

Enhanced Variable Power Broadcasting Based on Local Information in Mobile Ad Hoc Networks

Wilson Woon and Kwan L. Yeung
Department of Electrical and Electronics Engineering,
The University of Hong Kong,
Pokfulam, Hong Kong
Email: {thwoon, kyeung}@eee.hku.hk

Abstract—Broadcasting in mobile ad hoc networks (MANETs) usually involve fixed transmission power that covers, for example an area within 250 meters. However, it is often unnecessary to broadcast using fixed power because a node that needs to be covered may just be 100 meters away. By reducing the transmission power enough to cover this node, energy expenditure would be reduced, thus, prolonging the lifetime of a battery-powered MANET. Existing works on variable power broadcasting based on local information are effective in achieving this objective. However, they are not optimized and can be improved by dynamically adjusting the transmission power based on where a broadcast message comes from. This paper proposes simple mechanisms based on local knowledge to adjust the transmission power dynamically and incorporates a timer suppression mechanism to further enhance the effectiveness of the protocols in reducing energy expenditure. Results of simulation studies confirm the effectiveness of the proposed enhancements.

I. INTRODUCTION

Mobile ad hoc networks (MANETs) are popular wireless networks that received tremendous attentions over the last decade due to their inherent flexibility and simple implementation. They are created on the fly without the need for any centralized networks. A typical MANET usually consists of hundreds to thousands of inexpensive, small-sized, and battery-powered wireless nodes that collaborate with each other to perform a network function. Notably, sensor networks are a major application of MANETs, which can be used for habitat monitoring, object tracking, fire detection, environmental monitoring, and so on.

Network-wide broadcasting or simply broadcasting is a process of delivering a message to all nodes in a MANET. Besides that, it also plays an important role in establishing routes for on demand routing protocols [10], building routing tables for table-driven routing protocols, and address assignment. Since nodes in a MANET are battery-powered, efficient broadcasting protocols are essential in prolonging the lifetime of nodes. Existing work on broadcasting mainly focuses on reducing the number of nodes involved in the broadcasting process while ensuring high delivery ratio [1-4]. Notably, one common feature of these protocols is the use of fixed transmission power.

If nodes are allowed to adjust their transmission power to cover nodes that need to be covered only, minimizing the number of nodes involved in the broadcasting process may not be the most energy efficient. Some existing work on broadcasting with variable transmission power can be found in [5-8]. The problem of finding the most energy efficient broadcasting can be solved either using centralized algorithms [7,8] where each node is equipped with knowledge of global network topology, or distributed algorithms [5,6] with local information. This paper focuses on distributed algorithms. There are two related protocols, namely PABLO (Power Adaptive Broadcasting with Local Information) [5] and INOP (Inside-Out Power Adaptive Approach) [6]. With either PABLO or INOP, prior to any broadcast message arrival, the optimal transmission power of a node is pre-determined based on its local knowledge of neighbors within its 2-hop. Once the transmission power is set, no matter from where a broadcast message arrives, a node will simply use its pre-determined power to rebroadcast the message when a local hold-off timer expires. This approach can be inflexible and not optimized. Fig. 1 illustrates the inefficiency of PABLO and INOP. Assume the optimized transmission power of node 2 obtained from PABLO or INOP (independent of where the message arrives) is $P_{1,2}$. When node 2 receives a message from node 1, it will rebroadcast using $P_{1,2}$ to cover node 3. Then, node 3 rebroadcasts to cover node 4. Clearly, this broadcasting consumes more energy because node 2 should use $P_{2,3}$ as its transmission power instead to cover node 3, assuming $P_{2,3} < P_{1,2}$.

This paper proposes some simple mechanisms to dynamically adjust the transmission power of nodes by considering only not-yet-covered (uncovered) nodes. From Fig. 1, if node 2 receives a message from node 1, it downward adjusts its transmission power to $P_{2,3}$ as only node 3 is to be covered. Further descriptions of the proposed solutions and other enhancements are given in Section III. Although MANETs support nodal mobility, in this paper our broadcasting protocols are designed with sensor networks in mind, where mobility support is not a major concern.

This paper is organized as follows. Section II gives detailed descriptions of PABLO and INOP while Section III describes the proposed optimizations and enhancements of the original PABLO and INOP. Section V provides descriptions of the

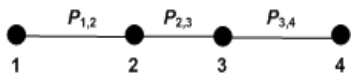


Fig. 1. A simple network that shows the limitations of PABLO and INOP

simulation settings and results. Finally, Section VI concludes the paper.

II. LITERATURE REVIEW

Some existing work on broadcasting with variable transmission power can be found in [5-8]. The problem of finding the most energy efficient broadcasting can be solved either using centralized algorithms [7,8] where each node has knowledge of global network topology, or distributed algorithms [5,6] with local information confined to 2-hop neighbors. This paper focuses on distributed algorithms. There are two related protocols, namely PABLO (Power Adaptive Broadcasting with Local Information) [5] and INOP (Inside-Out Power Adaptive Approach) [6].

Without loss of generality, we assume a common channel model that follows the power law model [9]:

$$P_{recv} = \frac{P_{tx}}{r^n} \quad (1)$$

where P_{recv} is the strength (or power) of the signal when it arrives at a receiver, P_{tx} is the transmission power, r is the transmission range, and n is the power loss exponent that takes a value between 2 and 4. Each node is assigned a default maximum transmission power, P_{max} to send HELLO message. This value is included in the header of the HELLO message. In order to adjust the transmission power, each node needs to determine the transmission power that it needs to reach each of its neighbors. When a node receives a HELLO message from a neighbor, it simply extracts the power level with which the packet is transmitted (P_{max}) and compute the required transmission power, P_{req} , to reach this neighbor using the following equation:

$$P_{req} = \left[\frac{P_{max}}{P_{recv}} \right] \times P_{threshold} \quad (2)$$

where $P_{threshold}$ is the minimum power for a packet to be received correctly.

The main idea of PABLO [5] is to find a nearer neighbor (relay node) to cover the furthest neighbor such that the total transmission power in reaching the furthest neighbor (via relays) is lower than direct transmission. In other words, PABLO follows an outside-in approach. Fig. 2 illustrates the operation of PABLO. In this example, the neighbors of node 10 in the set $N(10)$ includes nodes 9, 5, 0, and 3 where the furthest neighbor is node 9, followed by node 5, 0, and 3. The transmission power to reach nodes 9, 5, 0, and 3 is $P_{10,9} = 0.232021$, $P_{10,5} = 0.230102$, $P_{10,0} = 0.190218$, and $P_{10,3} = 0.103364$ respectively. Since node 9 is the furthest neighbor, node 10 evaluates whether node 0 is a suitable relay node to node 9. Since $P_{10,0} + P_{0,9} < P_{10,9}$, node

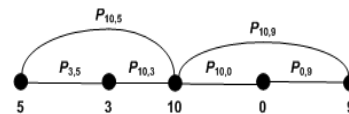


Fig. 2. A network that shows the optimization used in PABLO and INOP

10 will set its transmission power to $P_{10,0}$ and removes node 9 from $N(10)$. The next furthest neighbor is node 5. However, node 3 cannot be used as a relay node because $P_{10,3} + P_{3,5} > P_{10,5}$. Therefore, node 10 sets its transmission power to $P_{10,5}$ or 0.230102. Using PABLO, node s obtains its optimized transmission power based on Algorithm 1 below.

- 1) Find $k \in N(s)$ such that $(P_{sk} \geq P_{si}) \forall i \in N(s)$
- 2) Find $r \in N(s) - \{k\}$ such that $(P_{sr} + P_{rk}) \leq (P_{sq} + P_{qk}) \forall q \in N(s) - \{k\}$
- 3) if $(P_{sr} + P_{rk}) < P_{sk}$
- 4) Eliminate k from $N(s)$ // since k can be reached via r
- 5) else
- 6) Transmit directly to k

Algorithm 1: PABLO [5]

It should be noted that the above procedures (as well as Algorithm 2 below) are carried out independent of (in fact, prior to) actual broadcast message arrival.

In an active broadcast session, PABLO suppresses redundant broadcasting. As mentioned earlier, each node has knowledge of up to 2-hop neighbors. When receiving a broadcast message, a node could identify its immediate neighbors that have also received the same message. If all immediate neighbors have been covered, its rebroadcasting is inhibited. To give sufficient time to eliminate neighbors that have been covered, a rebroadcasting node chooses a random “backoff” timer before rebroadcast. If the timer expires and all neighbors have been covered, the rebroadcasting is canceled. Otherwise, the message is broadcast using its pre-determined transmission power.

PABLO sets the random broadcast timer (RBT) of node s as follows:

$$RBT = \left[\frac{MaxRange - distance}{MaxRange} + \frac{MaxReach - unreached}{MaxReach} \right] \times rand() \times MaxDel \quad (3)$$

In the equation above, $MaxRange$ denotes the maximum transmission range of a node, and $MaxReach$ the maximum number of neighbors that a node can cover. It is expected [5] that the transmission range and number of immediate neighbors of all nodes should not exceed 250 meters and 20 respectively, $MaxRange = 250$ and $MaxReach = 20$. $MaxRange = 250$ is obtained by assuming $n = 2$ in (1) and $P_{threshold} = 3.652 \times 10^{10}$ and $P_{max} = 0.28183815$ in (2). $distance$ denotes the distance of node s from the message originating node while $unreached$ represents the number of not-yet-covered neighbors of node s

(as seen by node s). $rand()$ is a random number generator that generates a uniformly distributed number between $[0,1]$ and $MaxDel$ is the pre-determined maximum rebroadcast “back-off” time.

INOP [6] is proposed as an enhancement of PABLO. While PABLO is based on outside-in, INOP on the other hand, is based on inside-out approach. It starts from the nearest node and moves outward. It considers the cumulative power to reach a node and covers all nodes up to this distance during broadcasting. Fig. 2 is again used for illustration of INOP. Since node 3 is the nearest neighbor of node 10, the cumulative power to reach node 3 is $P_{10,3}^{cumu} = P_{10,3}$. If node 10 were to transmit using transmission power $P_{10,3}$, then neighbor 3 will be covered. Next, the cumulative power to reach the second furthest neighbor, that is node 0, is calculated as the minimum of (a) the power to reach node 0 directly from node 10 and (b) the sum of the cumulative power to reach the previous nearest neighbor, i.e. node 3, $P_{10,3}^{cumu}$, and the power to reach node 3 from a relay node. Since there is no relay node, the cumulative power now becomes $P_{10,0}^{cumu} = P_{10,0}$. The next nearest neighbor is node 5. Since $\min((P_{10,0}^{cumu} + P_{3,5}), P_{10,5})$ is $P_{10,5}$ the cumulative power to reach node 5 is $P_{10,5}^{cumu} = P_{10,5}$. The next neighbor is node 9. By evaluating $\min((P_{10,5}^{cumu} + P_{0,9}), P_{10,9})$, the cumulative power to reach node 9 is $P_{10,9}^{cumu} = P_{10,9}$. Finally, node 10 sets its transmission power to reach the furthest neighbor 9 or $P_{10,9}$. Using INOP, node s obtains its optimized transmission power based on the procedures below.

- 1) Sort $N(s) = \{1, 2, 3, \dots, n\}$ such that $P_1 \leq P_2 \leq \dots \leq P_n$
- 2) $n = |N(s)|, F = \emptyset$
- 3) for ($i = 1; i \leq n; i++$)
- 4) Let $R \subset N(s)$ such that $\forall q \in R, P_{sq} < P_{si}$ and $P_{qi} < P_{si}$
- 5) if ($R = \emptyset$)
- 6) $P_{si}^{cumu} = P_{si}$
- 7) else
- 8) Find $r \in R$ such that $P_{ri} \leq P_{qi} \forall q \in R$
- 9) $P_{si}^{cumu} = \min((P_{s(i-1)}^{cumu} + P_{ri}), P_{si})$
- 10) if ($P_{si}^{cumu} = P_{si}$)
- 11) $F = F \cup \{i\}$
- 12) Source s transmits with P_{si} such that $i \geq j, \forall j \in F$

Algorithm 2: INOP [6]

Similar to PABLO, INOP also uses 2-hop neighborhood information in eliminating redundant broadcasting in an active broadcast session. During “backoff” period, INOP listens to the channel for redundant messages and eliminate nodes that have been covered. When the “backoff” period expires and there are not-yet-covered neighbors, the message is rebroadcast. INOP sets the “backoff” timer or RBT based on the following ratio:

$$RBT = \left\lceil \frac{1}{\|N(s)\|} \right\rceil \times rand() \quad (4)$$

where $\|N(s)\|$ is the size of set $N(s)$, or the number of not-yet-covered neighbors of the node under consideration.

III. PROPOSED ENHANCEMENTS

PABLO and INOP are effective in reducing the overall energy expenditure but they can be further enhanced. A major deficiency, as we have pointed out using the example in Fig. 1, is that optimized transmission powers of individual nodes are determined, using Algorithm 1 or 2, *prior to* any broadcast message arrival, and are intended to be generic for any broadcast session to use. But for a specific session, the pre-determined transmission power may not be optimal. In Fig. 1, since node 2 receives the message from node 1, it does not need to cover node 1 during rebroadcasting. As such, node 2 can reduce its transmission power to $P_{2,3}$ for covering (the next furthest away) node 3 only.

In this section, several new algorithms are proposed to set the transmission power of a node in real-time and based on where an actual broadcast message arrives. We also show that the extra complexity involved is marginal.

A. E-PABLO and E-INOP

Firstly, two simple extensions of PABLO and INOP, denoted as E-PABLO and E-INOP, are designed. Both of them retain most of the features of the original schemes. Initially, all nodes set their optimized transmission power using PABLO or INOP. When a message arrives, a node (s) determines the set of uncovered neighbors. Next, it starts its RBT and during this interval, it listens to the channel for subsequent broadcast messages. When the RBT expires, node s checks whether there is any uncovered neighbor. It will only rebroadcast the message if there is at least one uncovered neighbor. Based on the current not-yet-covered neighbor information (i.e. $N(s)$), node s uses Algorithm 1 or Algorithm 2 to readjust its transmission power.

In Fig. 1, the pre-determined optimized transmission power of node 2 is $P_{1,2}$, obtained from PABLO or INOP. Assume $P_{2,3} < P_{1,2}$. When a broadcast message is received from node 1, using E-PABLO or E-INOP, node 2 removes node 1 from its not-yet-covered set $N(2)$, which now contains only node 3, and starts its RBT using either (3) or (4). During “backoff”, node 2 may receive the same broadcast message from other neighbors and updates $N(2)$, if necessary. When the timer expires and $N(2)$ is empty, node 2 is inhibited from sending. Otherwise, it determines its transmission power using Algorithm 1 or 2 based on the current $N(2)$ and rebroadcasts the message. In this example, the actual transmission power of node 2 can thus be lowered from $P_{1,2}$ to $P_{2,3}$.

B. ES-PABLO and ES-INOP

The heuristics behind the RBT settings of both PABLO and INOP is to ensure that nodes with higher rebroadcast priority timeout earlier. For example, the RBT adopted by INOP is to give priority to nodes with most not-yet-covered neighbors rebroadcast first so as to quickly cover as many nodes as possible. Although this type of heuristics is effective

in reducing the number of broadcasting [2,4], a recent study [4] shows that a simple timer suppression mechanism is effective in further reducing the number of broadcasting. Specifically, when a node is in “backoff” state and receives the same (redundant) broadcast message again, its “backoff” timer RBT is reset and re-started using either (3) or (4) based on $\|N(s)\|$, the current number of not-yet-covered nodes in $N(s)$. This timer suppression mechanism tends to ensure nodes with more neighbors to timeout earlier by also taking the change during “backoff” period into account. The suppressed timer will timeout eventually (if $\|N(s)\| > 0$) and the rebroadcasting will take place to reach any uncovered nodes.

Both ES-PABLO and ES-INOP incorporate the timer suppression mechanism while retaining the features of E-PABLO and E-INOP respectively. Intuitively, fewer nodes rebroadcast means less energy consumption. In addition, the ability to dynamically adjust the transmission power based on latest neighborhood information is also expected to lower the energy consumption. The timer suppression mechanism further reduces the number of broadcasting nodes, thus, resulting in lower energy expenditure.

C. V-ESBA

The Enhanced Scalable Broadcast Algorithm (ESBA) [4] is proposed to solve the limitations with the original SBA protocol [2] and further enhances its effectiveness in reducing the number of broadcasting nodes by incorporating a timer suppression mechanism similar to the one described in the previous sub-section. This protocol is proven to be effective in reducing redundant rebroadcasting [4]. A node s sets its RBT timer as follows:

$$RBT = \frac{1}{d^*(s)} + \frac{1}{d_{max}^*(s)} \quad (5)$$

where $d^*(s)$ is the degree of node s based on the latest not-yet-covered nodes information and $d_{max}^*(s)$ is the maximum degree of any neighbor of node s based on the latest not-yet-covered nodes information.

However, it is based on fixed transmission power. Therefore, it may not be energy efficient. Instead of adopting the techniques similar to Algorithm 1 or 2, which requires more effort in adjusting the transmission power, Variable ESBA (V-ESBA) extends ESBA by simply reducing the transmission power of rebroadcasting nodes to cover the current furthest uncovered neighbor. In other words, a node that receives a message starts its RBT as specified in [4] and rebroadcast the message upon expiration with enough power to cover the furthest uncovered neighbor. This provides a simple way to cut down the energy bill of ESBA. Here, we are interested in determining how well V-ESBA reduces energy consumption as compared with the enhancements mentioned in the previous sub-sections.

D. M-PABLO and M-INOP

M-PABLO and M-INOP retains most of the original features of E-PABLO and E-INOP respectively except they utilize the simple idea of readjusting the transmission power proposed

TABLE I
COMMON SIMULATION PARAMETERS

Simulation Parameter	Value
Simulator	NS2.33
MAC Protocol	IEEE 802.11 without RTS/CTS/ACK
Bandwidth	2 Mbps
Network Size	1000 m \times 1000 m
Default Transmission Range	250 m
Data Packet Size	512 bytes
Simulation Duration	10 seconds
Data Broadcast Rate	1 packet per second (pps)
No. of Nodes	20, 40, 60, 80, and 100
Confidence Interval	95%

in the previous sub-section. Initially, both M-PABLO and M-INOP use the transmission power obtained from PABLO and INOP respectively. The source node would broadcast using this transmission power. When the RBT timers of rebroadcasting nodes expire, they readjust the transmission power to cover the current furthest not-yet-covered neighbor.

E. MS-PABLO and MS-INOP

MS-PABLO and MS-INOP are similar to ES-PABLO and ES-INOP respectively. However, a rebroadcasting node will rebroadcast a message upon expiration of its RBT to cover the current furthest not-yet-covered neighbor.

F. P-PABLO and P-INOP

Another interesting enhancement is to incorporate the RBT proposed in [4] and used in ESBA and V-ESBA into ES-PABLO and ES-INOP. That is, the proposed P-PABLO and P-INOP retains all features of ES-PABLO and ES-INOP respectively except for the RBT timer which follows the one used by ESBA and V-ESBA.

IV. PERFORMANCE EVALUATIONS

This section first presents the simulations settings for comparing the performances of PABLO, INOP, ESBA, and their associated enhancements specified in section III. This is followed by detailed analysis of the results. Table I summarizes the general simulation parameters. The maximum transmission power, P_{max} is set to 0.2818384, which covers a range of 250 meters with the threshold power at the receiver $P_{threshold} = 3.652 \times 10^{-10}$, and the power loss exponent $n = 2$. The $MaxDel$ in equation (3) is 1 second. The network size ranges from 20 to 100 nodes and there are 10 different topologies for each network. In the simulations, each node in a network gets to become the source node that broadcast data messages every one second until the end of the simulation time which is 10 seconds. For example, in the simulation that involves 20 nodes, node 0 will become the source in the first sample, node 1 in the second, and so on. This is the same for networks of 40, 60, 80, and 100 nodes. In each simulation run, ten broadcast messages, each identified by a sequence number, are generated with an inter-message interval of one second.

The performance metrics used for comparing the protocols are:

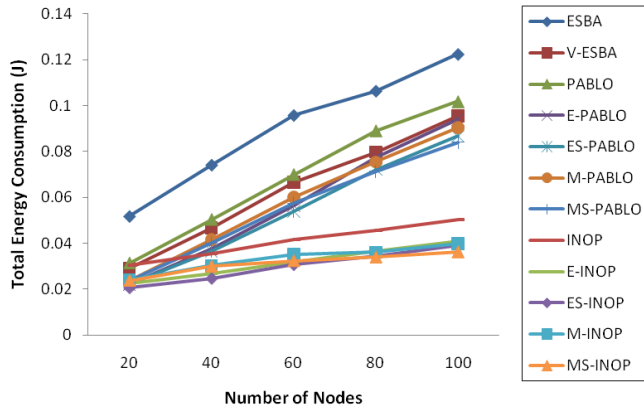


Fig. 3. Total Energy Consumption (J) vs. Number of Nodes

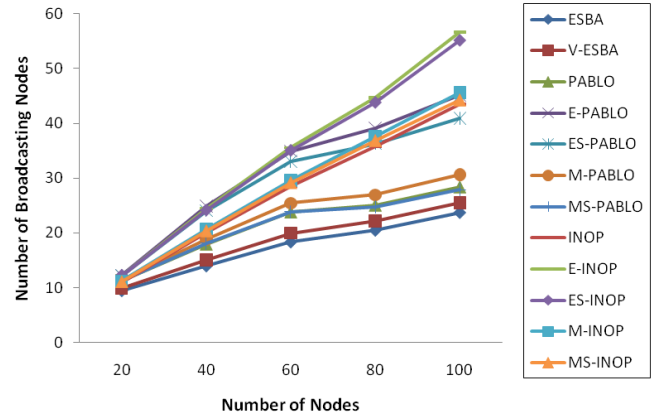


Fig. 4. Number of Broadcasting Nodes vs. Number of Nodes

- Total Energy Consumption: the sum of transmission power that is used by nodes to perform the broadcasting. Energy consumed is in Joules (J) and is represented as a product of the transmission power (in Watts) and the time duration of transmission. Since the packet size is fixed, the time duration for a single transmission is the same for every packet.
- Number of Broadcasting Nodes: total number of nodes involved in broadcasting the data messages.
- Broadcast Latency: time required for the slowest/last node to get a successful packet.
- Packet Delivery Ratio (PDR): ratio between the number of packets that are successfully received and the total number of packets sent.

The results for total energy consumption are presented in Fig. 4. It clearly shows that ESBA consumes the highest energy. This is predictable since every node uses the fixed maximum transmission power. Fig. 4 also confirms a previous finding [6] that INOP is significantly more effective than PABLO in reducing the total energy consumption. By allowing transmission power adjustment in real-time, the two proposed enhancements, E-PABLO and E-INOP, are effective in reducing energy expenditure as they outperform their respective predecessor. Further equipping with the timer suppression mechanism, ES-PABLO and ES-INOP further improve the total energy consumption. One interesting finding in this study is that V-ESBA actually outperforms PABLO, which shows that simple transmission power adjustment is sometimes sufficient to produce significant energy savings. The same is observed for M-PABLO, MS-PABLO, M-INOP, and MS-INOP where they outperformed E-PABLO, ES-PABLO, E-INOP, and ES-INOP respectively in networks of 80 and 100 nodes. The results for P-PABLO and P-INOP are not shown in this paper because they do not yield any performance improvement.

Fig. 5 presents the number of broadcasting nodes for all protocols. It shows that ESBA is the most effective protocol in reducing redundant broadcasting. Another observation is that the proposed protocols, namely E-PABLO, E-INOP, and V-

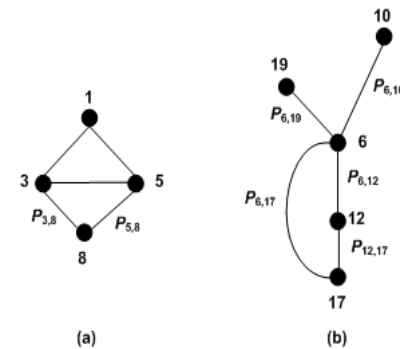


Fig. 5. Examples showing the proposed enhancements require additional broadcasting nodes

ESBA, require more broadcasting nodes than PABLO, INOP, and ESBA respectively. From Fig. 4, these enhancements are more energy efficient because they reduce the transmission power by using shorter transmission range.

Fig. 6 (a) shows why V-ESBA requires more broadcasting nodes than ESBA. Let $P_{3,8} < P_{5,8}$. In ESBA, assume node 1 is the source node and node 3 rebroadcast the message from node 1 using the default transmission power that covers nodes 8 and 5. However, in V-ESBA, node 3 adjusts its transmission power to cover the further uncovered neighbor which is node 8. When node 3 rebroadcast the message, node 5 does not know that node 8 has been covered and then perform the (redundant) rebroadcasting. In Fig. 6 (b), assume the transmission power of node 6 obtained from PABLO or INOP is $P_{6,10}$, where node 10 is the furthest neighbor. When node 6 broadcasts a message, it covers all neighbors. Now, assume nodes 10 and 19 have been covered by a previous broadcasting. In E-PABLO and E-INOP, these two nodes are removed from $N(6)$. By using the updated $N(6)$, node 10 adjusts its transmission power to $P_{6,12}$, thus having an additional broadcasting by node 12 to reach node 17. Even though the proposed protocols required additional broadcasting, they incurred lower energy consumption because the saving in using a smaller transmission power outweighs the loss in using more broadcasting nodes. However, for M-

PABLO, MS-PABLO, M-INOP, and MS-INOP, they incurred lower energy expenditure than E-PABLO, ES-PABLO, E-INOP, and ES-INOP respectively in networks of 80 and 100 nodes due to fewer broadcasting nodes. Another observation is that the proposed timer suppression mechanism in ES-PABLO and ES-INOP is effective in reducing the number of broadcasting, thus lower the total energy consumption as shown in Fig. 4.

In terms of broadcast latency, all protocols show a similar decreasing trend as the number of nodes increases, as indicated in Fig. 7. However, this contradicts with the intuition that broadcast latency should increase with the number of nodes (as the hop-distance tends to increase with nodes). The reason for the decreasing trend is due to the RBT or “backoff” timer settings. The heuristics for the timer settings is to allow nodes with higher priority broadcast earlier. For example, in INOP, the RBT as specified in (4) gives nodes with larger $\|N(s)\|$ shorter “backoff” delays. In a sparse network like 20 nodes, a node may have only one uncovered neighbor but its RBT in (4) is 1 second. For the case of dense network like 100 nodes, the chances of having few uncovered neighbors is low, thus resulting in shorter RBTs. The timer suppression mechanism appears to incur long broadcast latency. However, Fig. 7 shows otherwise and the increase in delay is only marginal in most cases. This is because suppressed nodes may find all neighbors already covered when their RBT expires.

Finally, Fig. 8 presents the results on the packet delivery ratio (PDR) achieved by all protocols. Again, the trend is consistent for all protocols. In the absence of mobility and channel errors, all protocols achieved delivery ratio of 98% and above. The loss is due to the impact of hidden terminals and simultaneous packet transmissions.

V. CONCLUSION

Existing protocols based on variable transmission power using local information include PABLO and INOP. INOP was proposed to solve the inefficiency of PABLO [6]. One notable similarity of both protocols is that the optimized transmission power is found before actual packet transmission takes place. We have proven that this approach is not optimized and can be improved by considering where the messages come from when adjusting the transmission power. We also found that timer suppression can be used to further reduce the number of rebroadcasting nodes. As a result, 10 broadcasting algorithms, E-PABLO, E-INOP, ES-PABLO, ES-INOP, M-PABLO, M-INOP, MS-PABLO, MS-INOP, P-PABLO, and P-INOP were proposed to address the limitations with the original PABLO and INOP. Simulation results confirmed that all proposed solutions except for P-PABLO and P-INOP are effective in further reducing the total energy consumption. Our future work would involve studying the tradeoff between performance gains and complexity.

REFERENCES

[1] B. Williams and T. Camp, “Comparison of Broadcasting Techniques for Mobile Ad Hoc Networks,” *Proc. of MOBIHOC*, 2002.

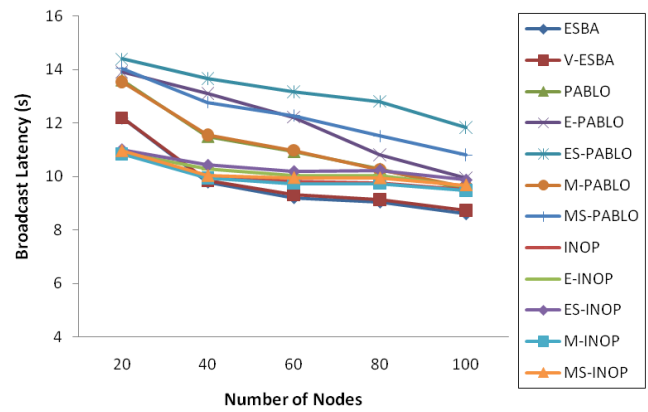


Fig. 6. Broadcast Latency (s) vs. Number of Nodes

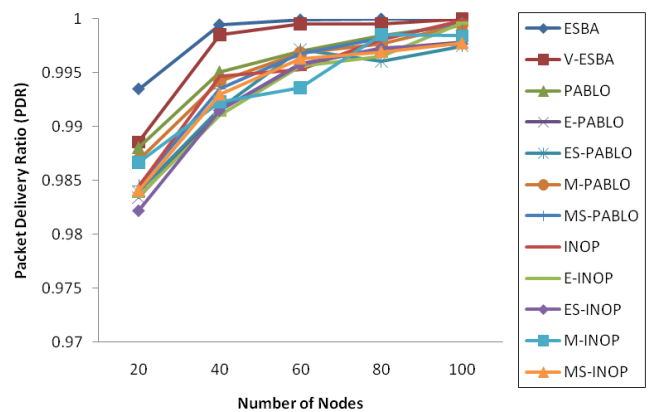


Fig. 7. Packet Delivery Ratio (PDR) vs. Number of Nodes

[2] W. Peng and X. C. Lu, “On the Reduction of Broadcast Redundancy in Mobile Ad Hoc Networks,” *Proc. of MOBIHOC*, 2000.

[3] Y. C. Tseng, S. Y. Ni, Y. S. Chen, and J. P. Sheu, “The Broadcast Storm Problem in a Mobile Ad Hoc Network,” *Wireless Networks*, vol. 8, no. 2-3, pp. 153-167, Mar. 2002.

[4] W. Woon and K. Y. Yeung, “Self-Pruning Broadcasting for Mobile Ad Hoc Networks,” *IEEE Globecom*, Hawaii, USA, Dec. 2009.

[5] X. Chen, M. Faloutsos, S. Krishnamurthy, “Power Adaptive Broadcasting with Local Information in Ad Hoc Networks,” *Proc. of IEEE ICNP*, 2003.

[6] A. Chiganmi, M. Baysan, K. Sarac, and R. Prakash, “Variable Power Broadcast Using Local Information in Ad Hoc Networks,” *Ad Hoc Networks*, vol. 6, no. 5, pp. 675 - 695, July 2008.

[7] J. Cartigny, F. Ingelrest, D. Simplot-Ryl, and I. Stojmenovic, “Localized LMST and RNG based minimum-energy broadcast protocols in ad hoc networks,” *Ad Hoc Networks*, vol. 3, no. 1, pp. 1 - 16, Jan. 2005.

[8] N. Li and J. C. Hou, “A Scalable, Power-Efficient Broadcast Algorithm for Wireless Networks,” *Wireless Networks*, vol. 12, no. 4, pp. 495 - 509, Jul. 2006.

[9] T. Rappaport, “Wireless Communications, Principles and Practices,” Prentice Hall, 1996.

[10] H. S. Chiu, Kwan L. Yeung, and K. S. Lui, “J-CAR: An Efficient Joint Channel Assignment and Routing Protocol for IEEE 802.11-based Multi-Channel Multi-Interface Mobile Ad Hoc Networks,” *IEEE Trans. on Wireless Commun.*, vol. 8, no. 4, pp.1706 - 1715, Apr. 2009.