

Accepted Manuscript

Impact of Transmission Power Control in multi-hop networks

R. Kotian, G. Exarchakos, S. Stavros, A. Liotta

PII: S0167-739X(16)30391-0

DOI: <http://dx.doi.org/10.1016/j.future.2016.10.010>

Reference: FUTURE 3182

To appear in: *Future Generation Computer Systems*

Received date: 12 December 2015

Revised date: 31 May 2016

Accepted date: 6 October 2016



Please cite this article as: R. Kotian, G. Exarchakos, S. Stavros, A. Liotta, Impact of Transmission Power Control in multi-hop networks, *Future Generation Computer Systems* (2016), <http://dx.doi.org/10.1016/j.future.2016.10.010>

This is a PDF file of an unedited manuscript that has been accepted for publication. As a service to our customers we are providing this early version of the manuscript. The manuscript will undergo copyediting, typesetting, and review of the resulting proof before it is published in its final form. Please note that during the production process errors may be discovered which could affect the content, and all legal disclaimers that apply to the journal pertain.

- The impact of a single node performing Transmission Power Control (TPC) frequently for a short period of time on a relatively dense network is evaluated.
- A prominent node performing TPC severely impacts the network more than the non-prominent nodes.
- Design of TPC must take data rate, traffic flow, node deployment and the working of routing and MAC protocol into consideration.

Impact of Transmission Power Control in Multi-hop Networks

Authors: R. Kotian¹, G. Exarchakos¹, S. Stavros², A. Liotta¹

¹Technical University Eindhoven (TU/e), Department of Electrical Engineering,
5600 MB, Flux Building 19, Eindhoven, The Netherlands

²Telecommunication Systems Research Lab (TSRL), Open University of Cyprus, Cyprus
{r.kotian,g.exarchakos,a.liotta}@tue.nl stavros.stavrou@ouc.ac.cy

Abstract

Many Transmission Power Control (TPC) algorithms have been proposed in the past, yet the conditions under which they are evaluated do not always reflect typical Internet-of-Things (IoT) scenarios. IoT networks consist of several source nodes transmitting data simultaneously, possibly along multiple hops. Link failures are highly frequent, causing the TPC algorithm to kick-in quite often. To this end, in this paper we study the impact that frequent TPC actions have across different layers. Our study shows how one node's decision to scale its transmission power can affect the performance of both routing and MAC layers of multiple other nodes in the network, generating cascading packet retransmissions and forcing far too many nodes to consume more energy. We find that crucial objectives of TPC such as conserving energy and increasing network capacity are severely undermined in multi-hop networks.

Keywords

Internet-of-Things; Transmission Power Control; Routing Protocol; MAC Protocol; Energy efficiency

1. Introduction

The fast-deployment of the Internet of Things (IoT) enables an unprecedented set of new applications, in contexts as varied and challenging as the smart-city, environmental monitoring, smart agriculture, e-Health [1] and smart energy (to mention but a few) [2], [3], [4]. The IoT network as a whole is projected to scale to millions of nodes within the next few years [5]. And, although not all these nodes will have to simultaneously communicate among each other, there will typically be hundreds of nodes within relevant multi-hop cluster networks. Thus, it is crucial to look at how existing communication protocols will cope in such high-density communication scenarios in typical wireless sensor network applications such as in monitoring, analysis and sharing of physical parameters [6], [7].

Major impediments to the successful operation of this complex interconnected system are (a) the limited battery capacity of the individual nodes, and (b) the ever-changing radio propagation medium. The uncertainty in the channel conditions are due to the interference from other sources such as Wi-Fi routers, microwave ovens [8]. In addition, natural obstacles such as office walls decays the signal strength [9]. Furthermore, environmental conditions such as humidity and temperature in

the indoor and outdoor locations significantly weaken the communication link reliability [10][11][12].

To enhance the link reliability between the pair of sensor nodes, boosting the transmission power level seems to be a natural choice [13]. However, the transceiver of several sensor nodes such as TelosB [14], MicaZ [15], etc. consumes more energy than other units such as ROM, sensors, CPU of the nodes [16]. Therefore, minimizing the power and simultaneously strengthening the communication link are conflicting goals [17].

A solution to this issue is to employ adaptive TPC techniques that scale the transmission power up or down at run-time whenever the link quality falls below or above a predefined threshold, respectively. As already pointed out, the radio propagation medium varies significantly and abruptly over time due to unavoidable environmental disturbances such as human activity, fading of the signals etc. [13]. This reinforces the need of TPC by the nodes to maintain a desired level of reliability. Therefore, the main objective of TPC is to achieve optimal transceiver power – a power level that does not break the already established link between a pair of nodes nor increases the contention in the network. However, several decisions about transmission or routing of a packet from MAC and routing protocols rely on link quality metrics which are directly influenced by TPC algorithms. We expect that TPC causes more damages than benefit the network.

The main objective of this experimental study is to identify the response of a MAC as well as that of routing protocols to transmission power scaling in a low power wireless networks (IEEE802.15.4). Two routing protocols are tested representing two different route discovery approaches:

- Route rediscovery if and only if route unknown. That is, the routing table are not updated based on link quality degradation but rather when a route to a destination is not known. This represents a more static situation and we use ContikiMesh as representative of this category.
- Continuous route optimization. Routing table is continuously updated based on link quality changes. That presumes that the links are continuously monitored and route are rebuilt whenever necessary. CTP is a representative of this category.

Both ContikiMesh and CTP are running over ContikiMAC. To tackle these research questions, the experiments were carried out in a popular simulation environment known as COOJA [18] and we extensively study and examine:

1. The impact of one node in transmission power scaling mode switching power levels frequently for a short period of time on a relatively dense and sizeable static network consisting of 10 nodes. The network performance is tested in conjunction with ContikiMAC protocol and two different routing protocols – ContikiMesh and CTP separately.
2. The overall network performance in terms of Packet Delivery Ratio (PDR), Latency, Energy Consumption as well as the benefits for a node in transmission power scaling mode.

The remainder of the paper is organized as follows. In section 2, we highlight the shortcomings of the analysis under which popular TPC algorithms are evaluated. Section 3 briefly describes two different routing protocols – ContikiMesh, CTP – and the ContikiMAC protocol. Section 4 provides

the information about the simulation parameters and experimental setup. Section 5 discusses and summarizes the results. Lastly, in section 7 we conclude the paper.

2. Related Work

In this section, we discuss the general shortcomings in the evaluation of the TPC. We discuss about the issues that can impact the working of TPC. We also explain how TPC can impact the normal operation of MAC and routing layers. We then highlight the drawbacks of some of the known TPC algorithms provided in Table 2. We then attempt to differentiate our work from the analysis done by other authors.

2.1 General Limitations

Several TPC algorithms shown in Table 2 have been proposed in the literature. However, majority of them are tested in a single-hop network. Furthermore, these algorithms running in nodes are tested in a round-robin fashion where only one node transmits certain amount of packets at certain power level and rest of the nodes are listening. Finally, inference is drawn on per link basis.

Large realistic IoT network does not operate in round-robin fashion, low contention environment. A real-world network rather has many multiple source nodes transmitting data simultaneously possibly along several hops. Imagine an IoT network deployed in vineyard that monitors humidity, temperature and transmits the data along multi-hop to the aggregation unit [19]. Large geographical area leaves the resource constraint nodes with limited transmission range no choice but to multi-hop its data. Thus, avoiding routing protocol is not possible.

When there is a dense network, without a MAC protocol performing CSMA/CA, there is going to be a lot of packet drops due to collision. It is already known fact that node in the listening or idle state still consumes more energy [20]. Therefore, MAC protocol with integrated sleep/wakeup capability reduces the listening/idle time of the node.

To have a large distributed and yet efficient IoT network, one has to exploit the benefits provided by the routing and MAC layers. On the contrary, the induction of TPC in dense multi-hop network can flip the merits of MAC and routing layer into demerits. Usually TPC resides between the routing and MAC layer [21]. TPC's impact on other layers or other layers impact on the performance of TPC has to be thoroughly examined in a realistic scenario (multi-hop network with multiple source nodes). Therefore, TPC algorithms cannot be evaluated in isolation.

2.2 Implication of TPC on MAC layer

MAC protocols help a pair of nodes to sync their communication. This aid the transmitting node to transmit data at the time interval when recipient is in the listen mode. The transmitting node transmits to its recipient based on the communication pattern of its neighbour. Under fixed T_x , the MAC's communication sync strategy works well in avoiding collision and mitigates the hidden terminal problem, but is known to aggravate exposed terminal issue [11]. In a dynamic environment as the link quality between any pair of nodes falls, a boost in T_x is triggered by TPC algorithm running in either of the nodes to compensate the error. This raise in T_x disrupts the communication sync of other neighbouring pair of nodes. As a result, other nodes in the vicinity may retransmit with high T_x several times than it normally does before it is successfully received by its recipient. This may further increase the contention. Hence, the network exhibits a vicious behaviour and one can see

degradation in the overall performance as discussed in [11]. Therefore, design of TPC algorithms must take the functioning of MAC protocols into account. An overview of how generic TPC algorithm may turn the good feature of MAC protocol – ContkiMAC against the network is presented in section 3.2.

2.3 Implication of TPC on Routing layer

Normally, the user of the IoT network determines the reliability and the reduction in the energy cost to be achieved. Typical TPC translates this to optimum T_x level by correlating RSSI and/or LQI to the PDR. Unfortunately, these correlation do not yield good result due to the sensitivity of RSSI and LQI [13]. This kind of translation without taking the topology of the network into consideration might result in the node not converging to optimum T_x level. As a result, node frequently might use a power level that is either small or large. A wrong usage of the power level would force the routing protocol to choose the inefficient path – longest path, less energy efficient path [22], high interference path [23] or high Expected Transmission Path (ETX) [24].

Many of the routing protocols generate control messages for the maintenance or fault tolerance purposes. For example, the performance of various routing protocol such as RPL, AODV, CTP and DSR in terms of PDR, latency, number of control messages, power consumption and fault tolerance in the fire emergency scenario is discussed in [25]. The performance of the test case is evaluated for static and fixed transmission power. Table 1 provides the partial results of that test-case.

Similarly, the performance of routing protocols such as Optimized Link State Routing (OLSR) and Dynamic Manet On-Demand (DYMO) in realistic urban test-case is discussed[26]. The introduction of TPC in both test-cases can increase the contention and may raise the value of the metrics (latency, control messages, battery consumption) which is not desired.

Table 1: Performance metrics of various routing protocol derived from[25]

Metrics	CTP	AODV
Delivery Ratio (%)	96.98	100
Average Latency (secs)	12.21	9.30
No. of Control messages	1508	3163
Power consumption (% battery Left)	58.09	53.32
Fault Tolerance (secs)	199	78

A node that scales down the power may cause its already established link to break. Scaling up the power by the node may cause the same problem but in the neighbouring nodes due to contention. Depending on how adaptive the routing protocols are, this link breakage will be identified. To repair the link, routing algorithm will spit out extra control messages causing higher contention and energy consumption. An overview of how generic TPC algorithm may turn the good feature of MAC protocol – ContkiMAC against the network is presented in section 3.2.

2.4 Limitations of Popular TPC

Keeping the implications of TPC on other layers (routing and MAC) and other layers/issue (Application layer and deployment) implication on TPC in mind, we present the shortcomings of TPC proposed in the literature. Due to limitation of space we consider only few algorithms.

Table 1: Design choice considered in various TPC algorithms

TPC Algorithms	MAC Protocol	Routing Protocol	Multi-hop Network	Multiple Nodes sending data simultaneously	Data Rate	Node Deployment	Data Traffic Pattern
ART[13]	TinyOS Default CSMA/CA	CTP	yes	yes	200 packets in 30 mins	10 nodes placed randomly. Multiple source nodes and 1 sink	Convergence
DTPC[27]	B-MAC	MintRoute	yes	No information	1 packet every 2 secs	22 nodes placed uniformly. Many source nodes and 1 sink	Convergence and Aggregation
RPAR[28]	B-MAC	RPAR	yes	yes	1 packet every 300 ms	130 nodes placed randomly. Multiple source nodes and 1 sink	Aggregation
MPC[29]	TinyOS Default CSMA/CA	No	No	Yes	1 packet every 3 secs	6 nodes placed randomly. Multiple source nodes and 1 sink	Convergence
ATPC[30]	TinyOS Default CSMA/CA	No information	yes	No	15 packets per secs	43 nodes placed randomly. Multiple source nodes and 1 sink	Convergence
P-TPC[31]	TinyOS Default CSMA/CA	No	No	No	20 packets per mins	24 nodes placed randomly. Single source nodes and multiple receiver	Point-to-point
ODTPC[32]	B-MAC	AODV	yes	No	1 packet every 5 secs	7 nodes placed randomly. One source node and 1 sink	Aggregation

From the Table 2, one can notice that only 3 (ART, RPAR and MPC) out of 7 TPC algorithms were tested in a multi-hop network with multiple source nodes sending data simultaneously. MPC and ART algorithms however are tested in a network with convergence traffic. This means that the source nodes are within the transmission range of the sink forming a kind of simple star topology. A real-world network would normally have a sink that is beyond the transmission range of majority of nodes thereby forcing the traffic to be aggregated at a certain point –sink.

Furthermore, a TPC in a network with aggregated network traffic has no noticeable performance improvements over fixed transmission power scheme as shown in [27]. Therefore, it is necessary to study extensively the effect of generic TPC model in a multi-hop network with different routing protocols and aggregated network traffic.

3. Background

In this section, we briefly explain the inner working of two routing protocols namely ContikiMesh and CTP and a MAC protocol known as ContikiMAC.

3.1 Transmission power control

The decision to scale the power either way by TPC algorithms is obtained by the link quality metrics such as Link Quality Indicator (LQI), Received Signal Strength Indicator (RSSI). These metrics are provided by the transceivers such as CC2420 of TelosB motes. RSSI are available in all the incoming packets and can be easily extracted. A typical approach to transmission power scaling by TPC algorithms is as follows. The receiving node calculates the RSSI from the incoming packets of the source node and then analyses if it is within the threshold. As RSSI value can be correlated to the transmission power, a fall below the lower bound or above the higher bound of threshold immediately triggers the receiving node (R_n) in sending back an ACK packet to the source node (S_n). This ACK packet contains the recommended transmit power (T_x) for that specific S_n node that it must use for transmitting future data packets. This generic technique as seen in AODTPC [33] ensures that the fading link quality is quickly rectified by raising or lowering the transmit power to compensate the variation. Thus, AODTPC claims to achieve higher Packet Delivery Ratio (PDR).

Other generic methods, found in P-TPC [31], ART [13], calculate the PDR or the number of failed transmissions (determined by not receiving ACK for the packet sent) for a given window w . If either one falls below or above a respective threshold, appropriate T_x level is selected for the future transmission of packets.

Although the metrics such as LQI, RSSI are inexpensive, easy to collect and can reflect the channel condition quickly, they are sensitive to environmental disturbances and the probability of it deviating from the predefined threshold level remains high[34]. This causes the node to change the T_x quite often and might fail to converge sooner to appropriate power level. Algorithms such as P-TPC, ART on the other hand do not rely on sensitive RSSI, LQI metrics. However, this makes them less responsive than its counterparts, forcing them to retain a higher or lower power level longer than needed.

Both of these generic approaches do not take into account the interference they might produce by frequent fluctuation of the power in a dense network consisting of multiple source nodes. Interference results in collisions and ultimately impacts the latency and PDR. Moreover, most of the

TPC algorithms are not tested in a multi-hop scenario. The situation of collisions is magnified when the underlying routing protocols, e.g. Collection Tree Protocol (CTP) [35], are designed to periodically transmit control messages to keep the network connected. Medium Access Control (MAC) protocols, such as ContikiMAC [36], are designed to retransmit the packets (control packets from routing protocol, data packets such as temperature or humidity from the application layer) until it receives the ACK from the receiving nodes, thus, increasing the energy consumption.

3.2 MAC Layer Protocol: ContikiMAC

The reason for not changing the MAC protocol as we do with routing protocols is because ContikiMAC is known to outperform its popular predecessors such as B-MAC and variants of X-MAC (X-MAC-C, X-MAC-CP, and X-MAC-P) in terms of latency, retransmission, energy consumption and PDR[37].

The main features of this protocol is as follows[36]:

- It is an asynchronous sender initiated radio duty cycle protocol. Asynchronous meaning there is no common wake-up schedule established between a pair of nodes before the communication between them starts.
- Node running ContikiMAC continuously sends the entire data frame until ACK is received from the recipient. This is unlike X-MAC where strobes are used by the sender and only after receiving corresponding strobe-ACK, the sender transmits the entire data frame.
- ContikiMAC does not have fixed wake-up and sleep schedule. Instead it has an adaptive scheme where the node performs two successive Clear Channel Assessments (CCA) to determine if there is an incoming data based on the RSSI. If the CCA finds the channel is clear, the nodes go to sleep. Else, the concerned node stays awake and executes a fast sleep optimization method. This method determines if the RSSI is due to noise or because of incoming data. If former is the case the node goes to sleep.

When the node incorporates TPC, the second feature of ContikiMAC can cause turbulence in the network. Imagine, a node continuously transmitting data for which it does not receive any ACK from the concerned receiver node (for e.g. due to channel busy on its side). A TPC based on the generic model as described in section 1 will increase its T_x level causing contention in its vicinity. As neighbouring nodes finds that the channel is busy, it might defer its communication. This might increase the latency. Worse, the exposed terminal problem becomes more prominent resulting in lower network capacity. As the retransmission increases the channel utilization the third feature of the ContikiMAC keeps the node in wake-up mode for slightly higher period of time at the expense of more energy consumption.

3.3 Routing Layer Protocol: ContikiMesh and CTP

For this experiment we have chosen two routing protocols-ContikiMesh and CTP. The objective is to test the impact of real-time TPC on two variants of routing protocols - dynamic (CTP) and less dynamic (ContikiMesh) in terms finding and repairing broken links. ContikiMesh and CTP are known to generate least and highest amount of control packets respectively. Hence, both of these protocols provide the opportunity to study impact of TPC on the whole network under varying routing overheads. Furthermore, CTP is a widely used protocol for static stationary network [38][39].

ContikiMesh on the other represents a multi-hop network deployed in open environment with little or no interference and traffic movement [40].

ContikiMesh is a lightweight protocol provided by the ContikiOS. Following are the important features of ContikiMesh[41]

- It uses two modules namely route-discovery and multi-hop to find the potential neighbours and multi-hop the data to specified receiver residing somewhere in the network.
- Once the route-discovery phase is completed there is no periodic transmission of control messages.
- It does not use any link quality estimation technique to cope with dynamic environment.

CTP is a distance vector protocol that is capable of computing any-cast routes to a single or small group of sink in a network. The three most important features of CTP are as follows [35]

- Wireless links are unstable and exhibit bursty behaviour over short time period. This suggests that the accuracy of link quality estimation can be high if it is agile. For this purpose, it uses information from the three layers namely physical, data link and network layers.
- It incorporates data-path validation scheme that reliably detects the path from the source to destination. The main task of this scheme is to avoid looping condition causing network congestion. For this, CTP uses probe packets to quickly detect the problem when the packets do not make progress towards the destination.
- Typical routing protocols transmit control messages at a fixed time interval. CTP however uses adaptive beaconing mechanism. When the topology is inconsistent, CTP transmits the control message faster and decreases it significantly when the network is stable.

Because of the third feature of CTP, TPC can cause more retransmission. Based on the default functioning of ContikiMesh and CTP in ContikiOS 2.7, we classify them as adaptive and non-adaptive protocols respectively. The adaptiveness of the protocols is tested in a scenario described in Figure 1.

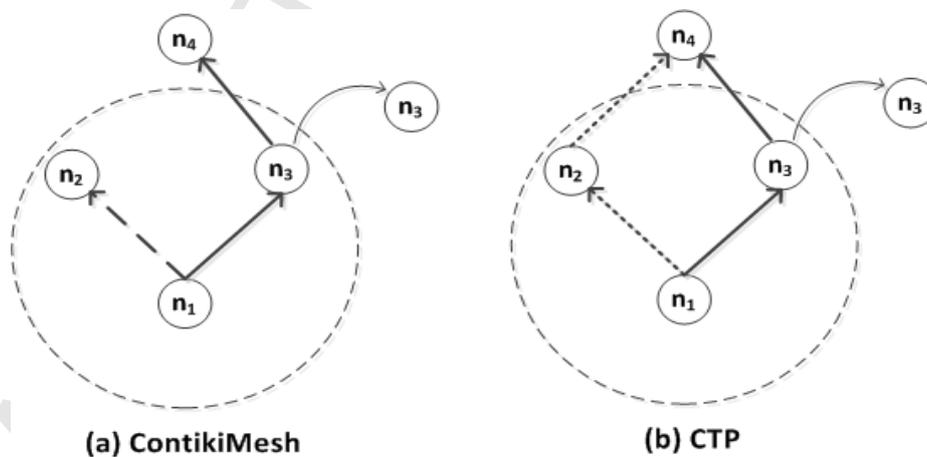


Figure 1: Routing strategies by ContikiMesh and CTP

In Figure 1 (a) and (b), node n_1 is the source node; n_2 and n_3 are the relay nodes. n_4 is the sink node. n_1 would either select n_2 or n_3 as a relay node to transmit its data to the sink. For explanation

purpose, let's assume n_1 opt n_3 (shown by thick arrow line) as its relay to send the data. To simulate the link breakage between n_1 and n_3 , we move n_3 out of the transmission range (dashed circle) of n_1 (shown by thin arrow line). The link breakage is detected at n_1 and it redirects its traffic to n_2 (shown by dashed arrow line). However, n_2 does not forward the data of n_1 to n_4 . By placing n_3 back to its original position, the sink n_4 continues to receive the data of n_1 through n_3 . However, in the case of CTP for the same network topology the removal of node n_3 is detected at n_1 and it forwards the data to the sink via n_2 in few minutes by sending additional control messages.

Although ContikiMesh can save energy as there are fewer control messages, a link break can severely reduce the PDR of n_1 and also other leaf nodes (if present) that uses n_1 as the relay node.

CTP on the other hand is more dynamic and has potential to detect routing problems. However, the disturbance in the network caused by TPC can force it to send higher control packets which consume more energy. In addition, under the flood of control messages the probability of data packets not reaching its destination is higher.

4. Experimental and Simulation Setup

In the following experiments, the goal was to study the impact of transmission power scaling on routing protocols. Therefore, we used a network emulation tool (Cooja Simulator [18]) to eliminate factors, other than power scaling, from twisting the results.

4.1 Constraints and Requirements

While designing a TPC algorithm, importance should also be given to the applicate data rate. Consider a case where TPC algorithm requires considerable amount of historical link quality data to decide the future T_x to be employed. If the data rate is fast, the duration of time pervious T_x is low and the adaptation to variation is fast. Opposite is the case when the data rate is slow. The importance of performing TPC adaptation quickly is discussed in [42], [43]. Having a high data rate does not necessarily translate into increase in the throughput. This is even truer in a network with real-time TPC.

The way the nodes are deployed can also affect the performance of TPC. For example, in a dense network where the nodes are randomly deployed, a node cannot use a single global transmission power to reach its neighbours. Hence, all the nodes must perform prolonged initialization phase to determine the optimum T_x level on a per link basis. On the other hand, in a network of uniformly deployed nodes, nodes may employ less intensive initialization phase thereby saving energy. Furthermore, in a dense network with randomly distributed nodes the power level variations may be extreme. This can cause more contention than in a network with uniformly distributed nodes.

The placement of the sink also plays a crucial role. It is experimentally proven that merits of TPC is noticeable only in a network with convergence data traffic and not in a network where the traffic flow is aggregated[27].

4.2 Network Topology

The experiments were carried out on a small sized homogenous spatially dispersed network as shown in the Figure 2. Each experiment lasted for 15 minutes. There are in total 10 nodes, one sink and nine independent sensor data source nodes. The longest path from data sources to the sink

depends on the transmission power per node; for our experiments, it varies from 5 to 6 hops. The description of various network flows is in [44]. The squares in the grid of Figure 2 have edges 10m long. The average density of the network stands at around 65% and is calculated by using the equation 1

$$\sum_{i=1}^n \frac{D_i N}{100} \quad (1)$$

Here, D_i is the total number of neighbouring nodes for a given node with a specific transmission range and N is the total number of nodes in the network. The link between any pair of nodes is considered to be asymmetric.

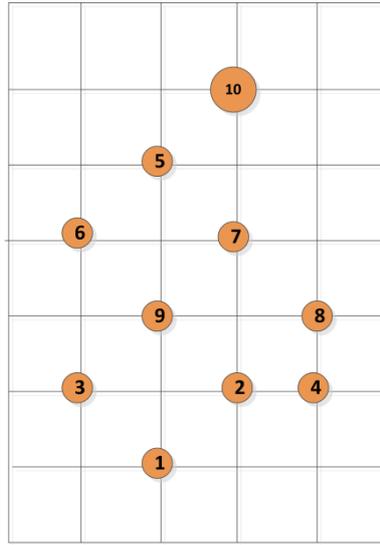


Figure 2: Topology of the Network

To increase the confidence of our results and more reliably correlate transmission range with routing protocols performance, we placed the nodes at equal distances between each other. This ensures that the degradation of the network, if any, with respect to the reliability, latency, energy and packet loss is not due to the random placement of the nodes but the node transmission power scaling. The network size was kept limited for two reasons: (a) the routing paths are long enough to simulate the impact of transmission power scaling of a node to the remaining nodes of the path, and (b) the logged data were also stayed manageable making the traceability of network performance patterns more effective.

4.3 Transmission Power Scaling

The default transmission power of all the nodes is set to $T_x=15$. This corresponds to a transmission range of 22 meters, output of -7dBm and energy consumption of 12.5mA as given by data sheet of CC2420 transceiver[16]. Transmission power upscaling boosts the power level to $T_x=19$. This translates to increase in the transmission range by 29 meters, output of -5dBm and energy consumption of 13.9mA [16]. A lower power level 11 with transmission range of 16 meters could have been used. However, this would place the nodes at the very edge of the radio range increasing the packet drop rate. This would erroneously affect the study.

Nodes 2, 7, 8 and 9 that have various interference coverages were chosen to perform the TPC. The interference area is divided into 1-hop region and 2-hops region. For example, in the Figure 2, when the node 2 employs power level 15, the neighbouring nodes (1, 3, 4, 7, 8 and 9) that are within 22 meters region are said to be in 1-hop interference region and all the nodes beyond 22 meters range are said to be 2-hops interference region. Table 3 provides the details of the interference coverages of various nodes.

Table 2: Interference coverage of the nodes with two different transmission power levels

		Node 2	Node 7	Node 8	Node 9
Default $T_x=15$	1-hop interference region (%)	67	78	45	89
	2-hops interference region (%)	33	22	54	11
Increased $T_x=19$	1-hop interference region (%)	78	89	67	89
	2-hops interference region (%)	22	11	33	11

Figure 3 illustrates a sample transmission power scaling pattern. Data source nodes send sensor data to the sink randomly in 2 to 4 seconds time interval. That is, for every 10 seconds, these nodes generate 5-6 data packets. At the end of every 10 seconds periods, the nodes 2, 7, 8 and 9 randomly decide to upscale (15 to 19), downscale (19 to 15) or maintain their transmission power. No specific TPC technique, e.g. Iterative method [45] or ATPC [30], is used. However, the focus is on the effect of transmission power scaling action; a study over the effect of random such actions is sufficient since, to a good extent, TPC based on the link quality metrics (Received Signal Strength Indicator and/or Link Quality Indicator) is mimicked.

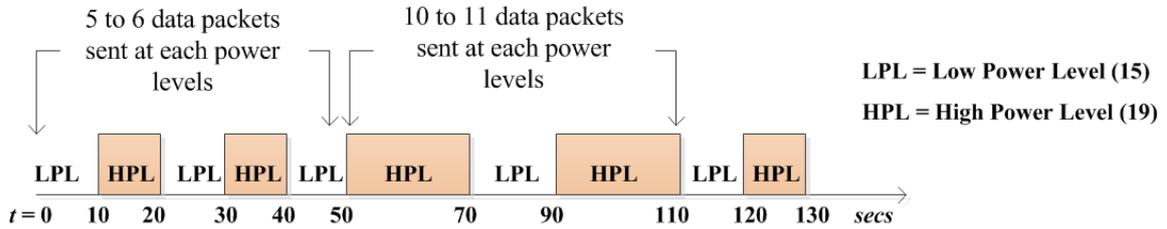
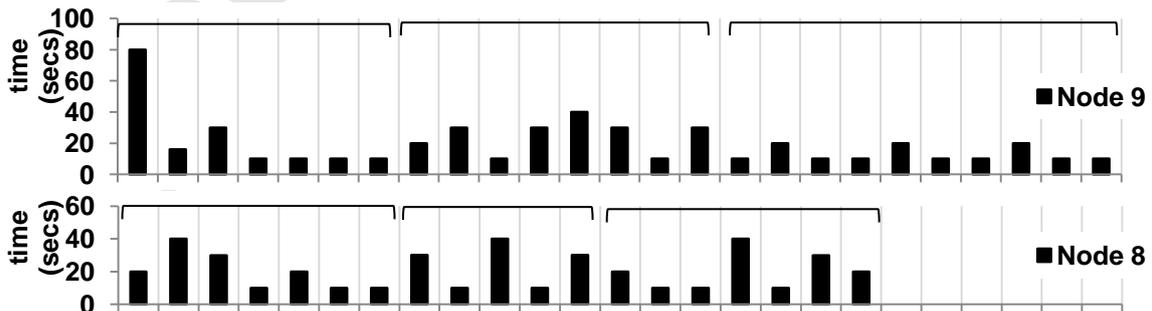


Figure 3: Sample transmission power scaling pattern

Due to randomness, the 4 nodes that were scaling their transmission power had different patterns. Figure 4 illustrates the duration a node maintained its high transmission power once it up scales for the ContikiMesh experiments. The upscaling events (x axis) are ordered based on moment of occurrence. The duration of downscaling is not shown. On the one hand, node 9 seems to have the most upscaling events with a long high power start-up. On the other hand, Node 2 has few upscaling events of short duration except for one outlier 130sec long in the middle of that sequence.



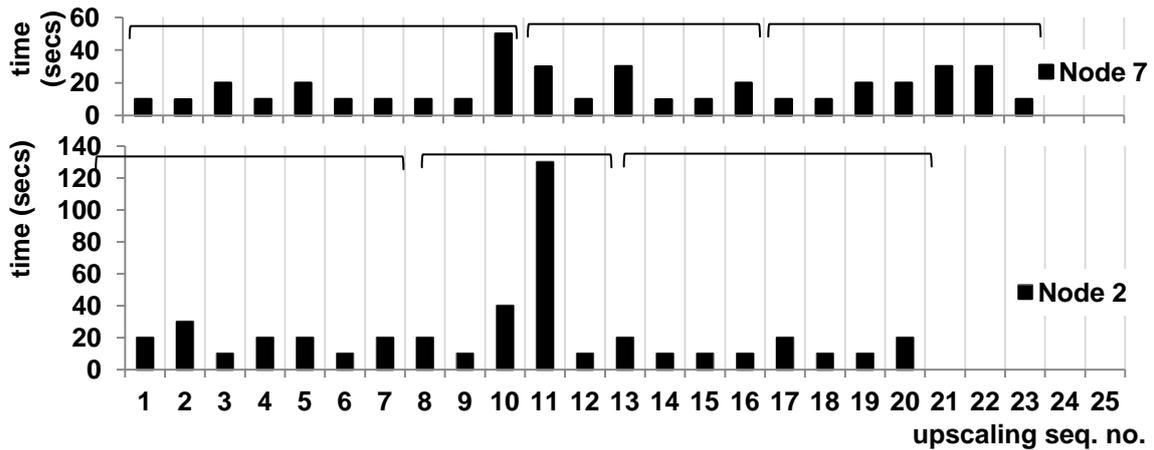


Figure 4: Duration of high transmission power after each upscaling event for ContikiMesh experiments.

Splitting the experimentation time of 15 minutes into three intervals of 5 minutes each, we generated Table 4. The table illustrates the amount of upscaling events and the duration of high transmission power for each of the 4 nodes. These intervals are also marked in Figure 4.

Table 3: Upscaling events and duration of high transmission power per 5 min interval of ContikiMesh experiments

	Interval	Node 2	Node 7	Node 8	Node 9
No. of times TPC was high ($T_x=19$)	0-5 mins	8	10	7	7
	5-10 mins	5	6	5	8
	10-15 mins	8	7	7	10
Duration of high transmission power (mins)	0-5 mins	2.33	1.99	2.33	2.66
	5-10 mins	3.49	1.82	1.99	3.16
	10-15 mins	1.82	2.16	2.33	2.16

The second set of experiments refers to Collection Tree routing protocol (CTP). The configuration is the same except for those parameters affected by randomness. Hence, the transmission power scaling pattern is shown in Figure 5 and Table 5.

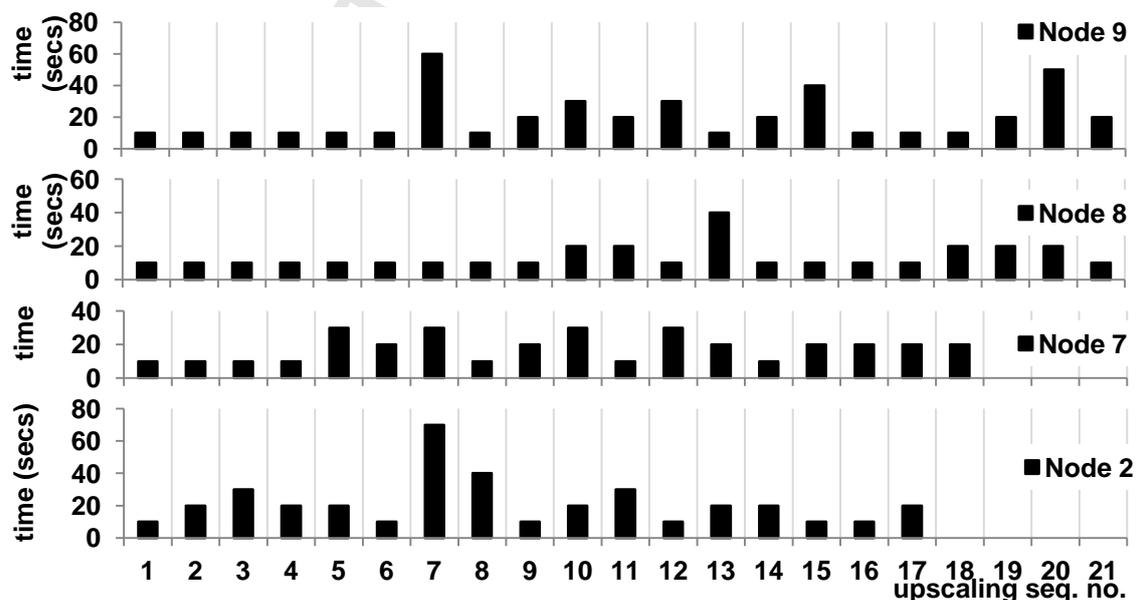


Figure 5: Duration of high transmission power after each upscaling event for CTP experiments.

Table 4: Upscaling events and duration of high transmission power per 5 min interval of CTP experiments.

	Interval	Node 2	Node 7	Node 8	Node 9
No. of times TPC was high ($T_x=19$)	0-5 mins	3	5	5	6
	5-10 mins	7	6	7	8
	10-15 mins	7	7	9	7
Duration of high transmission power (mins)	0-5 mins	1	1.16	0.83	0.9
	5-10 mins	3.16	1.99	1.49	3.33
	10-15 mins	1.99	2	2.49	2.66

Figure 5 and Table 4 reveal a less dynamic transmission power scaling situation for CTP experiments compared to ContikiMesh ones. That is, fewer power level changes and of shorter duration per level.

4.4 Emulation features

Cooja was configured at 100% simulation speed. The radio messages of the network were propagated based on Unit Disk Graph Model (UDGM) and were captured using inbuilt tool of Cooja. Based on UDGM, the strength of the signal fades with the distance between source and destination nodes emulating link failures [46]. Below the chosen routing protocols, ContikiMesh and Collection Tree Protocol (CTP), the MAC layer was ContikiMAC.

All the nodes in the network have a start-up delay of 1000ms and emulate the TelosB sensor motes equipped with CC2420 transceiver [14]. To check the channel condition, all the nodes perform Clear Channel Assessment (CCA) at the MAC layer [36]. The battery depletion of the nodes have significant impact on the listening and transmission aspect of the nodes [47]. Therefore, battery depletion of the nodes is not simulated and nodes experience no power outage. Hence, we make sure that no disturbance is caused by factors other than transmission power scaling. Likewise, the receiver sensitivity of all nodes is unchanged and is set to the default levels of -90dBm.

4.5 Benchmark and assessment metrics

Besides experimenting with the two routing protocols and different nodes (2, 7, 8, and 9) in power scaling mode, we have created a benchmark of no power scaling over the same network. That is, two sets of 5 experiments per routing protocol, 10 experiments in total. The network topology and other configurations (except for random seeds and randomized parameters) are identical for both sets. The results of those two sets are separately analysed and finally compared together.

As explained above, transmission power scaling has a direct effect on the internal interference of the whole system. Indirectly, packet retransmissions, queue lengths and energy consumption are influenced by the interference in the system. The question of this study is whether the adaptivity capabilities of routing protocols worsen or ease the situation. We have picked the following assessment metrics:

- *Number of packets* generated to assess how increased interference from the transmission power scaling node may cause collisions and trigger more retransmissions and or routing control packets.
- *End-to-end packet delivery ratio (PDR)* as a way to assess buffer overflows due to excessive amount of interference that forces packets stay in the queue for long and new ones to be

dropped. This is the number of packets received at the sink over the total number of packets sent.

- *End-to-end latency* as assessment of the queue lengths caused either by high interference and many retransmissions or by path rediscovery when paths are destroyed due to power scaling. We measure the average latency as the average time difference of a packet from its first transmission trial from the source node until the reception from the sink.
- *Radio duty cycle* to assess whether the energy savings from TPC outperform the energy costs introduced by increased signalling. It is the percentage of time the transceiver was on for the entire duration of the experiment.

5. Results and Discussion

The results provided are broadly classified into two main sub sections – impact of transmission power scaling on ContikiMesh, and on CTP. In both these sub sections, the analysis focuses on three main aspects: (a) impact of a single node’s transmission power scaling to retransmissions of packets from nodes, (b) the impact to network level performance metrics, and (c) the benefits, if any, for the node in power scaling mode.

For clarity and brevity, the labels NO-TPC, 2-TPC, 7-TPC, 8-TPC and 9-TPC in the following histograms denote the results of the experiments as detailed in section 5.4. The benchmark (no node in transmission power scaling mode) experiment is tagged as NO-TPC. The tag x-TPC maps to the experiment during which node x is in transmission power scaling mode.

5.1 Transmission Power Scaling over ContikiMesh

This section presents the traffic generated due to transmission power scaling in presence of ContikiMesh. The analysis of packets generated across the network includes the on-board sensors data, control packets generated by the routing protocols, and the ACK packets produced by the 802.15.4 protocol upon reception of the packets. Figure 6 illustrates the total and retransmitted number of packets generated in the network (a) as well as a breakdown of the total number of packets into 5min segments. The first segment of every experiment demonstrates a spike in the number of packets generated in the network compared to the remaining segments. The reason can be traced back to the route discovery phase of the ContikiMesh routing algorithm executed at the bootstrap of the network.

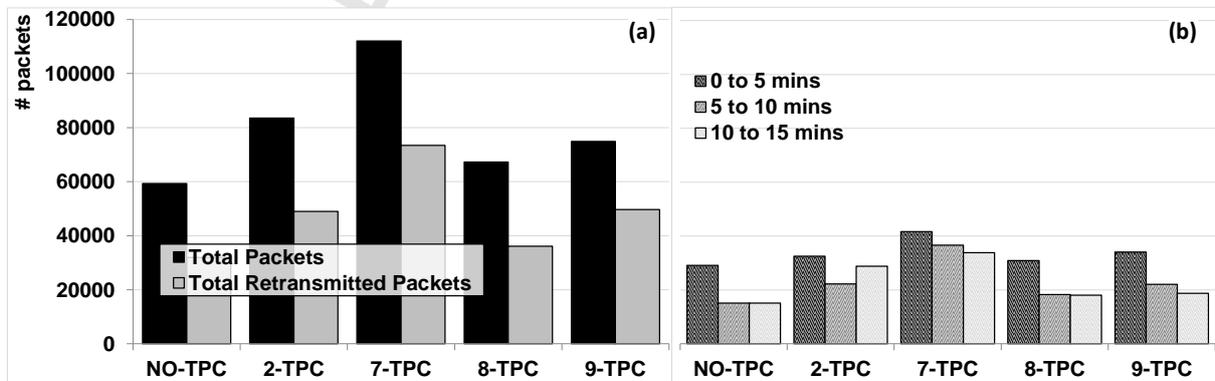


Figure 6: Packets transmitted during the 5 ContikiMesh experiments (NO-TPC & x-TPC). (a) Total number of packets and total number of retransmitted packets, (b) Breakdown of total number of packets transmitted into 5min segments

Compared to NO-TPC, the total number of packets generated in the network increases by 29%, 47.1%, 11.8% and 20.8% when nodes 2, 7, 8 and 9 perform TPC respectively. Similarly, transmission power scaling influences the amount of retransmitted packets as well. In fact, the total number of duplicates packet raises by 34.7%, 56.4%, 11.5% and 35.6% respectively; mostly higher than the increase to the total amount of packets.

When a specific node x up scales its power level (19), the neighbouring nodes recognize the channel usage and defer the transmission of packets in their buffer. This results in fast buffer capacity consumption from own sensor data packets and relayed data packets from other nodes, too. All the deferred transmissions are retried as soon as the node down scales its power to the default level (15). That momentarily increases collisions and nodes retransmit until ACK is received, since in NO-TPC experiment, all the network features such as transmission power levels, data transfer rate, network topology etc. remain unchanged.

Amongst the experiments with transmission power scaling, the network of 8-TPC experiences the least amount of packets in the network; 7-TPC produces the highest followed by 2-TPC and 9-TPC. Duration of each power level, frequency with which the levels interchange and the interference produced by the transmission power are responsible for the variation in the amount of packets generated. Finally, retransmission trials increase with the betweenness centrality of the node in power scaling mode. Figure 7 and Table 6 illustrate the influence of betweenness centrality of nodes in transmission power scaling mode on number of packets transmitted.

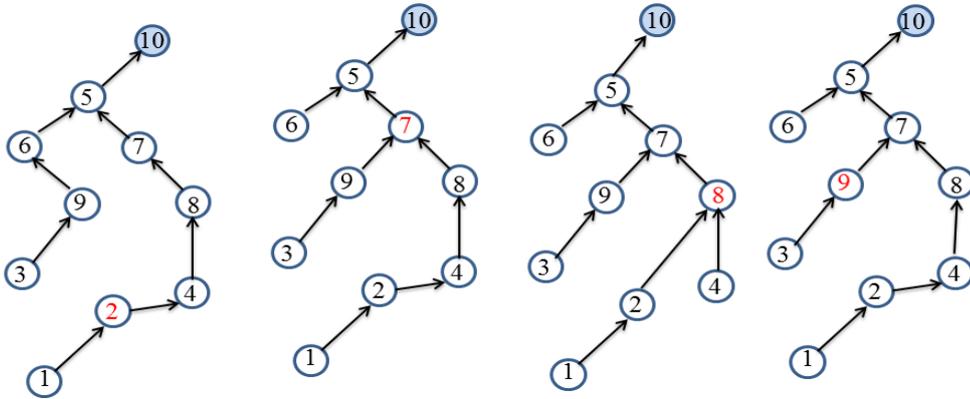


Figure 7: Routing paths built with ContikiMesh when nodes 2, 7, 8, 9 are in transmission power scaling mode.

Table 5: Influence of betweenness centrality of nodes in transmission power scaling mode on packets generated

	2-TPC	7-TPC	8-TPC	9-TPC
Relays the traffic of nodes	1	1, 2, 3, 4, 9, 8	1, 2, 4	1, 2, 3
Total packets sent compared to NO-TPC	+29%	+47.1%	+11.8%	+20.8%
Total control packets sent compared to NO-TPC	+34.7%	+56.4%	+11.5%	+35.6%

Although nodes 8 and 9 relay traffic from the same amount of nodes (3), the interference produced by node 8 in two different power levels (15 and 19) is the least compared to all other nodes (see Table 3). Although, as shown in Table 4, the transmission power level of node 8 changes more frequently the duration per level is shorter than that of node 9. Therefore, neighbourhood size

increases with the transmission power upscaling directly affecting the number of collisions and retransmissions.

As of 7-TPC versus 9-TPC, the former experiences shorter duration per transmission power level and lower number of level interchanges than the latter. The interference produced during power level 19 is the same to 9-TPC (see Table 3). However, in 7-TPC, node 7 relays double the traffic that node 9 relays in 9-TPC. This result suggests that collisions and retransmissions increase with the betweenness centrality of a node in transmission power scaling mode even if the power upscaling is short and infrequent.

2-TPC comes to strengthen the conclusion that long duration at high transmission power levels creates an explosion of deferred packet (re-)transmissions once the power is downscaled to low levels. As Figure 6 depicts, 2-TPC experiences 9% extra total number of packets even with lower interference region, duration per power level, number of level interchanges and amount of relayed traffic. 2-TPC generates more data packets in the last time segment (28785) compared to 9-TPC (19740), an increase of 34%. This is because, during the previous segment (5-10mins), node 2 remains in high power level (19) for 130 seconds. This forces nodes in the vicinity to defer for long before they are able to resend the packets and to cause an explosion of (re-)transmissions once the power level is downscaled.

In terms of the types of packets transmitted, we split Figure 6(b) into control signalling and data packets illustrated in Figure 8 (a) and (b) respectively. Figure 8(a) confirms that the increased packet transmission in the beginning of every experiment is due to ContikiMesh route discovery signalling. Note that Figure 8 includes retransmissions.

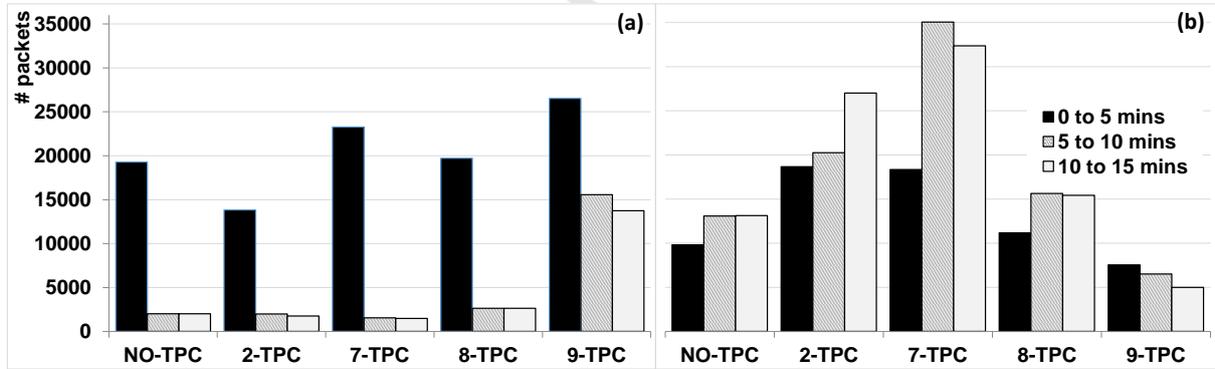


Figure 8: Total number of packets per packet type (control and data) per experiment and per time segment of 5 mins for ContikiMesh experiments. (a) Control packets, (b) Data packets.

9-TPC, as opposed to the other experiments, experiences more control packets than data packets even though data packets were generated at the same rate as any other experiment. Contrary to the other experiments, the node in transmission power scaling mode (node 9) booted with high transmission power level and stayed in that level for 80sec. Therefore, the route discovery mechanism built the routing tables based on that power level. When later the node down scaled the power level, some of the routes had to be rediscovered. Moreover, this experiment has the most frequent power level interchanges among all. This deteriorated the situation as the route discovery was not allowed to converge and stabilize. The frequent power level changes also triggered multiple explosions of retransmissions from other nodes adding delay to the route discovery mechanisms or

even loss of route discovery packets and unsuccessful route build-up. This resulted not only to high number of control packets but also to undiscovered routes and, hence, fewer packet transmissions.

End-to-end network performance metrics: With regards to network performance metrics (end-to-end PDR & latency and duty cycle), the picture is mixed as shown in Figure 9. NO-TPC experiment demonstrates that constant $T_x=15$ for all the nodes is a more reliable option (higher PDR), more energy efficient (lower duty cycle) and average in end-to-end latency.

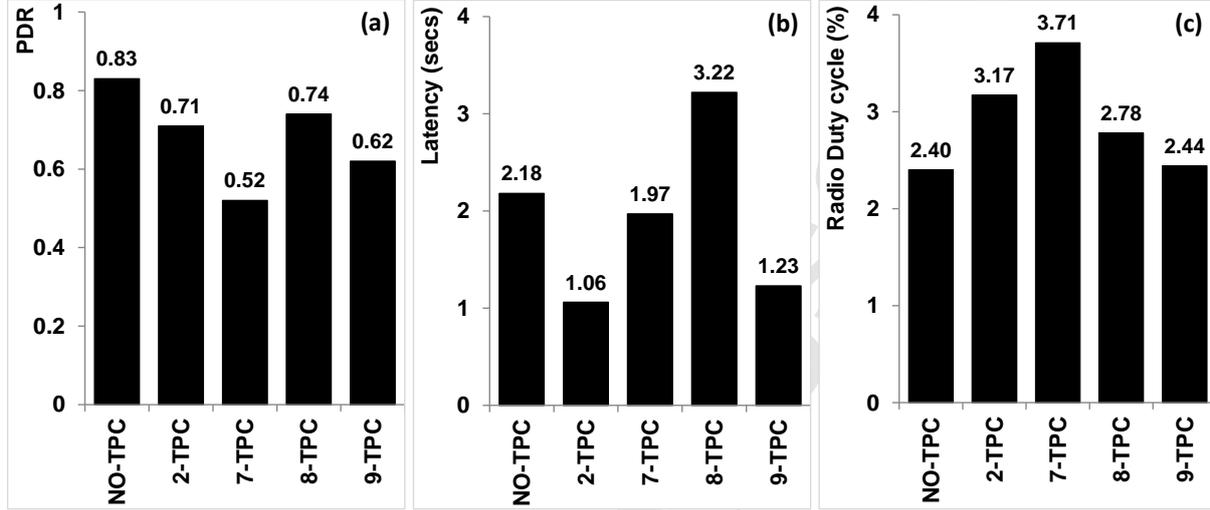


Figure 9: Average network performance metrics for ContikiMesh experiments. (a) End-to-end packet delivery ratio, (b) end-to-end latency, (c) radio duty cycle

As mentioned above, four important features of transmission power scaling vary the degree of the impact on the performance of the network. Of these four features, duration per transmission power level, neighbourhood size per level and interference region and the prominence betweenness centrality of the node in transmission power scaling mode have major impact on the performance. This is noticeable in Figure 9(a). Although node 7 in 7-TPC experiment exhibits the smallest duration per level, it relays the traffic of most of the nodes. Node 9 in 9-TPC relays less traffic but the interference produced during high transmission power level is the highest among all the experiments. Nodes 2 in 2-TPC and 8 in 8-TPC relay less traffic and have smaller interference neighbourhood. Hence PDR is higher than the other two experiments.

As shown in the Figure 9(b), in three out of four With-TPC scenarios, the latency is lesser compared to No-TPC. Small latency time is desirable property to have. However, recall that we calculate latency only for the number of packets that successfully arrive at the sink. As the environment is stable in No-TPC case, the amount of packets that reach the sink is high. This is visible in the PDR value. Higher interference region due to high TPC causes collision. Therefore, the amount of packets that reaches the sink is less. If we observe the PDR in the Figure 11 when node 8 performs TPC, we find that it is higher than any other With-TPC scenarios. This means that amount of packet reaching the sink is high and therefore the latency is naturally high as depicted in the Figure 12.

Figure 9(c), combined with Table 3; indicate that the duty cycle increases with the interference neighbourhood. As collisions increase, packets are retransmitted and ore energy is consumed. Since NO-TPC experiment experiences fewer collisions, the radio duty cycle of the network is 20.5% lower compared to all other experiments.

Selfish benefits of transmission power control: We have seen in the previous section how a single node that increases its transmission power to selfishly enhance its own one-hop link quality ends up adversely affecting the performance of the entire network. Here, we study what is the impact of this decision by a specific TPC node on its own overall performance. As shown in Figure 10(a), the average end-to-end PDR based on packets generated by nodes in transmission power scaling mode between all x-TPC experiments is 0.53. However, the average end-to-end PDR of the same nodes in NO-TPC experiment is 0.76, 30% higher. Similarly, as shown in Figure 10(b), the average end-to-end latency for the given nodes in x-TPC experiments is 2.05 secs and by 30% lower (1.27 secs) in NO-TPC experiment. The average duty cycle, Figure 10(c), is 7.52% and 2.8% in x-TPC and NO-TPC experiments respectively.

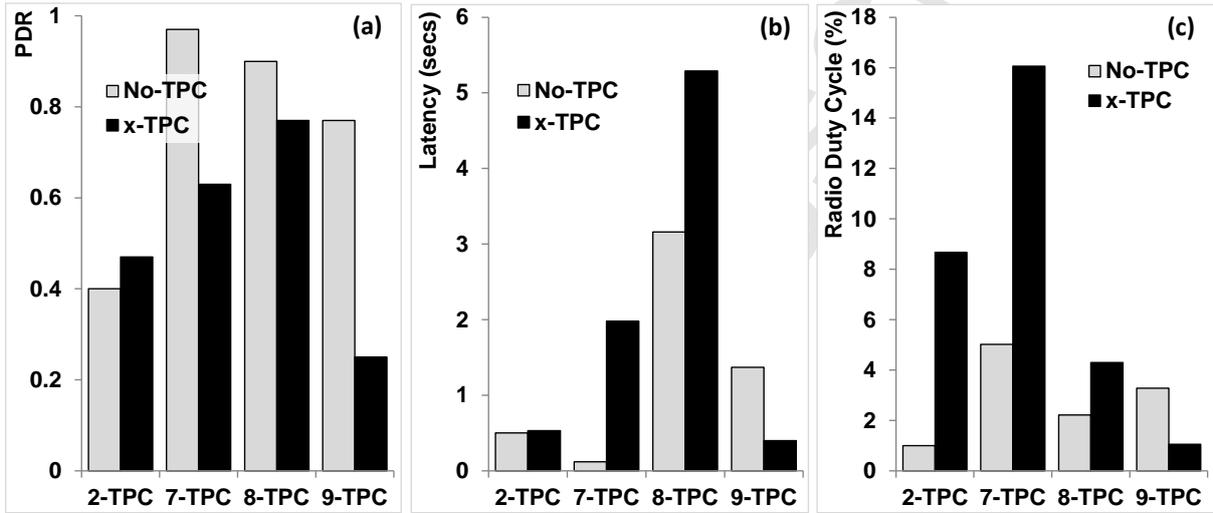


Figure 10: Average node performance metrics for ContikiMesh experiments. (a) End-to-end packet delivery ratio, (b) end-to-end latency, (c) radio duty cycle

The experiments and results above give no evidence that transmission power control provides benefits in the network or the node in transmission power scaling mode when ContikiMAC and ContikiMesh is used.

5.2 Transmission power scaling over CTP

CTP is a dynamic routing protocol trying eagerly to re-establish more optimal routing paths taking into account the link quality between neighbouring nodes. Figure 11 provides the routing paths in the network as established during the experiments. The dashed lines show the possible different paths a packet from a node can take depending on the quality of each link. The solid line shows the route from the respective nodes even in the presence of nodes 2, 7, 8 and 9. This comes with the cost of extra signalling, control packets to maintain a good path. Transmission power control generates two opposing forces to the performance of a network. From one hand, link quality may be improved and on the other hand more interference due to extra signalling might deteriorate the performance. We follow the same analysis strategy as with ContikiMesh.

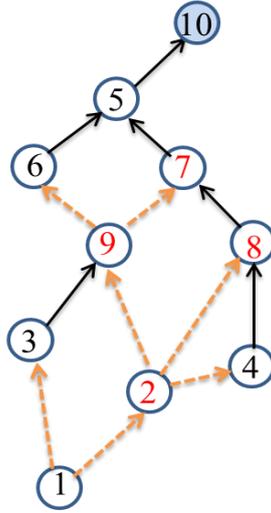


Figure 11: CTP routing path of the network when nodes (2, 7, 8 or 9) up/down-scale their transmission power

Traffic generated due to transmission power scaling. The total number of packets generated by CTP experiments, Figure 12(a), shows a clear difference from the ContikiMesh experiments. The total and retransmitted number of packets is higher. Especially, the ratio retransmitted/total number of packets is almost twice as big as in the ContikiMesh experiments. As pointed out from Figure 5 and Table 5, the transmission power scaling is less dynamic in CTP experiments, yet the number of generated packets higher. Moreover, from Figure 12(b), it is also clear that the route discovery is not taking place only at bootstrap time since the first time 5min segment experiences the fewest packets compared to the other two segments.

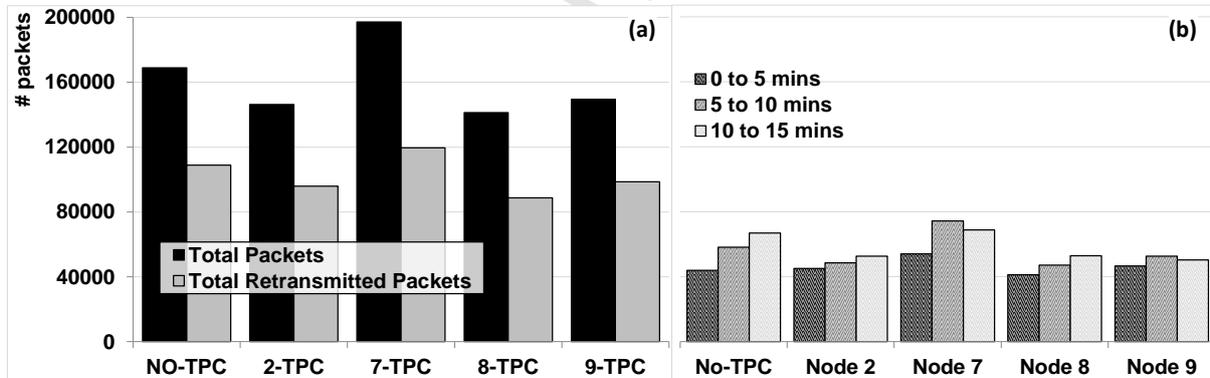


Figure 12: Packets transmitted during the 5 CTP experiments (NO-TPC & x-TPC). (a) Total number of packets and total number of retransmitted packets, (b) Breakdown of total number of packets transmitted into 5min segments

Based on Table 7, transmission power scaling seems to have a positive effect on the number of packets for some experiments. There is a decrease in the total number of packets sent during 2-TPC, 8-TPC and 9-TPC experiments. 7-TPC experiences the least power level changes, yet, the highest increase in total number of packets. As shown in Table 3 and Table 7, in 7-TPC experiment, node 7 relays the traffic from the majority of nodes and the interference produced by it is the second highest compared to all other nodes in transmission power scaling mode. Although node 9 in 9-TPC has fewer relay nodes, the interference produced by it during TPC is the highest, resulting in higher packet transmission compared to NO-TPC. The opposite is the case with 8-TPC, less interference fewer retransmissions. As with ContikiMesh experiments, betweenness centrality and interference

neighbourhood size of the node in transmission power scaling mode seem to play an important role in the number of packets generated. However, the results of both sets of experiments indicate that betweenness centrality has a stronger effect than interference.

Unlike ContikiMesh with its fixed beaconing procedure, CTP detects prevailing conditions in the radio propagation medium and employs adaptive beaconing of control messages. CTP mitigates the problem of self-interference by limiting transmission rate; i.e. the expected time of the packet to be sent is p , then CTP delays the packet transmission in the range of $1.5p$ to $2.5p$. This adaptive behaviour allows transmission power scaling generate benefits for the network in low self-interference situations i.e. 2-TPC and 8-TPC.

Table 6: Influence of betweenness centrality of nodes in transmission power scaling mode on packets generated

	2-TPC	7-TPC	8-TPC	9-TPC
Relays the traffic of nodes	1	1, 2, 3, 4, 9, 8	1, 2, 4	1, 2, 3
Total packets sent compared to NO-TPC	-13.39%	+34.73%	-28.31%	+5.78%
Total control packets sent compared to NO-TPC	-16.08%	+54.15%	-32.20%	-3.41%

As depicted in Figure 11(b), there is a hike in packets in the first five minutes due to route discovery which largely constitute control packets. Figure 13(a) illustrates the number of control packets transmitted per 5-min segments. With the exception of 2-TPC, the first 5min segments indeed experience more control packets compared to the following two segments. The situation is reversed with the data packets, as shown in Figure 13(b). As shown in Figure 5 and Table 5, node 2 in 2-TPC (5-10min segment) experiences a prolonged high transmission power level. This creates reliable links to nodes further away from node 2. It reduces the ETX to nodes at a longer distance and, hence, rewiring and route re-discovery is triggered so that shorter paths are built. This process generates more control packets but shortens the paths and reduces retransmissions; thus, fewer data packets.

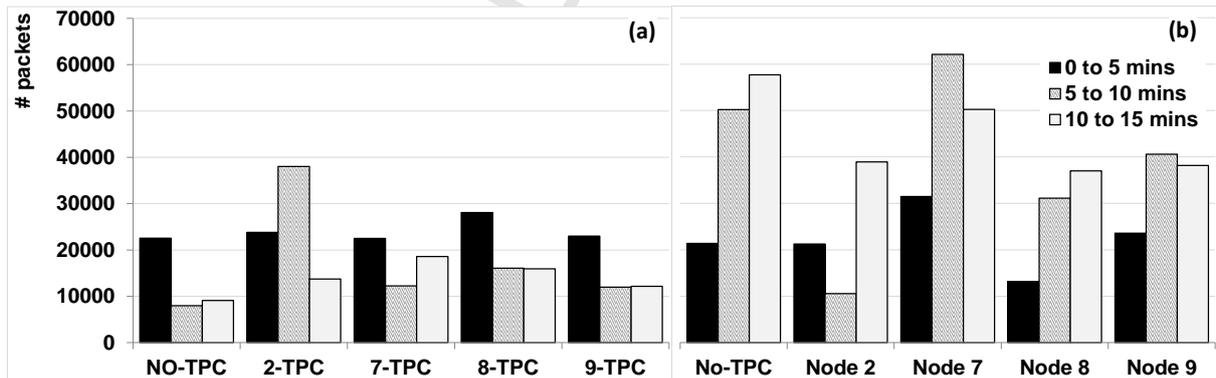


Figure 13: Total number of packets per packet type (control and data) per experiment and per time segment of 5 mins for CTP experiments. (a) Control packets, (b) Data packets.

End-to-end network performance metrics. The adaptive beaconing of CTP cooperates with transmission power scaling and yields improvements with regards to network performance compared NO-TPC situation. Figure 14(a), 14(b) and 14(c) illustrate the average end-to-end packet delivery ratio, average end-to-end latency and the average radio duty cycle, respectively. PDR has improved (+16.6%) in x-TPC experiments, the latency is reduced (-18.6%) and the radio duty cycle is mostly reduced (-8%, except for 7-TPC).

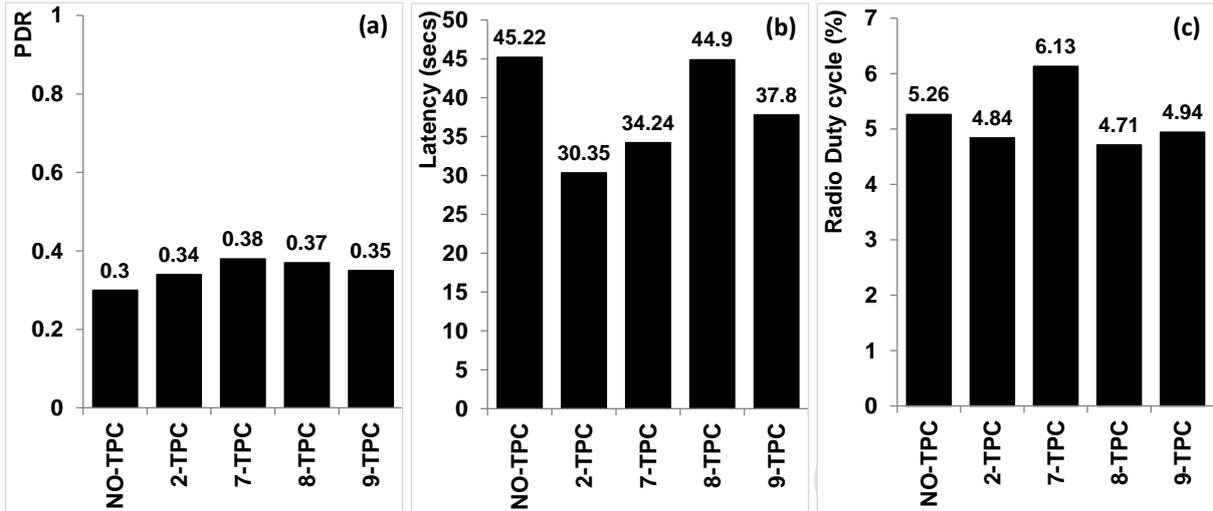


Figure 14: Average network performance metrics for CTP experiments. (a) End-to-end packet delivery ratio, (b) end-to-end latency, (c) radio duty cycle.

Selfish benefits of transmission power control. While transmission power scaling has a positive effect on the overall performance of the network, Figure 15 illustrates the marginal benefits (PDR, latency and duty cycle) it has on the node in transmission power scaling mode. It is important here to note that transmission power scaling was a random process and did not simulate careful decisions to upscale or downscale transmission power based on some measurable metric.

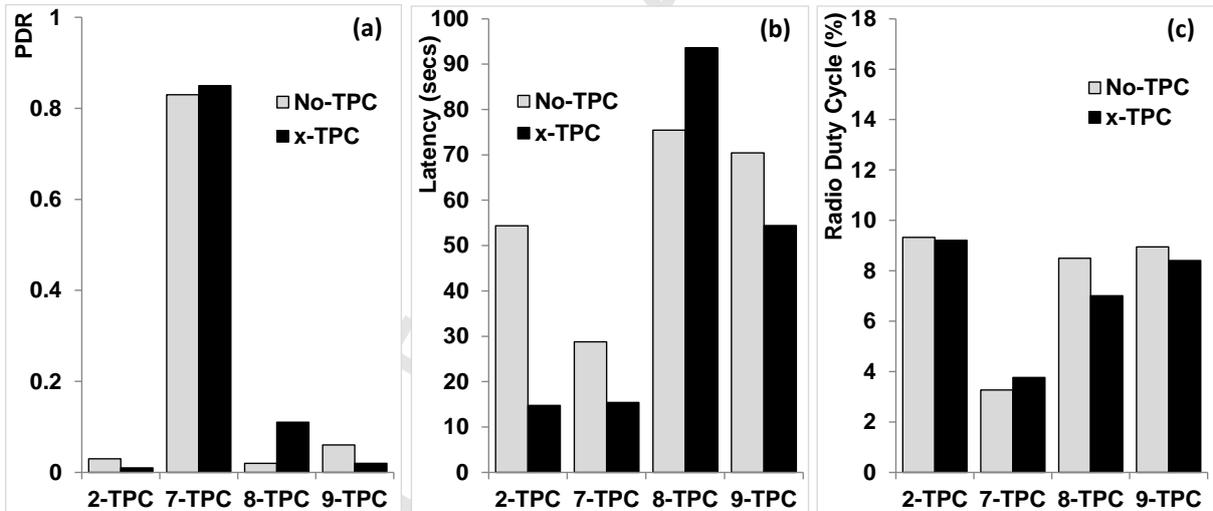


Figure 15: Average node performance metrics for CTP experiments. (a) End-to-end packet delivery ratio, (b) end-to-end latency, (c) radio duty cycle.

5.3 Transmission Power impact on ContikiMesh and CTP

Between the two sets of experiments (ContikiMesh versus CTP), transmission power scaling seems more compatible with the latter. The periodicity of the control packets in CTP prevents the protocol from overreacting to transmission power level and link quality changes. In fact, CTP was designed to precisely handle these situations.

ContikiMesh triggers route re-discovery once it is too late and the route to a destination is lost. Downscaling the transmission power may make routes built during high power level undiscoverable. Hence, the nodes wait until either the routes are rebuilt or power level re-upscaled. This waiting

period forces nodes, especially those with high betweenness centrality, generate a burst of packets once the route is available again. This burst increases temporarily the internal interference which results to CCA attempts and retransmissions. The phenomenon is worse when the scaling is frequent because even route discovery processes might be split between different power levels. That results to partial invalid routes. Therefore, transmission power scaling under ContikiMesh generate no benefit for either the network or the node in scaling mode.

On the other hand, the results on CTP demonstrate slight and under certain conditions improvements in the total number of packets, PDR, latency and duty cycling for both network and node performance. The experiments suggest that the transmission power should (a) be lower as the betweenness centrality increases, (b) change as rarely as possible, and (c) scale up and down iff the routing protocol is dynamic (not on-demand route discovery) and periodically update the routes.

For completeness, Table 9 provides the standard deviation of metrics for all the experiments. The fluctuation in metrics across all the experiments indicates that the impact of TPC is prominent in static multi-hop network.

Table 9: Standard deviation of metrics for two routing protocols

Metrics	NO-TPC		x-TPC							
	Contiki		ContikiMesh				CTP			
	Mesh	CTP	2-TPC	7-TPC	8-TPC	9-TPC	2-TPC	7-TPC	8-TPC	9-TPC
PDR	0.3	0.41	0.32	0.28	0.28	0.43	0.47	0.41	0.43	0.46
Latency (secs)	1.95	21.58	1.34	2.07	2.73	0.91	27.80	29.11	38.08	27.39
Radio Duty Cycle (%)	1.16	3.19	2.24	4.40	1.39	1.05	2.69	1.72	2.6	2.70

The design of TPC is strongly influenced by the traffic flow, data rate, the nature of node deployment, the routing and MAC protocols. Failure to study these influences can lead to a TPC which can function only in a specific scenario making its usage limited. Lastly, we do not recommend reactive TPC algorithm to be designed based on the RSSI or LQI. The reason for this is that these parameters are sensitive to environmental changes and therefore can trigger TPC frequently causing the network performance to drop. Moreover, RSSI and/or LQI varies from location to location. Hence, TPC based on these metrics are not location agnostic requiring considerable amount of fine-tuning of the threshold level before the network becomes operational. In addition, a reactive TPC algorithm can often keep the transmission power at high level until the next batch of link quality metrics are obtained. From our experiments we found that higher the duration of high transmission power, higher is the chance of collision.

6. Conclusions and Future Work

In this paper, we studied the impact of frequent TPC on a static multi-hop network consisting of simultaneously transmitting nodes. The data traffic in the network flows on a routing tree aggregated to a single sink. We found that when a node scales its transmission power, the impact to the entire network performance in terms of PDR, latency and energy consumption is significant. This is true in both ContikiMesh and CTP over ContikiMAC. In the former case, the PDR and latency of the network drop by 21% and 14%, respectively, and the energy consumption increases by 20% compared to NO-TPC scenario. Fall in the latency here is because lesser number of packets arrives at

the destination. Compared to No-TPC scenario, there is increase in the amount of total packets and retransmitted packets sent by 29% and 38% respectively. With CTP and ContikiMAC we find that the PDR of the network increases by 16% and there is a decrease in the energy spent by 1.9% and the latency by 43% compared to No-TPC scenario. There is also a drop in total and retransmitted packets by 6.13% and 7.5% respectively.

Although among two routing protocols, CTP is more immune to the turbulence caused by TPC in the network, overall PDR achieved for the amount of energy consumed is not encouraging. One way to reduce additional routing messages is to use a variant of AODV routing protocol known as Gossiping based AODV [48] or utilize a routing protocol based on congestion metric [49]. Performing the same experiments and evaluation methodology for AODV and RPL is planned for future work. Another way to overcome this drawback is to use TPC in conjunction with a scheduler that carefully allocates different timeslots and/or channels for every pair of nodes in the network. Although this technique requires complex scheduling techniques [50] that adheres to strict deadlines, it may be the only way to reduce collision and retransmissions. We carried out our experiments in a simulator with an ideal circular radio model on a homogenous network. In reality the radio ranges are highly irregular and the multi-path effect and a random node deployment can produce a result far inferior to what we have already got. Conducting the same experiment with real hardware and deployment such as office space is also planned for future.

Our conclusion is that TPC inadvertently influences routing and MAC layers. In addition, data rate, traffic flow and node deployment also has a significant impact on the working of TPC. Failure to study these factors can lead to a design of TPC that can do more harm than benefit the network. An isolated design of TPC may lead to non-generic TPC that works only under specific condition and location. This requires time consuming fine-tuning of TPC before the network becomes operational.

Acknowledgment

This research has been partly funded by the European Union's Horizon 2020 project INTER-IoT (grant number 687283) and by ARTEMIS project DEMANES (Design, Monitoring and Operation of Adaptive Networked Embedded Systems (grant number 295372)).

Bibliography

- [1] G. Fortino, A. Guerrieri, F. Bellifemine, and R. Giannantonio, "Platform-independent development of collaborative wireless body sensor network applications: SPINE2," in *IEEE International Conference on Systems, Man and Cybernetics, 2009. SMC 2009*, 2009, pp. 3144–3150.
- [2] A. Liotta, D. Geelen, G. van Kempen, and F. van Hoogstraten, "A survey on networks for smart-metering systems," *Int. J. Pervasive Comput. Commun.*, vol. 8, no. 1, pp. 23–52, 2012.
- [3] D. Geelen, G. Van Kempen, F. Van Hoogstraten, and A. Liotta, "A wireless mesh communication protocol for smart-metering," in *Computing, Networking and Communications (ICNC), 2012 International Conference on*, 2012, pp. 343–349.
- [4] G. Fortino, A. Guerrieri, G. M. P. O'Hare, and A. Ruzzelli, "A flexible building management framework based on wireless sensor and actuator networks," *J. Netw. Comput. Appl.*, vol. 35, no. 6, pp. 1934–1952, Nov. 2012.
- [5] A. Liotta, "The cognitive NET is coming," *IEEE Spectr.*, vol. 50, no. 8, pp. 26–31, Aug. 2013.

- [6] H. Ghasemzadeh, P. Panuccio, S. Trovato, G. Fortino, and R. Jafari, "Power-Aware Activity Monitoring Using Distributed Wearable Sensors," *IEEE Trans. Hum.-Mach. Syst.*, vol. 44, no. 4, pp. 537–544, Aug. 2014.
- [7] G. Fortino, A. Guerrieri, W. Russo, and C. Savaglio, "Middlewares for Smart Objects and Smart Environments: Overview and Comparison," in *Internet of Things Based on Smart Objects*, G. Fortino and P. Trunfio, Eds. Springer International Publishing, 2014, pp. 1–27.
- [8] N. Baccour, A. Koubâa, L. Mottola, M. A. Zúñiga, H. Youssef, C. A. Boano, and M. Alves, "Radio link quality estimation in wireless sensor networks," *ACM Trans. Sens. Netw.*, vol. 8, no. 4, pp. 1–33, Sep. 2012.
- [9] S. Kostin, R. M. Salles, and C. L. de Amorim, "An approach for wireless sensor networks topology control in indoor scenarios," *Proc. 5th Int. Lat. Am. Netw. Conf. - LANC 09*, p. 1, 2009.
- [10] K. Bannister, G. Giorgetti, and S. K. Gupta, "Wireless sensor networking for hot applications: Effects of temperature on signal strength, data collection and localization," in *Proceedings of the 5th Workshop on Embedded Networked Sensors (HotEmNets 08)*, 2008.
- [11] J. Thelen, "Radio Wave Propagation in Potato Fields," in *In Proceedings of the First Workshop on Wireless Network Measurements - WinMee 2005*, 2004, p. 5.
- [12] M. Zennaro, H. Ntareme, and A. Bagula, "Experimental evaluation of temporal and energy characteristics of an outdoor sensor network," *Proc. Int. Conf. Mob. Technol. Appl. Syst. - Mobil. 08*, no. Mobility, p. 5, 2008.
- [13] G. Hackmann, O. Chipara, and C. Lu, "Robust topology control for indoor wireless sensor networks," *Proc. SenSys*, pp. 1–14, 2008.
- [14] Willow Technologies, "The TelosB Mote Platform," San Jose, California.
- [15] MEMSIC Inc, "M ICAz," San Jose, California.
- [16] Chipcon, "2.4 GHz IEEE 802.15.4 / ZigBee-ready RF Transceiver," 2004.
- [17] I. Broustis, J. Eriksson, S. V Krishnamurthy, and M. Faloutsos, "Implications of Power Control in Wireless Networks: A Quantitative Study," *Pam*, vol. 4427, no. 0237920, pp. 83–93, 2007.
- [18] F. Osterlind, A. Dunkels, J. Eriksson, N. Finne, and T. Voigt, "Cross-Level Sensor Network Simulation with COOJA," in *Proceedings. 2006 31st IEEE Conference on Local Computer Networks*, 2006, pp. 641–648.
- [19] A. Medela, B. Cendon, L. Gonzalez, R. Crespo, and I. Nevares, "Future Network and Mobile Summit (FutureNetworkSummit), 2013," *Future Network and Mobile Summit (FutureNetworkSummit), 2013*, pp. 1–10, 2013.
- [20] G. Anastasi, M. Conti, M. Di Francesco, and A. Passarella, "Energy conservation in wireless sensor networks: A survey," *Ad Hoc Netw.*, vol. 7, no. 3, pp. 537–568, May 2009.
- [21] P. Santi, "Topology control in wireless ad hoc and sensor networks," *ACM Comput. Surv.*, vol. 37, no. 2, pp. 164–194, Jun. 2005.
- [22] H. Miao, Y. Jin, Q. Ge, and Y. He, "Expected Transmission Energy Routing Metric for Wireless Mesh Sensor Network," in *2010 International Conference on Computational Intelligence and Software Engineering*, 2010, pp. 1–4.
- [23] X. Zhang, Q. Liu, D. Shi, Y. Liu, and X. Yu, "An Average Link Interference-Aware Routing Protocol for Mobile Ad Hoc Networks," in *2007 Third International Conference on Wireless and Mobile Communications (ICWMC'07)*, 2007, pp. 10–10.
- [24] M. Caleffi, G. Ferraiuolo, and L. Paura, "Augmented Tree-based Routing Protocol for Scalable Ad Hoc Networks," in *2007 IEEE International Conference on Mobile Adhoc and Sensor Systems*, 2007, pp. 1–6.
- [25] I. Radoi, A. Shenoy, and D. Arvind, "Evaluation of Routing Protocols for Internet-Enabled Wireless Sensor Networks," in *2012 8th International Conference on Wireless and Mobile Communications*, 2012, no. c, pp. 56–61.
- [26] J. Haerri, "Performance comparison of AODV and OLSR in VANETs urban environments under realistic mobility patterns," in *Med-Hoc-Net 2006, 5th IFIP Mediterranean Ad-Hoc Networking Workshop*, 2006, pp. 1 – 8.

- [27] J. Jeong, D. Culler, and J.-H. Oh, "Empirical Analysis of Transmission Power Control Algorithms for Wireless Sensor Networks," in *2007 Fourth International Conference on Networked Sensing Systems*, 2007, pp. 27–34.
- [28] O. Chipara, Z. He, G. Xing, Q. Chen, X. Wang, C. Lu, J. Stankovic, and T. Abdelzaher, "Real-time Power-Aware Routing in Sensor Networks," in *2006 14th IEEE International Workshop on Quality of Service*, 2006, pp. 83–92.
- [29] K. Withephanich and M. J. Hayes, "On the applicability of model predictive power control to an IEEE 802.15.4 wireless sensor network," in *IET Irish Signals and Systems Conference (ISSC 2009)*, 2009, pp. 32–32.
- [30] S. Lin, J. Zhang, G. Zhou, L. Gu, J. A. Stankovic, and T. He, "ATPC: Adaptive Transmission Power Control for Wireless Sensor Networks," in *Proceedings of the 4th international conference on Embedded networked sensor systems - SenSys '06*, 2006, pp. 223–236.
- [31] G. Hackmann and C. Lu, "Practical control of transmission power for Wireless Sensor Networks," in *2012 20th IEEE International Conference on Network Protocols (ICNP)*, 2012, pp. 1–10.
- [32] J. Kim, S. Chang, and Y. Kwon, "ODTPC: On-demand Transmission Power Control for Wireless Sensor Networks," in *2008 International Conference on Information Networking - ICOIN 2008*, 2008, pp. 1–5.
- [33] M. M. Y. Masood, G. Ahmed, and N. M. Khan, "A Kalman filter based adaptive on demand transmission power control (AODTPC) algorithm for wireless sensor networks," *2012 Int. Conf. Emerg. Technol.*, pp. 1–6, Oct. 2012.
- [34] V. Shrivastava, D. Agrawal, A. Mishra, S. Banerjee, and T. Nadeem, "Understanding the limitations of transmit power control for indoor w lans," in *Proceedings of the 7th ACM SIGCOMM conference on Internet measurement - IMC '07*, 2007, p. 351.
- [35] O. Gnawali, R. Fonseca, K. Jamieson, D. Moss, and P. Levis, "Collection tree protocol," in *Proceedings of the 7th ACM Conference on Embedded Networked Sensor Systems - SenSys '09*, 2009, p. 1.
- [36] A. Dunkels, "The ContikiMAC Radio Duty Cycling Protocol," Swedish Institute of Computer Science, Dec. 2011.
- [37] M. Michel and B. Quoitin, "Technical Report : ContikiMAC vs X-MAC performance analysis," 2015.
- [38] L. (Eds. . Langendoen, K., Hu, W., Ferrari, F., Zimmerling, M., Mottola, Ed., *Real-World Wireless Sensor Networks - Proceedings of the 5th | Koen Langendoen | Springer*. Como, Italy: Springer International Publishing, 2014.
- [39] O. Chipara, C. Lu, T. C. Bailey, and G.-C. Roman, "Reliable clinical monitoring using wireless sensor networks," in *Proceedings of the 8th ACM Conference on Embedded Networked Sensor Systems - SenSys '10*, 2010, p. 155.
- [40] A. Mainwaring, D. Culler, J. Polastre, R. Szewczyk, and J. Anderson, "Wireless sensor networks for habitat monitoring," in *Proceedings of the 1st ACM international workshop on Wireless sensor networks and applications - WSNA '02*, 2002, p. 88.
- [41] A. Dunkels, "Rime — A Lightweight Layered Communication Stack for Sensor Networks," in *Proceedings of the European Conference on Wireless Sensor Networks (EWSN)*, 2007, pp. 1–2.
- [42] R. Kotian, G. Exarchakos, and A. Liotta, "Assessment of Proactive Transmission Power Control for Wireless Sensor Networks," in *Proc. 9th Int. Conf. Body Area Networks*, 2014, pp. 253–259.
- [43] M. Chincoli, A. Syed, G. Exarchakos, and A. Liotta, "Power Control in Wireless Sensor Networks with Variable Interference," *Mob. Inf. Syst.*, 2016.
- [44] N. a Pantazis, S. a Nikolidakis, D. D. Vergados, and S. Member, "Energy-Ef fi cient Routing Protocols in Wireless Sensor Networks : A Survey," *IEEE Commun. Surv. Tutor.*, vol. 15, no. 2, pp. 551–591, 2013.

- [45] L. H. a Correia and J. M. S. Nogueira, "Transmission power control techniques for MAC protocols in wireless sensor networks," *Netw. Oper. Manag. Symp. 2008 NOMS 2008 IEEE*, pp. 1049–1054, 2008.
- [46] H. Ali, "A Performance Evaluation of RPL in Contiki," Blekinge Institute of Technology, Sweden, 2012.
- [47] A. Forster, D. Puccinelli, and S. Giordano, "Sensor node lifetime: An experimental study," in *2011 IEEE International Conference on Pervasive Computing and Communications Workshops (PERCOM Workshops)*, 2011, pp. 202–207.
- [48] S. Galzarano, C. Savaglio, A. Liotta, and G. Fortino, "Gossiping-Based AODV for Wireless Sensor Networks," in *2013 IEEE International Conference on Systems, Man, and Cybernetics*, 2013, pp. 26–31.
- [49] G. Di Fatta, S. Gaglio, G. Lo Re, and M. Ortolani, "Adaptive routing in active networks," in *OPENARCH 2000: Third IEEE Conference on Open Architectures and Network Programming*, 2000, pp. 8–11.
- [50] G. Exarchakos, I. Oztelcan, D. Sarakiotis, and A. Liotta, "plexi: Adaptive re-scheduling web service of time synchronized low-power wireless networks," *J. Netw. Comput. Appl.*, 2016.

Roshan Kotian is a PhD researcher in the Electrical Engineering department of Technical University of Eindhoven at Eindhoven, The Netherlands. He received Professional Doctorate in Engineering (PDEng) from TU/e in 2012. His research interests include Wireless Sensor Network, Internet-Of-Things, and application of Machine Learning in the Sensor Network.

Georgios Exarchakos is an assistant professor of dependable communications. His primary focus areas are network dynamics, internet of things, network resource management, and smart cross layer optimizations. Georgios joined the Department of Electrical Engineering at Eindhoven University of Technology in 2009 as postdoctoral researcher of network management. Since 2011, as assistant professor at the same department has managed two multinational EU projects and has been teaching computer networks and network overlays. He is co-author of *Networks for Pervasive Services: six ways to upgrade the Internet* (Springer) and editor of two edited books (Springer, IGI-Global). George is the author of more than 50 journal articles and conference papers.

Dr. Stavros Stavrou holds a PhD degree in Electronic Engineering from the University of Surrey, (U.K). Between 1997- 2009 he was a faculty member of the University of Surrey where he lead with his research team a number of wireless planning activities related to Terrestrial and Satellite systems, MIMO and UWB communications. He currently leads the Telecommunication Systems Research Lab (TSRL) at the Open University of Cyprus. His current research interests span through different topics in telecommunications and their applications, including localization, wireless and sensor networks, smart cities and IoT networks. He is also a Fellow of the Higher Education Academy U.K.

Antonio Liotta holds the Chair of Communication Network Protocols at the Eindhoven University of Technology (NL), where he leads the Smart Networks team. Antonio is Editor-in-Chief of the book series *Internet of Things: Technology, Communications and Computing* (Springer) and associate editor of the *Journal of Network and System Management* (Springer) and of the *International Journal of Network Management* (Wiley). He investigates topical issues in the area of computer and multimedia networking, with emphasis on cognitive systems for optical, wireless and sensor networks. He is the author of *Networks for Pervasive Services: six ways to upgrade the Internet* (Springer). His work has recently been featured in *IEEE Spectrum* in *The cognitive Net is coming*.



Roshan Kotian



Georgios Exarchakos



Dr. Stavros Stavrou



Antonio Liotta