Empirical Analysis of Transmission Power Control Algorithms for Wireless Sensor Networks

Jaein Jeong
David E. Culler
Jae-Hyuk Oh

Electrical Engineering and Computer Sciences
University of California at Berkeley

Technical Report No. UCB/EECS-2005-16
http://www.eecs.berkeley.edu/Pubs/TechRpts/2005/EECS-2005-16.html

November 21, 2005

Empirical Analysis of Transmission Power Control Algorithms for Wireless Sensor Networks

Abstract. Using a real wireless sensor network testbed, we have demonstrated and analyzed the performance of fixed and dynamic transmission-power-control algorithms. Several transmission power control algorithms have been proposed, but little empirical data is available. We test several popular algorithms on convergence and aggregation traffic patterns, which are commonly used in multi-hop wireless sensor networks for data collection. For the convergence traffic pattern, we have found that the dynamic transmission-power-control algorithm enabled a wireless sensor network to sustain the desired end-to-end throughput while consuming less energy than the fixed transmission-power-control algorithm.

1 Introduction

Wireless sensor networks are becoming possible choices for supporting data communication in industrial applications (e.g. HVAC, building control). Since these applications can span a long distance possibly without any line of sight between communicating sensor nodes, it is required that the sensor networks support multi-hop routing. Many of these commercial control systems are based on wired links and have higher requirement for the data throughput. Thus, it is desirable that the wireless sensor networks deliver the traffic at high data rate while operating on the limited energy resources. When we assume that all sensor nodes are scheduled to report their sensor data at the same rate, we can achieve high data throughput by reducing the communication contention. One way to do this is to adjust the transmission power of the radio transceiver of each sensor node. Setting the radio transmission power to a maximum value doesn't necessarily mean the best possible throughput because the interference increases as well as the range of a sensor node. A number of previous studies tried to achieve the best possible throughput or network lifetime by adjusting the radio transmission power of individual node and we call these as dynamic transmission power control algorithms. However, these works have limitations: First, previous approaches were demonstrated using either idealized simulators [RRH00, KKW+03, EKCD00, MBH01, WLBW01] or hardware platforms which do not have the resource limitations of wireless sensor networks [SKH04, ESWW00]. Second, they did not consider various traffic patterns in wireless sensor networks. This is because their protocols are designed for wireless LAN [EKCD00, RRH00, WLBW01, ESWW00, MBH01], or they just used the traffic pattern that can be readily supported by their experimental platform [KKW+03, SKH04].

In this paper, we compare the dynamic transmission power control algorithms introduced in the literature, and describe a representative algorithm which can be used for wireless sensor networks. Then, we present a performance analysis of the implementation of the algorithm on a large Mica2dot-based wireless sensor network testbed using convergence and aggregation traffic patterns which are commonly used for data collection in multi-hop wireless sensor networks. We have found that using the dynamic transmission power control algorithm saves the energy consumption while making little difference to the throughput for the case of convergence traffic pattern. We have also found that the sensor network had higher end-to-end throughput and fairness with the aggregation traffic pattern than with the convergence traffic pattern.

We review the related work in Section 2. We describe our proposed algorithm in Section 3.1 and various traffic patterns for wireless sensor networks in Section 3.2. We present experimental methodology in Section 4 and experiment results in Section 5. And we conclude this paper in Section 6.

2 Related Work

There are some previous work that tried to achieve the best possible performance by changing the radio transmission power. The basic ideas of these dynamic transmission power control algorithms are similar to one another. The algorithms find how many neighbors each node has and adjust the radio transmission power of each node so that the number of neighbors stays within the desired range. LINT(Local Information No Topology) / LILT(Local Information Link-State Topology) by Ramanathan *et al.* [RRH00], LMA(Local Mean Algorithm)/ LMN(Local Mean of Neighbors) by Kubisch *et al.* [KKW$^+$03] and PCBL(Power Control with BlackListing) by Son *et al.* [SKH04] are such examples. Whereas, these algorithms can vary depending on the parameters like how neighbors are chosen, what metric is optimized for, how the experiment is set up and what traffic patterns are considered.

In the literature, three different methods have been used for choosing a neighbor: (1) connectivity, (2) packet reception rate (PRR) and (3) received signal strength (RSS) based methods. A connectivity based method simply counts different sender IDs without differentiating link quality and is easy to implement, but it does not achieve the desired level of connectivity. A PRR based method and a RSS based method choose neighbors by filtering incoming neighbors that have their metrics (PRR or RSS) better than a certain threshold. LINT/LILT snoops the data packets each node receives to count the number of neighbors by checking the sender ID of the data packets, and it adjusts the radio transmission power if the number of neighbors is outside the bound of $[d_l, d_h]$ (RSS based). LMA/LMN determines the range of a node by counting from how many other nodes the node has received acknowledgement for the beacon message it has sent (Connectivity based). In the algorithm of ElBatt *et al.* [EKCD00], each node ranks neighboring nodes in the order of received signal strength and adjusts its radio transmission power so that the node covers only N-most neighbors

(RSS base). With PCBL, each sensor node adjusts its radio transmission power with the smallest possible value such taht $PRR > PRR_{threshold}$. And it also blacklists (filters) the nodes that have too low PRR values (PRR based).

The previous work set their objective as improving either throughput or energy consumption (network lifetime). We can have the following findings from their experiment results: First, the improvement of the throughput from these transmission power control algorithms is limited. The maximum throughput of LINT/LILT is no higher than that of the fixed transmission power control algorithm (NONE) although LINT/LILT had higher throughput than NONE for the node density of 0.5 or higher (Fig. 8 at page 9). ElBatt *et al.* showed that their radio transmission power control algorithm achieved the best possible end-to-end throughput as they increased the number of neighbors, but they did not show how well the algorithm performed compared to the case when they didn't use the algorithm. Son *et al.* showed that the best possible packet delivery rate of PCBL is as much as that of M-BL (fixed transmission power algorithm at maximum power with blacklisting) and M-BL has highest packet delivery rate among the fixed transmission power control algorithms. Second, the previous work showed using radio transmission power control algorithms saved the energy consumption. LMA and LMN had about 37000 and 42000 seconds for the network lifetime compared to 26000 sec of the fixed transmission power control algorithm. This means 42% and 62% improvement (Fig. 4 and 5 at page 4). ElBatt *et al.* showed that setting the radio transmission power differently for different destinations consumed less power compared to setting the radio transmission power the same level for all destinations. Son *et al.* showed that PCBL consumed less energy than M-BL and more energy than TPP-P0 (fixed transmission power at minimum power), but PCBL consumed the least energy when they considered the energy consumption per successful packet delivery.

As for the experiment platform, the previous work was demonstrated demonstrated with either idealized simulators [RRH00, KKW+03, EKCD00, MBH01, WLBW01] (LINT/LILT with Rooftop C++ toolkit simulator, LMA/LMN with OMNet++ simulation tool and the work of ElBatt *et al.* with OPNET simulation model). Or it was tested with hardware platforms which do not have the same resource limitations of wireless sensor networks [SKH04, ESWW00] (PCBL with PC104 testbed).

There are some optimization techniques based on IEEE 802.11 specific protocols, but they are not suitable for wireless sensor networks due to their resource constraints. Monks *et al.* [MBH01], Wu *et al.* [WTS03] and Jung *et al.* [JV02] adjusted the radio transmission power of RTS/CTS packets to increase the throughput or to reduce the energy consumption. This puts severe overhead on sensor nodes that have smaller packet size although it helps avoiding channel contention. Ebert *et al.* [ESWW00] changed the radio transmission power depending on the packet size or they fragmented a packet into smaller fragments for optimum energy consumption. This idea is not suitable for our sensor network platform using fixed, small size packets.

Table 1 summarizes different transmission-power-control algorithms.

Table 1. Comparison of different transmission-power-control algorithms

Ramanathan *et al.*

Metrics	Throughput, transmission power, delay
Neighbors	Received-signal-strength based
Experiment	Simulation (Rooftop C++ Toolkit)
Traffic	Point-to-point routing between two random points

Kubisch *et al.*

Metrics	Network lifetime, connectivity
Neighbors	Connectivity based
Experiment	Simulation (OMNet++)
Traffic	Request/reply between two nodes using point-to-point routing

ElBatt *et al.*

Metrics	Throughput, transmission power
Neighbors	Received-signal-strength based
Experiment	Simulation (OPNET)
Traffic	Random point-to-point routing

Son *et al.*

Metrics	Packet reception rate (Throughput)
Neighbors	Packet-reception-rate based
Experiment	PC 104 nodes with Mica2 as radio transceiver
Traffic	Point-to-point routing using Directed Diffusion

Monks *et al.*

Metrics	Throughput
Neighbors	Received-signal-strength based
Experiment	Simulation (ns-2)
Traffic	Single-hop local communication

Ebert *et al.*

Metrics	Energy consumption, delay
Neighbors	Packet-reception-rate based
Experiment	Two laptops with wireless LAN
Traffic	Single-hop local communication

Wattenhofer *et al.*

Metrics	Coverage (# of neighbors)
Neighbors	Connectivity, angle-of-arrival (AOA)
Experiment	Simulation (ns-2 based on WaveLAN-I radio)

3 Design

3.1 Description of Transmission Power Control Algorithm

We present a dynamic transmission-power-control algorithm that attempts to improve the throughput, fairness and energy consumption of multi-hop wireless sensor networks. The algorithm works in two steps:

1. Find how many neighbors a sensor node has.
2. Adjust the radio transmission power of the sensor node so that the number of neighbors stays within the desired range.

As a way of choosing a neighbor, a PRR-based or RSS-based method is preferred over a connectivity-based method. This is because a PRR-based or RSS-based method achieve the desired level of connectivity by filtering incoming neighbors that have their metrics (PRR or RSS) better than a certain threshold. A PRR based protocol has advantages in that PRR is directly related to the quality of a link and it doesn't require any special hardware support. However, it has the software overhead of maintaining the neighbor table. A RSS based protocol can determine the quality of a link without much overhead of maintaining neighbor table with assistance from the radio hardware. One drawback is that RSS is sensitive to the background noise and does not accurately measure the quality of a link. Our algorithm uses RSS to choose a neighbor.

In order to adjust the radio-transmission power to the right level, we use the notion of an effective neighbor. A node is an effective neighbor of n_a if n_a knows that it can hear n_a, and the number of effective neighbors of n_a is the sum of all the effective neighbors of n_a. A sensor node can tell its number of effective neighbors (N) using the following protocol (Fig 1).

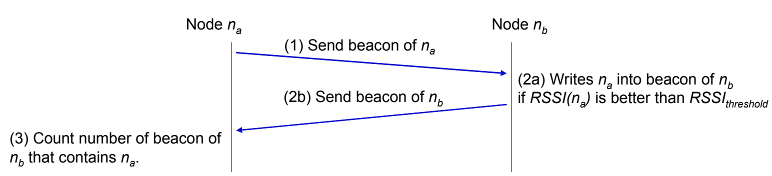

Fig. 1. Finding number of effective neighbors

1. Each node n_a sends a beacon message.
2. A node that hears the beacon message from n_a with the link quality better than a pre-defined threshold $RSSI_{threshold}$ records the source ID of the message. When the node sends its beacon message, it piggybacks the list of neighbors on the beacon message.
3. Node n_a hears a beacon message from another node and it can tell whether the node has heard n_a by looking at the neighbor list which is piggybacked on the beacon message. Node n_a counts all the nodes that have heard n_a.

After finding the number of neighbors, our algorithm adjusts the radio transmission power so that number of effective effective neighbors N converges to a predefined value N_{target} (Fig 2).

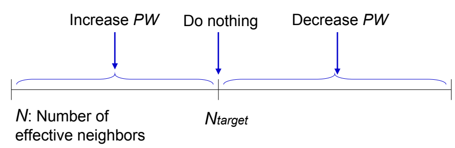

Fig. 2. Adjusting radio-transmission power

The transmission-power-control algorithm initializes the transmission power P, the step D and the sign S as follows:

$$P_0 = P_{max},\ D_0 = D_{init},\ S_0 = 1$$

At step 1, the algorithm either increases or decreases the transmission power P_1 depending on whether the number of neighbors N is larger or smaller than N_{target}:

$$\begin{aligned} S_1 &= sign(N_1 - N_{target}) \\ D_1 &= D_{init} \\ P_1 &= \min\{P_0 - S_1 D_1, P_{max}\} \end{aligned}$$

where $sign(x)$ is a function which is defined as follows:

$$sign(x) = 1\ (x \geq 0),\ \ -1\ (x < 0)$$

At step $k >= 2$, the algorithm either increases or decreases the transmission power as it did at step 1. In addition, the algorithm divides the step D_k by 2 if the sign S_k is different from the previous step S_{k-1}. This algorithm makes the step D_k smaller as N_k moves close to N_{target} so that the radio-transmission power converges to a desired value:

$$\begin{aligned} S_k &= sign(N_k - N_{target}) \\ D_k &= \max\{\frac{3 + S_k S_{k-1}}{4} D_{k-1}, 1\} \\ P_k &= \max\{\min\{P_{k-1} - S_k D_k, P_{max}\}, P_{min}\} \end{aligned}$$

3.2 Traffic Patterns

A sensor network application can use different traffic patterns. The traffic of a wireless sensor network can be either single hop or multi hop. Multi-hop traffic patterns can be further divided depending on the number of sender and receiver nodes, or whether the network support in-network processing. Based on these criteria we can categorize the traffic patterns into (a) local communication, (b) point-to-point routing, (c) convergence, (d) aggregation and (e) divergence. Figure 3 illustrates different traffic patterns.

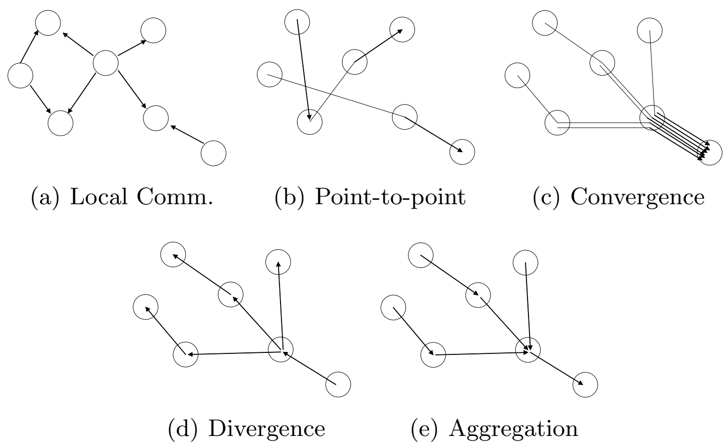

Fig. 3. Different Traffic Patterns

Local communication is used to broadcast the status of a node to its neighbors and is also used to transmit the data between the two nodes directly. Point-to-point routing is used to send a data packet from an arbitrary node to another arbitrary node, and this is commonly used in wireless LAN environment. With the convergence traffic pattern, the data packets of multiple nodes are routed to a single base node. The convergence traffic pattern is commonly used for data collection in wireless sensor networks. With the aggregate traffic pattern, the data packets can be processed in the relaying nodes and the aggregate value is routed to the base node rather than the raw data. Finally, divergence traffic is used to send a command from the base node to other sensor nodes.

Previous works for dynamic transmission power control do not consider different traffic patterns in wireless sensor networks because they are based on wireless LAN where point-to-point routing or local communication is more common [EKCD00, RRH00, WLBW01, ESWW00, MBH01], or they just used the traffic pattern that can be readily supported by their experimental platform [KKW$^+$03, SKH04]. In this paper, we study the behavior of the dynamic transmission power control algorithm for the convergence and aggregation traffic patterns which are used for data collection in wireless sensor networks.

4 Experimental Methodology

4.1 Platform

In order to monitor the wireless communication behavior of a real sensor network, we use the Smote testbed that consists of 78 Mica2dot sensor nodes. Each sensor node can be programmed and monitored through the Ethernet programming board connected to the sensor node (Figure 4(a)). Smote provides a convenient a convenient test environment by exposing the UART (Universal Asynchronous Receiver/Transmitter) of each mote as a TCP port. A testbed user can easily monitor the behavior of the overall sensor network as well as each individual sensor node by hearing the messages from the TCP connections. We used 22 sensor nodes including the base node as shown in Figure 4(b). The Mica2dot sensor node has Chipcon CC1000 radio transceiver, which allows a user to change the radio-transmission-power output between -20 dBm to 10 dBm (corresponding current consumption 5.3 mA to 26.7 mA) by setting the radio-transmission-power register values between 1 to 255 [Chi].

(a) A Smote node (b) Node Layout for experiments

Fig. 4. Smote, a wireless sensor network testbed

As for the software platform running on this testbed, we use the TinyOS B-MAC protocol [PHC04], which is a CSMA/CA-based MAC protocol, and the TinyOS *MintRoute* multi-hop routing protocol. MintRoute exposes *Receive* interface which can be used to measure the end-to-end throughput at the base node. *MintRoute* also exposes *Intercept* and *Snoop* interfaces, and this allows monitoring neighborhood information (e.g. number of neighbors) without affecting the function of the routing module. *MintRoute* itself does not support aggregation, so we emulate aggregation by intercepting the forwarded messages.

4.2 Performance Metrics

For the experiments, we measure the following performance metrics: end-to-end data throughput, energy consumption, neighbor distribution, routing status (e.g. number of hops) and reduction for aggregation traffic.

Throughput The per-node end-to-end throughput for node i is defined as the rate of the number of data packets from node i that have arrived at the base node ($nPackets$) over the elapsed time since the first packet from node i has arrived ($Duration$). We use the average per-node throughput to compare the results of different experiment runs.

$$\text{Per-node throughput for } i = \frac{nPackets(i)}{Duration}$$

The fairness index of the throughput shows how evenly each sensor node delivers the data traffic. Suppose the per-node throughput for node i is given as x_i. The fairness index for the throughput can be defined as follows [PD00]:

$$f(x_1, x_2, \cdots, x_n) = \frac{(\sum_{i=1}^{n} x_i)^2}{n \sum_{i=1}^{n} x_i^2}$$

The fairness index $f(x_1, x_2, \cdots, x_n)$ is a random variable that takes a value between 0 and 1 (0 means the smallest fairness and 1 means the greatest fairness).

Energy consumption The energy consumption of a sensor node can be attributed to several factors (e.g. CPU activity, LED, sensor board access, EEPROM access, radio listen and radio transmission) [SHC$^+$04], but only the energy consumption due to the radio transmission varies depending on the settings. Based on this fact, we can compare the relative performance of the different algorithm configurations without taking much effort for setting up the oscilloscope or the current meter for each sensor node.

Assuming that all the sensor nodes in the testbed are operating at the same supply voltage V and sending packets for the same time period Δt, the energy consumed by a sensor node can be calculated as $IV\Delta t$. This assumption is based on the fact that the Mica2dot sensor nodes in Smote testbed are powered through wall-plugged Ethernet programming boards and the packet size (36 bytes) and the transmission rate (19.2 kbps) are the same for all the sensor nodes. The current draw (I) for the radio transmission of a Mica2dot sensor node can be determined by table look-up [Chi] and be reported on a status message over UART each time the sensor node sends a data message.

The energy consumption $IV\Delta t$ is for sending a message over one hop from the originating node. More meaningful data would be the energy consumption that includes both originating and forwarded messages. To measure the energy consumption caused by both originating and forwarded messages, we associate a counter value M with each sensor node and this counter value is incremented each time a sensor node sends or forwards a data message. The counter value M is copied to the status message and reset to 0 when the sensor node sends a status message. Then, E_k, the energy consumption of a node k since the last transmission of its originating message is

$$E_k = IV\Delta t \cdot M$$

Since V and Δt are assumed to be the same for all the sensor nodes, we can compare the results of different runs by reading the product of the current draw I and the counter value M, which we define as energy cost.

$$\text{Energy Cost} = I \cdot M$$

Neighbor distribution A radio-transmission-power-control algorithm changes the radio-transmission power to increase or to decrease the number of neighbors. Neighbor distribution will show how the algorithm adapts to an optimum state when it is given different initial parameters.

Per-hop statistics The routing tree on which we run experiments is dynamically built and each experiment run can have different routing trees. In order to understand the effects of this dynamics in a more detailed level, we measure the per-hop statistics like number of hops and throughput per hop. We calculate the number of hops from the base node by using the parent node ID in the status message and the routing tree implied by the (node, parent node) relations. And we calculate the throughput per hop by averaging the throughput of the sensor nodes that have the same number of hops.

Traffic Reduction for Aggregation Traffic As for aggregation traffic pattern, the total traffic that goes into the base node should be smaller than the total originating traffic because non-terminal nodes aggregate traffics from their child nodes. To verify this, we measure the traffic that goes into the base node and the originating traffic that is being delivered over the aggregate packets.

5 Experiment Results

5.1 Experiment Configurations

For the experiment, we measure the performance of the dynamic transmission-power-control algorithm comparing it with that of the fixed transmission-power-control algorithm. For each run of the experiment, all the sensor nodes except the base node originate data packets at the rate of 1 packet per 2 seconds, and each run lasts 20 minutes. We use 22 sensor nodes including the base node as shown in Figure 4(b).

For the fixed transmission-power-control algorithm, we try different radio-transmission-power values (PW_{init}):

– PW_{init}: 64, 128, 192, 255

For the dynamic transmission-power-control algorithm, we try different values for the number of target neighbors (N_{target}). As for the threshold value for the received-signal-strength ($RSSI_{threshold}$), we set it as 50.

– N_{target}: 3,6,9,12,15
– $RSSI_{threshold}$: 50

5.2 Convergence Traffic Results

Throughput and Energy Consumption For the convergence traffic, the dynamic transmission power control algorithm makes little difference to throughput while it saves the energy consumption over the fixed transmission power control algorithm. This is shown in Figure 5 which displays the average per-node throughput (left axis) and the average per-node energy cost (right axis). If we pick the points which maximize the throughput, the two algorithms have the maximum throughput as follows:

– Fixed: throughput 0.390 packets/sec, energy cost 38.58 at $PW = 255$
– Dynamic: throughput 0.409 packets/sec, energy cost 20.65 at $N_{target} = 12$

We can see that the dynamic transmission-power-control algorithm achieves 4.65% higher throughput and 86.8% less energy consumption than the fixed transmission-power-control algorithm when both algorithms achieve the maximum throughput. We can observe the similar results from some previous work [SKH04, KKW$^+$03].

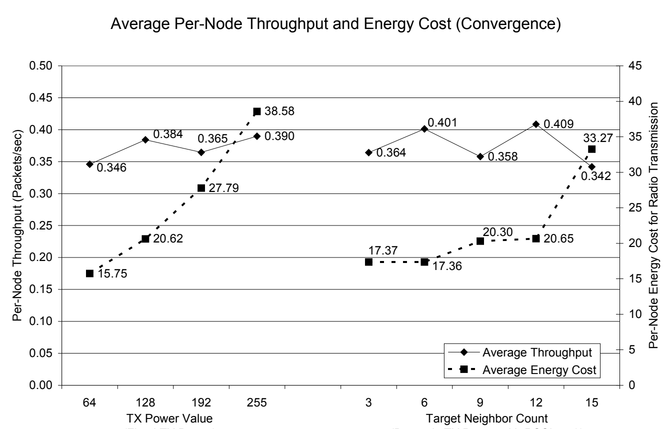

Fig. 5. Average Per-Node Throughput and Per-Node Energy Cost for Convergence Traffic

Table 2 shows the average per-node throughput and the fairness index for the fixed and dynamic transmission-power-control algorithms. We can see that the algorithms achieve best fairness index at the point where the algorithms have their maximum throughput.

Table 2. Average Per-Node Throughput and Fairness for Convergence Traffic

	PW(Fixed)				N_{target}(Dynamic)				
	64	128	192	255	3	6	9	12	15
Throughput	0.346	0.384	0.365	0.390	0.364	0.401	0.358	0.409	0.342
Fairness	0.903	0.938	0.934	0.941	0.897	0.943	0.911	0.970	0.900

Node Distribution and Per-Hop Statistics Figure 6 shows that the number of effective neighbors tends to increase as the number of target neighbors (N_{target}) increases. Due to radio characteristics and testbed topology, there is a limitation in controlling the number of neighbors. As we adjusted N_{target} between 3 and 15, we observed the number of neighbors in the range between 7.52 and 11.86.

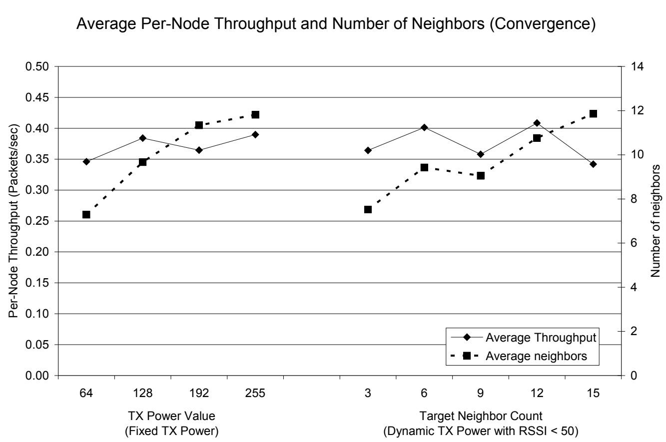

Fig. 6. Average Per-Node Throughput and Number of Neighbors (Convergence)

Table 3 shows trends of the average throughput and the per-hop throughput as we increase the number of target neighbors (N_{target}) with the dynamic transmission power control algorithm. We can see that the depth of the routing tree changes as we change N_{target} and it has the smallest value 2 when N_{target} is either 9 or 12. As for the relation between the average throughput and the per-hop throughput, we can find two things: First, the per-hop throughput becomes smaller as the number of hops increases. Second, communication contention becomes higher as we increase N_{target} and it has negative effects after a certain point. We can see that the throughput at hop 1 is about 0.41 when N_{target} is 3,6,9 or 12, but the throughput at hop decreases to 0.342 at $N_{target} = 15$.

Table 3. Throughput per Hop for Convergence Traffic

	N_{target}(Dynamic)				
	3	6	9	12	15
Average	0.364	0.401	0.358	0.409	0.342
Hop 1	0.416	0.427	0.414	0.417	0.359
Hop 2	0.351	0.379	0.296	0.393	0.362
Hop 3	0.197	0.396	-	-	0.167
Hop 4 or more	-	-	-	-	-

5.3 Aggregation Traffic Results

Throughput and Energy Consumption For the aggregation traffic, the dynamic transmission power control algorithm does not have the performance improvement compared to the fixed transmission power algorithm. Figure 7 shows the average per-node throughput and the average per-node energy cost for the fixed and dynamic transmission-power-control algorithms with aggregation traffics. If we pick the points that maximize the throughput, the two algorithms have the throughput and the energy cost as follows:

– Fixed: throughput 0.414 packets/sec, energy cost 22.98 at $PW = 128$
– Dynamic: throughput 0.421 packets/sec, energy cost 29.05 at $N_{target} = 12$

This result translates to small increase in throughput with 26.4% more energy consumption.

Comparison of Convergence and Aggregation Traffic The dynamic transmission-power-control algorithm achieves higher throughput and higher fairness with the aggregation traffic than it is applied to the convergence traffic (Table 4).

– Convergence Traffic
  • Throughput: max 0.409 packets/sec, min 0.342 packets/sec
  • Fairness index: max 0.970, min 0.897
– Aggregation Traffic
  • Throughput: max 0.421 packets/sec, min 0.393 packets/sec
  • Fairness index: max 0.975, min 0.966

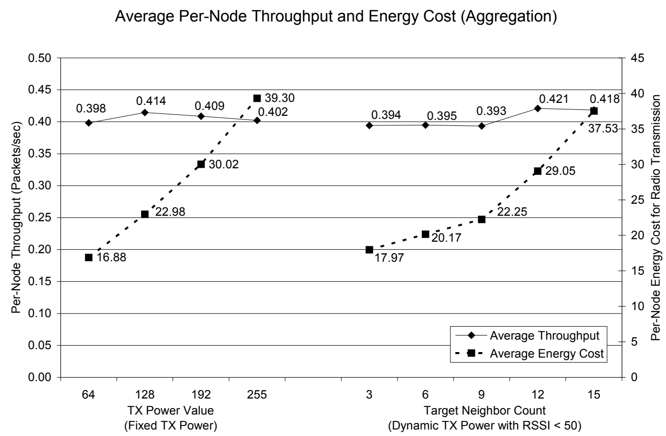

Fig. 7. Average Per-Node Throughput and Energy Cost (Aggregation)

Table 4. Throughput and Fairness for Aggregation and Convergence Traffics

	N_{target} (Aggregation)					N_{target} (Convergence)				
	3	6	9	12	15	3	6	9	12	15
Throughput	0.394	0.395	0.393	0.421	0.418	0.364	0.401	0.358	0.409	0.342
Fairness	0.966	0.968	0.969	0.974	0.975	0.897	0.943	0.911	0.970	0.900

Table 5. Per-hop Throughput for Aggregation and Convergence Traffics

	N_{target} (Aggregation)					N_{target} (Convergence)				
	3	6	9	12	15	3	6	9	12	15
Average	0.394	0.395	0.393	0.421	0.418	0.364	0.401	0.358	0.409	0.342
Hop 1	0.433	0.434	0.398	0.450	0.445	0.416	0.427	0.414	0.417	0.359
Hop 2	0.361	0.380	0.388	0.391	0.402	0.351	0.379	0.296	0.393	0.362
Hop 3	0.343	0.253	-	0.428	-	0.197	0.396	-		0.167
Hop 4 or more	-	-	-	-	-	-	-	-	-	-

Figure 8 shows traffic reduction by using aggregation traffic patterns. For the aggregation traffic pattern, the traffic that goes into the base node has decreased to 40.5% to 57.5% of the originating traffic. We can observe the effects of the traffic reduction from the per-hop throughput data (Table 5). For the case of convergence traffic, increasing the number of target neighbors (N_{target}) increases the contention and puts negative effects on the performance after a certain point ($N_{target} = 15$)). For the case of aggregation traffic, however, increasing the number of target neighbors didn't hurt the performance.

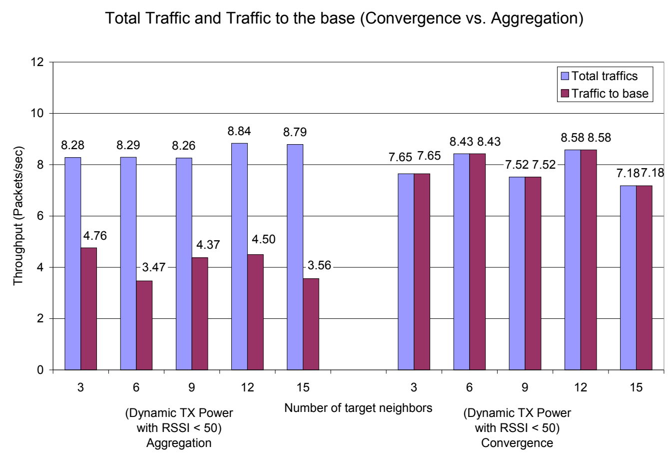

Fig. 8. Aggregate Traffic to the base node

6 Conclusion

In the literature, several dynamic transmission power control algorithms have been proposed in order to improve the throughput and the energy consumption of mobile wireless networks. In the context of wireless networks, however, little empirical data is known that supports the previous approaches. The previous approaches are based on either simulation data or the hardware platform without the resource constraints of wireless sensor networks. In this paper, we have evaluated the performance of a dynamic transmission power control algorithm using Mica2dot-based Smote testbed and compared it with the fixed transmission power control algorithm. From the experiments, we have found the followings: As for the convergence traffic, we have found that the dynamic transmission power control algorithm makes little improvement whlie it saves the energy consumption compared to the fixed transmission power control algorithm. As for the aggregation traffic pattern, the dynamic transmission-power-control algorithm does not have the same performance improvement compared to the fixed transmission-power-control algorithm. We have also found that aggregation traffic achieves higher throughput than the convergence traffic by reducing the data traffic that travels along the routing tree.

References

[Chi] Chipcon. Cc1000 single chip very low power rf transceiver data sheet. *http://www.chipcon.com/files/CC1000_Data_Sheet_2_2.pdf*, page 32.

[EKCD00] Tamer A. ElBatt, Srikanth V. Krishnamurthy, Dennis Connors, and Son Dao. Power management for throughput enhancement in wireless ad-hoc networks. *IEEE ICC*, 2000.

[ESWW00] Jean-Pierre Ebert, Bjorn Stremmel, Eckhardt Wiederhold, and Adam Wolisz. An energy-efficient power control approach for wlans. *Journal of Communications and Networks (JCN)*, September 2000.

[JV02] Eun-Sun Jung and Nitin H. Vaidya. A power control mac protocol for ad hoc networks. *ACM MOBICOM*, September 2002.

[KKW+03] Martin Kubisch, Holger Karl, Adam Wolisz, Lizhi Charlie Zhong, and Jan Rabaey. Distributed algorithms for transmission power control in wireless sensor networks. *IEEE WCNS*, 2003.

[MBH01] Jeffrey P. Monks, Vaduvur Bharghanvan, and Wen-Mei W. Hwu. A power controlled multiple access for wireless packet networks. *IEEE INFOCOM*, 2001.

[PD00] Larry L. Peterson and Bruce S. Davie. *Computer networks: a systems approach*, page 456. Morgan Kaufmann Publishers Inc., 2000.

[PHC04] Joseph Polastre, Jason Hill, and David Culler. Versatile low power media access for wireless sensor networks. *ACM Sensys*, November 2004.

[RRH00] Ram Ramanathan and Regina Rosales-Hain. Topology control of multihop wireless networks using transmit power adjustment. *IEEE INFOCOM*, 2000.

[SHC+04] Victor Shnayder, Mark Hempstead, Borrong Chen, Geoff Werner Allen, and Matt Welsh. Simulating the power consumption of large-scale sensor network applications. *ACM SenSys*, November 2004.

[SKH04] Dongjin Son, Bhaskar Krishnamachari, and John Heidemann. Experimental study of the effects of transmission power control and blacklisting in wireless sensor networks. *IEEE SECON*, October 2004.

[WLBW01] Roger Wattenhofer, Li Li, Paramvir Bahl, and Yi-Min Wang. Distributed topology control for power efficient operation in multihop wireless ad hoc networks. *IEEE INFOCOM*, 2001.

[WTS03] Shu-Lin Wu, Yu-Chee Tseng, and Jang-Ping Sheu. Intelligent medium access for mobile ad hoc networks with busy tones and power control. *IEEE Journal on Selected Areas on Communications*, September 2003.