ISSN:0975-9646

Krishna Siva Prasad.M et al, / (IJCSIT) International Journal of Computer Science and Information Technologies, Vol. 3 (4) , 2012,4840-4842

Weak Bloom Filtering for Large-Scale Dynamic Networks

Krishna Siva Prasad.M , Y.Adilakshmi

Department of CSE
Gudlavalleru Engineering College,
Gudlavalleru,(A.P.), India

***Abstract* — In order to forward decisions in networks where protocols play an important role we need to provide the information about the destination nodes provided by routing table states. When dynamic networks are considered the path to reach the destination may change, and corresponding states become invalid and need to be refreshed . In large, complex and highly dynamic networks, this is quite cumbersome. For such networks, we propose the concept of *weak state*, associated with a bloom filter which is interpreted as a probabilistic hint. Bloom filters provide space efficient storage of sets at the cost of a probability of false positives on membership queries. Weak Bloom Filtering (WBF) is a novel routing protocol that uses weak state along with *random directional walks* for forwarding packets. When a packet reaches a node that contains a weak state about the destination with higher confidence than that held by the packet, the walk direction is biased. The packet reaches the destination via a sequence of directional walks, punctuated by biasing decisions.**

***Keywords*— Data Structures, Bloom Filters, Distributed Systems, unstructured routing, weakstate**.

I. INTRODUCTION

IN THIS PAPER, we consider the problem of designing robust and scalable routing protocol for large and dynamic networks like large-scale mobile ad hoc networks (MANETs) with the help of bloom filters. A Bloom filter is traditionally implemented by a single array of N bits, where N is the filter size [1]. When a filter is created all bits are reset to zeroes. A filter is also parameterized by a constant k that defines the number of hash functions used to activate and test bits on the filter. Each hash function should output one index in M. When inserting an element e on the filter, the bits in the k indexes h1(e), h2(e), . . . , hk(e) are set. In particular, a filter with M = 15 bits and k = 3 hash functions could become as follows, after the insertion of one element:

0	0	1	0	0	0	0	0	0	0	1	0	1	0	0

The same procedure is used to insert other elements, each time setting the bits given by the corresponding k indexes. Here for large and dynamic networks, we use bloom filters which acts as probabilistic routing tables. Such state information is called *weak state*. Weak state can be locally refreshed by reducing the associated confidence value, a measure of the probability that the state is accurate. When the information is to be passed to the destination then basing on the corresponding probabilistic hints the information is sent to the destination.In forwarding the information the routing tables play an important role, where for each and every entry

II. RELATED WORK

Traditional state concept can be classified into two broad categories: hard and soft state approaches. *Hard state* is maintained at a remote node until it is explicitly removed using state-teardown messages by the node that installed the state. Since the state is removed explicitly, reliable communication is essential. *Soft state*, which was first coined in , *times out* unless it is refreshed within a timeout duration. The node that installed the state periodically issues refresh messages. Once a message is received by the node maintaining the soft state, the timer corresponding to the state is rescheduled. If the timer expires, the state times out and is removed from the system. Soft state does not require explicit removal messages, unlike hard state. Hence, reliable signaling is not required[1].

In both hard state and soft state, the state information is regarded as absolute truth. We refer to such state information as having *strong* semantics or that it is an example of *strong state*. When the original state changes, the strong state value at the remote nodes should be explicitly refreshed in both approaches(hard or soft).*Weak state*, on the other hand, has “weak” or probabilistic semantics[1]. The state can be refreshed *locally* by weakening or decaying the confidence value associated with the state over time. The confidence value is an estimate of the probability that the true state is valid. Once the confidence in the state is below a threshold value, the state is removed from the system.

As, Bloom filters are now being used in a large number of systems, including peer-to-peer systems , web caches , database systems and others these are used with weak states because Bloom Filter allows efficient access when this data structure is stored on disk; a similar approach is used in this network routing algorithms[2].

III. WEAK BLOOM FILTERING

Weak state is realized and implemented using Bloom filters. Packet forwarding is done using the information present in the Bloom filters corresponding to each and every node.

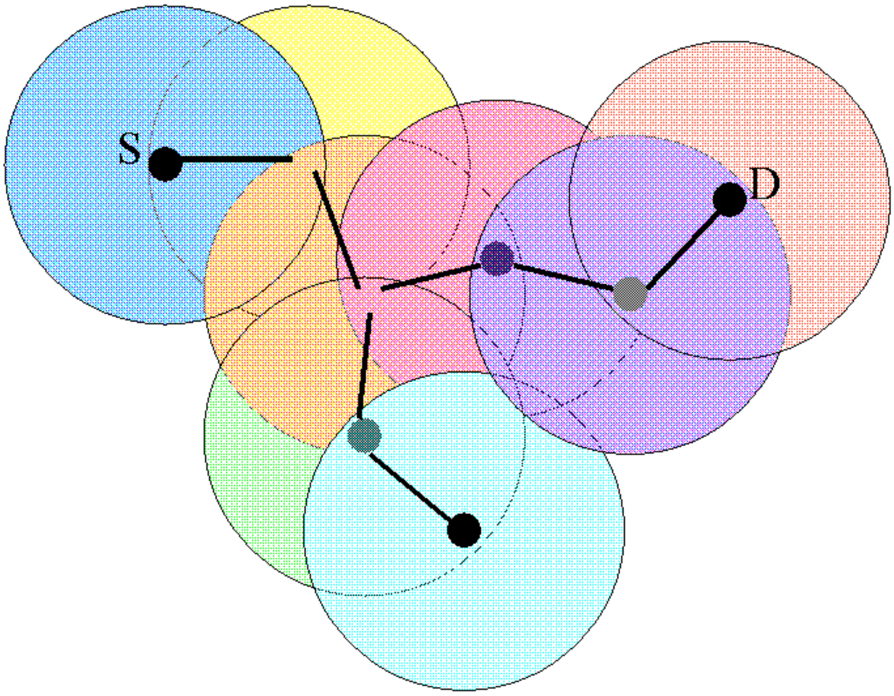


Fig.., 1Path tracing from Source S to Destination D in different GeoRegions

A. Preliminaries of Bloom filters

A set X of n items is represented by a Bloom filter using a vector of m bits which are initially set to 0. A Bloom filter uses k independent hash functions R1,R2,……. ,Rk with a range {1, ...,m}. When inserting an item x to X, all bits of Bfaddress(x) (consisted of Ri(x) for $1 \leq i \leq k$) will be set to 1. To answer a membership query for any item x, users check whether all bits Ri(x) are set to 1. If not, x is not a member of X. If yes, we assume that x is a member of X, although we might be wrong in some cases. Hence, a Bloom filter may yield a *false positive* which suggests that the item x is in X even though it is not [2].

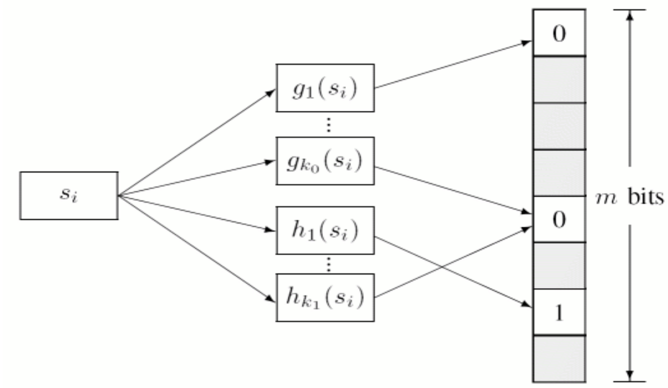


Fig 2.., Insertion of an element into a Generalized Bloom Filter

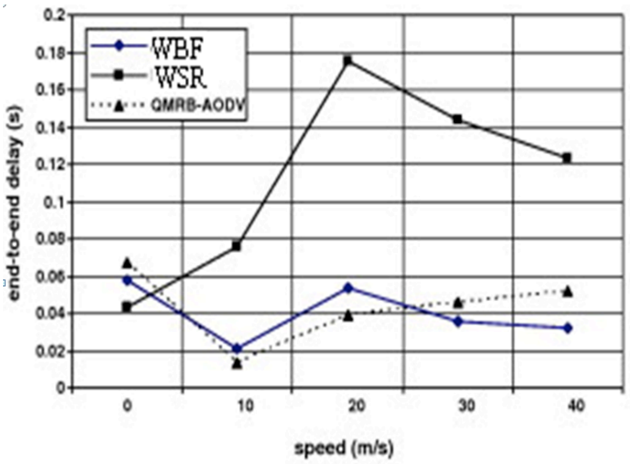
Assume that a hash function selects each array position with equal probability.If *m* is the number of bits in the array, the probability that a certain bit is not set to one by a certain hash function during the insertion of an element is then $(1\text{-}1/m)$. The probability that it is not set by any of the hash functions is $(1\text{-}1/m)^k$. If we have inserted *n* elements, the probability that a certain bit is still 0 is $(1\text{-}1/m)^{kn}$, the probability that it is 1is therefore $1\text{-}(1\text{-}1/m)^k$

B. Assumptions

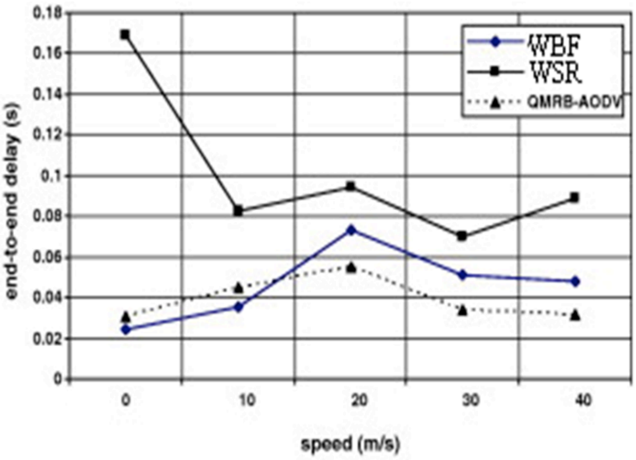
The assumptions WBF makes are similar to those made by traditional location based routing protocols: Nodes know their positions on a 2-D plane, either using a GPS device or through any other localization techniques. By using periodic single hop beacon messages, each node also knows its neighbors and their positions. The nodes have uniform omni-directional antennas. The source nodes in general do not know the location of the destination nodes. We consider the scenario where the nodes move independently and the network density is high enough for connectivity at any time. The maximum node speed is known. Though this value can be large, we assume that the average displacement in unit time is small in comparison to the maximum distance between any two points in the area covered by the network.

C. Weak State Realization

In WSR, a *weak state* corresponds to a mapping from a persistent node ID or a collection of IDs (*SetofIDs*) to a geographical region (*GeoRegion*) in which the node (or the set of nodes) is believed to be currently located. The state information captures the uncertainty in the mapping. An explicit mapping from a SetofIDs to a GeoRegion can be used to "bias" the random directional walks of packets being forwarded. If the destination ID is an element of the SetofIDs, the packet can be biased toward the center of the associated Geo-Region (subject to other conditions described later in this section).The bias can be reinforced as the packets get closer to their destinations. This is achieved by maintaining more definite, or stronger, mappings for nodes located closer to the destination. We capture the uncertainty in the mappings by weakening the two components of these mappings: SetofIDs and GeoRegion. We also aggregate the location information about a number of nodes.



(a) 5 mobile nodes



(b) 10 mobile nodes

Fig 3..,Comparison of WBF and WSR

Algorithm 1 Algorithm for biasing packets in WBF
ForwardPacket (K)
1: //Consider the bias previously given to the packet
2:m□destination(K)
3:Ø□Temporal(K)
4:R□Spatial(K)
5:(x,y) □TargetLocation(K)
6: //Find the strongest local mapping indicating the whereabouts of the node
7: **for all** mapping j **do**
8: Øi□Lookup(I,m)
9: **if**(Øj>Ø) OR (Øj=Ø AND Ri<R) **then**
10: Ø□Øj
11: R□Rj
12: (x,y) □Centeri
13: **end if**
14: **end for**
15:Temporal(K) □Ø
16:Spatial(K) □R
17:TargetLocation(K) □(x,y)
18: Use a geographic forwarding scheme to send the packet to Target Location(K)
Lookup(j,m)
1:θ □ 0
2: **for all q € {1,2,…..i}do**
3: θ□θ+WBFj[hq(m)]
4: **end for**
5: Return

IV. Conclusion And Future Work

We present Weak Bloom Filter (WBF) protocol, an unstructured forwarding paradigm based on the partial knowledge about the node locations. The nodes periodically announce their locations on random directions. The nodes use these announcements to create aggregated SetofIDs-to-GeoRegion mappings. A routing state consists of a weak Bloom filter (WBF) that contains a set of nodes and a geographical region where the nodes are believed to be located. WBF also yields the confidence that a node is an element of SetofIDs. When a node has a data packet, the packet is sent in a random direction with the belief that an intermediate node will give the packet a superior hint about the location of the destination node. The packet trajectory is then biased toward the center of the region indicated by this state value While we have considered WSR in the context of large, mobile, and connected ad hoc networks in this paper, we believe the weak state concept can be also adopted in networks that experience node disconnectedness, i.e., delay-tolerant networks (DTNs), though this would require different methods for state dissemination and packet forwarding. Our future plans include the investigation of such methods

V.References

[1] U. G. Acer, S. Kalyanaraman, and A. A. Abouzeid, "Weak state routing for large scale dynamic networks," in *Proc. ACM MobiCom*, 2007, pp. 290–301.

[2] P. Reynolds, A. Vahdat, Efficient peer-to-peer keyword searching., in: M. Endler, D. C. Schmidt (Eds.), Middleware, Vol. 2672 of Lecture Notes in Computer Science, Springer, 2003, pp. 21–40.

[3] S. C. Rhea, J. Kubiatowicz, Probabilistic location and routing., in: Proc. of the 21st Annual Joint Conference of the IEEE Computer and Communications Societies (INFOCOM 2002), 2002. [19] L. Fan, P. Cao, J. Almeida, A. Z. Broder, Summary cache: a scalable wide-area web cache sharing protocol, IEEE/ACM Trans. Netw. 8 (3) (2000) 281–293.

[4] L. F. Mackert, G. M. Lohman, R* optimizer validation and performance evaluation for distributed queries, in: Proceedings of the Twelfth International Conference on Very Large Data Bases (VLDB '86), Morgan Kaufmann Publishers Inc., San Francisco, CA, USA, 1986, pp. 149–159.

[5]. Broder, M. Mitzenmacher, Network applications of bloom filters: A survey, in: Proc. Of Allerton Conference, 2002.

[6].S. Cohen, Y. Matias, Spectral bloom filters, in: Proceedings of the 2003 ACM SIGMOD international conference on Management of data (SIGMOD '03), ACM Press, New York, NY, USA, 2003, pp. 241–252.

[7] U. Manber, S.Wu, An algorithm for approximate membership checking with application to password security, Inf. Process. Lett. 50 (4) (1994) 191–197.

[8] B. H. Bloom, "Space/time trade-offs in hash coding with allowable errors," *Commun. ACM*, vol. 13, no. 7, pp. 422–426, 1970.

[9] U. G. Acer, S. Kalyanaraman, and A. A. Abouzeid, "Weak state routing for large scale dynamic networks (extended paper)," Elect., Comput.,Syst. Eng. Dept., Rensselaer Polytechnic Institute, 2009 [Online].Available: http://networks.ecse.rpi.edu/~acer/wsr_extended.pdf .