-free destination-based routing.

2) The routing conversion method offers a new way to solve a destination-based routing problem, i.e., we can first formulate and solve the routing optimization problem using a simple explicit routing model, and then convert the explicit routing solution.

1.2 SCOPE OF PRESENT WORK

The basic idea is to avoid network congestion by adaptively balancing the load among multiple paths based on measurement and analysis of path congestion. If the shortest paths are not well defined, the effect of weighted traffic splitting becomes limited. Another category of traffic engineering is based on two-phase routing [7]. In such schemes, traffic is sent from each source to a set of intermediate nodes with predetermined split ratios. The intermediate nodes then deliver the traffic to the final destinations. According to link costs contained in link state information messages, shortest paths to each destination are calculated, and corresponding forwarding tables are installed in each router. If there exist multiple next-hops for a destination, the router splits traffic evenly among them, according to the Equal-Cost Multipath (ECMP) [4] split rule (it is common practice to forward packets belonging to the same flow (e.g., defined by available next-hops). Therefore, it can further adjust traffic distribution to improve load balancing. However, the performance of this scheme is still affected by the link cost setting. If the shortest paths are not well defined, there would not be significant improvement. The smart OSPF scheme extends the capabilities of OSPF by allowing source edge nodes to distribute traffic to the neighbor nodes with predetermined split ratios. The neighbor nodes then deliver the traffic to destinations along OSPF paths. Compared to traditional OSPF, S-OSPF provides more routing flexibility. However, the improvement is limited by link costs and topologies. To avoid forwarding loops, each source edge node cannot forward any traffic to its OSPF ancestors. Thus, source edge nodes may have very limited available neighbor nodes to adjust traffic distribution. The scheme extends ECMP to allow weighted traffic splitting at each node. Weighted

splitting achieves significant performance improvement over ECMP. However, the improvement also depends on the selected shortest paths. In this paper, we present an approach, called Generalized Destination-based Multipath Routing (GDMR), to achieve the same high performance as explicit routing. The key insight of our approach is that we show an arbitrary explicit routing can be converted to a loop-free destination-based routing without any performance penalty for a given traffic matrix. Moreover, if we adjust the traffic distribution of pair on node 3 (e.g., and), the traffic distribution of pair on node 3 would also be changed. This is because that destination-based routing distributes packets to the same destination with identical ratios. In contrast, explicit routing supports flexible routing for each individual flow (i.e., specifying arbitrary paths and tuning traffic split ratios for each individual flow). This greatly facilitates heuristic algorithm design. Thus, we design a heuristic algorithm to obtain the near-optimal explicit routing solution and then apply the routing conversion to get the destination-based routing solution.

Advantages:

- In our proposed Energy Efficient, a hop-by-hop power control mechanism is used to adjust the total power consumption of the network.
- This information is used by the Graphical User Interface component of the IDE to generate the attack reports.
- It has observed the different approaches used to bring secure energy efficiency in routing.

1.3 LITERATURE REVIEW

In this project, we present an approach, called Generalized Destination-Based Multipath Routing (GDMR), this becomes access critical reviews of computing literature to achieve this problem. We present a systematic approach including a heuristic algorithm to realize GDMR. index terms—destination-based routing, load balancing, therefore, the complexity and overhead associated with the MPLS approach can be avoided. recent proposals similarly classify and apply practicality in bridging the characteristics of explicit routing (e.g. MPLS) and destination-based routing. The main goal with load balancing is to make more use of available network resources in order to minimize the risk of traffic congestion. Hopefully this would lead to less delay and packet loss. It could however lead to additional propagation delay if the alternative routes are badly chosen. Some applications are very sensitive to delays. Others are more sensitive to packet loss. Multipath routing aims to exploit the resources of the underlying physical network by providing multiple paths between source-destination pairs.

Multipath routing has a potential to aggregate bandwidth on various paths, allowing a network to support data transfer rates higher than what is possible with any one path. The work in the area of multi-path routing has focused mainly on extending intra-domain routing algorithms (both RIP and OSPF) for multipath support. There are two aspects of a multi-path routing algorithm: computation of multiple loop-free paths and traffic splitting among these multiple paths. Extensive work has been done in both these areas. Distributed multi-path routing algorithms can be viewed as an extension of hop-by-hop routing algorithms.

2 EXISTING METHODOLOGY

There is a large body of literature on traffic engineering, it formulates the routing problem as an optimization problem and solves the problem to obtain the explicit routes for each source–destination pair to distribute traffic. They based on OSPF and ECMP protocols. The idea is to carefully fine-tune the link costs to adjust path selection in ECMP so as to optimize load balancing. These schemes bring performance improvement to ECMP compared to arbitrarily configured link costs. However, These schemes are hard to converge to near-optimal solutions in most cases. Even splitting traffic among next-hops further limits the performance of these types of schemes. As a result, such schemes are not guaranteed to achieve near-optimal load balancing. The scheme in [4] is also based on ECMP. Instead of distributing traffic among all available next-hops, it carefully selects a subset of allowable next-hops for each destination IP prefix.

Therefore, it can further adjust traffic distribution to improve load balancing. However, the performance of this scheme is still affected by the link cost setting. If the shortest paths are not well-defined, improvement is not significant. We present weighted ECMP in [7]. The scheme extends ECMP to allow weighted traffic splitting at each node. Weighted splitting achieves significant performance improvement over ECMP. However, the improvement still depends on the selected shortest paths. If the shortest paths are not well defined, the effect of weighted traffic splitting becomes limited. Another category of traffic engineering is based on two-phase routing. In such schemes, traffic is sent from each source to a set of intermediate nodes with predetermined split ratios. The intermediate nodes then deliver the traffic to the final destinations. Performance optimization is achieved by carefully picking a set of intermediate nodes and tuning the split ratios.

The advantage of the two-phase approach is that it handles highly dynamic and fluctuating traffic very well. However, the two-phase routing protocol is complex since it delivers traffic through IP tunnels, optical-layer circuits, or label switched paths in each phase. LB-SPR decreases the

complexity of two-phase routing by using the standard shortest path routing protocol for each phase. However, additional modules are still required to support LB-SPR, such as replacing the destination IP addresses with the IP addresses of the intermediate routers to redirect packets and forwarding packets to intermediate routers with predetermined ratios. There are dynamic schemes that try to adjust traffic distribution in real time to achieve load balancing. OSPF-OMP dynamically determines the split ratios of traffic distributed along multiple equal-cost paths to avoid congestion, where traffic load control messages are dynamically exchanged among the routers. The scheme in performs load balancing at the transport layer. It uses minimal congestion feedback signals from routers. The basic idea is to avoid network congestion by adaptively balancing the load among multiple paths based on measurement and analysis of path congestion. Flare is a flowlet-aware routing engine, it splits a TCP flow into bursts and sends each burst through a different path based on network congestion. With instant measurement and careful engineering, Flare can achieve load balancing without causing packet disorder.

Disadvantages:

- The practically it is not possible to replace the batteries of large number of deployed sensor in the environment.
- The node rebroadcast the RREQ with maximum power, if it is not the destination.
- Not all attacks on security protocols occur over a single session.

3 PROPOSED METHODOLOGY

The generalized destination-based multipath routing problem can be described as follows. Given a network with a traffic demand matrix, our objective is to obtain the best loop-free weighted destination-based routing configuration so that the network congestion ratio is minimized. (In this project, our aim to minimize the network congestion ratio to achieve good load balancing. However, our approach can also be applied to the routing optimization problems with different objective functions such as minimizing end-to-end delay.) A loop-free destination-based routing solution can be converted from an explicit routing solution. Thus, the GDMR problem can be solved in a new way, i.e., instead of obtaining the destination-based routing configuration directly, it can first solve a corresponding explicit routing optimization problem and then convert the obtained explicit routing solution to the desired loop-free destination-based routing solution. The corresponding explicit routing optimization problem is described as follows. Given a network with a traffic demand matrix, our objective is to

obtain the best explicit routing ratios, so that the network congestion ratio is minimized.

The objective of function to minimize the network congestion ratio. By solving the above linear programming (LP) problem using LP solvers, it can obtain the optimal explicit routing solution. Since the conversion from explicit routing to loop-free destination-based routing does not cause a performance penalty, the derived destination-based routing solution achieves the optimality of the original explicit routing problem. Assume there is an optimal explicit routing with minimum, and a destination-based routing derived from using (1) contains loops. Since destination-based routing is a special case of explicit routing in terms of forwarding strategy, there must exist an explicit routing with the smaller. This contradicts the given assumption that is an optimal explicit routing with minimum. The optimal destination-based routing solution of the proposed GDMR problem can also be obtained by solving an LP problem in the destination-based routing formulation (13) using LP solvers. The optimal solution of destination achieves the exact same as that of explicit routing formulation (11), based on the analysis. However, it is difficult to design a heuristic algorithm based on destination-based routing due to the constraints of destination-based routing, which include distributing packets along shortest paths and splitting packets to the same destination with identical ratios. Let us take an example. If we increase the cost of link to let traffic of pair be distributed along a single shortest path, the paths of pair would also be affected. There would also be only one shortest path for pair. We are unable to specify the paths for each node pair since destination-based routing distributes traffic along shortest paths. Moreover, if we adjust the traffic distribution of pair on node 3 (e.g., and), the traffic distribution of pair on node 3 would also be changed. This is because that destination-based routing distributes packets to the same destination with identical ratios. In contrast, explicit routing supports flexible routing for each individual flow (i.e., specifying arbitrary paths and tuning traffic split ratios for each individual flow). This greatly facilitates heuristic algorithm design. Thus, we design a heuristic algorithm to obtain the near-optimal explicit routing solution and then apply the routing conversion to get the destination-based routing solution.

Advantages:

- In our proposed Energy Efficient, a hop-by-hop power control mechanism is used to adjust the total power consumption of the network.
- This information is used by the Graphical User Interface component of the IDE to generate the attack reports.
- It have observed the different approaches used to bring secure energy efficiency in routing.

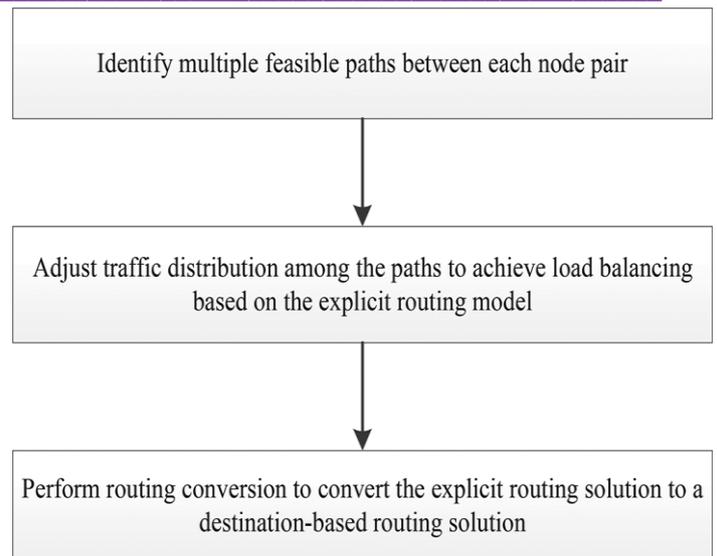


Fig 3.1. Framework of the heuristic algorithm.

Based on the analysis and discussion presented in previous sections, the proposed system develop a heuristic algorithm to obtain near-optimal solutions for large-scale networks. The proposed heuristic Algorithm optimizes destination-based routing in three steps. We first identify multiple loop-free paths for each source–destination pair. We then adjust traffic distribution among the paths to achieve load balancing based on the explicit routing model. Finally, we perform routing conversion to convert the explicit routing solution to a loop-free destination-based routing solution. The framework of the proposed heuristic algorithm is illustrated in Fig.3.1.

4.1 Multipath Routing Scheme

A host that provides multiple paths must first calculate the path sets between the source and destination. Two of the characteristics that can be used for determining a path, set are path quantity and path independence. Path quantity is the number of available paths between nodes. The higher the number better chances for load distribution. Uniform path sets are preferable over high variance path sets i.e. a path set with every node having 5 paths is preferred than one with nodes having 1 path for some path sets and 9 paths for other path sets. The second characteristic of path sets is path independence, which is illustrated with Fig. 4.1.1 Consider a path set with 2 paths (a, b, c, d) and (a, f, c, d) and other path set with 2 paths as (a, b, c, d) and (a, f, e, d). The second set is independent when compared to the first set. a f e d b c a f e d b c Fig.3.1.1 Illustration of multipath routing scheme So the second set would lead to better usage of resources and is less likely to be congested because at least one link in each path should be congested, whereas in the first set congestion

at link (c, d) is reflected in both the path sets. Multipath sets with these attributes facilitate for higher performance.

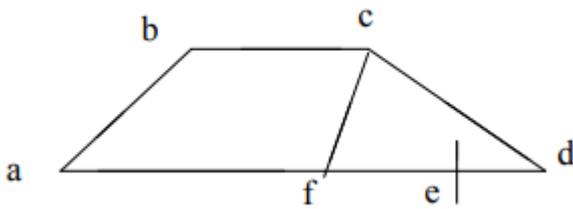


Fig 3.1.1 Illustration of multipath routing scheme

So the second set would lead to better usage of resources and is less likely to be congested because at least one link in each path should be congested, whereas in the first set congestion at link (c, d) is reflected in both the path sets. Multipath sets with these attributes facilitate for higher performance.

4.2 Routing protocol considerations

The destination-only routing, source-destination routes will typically be disseminated throughout the network by dynamic routing protocols. It is expected that multiple dynamic routing protocols will be adapted to the needs of source-destination routing architecture.

Loop-freeness considerations

Some existing routing protocols will be enhanced to propagate source-destination routing information. In this project the protocol may be configured to operate in a network where some, but not all, routers support source-destination routing and others are still using destination-only routing. Even if all routers within a network are capable of source-destination routing, it is very likely that on edges of the network they will have to forward packets to routers doing destination-only routing. Since a router implementing source-destination routing can have additional, more granular routes than one that doesn't implement it, persistent loops can form between these systems.

Thus specifications of source-destination routing protocols (either newly defined protocols or enhancements to already existing one) MUST take provisions to guarantee loop-free operations.

There are 3 possible approaches to avoid looping condition:

1. Guarantee that next-hop gateway of a source destination route supports source-destination routing, for example calculate an alternate topology including only routers that support source-destination routing architecture
2. If next-hop gateway is not aware of source-destination routing then a source-destination path can lead to it only if next-hop router is 'closer' to the destination in terms of protocol's routing metric; important particular case of the rule is if destination-only routing is pointing to the same next-hop gateway
3. Discard the packet (i.e. treat source-destination route as unreachable)

5 IMPLEMENTATION OF PROPOSED METHODOLOGY

5.1 System Architecture

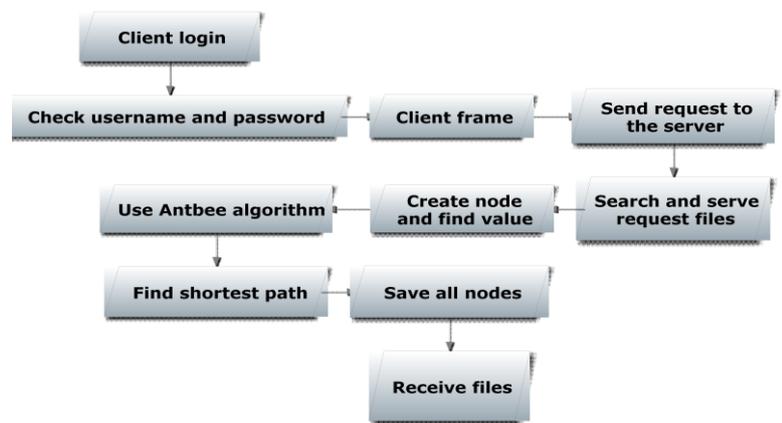


Fig 4.1 .Architecture of Load Balancing of IP-Network using Generalized Destination-Based Multipath Routing

According to the above analysis, we draw the conclusion that an arbitrary explicit routing can be converted to a loop-free destination-based routing without any performance penalty for a given traffic matrix. In other words, for a given traffic matrix, destination-based routing is able to achieve the high performance as explicit routing. An approximate to the traffic splitting is to hash the n-tuple packet header and then allocate flows to one of the output ports based on the hash results and the ratios. This load balancing performed through Antbee algorithm. By using Antbee algorithm the system finds the shortest path. Then it saves all files in the source node. Then it moves to the destination node. And then client receives files from destination node. The interest packet always follows the shortest path to the server. The shortest path problem is the problem of finding a path between two nodes in a graph such that the sum of the weights of its constituent edges is minimized.

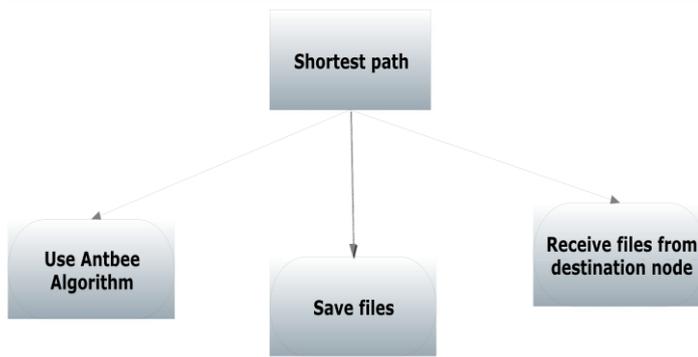


Fig 5.1.1 Shortest path problem

5.2 MULTIPATH ROUTING IN MPLS NETWORKS

5.2.1 Traffic Engineering with MPLS

The emergence of MPLS with its efficient support of explicit routing provides basic mechanisms for facilitating traffic engineering [3]. Explicit routing allows a particular packet stream to follow a predetermined path rather than a path computed by hop-by-hop destination-based routing such as OSPF or IS-IS. With destination-based routing as in traditional IP network, explicit routing may be provided by attaching to each packet the network-layer address of each node along the explicit path. This approach generally incurs prohibitive overhead.

In MPLS, a path (known as a LSP) is identified by a concatenation of labels which are stored in the nodes. As in traditional virtual-circuit packet switching, a packet is forwarded along the LSP by swapping labels. Thus, support of explicit routing in MPLS does not entail additional packet header overhead.

Traffic engineering with MPLS requires the components of constraint based routing [9] and an enhanced IGP. With MPLS when an enhanced IGP builds LSR's forwarding table, it takes into account LSPs originated by the LSR, so that LSPs can be used to carry traffic. IGPs using shortest path to forward 15 traffic attempt to conserve resources but can lead to congestion. This can be due to different shortest paths overlapping at some link or the traffic from a source to a destination exceeding the capacity of the shortest path.

Constraint based routing, along with some form of connection admission control, avoids placing too many LSPs on any link, thus avoiding one of the problems. Similarly, if the traffic between two routers exceeds the capacity of any single path, then multiple LSPs can be set up between them. The traffic is split between these based on specified or derived load ratios. For example, the ratios may be proportional to the bandwidths of the LSPs. Further, such

LSPs can be placed on different physical paths to ensure more even distribution of load. This also allows for graceful degradation in case one of the paths fails. MPLS allows enforcement of some administrative policies in online path computation. For example, resource color can be assigned to LSPs and links to achieve a degree of desired LSP placement. Suggests an example where regional LSPs are to be kept from traversing inter-region links. To enforce this scheme, all regional links may be colored green, and all inter-region links colored red. Regional LSPs are then constrained to use only green links. If an operator chooses, paths for LSPs may be determined offline, possibly based on global optimization and other administrative policies considerations. This allows network administrators to control traffic paths precisely

5.3 Network Model

In these Fig 5.4, the network is described as a connected graph $G(V,E)$ where V is the set of nodes and E is the set of directed links observation, we design an efficient routing conversion method and theoretically present its correctness.

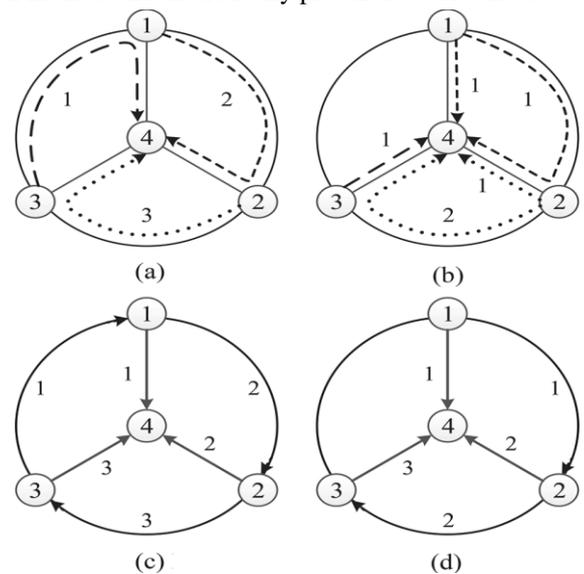


Fig. 4.3. Examples of routing conversion. (a) Explicit routing. (b) Convertible explicit routing. (c) Destination-based routing (contain a loop). (d) Loop-free destination-based routing.

Greedy Algorithm:

In many problems, a greedy strategy does not in general produce an optimal solution, but nonetheless a greedy heuristic may yield locally optimal solutions that approximate a global optimal solution in a reasonable time. When choosing the optimal cache locations on any SPT with

the greedy method, the core nodes with higher fan-out and more traffic will be appropriate candidates.

Greedy algorithms have five components:

1. A candidate set, from which a solution is created
2. A selection function, which chooses the best candidate to be added to the solution
3. A feasibility function, that is used to determine if a candidate can be used to contribute to a solution
4. An objective function, which assigns a value to a solution, or a partial solution, and
5. A solution function, which will indicate when we have discovered a complete solution

Greedy select the things which will take the minimum amount of time to complete while maintaining two variables current Time and numberOfThings. To complete the calculation, you must:

1. Sort the array A in a non-decreasing order.
2. Select each to-do item one-by-one.
3. Add the time that it will take to complete that to-do item into currentTime.
4. Add one to numberOfThings.

6 DESIGN OF EXPERIMENTAL SET-UP & INSTRUMENTATION

MODULES

GDMR
Sensor Node
Routing
Design Goals

6.1 MODULE DESCRIPTION

GDMR:

- ⊙ In this paper, for the first time, we propose a secure and efficient Cost-Aware Secure Routing protocol .
- ⊙ That can address energy balance and routing security concurrently in WSNs. In CASER protocol, each sensor node needs to maintain the energy levels of its immediate adjacent neighboring grids in addition to their relative locations.
- ⊙ Using this information, each sensor node can create varying filters based on the expected design tradeoff between security and efficiency.

Sensor Node:

- ⊙ Each sensor node can update the energy levels based on the detected energy
- ⊙ usage. The actual energy is updated periodically.
- ⊙ It also assume that data generation in each sensor node is a random variable.
- ⊙ Each sensor node can create varying filters based on the expected design tradeoff between security and efficiency.
- ⊙ In protocol, each sensor node needs to maintain the energy levels of its immediate adjacent neighboring grids in addition to their relative locations.

Routing:

- ⊙ It is developed a two-phase routing algorithm to provide both content confidentiality and source location privacy.
- ⊙ In phantom routing protocol each message is routed from the actual source to a phantom source along a designed directed walk through either sector-based approach or hop based approach.
- ⊙ To solve this problem, several schemes have been proposed to provide source-location privacy through secure routing protocol design.

Design Goals:

- ⊙ To maximize the sensor network lifetime, we ensure that the energy consumption of all sensor grids are balanced.
- ⊙ To achieve a high message delivery ratio, our routing protocol should try to avoid message dropping when an alternative routing path exists.
- ⊙ The adversaries should not be able to get the source location information by analyzing the traffic pattern.

RESULT AND DISCUSSION



Fig 7.2 Comparing the execution time

Existing system of explicit routing needs more time to transfer. The proposed system takes less times to computes.

CONCLUSION AND SUGGESTED FUTURE WORK

CONCLUSION

The key insight of our approach is that we show an arbitrary explicit routing can be converted to a loop-free destination-based routing without any performance penalty for a given traffic matrix. This has great value for practice in that the property of destination-based routing allows forwarding entries to be stored in RAM instead of TCAM, which greatly reduces hardware cost. We design an efficient routing conversion method and prove its correctness. We show that the desired loop-free destination-based routing solution can be obtained by solving an explicit routing problem and then doing the routing conversion. We also present a heuristic algorithm to realize GDMR. The performance of the proposed heuristic algorithm is verified in several practical networks using simulation. The results show that the proposed heuristic algorithm for GDMR provides extremely good load balancing that is comparable to the performance of optimal explicit routing. GDMR has very low complexity and completes all experiments in a very short time. We also show that GDMR is robust regarding fluctuations in traffic.

SUGGESTED FUTURE WORK

Future work of our project is to implement the high performance using different algorithms. We will mainly focus on security protocols and easy to replace the batteries of large number of deployed sensor in the environment.

ACKNOWLEDGEMENT

We take privilege to greet our beloved parents for their encouragements in every effort. Also, we are happy to thanks our college managements, principal, head of the department and my colleagues for their sincere support in all concerns of resources.

REFERENCES

- [1]. Wang and Z. Wang, "Explicit routing algorithms for internet traffic engineering," in *Proc. 8th Int. Conf. Comput. Commun. Netw.*, 1999, pp. 582–588.
- [2]. D. Awduche, J. Malcolm, J. Agogbua, M. O'Dell, and J. McManus, "Requirements for traffic engineering over MPLS," IETF RFC 2702, Sept. 1999.
- [3]. D. Thaler and C. Hopps, "Multipath issues in unicast and multicast next-hop selection," IETF RFC 2991, Nov. 2000.
- [4]. C. Hopps, "Analysis of an equal-cost multi-path algorithm," IETF RFC 2992, Nov. 2000.
- [5]. A. Elwalid, C. Jin, S. Low, and I. Widjaja, "MATE: MPLS adaptive traffic engineering," in *Proc. 20th Annu. IEEE INFOCOM*, 2001, vol.3, pp. 1300–1309.
- [6]. E. Rosen, A. Viswanathan, and R. Callon, "Multiprotocol label switching architecture," IETF RFC 3031, Jan. 2001.
- [7]. D. Awduche, A. Chiu, A. Elwalid, I. Widjaja, and X. Xiao, "Overview and principles of internet traffic engineering," IETF RFC 3272, May 2002.
- [8]. E. Osborne and A. Simha, *Traffic Engineering With MPLS*. Upper Saddle River, NJ, USA: Pearson Education, 2002.
- [9]. B. Fortz and M. Thorup, "Optimizing OSPF/IS-IS weights in a changing world," *IEEE J. Sel. Areas Commun.*, vol.20, no.4, pp.756–767, May 2002.
- [10]. K. Holmberg and D. Yuan, "Optimization of internet protocol network design and routing," *Networks*, vol.43, no.1, pp.39–53, 2004.
- [11]. A. Sridharan, R. Guerin, and C. Diot, "Achieving near-optimal traffic engineering solutions for current OSPF/IS-IS networks," *IEEE/ACM Trans. Netw.*, vol. 13, no. 2, pp. 234–247, Apr. 2005.
- [12]. N. McKeown *et al.*, "OpenFlow: Enabling innovation in campus networks," *Comput. Commun. Rev.*, vol.38, pp.69–74, Mar 2008.
- [13]. M. Kodialam, T. V. Lakshman, J. B. Orlin, and S. Sengupta, "Oblivious routing of highly variable traffic in service overlays and IP backbones," *IEEE/ACM Trans. Netw.*, vol. 17, no. 2, pp. 459–472, Apr. 2009.
- [14]. C. Jian and L. Chin-Tau, "Optimal link weights for IP-based networks supporting hose-model VPNs," *IEEE/ACM Trans. Netw.*, vol. 17, no.3, pp. 778–788, Jun. 2009.
- [15]. Cisco, San Jose, CA, USA, "Cisco Open Network Environment: Adaptable framework for the Internet of everything," [Online]. Available: <http://www.cisco.com>
- [16]. K. Kannan and S. Banerjee, "Compact TCAM: Flow entry compaction in TCAM for power aware SDN," in *Distributed Computing and Networking*. New York, NY, USA: Springer, 2013, pp.439–444.
- [17]. E. Oki and A. Iwaki, "Load-balanced IP routing scheme based on shortest paths in hose model," *IEEE Trans. Commun.*, vol. 58, no. 7, pp. 2088–2096, Jul. 2010.
- [18]. A. R. Curtis *et al.*, "DevoFlow: Scaling flow management for high performance networks," *Comput. Commun. Rev.*, vol.41, pp.254–265, Aug. 2011.
- [19]. Z. Cao, Z. Wang, and E. Zegura, "Performance of hashing-based schemes for internet load balancing," in *Proc. 19th Annu. IEEE INFOCOM*, 2000, vol. 1, pp. 332–341.
- [20]. A. Mishra and A. Sahoo, "S-OSPF: A traffic engineering solution for OSPF based best effort networks," in *Proc. IEEE GLOBECOM*, Nov. 2007, pp. 1845–1849.

APPENDIX “A”

ISP - Internet Service Provider

TE - Traffic Engineering

MPLS - Multi Protocol Label Switching

SDN - Software Defined Networking

TCAM - Ternary Content Addressable Memory

RAM - Random-Access Memory

IGP - Interior Gateway Protocol

IS-IS - Intermediate System to Intermediate System

OSPF - Open Shortest Path First

ECMP - Equal-Cost Multipath

GDMR - Generalized Destination-based Multipath Routing

IAAS - Infrastructure-as-a-Service

HPC - High Performance Computing

HP - Linear Programming