HALBRP: History-Aware Load Balanced Routing Protocol in Delay Tolerant Networks

Mohamed A. El-Zawawy¹,²

¹College of Computer and Information Sciences
Al Imam Mohammad Ibn Saud Islamic University (IMSIU)
Riyadh
Kingdom of Saudi Arabia

²Department of Mathematics
Faculty of Science
Cairo University
Giza 12613
Egypt
maelzawawy@cu.edu.eg

Abstract: Delay Tolerant Networks (DTNs) can be defined as groups of moving hosts that form networks with quickly changing dynamic topologies. In such networks, there is no pre-assumed control center or infrastructure. This is so as the network nodes are randomly mobile and located. The nature of such networks makes efficient routing in them an involved problem. This paper presents HALBRP, a new History-Aware Load-Balanced Routing Protocol in Delay Tolerant Networks. The proposed technique uses a precise method (presented in the paper) to measure the load of a node before adding any extra load (in the form of new packets) to the node. The method of calculating the load of a node takes into consideration the history of the node content of messages.

Key-Words: HALBRP; DTNs; History-Aware Routing; Load-Balanced Routing; Delay Tolerant Networks; Routing Protocols.

1 Introduction

The classical routing problem is to design an efficient route from source to destination relying on persistent node to node routes [18]. The nature of routing algorithms depends heavily on the type of the network under study. Therefore classical routing technique are not typically applicable to emerging types of modern networks such as Delay Tolerant Networks [3, 2, 1, 20].

Delay-Tolerant Networks (DTNs) can be considered as models for networks that are marked with often partitioning and infrequent affinity [14]. Energy and mobility aspects of DTNs cause them to frequently have intermittent failure [19]. This is partially due to the fact that the network topology of DTNs is typically an extremely dynamic one [17]. Therefore in DTNs, the challenge is that all long interruptions and delays in node to node routes must be taken into consideration by all network algorithms (including routing). This is the case for many applications and network machinery such as Information-centric networks (ICNs) [13].

It is the case in DTNs that the connection between nodes is intermittent meaning that the node to node routes does always established. This makes the routing problem in DTN a challenging one [18]. The ability to anticipate the node moves in the DTNs does not guarantee the easiness of building a routing technique for boosting the performance in the network. This is partially due to the lack of load data along the DTN [17, 15]. One of the reasons for diversity in routing protocols of DTNs is that a wide range of network kinds are classified as DTN [14].

In this paper, we present a new History-Aware Load-Balanced Routing Protocol (HALBRP) for DTNs. The protocol is History-aware in the sense that the load to each node is calculated in a way that considers the history of messages carried by the node. The protocol is also classified as load-balanced because nodes exchange messages in a manner that balance the load of network nodes. As a result the transfer of a message from one node to another one, in its way to its final destination, is controlled by the condition that the load of the receiving node does not exceed a
fixed load threshold. The rest of the paper is organized as follows. Section 2 presents in details the new routing protocol; HALBRP. Section 3 reviews some examples of modern routing methods in DTNs and also reviews modern applications and technologies of DTNs. Finally Section 4 concludes the paper.

2 Routing Protocol

In this section, we present a new routing protocol, HALBRP (History Aware Load Balanced Routing), for DTNs. The protocol is load-balanced in the sense that nodes exchange messages in a way that balance the load of moving nodes. The protocol is History-aware in the sense that load of a node is calculated in a way that considers the history of messages carried by the node. Therefore the transfer of a message from one node to another one is subject to the condition that the load of the receiving node does not exceed a fixed load threshold.

For each node, \( n \), we calculate the load of \( n \) using the following formula:

\[
\text{Node-Load}(n) = \sum_{m_n} 1 - \frac{\text{act}(m_n)}{\text{max}(m_n)}. \tag{1}
\]

Equation 1, is a summation over the messages \( m_n \) hosted by the node \( n \). The activity of a message \( m_n \) is the number of nodes hosted the message so far and is denoted by \( \text{act}(m_n) \). In the equation above, \( \text{max}(m_n) \) denotes the maximum number of nodes that \( m_n \) is allowed to visit. This number depends on the message level of trust and importance. The subtraction from 1 in Equation 1 expresses the fact that the higher the activity of the message, the less the time it is going to spend more on the network, and hence, the less the load caused by the message to the network, in general, and the hosting node, in particular.

Algorithm 1 presents the routing algorithm. This algorithm is to run on each node of the DTN. The frequency of executing the algorithm depends on the amount of message on the network and the mobility of the nodes. The algorithm first (line 2) finds a list of the nodes close enough to the main node, \( n \), via sensors. If this list is empty then the algorithm is finished. Otherwise, in line 6, the list is sorted according to the loads of its element nodes. For each node \( n' \) in the list, if \( n' \) is still close to the main node \( n \) (line 8) and if \( n \) and \( n' \) have not exchanged messages recently (line 9), then algorithm 1 calls Algorithm 2, Messages-exchange, to attempt exchanging messages between \( n \) and \( n' \).

Algorithm 2 presents our proposed method for message exchange between a pair of close nodes, \( n_1 \) and \( n_2 \). The load of nodes are caudated using Equation 1 (lines 3 and 6). The algorithm uses two variables \( i_1 \) and \( i_2 \) to mark whether \( n_1 \) and \( n_2 \), respectively, are overloaded. If this is the case the algorithm is finished (line 10). The messages of \( n_1 \) and \( n_2 \) are extracted to \( \text{list}_1 \) and \( \text{list}_2 \), respectively (lines 12 and 13). This extraction includes sorting the message in \( \text{list}_1 \) and \( \text{list}_2 \) according to their priority and history in the network.

For each message \( m \) in \( n_1 \) that is not in \( n_2 \) (lines 15 and 16), if transferring \( m \) to \( n_2 \) would not turn \( n_2 \) overloaded (line 17), the transfer is carried (lines 18 and 19). This is done provided that \( n_2 \) is not overloaded in the first place (line 14).

For each message \( m \) in \( n_2 \) that is not in \( n_1 \) (steps 25 and 26), if transferring \( m \) to \( n_1 \) would not turn \( n_1 \) overloaded (step 27), the transfer is carried (steps 28 and 29). This is done provided that \( n_1 \) is not overloaded in the first place (step 24).

If node \( n_1 \) has already received some messages (line 34), then messages of \( n_1 \) whose final destination is \( n_1 \) (lines 35 and 36) are removed (line 37) from \( n_1 \) to prevent that they get unnecessarily delivered by \( n_1 \) to any other node in the DRN. If node \( n_2 \) has already received some messages (step 41), then messages of \( n_2 \) whose final destination is \( n_2 \) (steps 42 and 43) are removed (step 44) from \( n_2 \) to prevent that they get unnecessarily delivered by \( n_2 \) to any other node in the DRN.

3 Related Work

Recently, routing in DTNs has gained huge research force. Among the typical DTN routing techniques classifications is that has two categories depending on the way adapted to find the target node; flood-
Algorithm 2 Messages-exchange

1: procedure MESSAGES-EXCHANGE
2: i₁, i₂ ← 0;
3: if Node-Load(n₁) ≥ max₁ then
4:    i₁ ← 1;
5: end if
6: if Node-Load(n₂) ≥ max₁ then
7:    i₂ ← 1;
8: end if
9: if i₁ = 1 ∧ i₂ = 1 then
10:    exit;
11: end if
12: list₁ ← calc-list(n₁);
13: list₂ ← calc-list(n₂);
14: if i₂ = 0 then
15:    for each message m ∈ list₁ do
16:      if m ∉ list₂ then
17:        if Node-Load(n₂ ∪ {m}) ≤ max₁ then
18:            n₂ ← n₂ ∪ {m};
19:            n₁ ← n₁ \ {m};
20:        end if
21:      end if
22:    end for
23: end if
24: if i₁ = 0 then
25:    for each message m ∈ list₂ do
26:      if m ∉ list₁ then
27:        if Node-Load(n₁ ∪ {m}) ≤ max₁ then
28:            n₁ ← n₁ ∪ {m};
29:            n₂ ← n₂ \ {m};
30:        end if
31:      end if
32:    end for
33: end if
34: if i₁ = 0 then
35:    for each message m ∈ n₁ do
36:      if destination(m) = n₁ then
37:        n₁ ← n₁ \ {m};
38:      end if
39:    end for
40: end if
41: if i₂ = 0 then
42:    for each message m ∈ n₂ do
43:      if destination(m) = n₂ then
44:        n₂ ← n₂ \ {m};
45:      end if
46:    end for
47: end if
48: end procedure

As a matter of fact, the routing component in some DTNs has records concerning required data traffic and the contact history between nodes. Therefore existing techniques of linear programming can be used to limit the delay of messages in DTNs. However this idea does not perform well for important well-known small applications. In [14], an efficient linear programming algorithm for routing in DTNs was presented using a relatively little number of LP constraints. This algorithm has the advantage of being applicable to column generation applications [14].

Among promising future architectures of Internet is Information-centric networks (ICNs). The idea of ICNs networks is to focus on delivering information among the ends hosts. This is not the case for today’s Internet architectures focusing on delivering messages...
among the end hosts. The potential of merging the two concepts of DTNs and ICNs were discussed in [13]. The combination was in the form of delay-tolerant information-centric network (DTICNs). This paper [13] shows good study of potential design of DTICNs and the involved research challenges.

Security is an important aspect of any DTN routing protocol. However this aspect did not attract enough researchers. Hence many types of attacks are threatening messages moving on DTN. In [19] a defense technique fighting several types of routing attacks was introduced for DTN. This technique extracts and uses routing data from the messages and acknowledgments (ACK). Techniques of evolutionary game theory are used for achieving analysis and enabling routing roles of nodes in the DTNs [19].

4 Conclusion

This paper presented a new routing technique for Delay tolerant networks. The proposed technique has the advantages of being load-balanced and history-aware. This leads to better and more efficient routing decisions by our protocol. There are many directions for developing our technique; for example combing the concept of Information-centric networks (ICNs) [13] with our protocol seems a promising direction.

References:


