Computer Communications xxx (2014) xxx–xxx

Contents lists available at ScienceDirect

Computer Communications

journal homepage: www.elsevier.com/locate/comcom

Self-Organized Things (SoT): An energy efficient next generation network management

Özgür Umut Akgül [*], Berk Canberk

Department of Computer Engineering, Istanbul Technical University, Ayazaga, 34469 Istanbul, Turkey

ARTICLE INFO

Article history:
Available online xxxx

Keywords:
Internet of Things (IoT)
Self-organized networks
Energy efficiency
Durability
Spatial optimization

ABSTRACT

The prodigious rise in consumer electronics and advanced low-cost manufacturing techniques have increased the human-device interaction in daily life. This interaction has led researches on the concept of Internet of Things (IoT) to raise the quality of life in all manner. However, communication between these "things" reduces their lifetimes because of the battery limitation. Due to this reason, energy efficiency has became one of the most important challenge in IoT. In many recent studies, even though this main challenge has been addressed with different aspects, most of all these solutions have compromised the coverage area for energy efficiency. Beside coverage losses, most of the proposed frameworks demand human control and interaction during fail recovery. The energy efficiency and minimization of human interactions should be handled simultaneously in IoT frameworks for more effective deployments. Additionally, the total covered area, distribution of devices and the distribution of events have to be taken into account to successfully manage the network. Having this motivation, we propose an energy efficient Self-Organized Things framework, SoT. The proposed SoT uses an optimization procedure in order to minimize the overall energy consumption of the *things*, while stabilizing the total coverage area. Moreover, the human dependency is overcomed in SoT by re-defining the next generation self-configuration, self-optimization and self-healing procedures of self-organizing network structure of Long-Term Evaluation (LTE) systems. In this self-management process of the SoT, specific spatial distributions of devices and intersections of their coverage areas are also analytically derived. Here the spatial distribution is used to determine the distribution of the active *things* in 2D space. To increase the efficiency, the remote devices are activated and the event observation rate is maximized. By definition the "event" is a special attribute that the *things* are observing. In addition to spatial distribution in SoT, a conflict parameter, that calculates the intersection of devices' coverage areas, is also proposed to enlarge the coverage area. Using this conflict parameter, the selection of overlapping *things* is prevented and the actual covered area is maximized. By increasing the actual covered area, the probability of observing an event is increased. Consequently, the performance of the proposed SoT framework is measured in terms of the total covered area per energy and network's lifetime. The through evaluation results verify that proposed framework increases the lifetime of network 150%.

1. Introduction

Through the last decade, while the technological capabilities of commercial electronic devices increases, they became cheaper. This economical and technological improvements have lead an enormous increase in the number of small electronic devices, which have communication capabilities consequently, the concept of Internet of Things, IoT, has emerged as a promising framework. At a high level IoT can be defined as establishing connections between electronic devices, "things", i.e. varying from every kind of sensor to smart phones and even cars. The integration of IoT into the daily life, provides many advantages to many fields, i.e. planning, medicine or security [1]. Many optimistic scenarios are presented about its advantages [2]. According to this scenario, due to the communication between a business man's alarm clock, coffee machine and car, a great morning planning can be executed. However, the realism of these scenarios is doubtful.

Behind all these optimistic scenarios about IoT, there exist many challenges like the ones stated in [3–5], i.e. users security and privacy or routing technology or communicating different kind

[*] Corresponding author.
E-mail addresses: akgulo@itu.edu.tr (Ö.Umut Akgül), canberk@itu.edu.tr (B. Canberk).

http://dx.doi.org/10.1016/j.comcom.2014.07.004
0140-3664/© 2014 Elsevier B.V. All rights reserved.

of devices or handling all these huge amount of data while keeping the energy consumption low. Even for the scenario in [2], if there is an interruption to the plan the whole easiness and efficiency will be lost. It is also indicated that the only way to gain complete knowledge and prevent interruptions is using more sensors. However, this increase of devices will cause an increase in raw knowledge and energy consumption.

The energy efficiency in IoT is a very popular research area because of both economical and environmental concerns, i.e. Capital Expenditures (CAPEX), Operational Expenditures (OPEX), lack of resources and the global warming. Beside them, as most of the *things* are user equipments with low battery power and the number of devices that is out of service due to some kind of external failure is negligible, the durability is highly dependent on the energy consumption. The service persistence, more generally durability, changes with the energy efficiency as for the most of the cases the main reason of device losses is based on the battery end. Secondly, self-management is also crucial for IoT concept [6]. A most rapid and possibly the most efficient solution is to define a self-organized framework for the IoT concept.

The tradeoff between energy and coverage area is presented in Fig. 1. Here, a "communication rate" can be defined as the dimensionless rate of active communication time over the total time interval. In this Figure, the coverage area changes under different traffic requests are presented for a generic IoT model without any scheduling or management improvements. Without energy optimization the maximum lifetime, that can only be achieved by 0 traffic rate, is 50 h. In Fig. 1, it can also be seen that increasing communication rate decreases the energy efficiency and also decreases the lifetime of devices. This inversely proportional relationship between the energy consumption and network durability should be carefully addressed with a self-management perspective. To increase the duration of IoT network, the energy efficiency has to be increased. However, for many cases energy efficiency can only be achieved by the loss of coverage area. To overcome this tradeoff, there exist several solutions in the current literature. Sending the unnecessary devices into sleep mode lowers the energy consumption and increases the lifetime of the thing. In [7], a bio-inspired self- organization scheme is presented. In this study the unnecessary devices are sent to sleep mode and by this method the energy consumption is decreased. In [8], an energy efficient sleep scheduling method proposed. It is showed that the proposed scheme in [8] can guarantee QoS needs. In [9], usage of Wi-Fi technology in IoT is investigated and evaluated the power consumption of *things*. Even though many researches is going on to overcome the problems in IoT, as stated in [5], IoT is still the "Wild West" of technology. So most of the studies are going on from mapping the classical communication solutions to IoT. In [10], the problems in internet of nano *things* is investigated and some possible solutions are presented. [11] proposes an immune inspired Distributed Node and Rate Selection (DNRS) in sensor networks. In this study a B-cell inspired DNRS makes the selection of the best node to transmission. In [12], Peng et al. presents a survey on the algorithms of two important concept in SON structure, self-configuration and self-optimization. A self-organized clustering protocol is presented to increase the life time of the sensor networks in [13]. Finally [14] is a survey about the self-organization concept and gives a complete study from the basic concept to the challenges and trends. However, none of these studies are considering the distribution of devices and distribution of events in 2D space.

Consequently, by considering all these aforementioned recent studies, and with the motivation of merging the self-management issues by optimizing the energy consumption with coverage area for IoTs, in this paper, we present a novel self-organized internet of *things* framework, named Self-Organized Things, SoT, by making the following contributions;

- By defining three main self-organizing concept, more specifically self-configuration, self-optimization and self-healing, the human intervention is decreased. Thus self-management is achieved.
- In self-configuration of the proposed SoT, two parameters, i.e. conflict parameter, ξ_i and spatial correlation parameter I_k, are defined. Even though there exist some works that uses this kind of parameters separately, to the best of our knowledge there exist no work that integrates both of them. We are presenting a novel approach that integrates both of these parameters with self-organizing framework and by this way increasing the coverage rate in the network.
- In self-optimization, a spatial correlation parameter, I_k and conflict parameter, ξ_i, are calculated as decision parameters to reach an energy efficient topology.
- We develop a sleep mode optimization mechanism to minimize the unnecessary energy consumption and stabilize the battery lifetime.
- In self-healing, the device losses due to battery limitation are compensated and with the energy efficiency increase, gained by self-optimization, we are increasing the durability of the network.

The rest of this paper is organized as follows. In Section 2, we present the network model in our work and also presenting a probabilistic approach to calculate number of active devices to guarantee a specific network performance. In Section 3 we give the self-organized framework, explaining all three concepts, i.e.

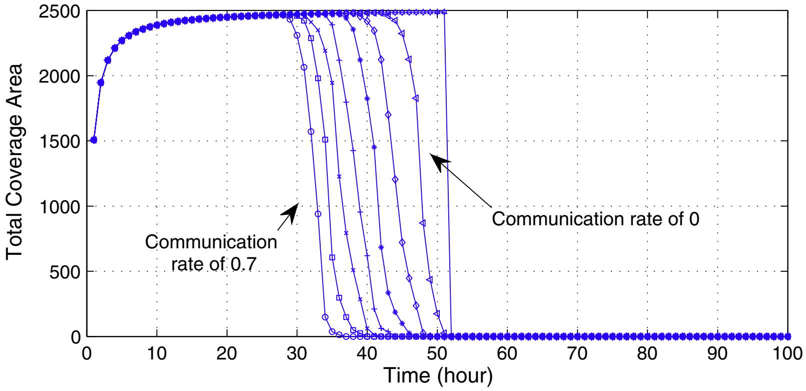

Fig. 1. Coverage area changes for different request rates.

self-configuration, self-optimization and self-healing. Finally in Section 4, we give the simulation results and evaluate the success of the proposed framework.

2. The network architecture

In this work, the considered topology for the network of *things* covers an area which contains many smart devices that observe certain events and communicate with each other or with a server about their observations. Such an area is presented in Fig. 1. Despite the large varieties, these devices can be investigated under two major device types, i.e. trigger based devices (TBD) and periodic signal devices (PSD). The accuracy of this aggregation of device types can be investigated using the famous example of the IoT network of a businessman who has a meeting at 8 am. Previous night, he sets his phones alarm clock to 7 am and installs his coffee machine and toast machine. At 6 am, telephone gathers the traffic flow speed via Internet and based on this knowledge, it changes the alarm clock. For example it pulls the alarm clock to 6^{30} am due to the low traffic flow speed. At 6^{35} am, while the businessman is in shower, telephone transmits a wake signal to the coffee machine and toast machine. At 6^{45} am, both toast machine and coffee machine transmits their ready signal to the phone and the businessman starts his breakfast. At 6^{55} am, telephone connects to the car and opens its air conditioner. At 7^{10} am, the businessman is on his car and traveling to his meeting. The reality of this scenario is doubtful, however, it helps to understand the difference between IoT devices. During the optimistic scenario of the businessman, the connection between coffee machine and telephone is a great example for a connection between a TBD device and a controller. This type of communication does not contain any specific data and more like a trigger signal that is used to wake TBD and wait it to observe the expected event. When it observes the event, which is the "hot coffee" for this specific case, it transmits another trigger back to the controller. The communication between the toast machine and the telephone is also a TBD communication example. However, the communication between the telephone and the car is the second type of communication, a PSD-to-Controller communication. The car contains many possible functions and the telephone has to define the specific attributes of the asked function like the name of the function and expected variables. For example, for the considered scenario, the telephone has to transmit that the air conditioner should be opened and it should be set to 22 °C degree. So different than TBD communications PSD has to transmit complete data packets to controllers and receive specific data packets. Battery consumption is the most important parameter for TBDs and the goal is to increase the duration of these devices while maintaining the coverage area. The second type of users, PSDs, have to send a periodic life signal to the base station. If there exist any server request for these devices they transmit the necessary data. In other cases they only transmit this life message. The coverage area is not important for this devices. Durability is the main concern for these devices and of course the battery consumption is the main reason of the battery exhaustion of these devices. With these objectives we defined three states for devices, namely *sleep*, *active* and *passive*. In sleep mode devices do not observe the area and all the unnecessary energy consumptions are turned off. In active mode, devices are working and observing the area. Finally in passive mode, the battery of devices are exhausted and devices are completely turned off. These three states are the only states that a device may exist. We accepted that the controller, e.g. mobile phone of the businessman for the specific example, could perform the distinction between TBDs with different tasks. In this study, the distinction part is not covered and accepted that all the TBDs are observing the same event.

Fig. 2 shows our example observation area. As seen in the Figure, there exist 8 *things*, 6 TBDs and 2 PSDs in this network. In a real scenario case there exist more than 20 devices in IoT. The main objective is to keep the coverage area as high as possible for a long time. A practical idea to increase the coverage area can be increasing the number of devices. Even though this increases the coverage area, because of the battery limitations the life time of all this devices will be short. In this practical case by increasing the number of devices, coverage improvements are maintained by the sacrifice of energy. However, as can been in Fig. 2, there could be cases in the network when putting a device in sleep mode does not create a high fall in the coverage area however decrease the energy consumption of the network. As an example, in Fig. 2, for 1st, 2nd and 3rd devices, mostly they are covering the same area. Instead of working all of them together, the 2nd device may be putted in sleep mode and 1st and 3rd devices may be kept in active mode. After batteries of 1st and 3rd devices are finished, the 2nd device may be activated and by this scheduling technique, the energy consumption can be decreased and the durability of the network can be increased by a small loss of coverage area.

In order not to decrease the performance or coverage area to increase the energy efficiency, this network configurations must be done according the objectives of devices. PSD devices do not have to keep in active mode as they do not have to make an observation in their area. While there is not any service requests, these devices can be sent to sleep mode. As these devises do not consume much energy during their sleep mode, the main energy killer process is the transmission. The main concern in this study for PSDs are their interference with TBDs. As they may both try to send information at the same time, interference could destroy both of messages. Transmission repetitions will consume power and cause decrease on the durability. Because of this reason, when a PSD gets active, other TBDs will have to get sleep mode to prevent interference. In this study interference between PSDs is not covered. For the second type of devices, TBDs, as these devices are not communicating without observed trigger, the energy optimization can only be achieved by putting these devices into sleep mode. However, as the main goal of these devices is to catch this specific trigger, as the number of active nodes decreases, the probability of catching this trigger decreases. This tradeoff between durability and the success of the observation is critical.

Most of the cases, the observed area does not have to cover all the region. Instead, a fraction of the region could be enough for these cases. For these cases, to increase the durability a lower number of devices, N, can be enough to fulfill the expected coverage rate. If x binary random variable denotes the existence of an event in the observation area, A, and C denotes the observed area, the probability of an event existing in C if existence of the event is known, $P(C|x)$, can be calculated using Eq. (1).

$$P(C|x) = \frac{P(C) \times P(x|C)}{P(x)} \tag{1}$$

as x and C are discrete events $P(x|C)$ will be equal to $P(x)$ and Eq. (1) will be equal to

$$P(C|x) = \frac{P(C) \times P(x|C)}{P(x)} = \frac{P(C) \times P(x)}{P(x)} = P(C) \tag{2}$$

As presented in Eq. (2), the probability of occurring an event in the observed area is equal to the observation probability of the area. The equivalence $P(C|x) = P(C)$ states that the probability of the event occurrence within the covered area is equal to the probability of covering the area. Using this equivalence, the event occurrence probability can be neglected during the performance calculations and the usage of probability of covering an area will be sufficient.

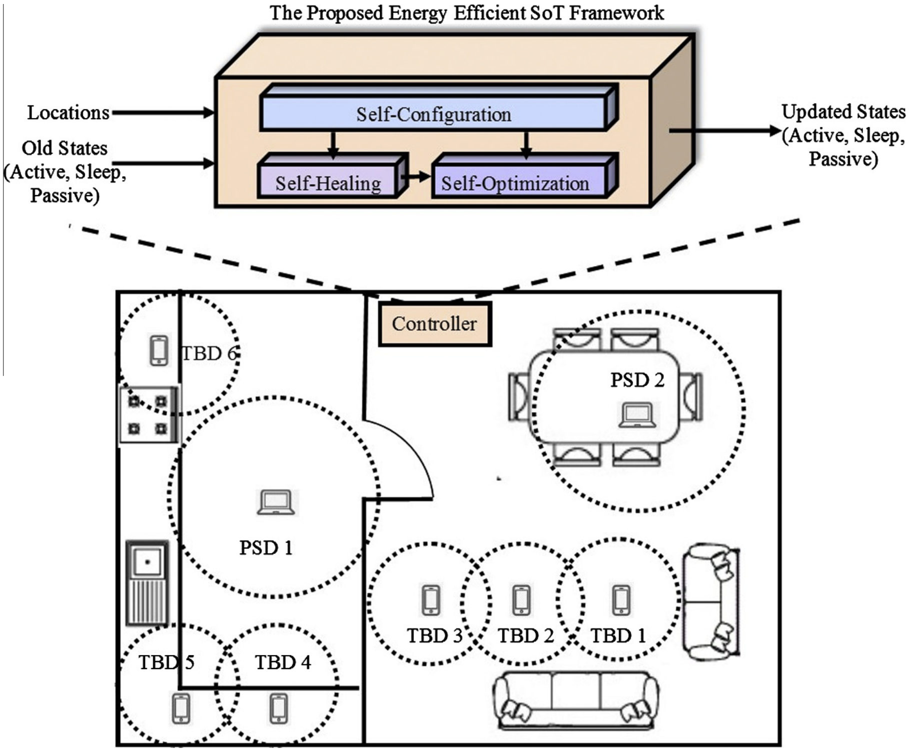
The Proposed Energy Efficient SoT Framework
Locations
Old States (Active, Sleep, Passive)
Self-Configuration
Self-Healing
Self-Optimization
Updated States (Active, Sleep, Passive)
Controller
TBD 6
PSD 2
PSD 1
TBD 3
TBD 2
TBD 1
TBD 5
TBD 4

Fig. 2. An example IoT network topology with SoT integration.

From this point of view the probability of observing an event is presented in Eq. (3).

$$P(C|x) = \frac{N \times \pi \times R^2}{A} \quad (3)$$

where R is the radius of coverage area for a single TBD. In Eq. (3), "$N \times \pi \times R^2$" term denotes the total observed area by N devices. However, as can be seen in Fig. 2, the actual total covered area does not always equal to the practical covered area as some of the devices covers the same region. As the actual covered area is smaller than the practical covered area, a normalization factor has to be used. In this study, we defined a normalization factor, ψ, to calculate the actually covered area. The actual covered area is equal to the ψ times practical coverage area. So the probability of observing an event will be equal to Eq. (4).

$$P(C|x) = \frac{\psi \times N \times \pi \times R^2}{A} \quad (4)$$

where ψ is the rate of the actual covered area over the practical covered area and calculated as Eq. (5).

$$\psi = \frac{\sum_{n=1}^{N} \pi \times \ R_n^2}{N \times \pi \times R^2} \quad \forall n \in N \quad (5)$$

Using Eq. (4), the number of TBDs needed to expect to observe P rate of all the events will be,

$$N = \frac{P \times A}{\psi \times \pi \times R^2} \quad (6)$$

where P is the probability of observing an event in observed region, more generally the rate of events that is expected to be observed, and ψ is the actual coverage area constant. As can be seen from the definition the expected number of observed events is equal to P, Eq. (7), and the standard deviation is equal to Eq. (8).

$$E(C|x) = P(C|x) \quad (7)$$

$$\sigma = \sqrt{P(C|x) - P(C|x)^2} \quad (8)$$

If we consider Eqs. (6)–(8) together, the number of active TBDs, to observe a constant rate of over all events, $E(C|x)$, is equal to N. However, the observation rate cannot be guaranteed due to the probabilistic distribution of event locations and a standard deviation, σ, may occur between the expected rate of observed events and the actual rate of observed events. The σ term also indicates the number of possible missed events by putting some of the devices in sleep mode.

Three states are defined for the devices, i.e. sleep mode, passive mode and active mode. Sleep mode is the minimum energy consumed state for a device while passive mode indicates that the device can no longer function due to the passive battery. The energy consumption for a device in passive mode is equal to 0. For a device in active state, three possible action are defined, being idle, receiving and transmitting. The idle action is the case when a device observes its region. This is especially defined for TBSs and it indicates that the device is waiting for an event, a trigger. The receive action and transmit action are the communication actions. The energy consumptions of all these two state and three action is presented in Table 1. As can be seen, wireless transceiver power consumptions that are presented in 802.11b are used for TBDs and 802.11a for PSDs [15,16]. All the used parameters are presented in Table 2.

Table 1
Power consumption model for different modes and devices.

Mode	TBDs (mW)	PSDs (mW)
Sleep	132	132
Idle	554	990
Receive	726	1320
Transmit	1089	1815

Please cite this article in press as: Ö.U. Akgül, B. Canberk, Self-Organized Things (SoT): An energy efficient next generation network management, Comput. Commun. (2014), http://dx.doi.org/10.1016/j.comcom.2014.07.004

Table 2
Symbols and their definitions.

Symbol	Definition
X	Binary random variable for the existence of a event
ψ	Actual coverage area constant
δ	Actual coverage area for the unit energy
ξ	Conflict parameter
ϕ	State of a device
A	Observed area
R	Coverage radius
D_{ij}	Euclidian distance between two devices
N	Number of devices
$N(t)$	Number of active sensors at time t
$E_T(t)$	Total energy consumption at time t
C	The observed area
P	Rate of observed events
(x, y)	Location coordinates of a device
I_K	Correlation parameter

3. The Self-Organized Things, SoT, framework

The proposed framework, SoT, has three different focuses;

- Decreasing the energy consumption.
- Increasing the durability of the network.
- Maintaining self-management and by this way decreasing the human interaction.

It is a known fact that the high energy consumption is decreasing the life time of the devices and also decreasing the durability. Due to this, the energy consumption is the main problem that needs to be taken care of to increase the durability of the network. To decrease the energy consumption, the number of active sensors can be decreased to a number, N, that is calculated for an expected rate of observed events. In literature there exist many algorithms that is working with a similar idea however the spatial distributions and the actually covered areas of these devices are not covered. As showed in Fig. 2, for many cases a random selection mechanism is ended with low event observation rates. To increase the observation rate while increasing the durability we are presenting a Self-Organized Things (SoT) framework. As can be seen in Fig. 2, the three main concepts of self-organized structure, self-configuration, self-healing and self-optimization, is presented in this work. As showed in Fig. 2, the inputs of SoT is the locations and the statuses of each devices. The objectives of our work can be modeled as an optimization process. The actual coverage area for the unit energy, δ, is presented in Eq. (9).

$$\delta = \frac{\psi \times N(t) \times \pi \times R^2}{E_T(t)} \tag{9}$$

where $E_T(t)$ denotes the total energy consumption in the network at time t and $N(t)$ is the number of active sensors at time t. Due to the traffic changes, triggers and service request, the total energy consumption of the network is a dynamically changing parameter. The number of active devices, N, changes with time as the conflict parameter, ψ, changes according to the active nodes. The objective of SoT framework can be presented as an optimization process as presented in Eq. (10).

$$\begin{aligned} &\text{Maximize} \quad \int \frac{\psi \times N(t) \times \pi \times R^2}{E_T(t)} \times dt \\ &\text{Subject to} \quad N(t) \geqslant N \end{aligned} \tag{10}$$

The optimization objective shows that the δ terms keeps increasing while the number of active nodes increases. However, if the number of active nodes passes a critical value, N_{cr}, conflicts begin and the ψ term decreases. As the energy consumption of an active node does not change with the active covered area, the total energy consumption of the network keeps increasing. After this critical N value the δ terms keeps decreasing. The maximum δ value can be reached for this N_{cr} parameter. However, there could be such a case when N_{cr} is lower than the necessary number of active devices, N. At this time, to fulfill the observation rate the number of active devices increases inspite of the loss of energy efficiency. The proposed SoT framework is maintaining this optimization problem according to a basic sleep and wake up methodology. To put a device to sleep state, SoT framework uses two decision parameters namely, the local indicator of spatial autocorrelation coefficient and the conflict parameter. As it is explained in the further section the conflict parameter is different from the actual coverage area constant, ψ. The conflict parameter presents a knowledge of how much unique area the active TBD is covering and unlike ψ, it is a device based constant. The local indicator of spatial autocorrelation (LISA) coefficient presents knowledge of active device distribution in the covered area. As an activation rule, the proposed framework tries to activate the TBDs that are remote from the remaining active TBDs. The higher LISA parameter indicates a remote TBD. After self-management process, SoT framework outputs the states of all devices.

3.1. Self-configuration

The SoT framework starts with the self-configuration structure. The inputs of this block are the states and the locations of the devices. The self-configuration structure is a pre-data-processing structure that prepares the input data for the optimization. The self-configuration algorithm is presented in Algorithm 1. As can be seen from Algorithm 1, two decision parameters are calculated in this structure. These are the conflict parameter, ξ, and the spatial correlation parameter, I_K.

Algorithm 1. Self-Configuration Algorithm

```
Require: $(x, y)$
Ensure: $\xi_i, I_k$
for $n \leftarrow 1$ to $N$ do
    if TBD then
        $\phi \leftarrow 1$
    else
        Send Life Signal
        $\phi \leftarrow 0$
    end if
end for
Calculate $\xi$
Calculate $I_K$
```

The self-configuration is putting all workable, not dead, TBDs to active state while it is putting all PSDs to sleep state. The coverage area is the main concern for TBDs. So in topology design, for the most optimal solution self-configuration is activating all the devices. Then the conflict calculation and spatial correlation calculation is calculated only for TBDs.

3.1.1. Conflict calculation

The main problem in network configuration is preventing the selection of devices that is covering the same area. For example in Fig. 2, if TBDs 1 and 3 is in active state than putting 2 into active state is not necessary as the most of its observation region is covered by 1 and 3. However, there could be also cases like TBDs 4 and 5 in Fig. 2, where the conflict between nodes are not very high and in case of necessity they can be both in active state. So there exist a

Please cite this article in press as: Ö.U. Akgül, B. Canberk, Self-Organized Things (SoT): An energy efficient next generation network management, Comput. Commun. (2014), http://dx.doi.org/10.1016/j.comcom.2014.07.004

decision problem in the conflicts. To overcome this decision problem, we define a conflict parameter that is calculated for each TBDs in the network.

The conflict parameter, ξ, is kept in the network controller and is calculated based on the locations of each device. The conflict parameter calculates how much unique area that a TBD covers. The ψ parameter presented in the previous section is a normalization factor that is used to determine the number of necessary devices. The ψ parameter is a general version of ξ as ξ is a device based parameter and ψ is a network based parameter. In Fig. 3, there exist two TBDs. The distance between these TBDs are R_{12} where the coverage area radius is R. There could be two possible cases, R_{12} is smaller than $2R$ or R_{12} is greater or equal to $2R$. The first case, $R_{12} \leqslant 2R$, indicates that there exist a conflict and this two device is covering the same area. Then the conflict parameter is calculated. Conflict parameter changes between 0 and 1. Value of 1 indicates that there is no conflict and the value of 0 indicates that this two devices are covering the same area. the total conflict parameter for ith device is the sum of all the conflict parameters between ith device and the jth device, as presented in Eq. (11).

$$\xi_i = \sum_j \frac{D_{ij} \times 1(D_{ij} < 2R) + 2R \times 1(D_{ij} \geqslant 2R)}{2R} \quad \forall i,j \in N \tag{11}$$

As can be understood from Eq. (11), conflict parameter indicates how much unique area the device is covering and higher parameter values are better in decision process.

3.1.2. Spatial correlation calculation

Even though the conflict parameter presents a great knowledge to make a decision, for many cases, it is not sufficient. For example in Fig. 2, for TBDs 4 and 5, the conflict parameters are equal. So the decision process that based on only conflict rate of the device will randomly activate one of these devices. However, for many cases this random decision is not efficient. For example if TBDs are trying to observe the motion in the region being far from the rest of the TBDs improves the change of observing the movement. For most of cases, giving the activation priority to the farther TBD is the most optimal decision. To measure the spatial distribution of devices in 2D space, we use spatial distribution parameter, I_k, for each device. In Fig. 4, an example distribution of devices in the 2D space is presented.

As definition, spatial distribution parameter, I_k, shows how a special attribute changes with location. In this study, we are trying to observe how the active state devices distributed in 2D space does. More specifically, the I_k parameter, which is defined for each device separately, gives an idea of device's position in the 2D space of active devices. I_k formula is presented in Eq. (12).

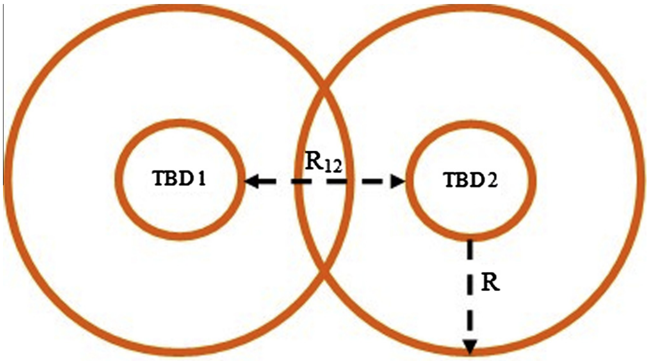

Fig. 3. Conflict parameter calculation.

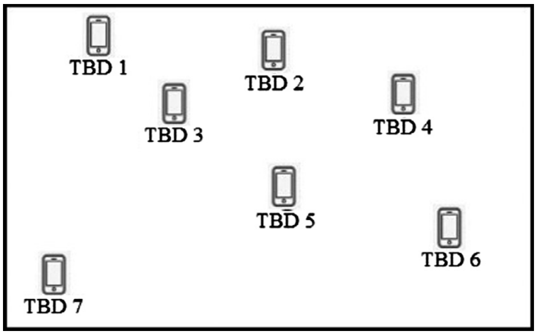

Fig. 4. Example case for spatial distribution of TBD devices.

$$I_k = \frac{\phi_k \times \sum \phi_i \times D_{ki}}{\sum_l \phi_l^2} \quad \forall k,i,l \in N \tag{12}$$

where ϕ denotes the state of the device and D_{ki} denotes the Euclidian distance between element i and k. We defined two values, 0 and 1, for the ϕ attribute. Three state defined for a device namely, active, sleep and dead. Neither in sleep state nor in dead state, devices does not communicate. So ϕ parameter gets value of 0 for both of these states and gets value of 1 for active state. w_{ij} parameter represents the distance between device i and device j. As can be easily understood the spatial correlation parameter, I_k, in Fig. 4 is now presenting the correlation of active nodes in 2D space and for each device I_k presents the correlation of other active devices around it. It can also be seen that if device is in sleep mode or passive mode, device's spatial correlation parameter will be equal to 0. It can be understood from Eq. (12) that the optimal decision is activating a device with higher spatial correlation coefficient as it indicates that device is covering a more different part of the region than the rest of the devices. For the example spatial distribution presented in Fig. 4, functioning TBDs 1–6 is not efficient as they are covering close areas. For many cases, to increase the change of event catching, functioning TBDs that is far from each other is more optimal. For example if TBDs are smoke sensors, if an event occur in the coverage area of TBD 3, it can also be observed by TBD 1 and TBD 2. So functioning these three devices together is inefficient as there exist no increase in change of event catching but there exist an increase in energy consumption. The most optimal configuration is activating TBD 7 and TBD 3 simultaneously as it increases the change of event catching.

3.2. Self-optimization

Self-optimization is the part where the most optimal configuration of the active devices is founded. Self-optimization receives the I_i parameters and ξ_i parameters of the devices in the network from the self-configuration. Specifications of these parameters can be presented as,

- ξ_i, Conflict parameter, gives the knowledge of how much unique area the device is covers.
- Higher ξ_i parameters are better as they show that device is covering an uncovered area.
- I_i, Spatial correlation parameter, gives the knowledge of how the active state devices change within 2D space.
- Higher I_i parameters are better as they present that device i is covering a less covered fraction of the region.
- I_i Does not give a complete knowledge about the conflicts between devices. If two device are far from the rest of the devices, even they are covering completely the same area, their I_i parameters are going to be large.

Please cite this article in press as: Ö.U. Akgül, B. Canberk, Self-Organized Things (SoT): An energy efficient next generation network management, Comput. Commun. (2014), http://dx.doi.org/10.1016/j.comcom.2014.07.004

One of the most important objective of this work is an energy efficient and high coverage rate topology configuration and we are measuring the optimality with this objective. To fulfill this objective, we are using both I_i and ξ_i parameters. The self-optimization algorithm is presented in Algorithm 2.

Algorithm 2. Self-Optimization Algorithm

```
**Require**: $\xi_i, I_k$
**Ensure**: $\phi$
List devices
Calculate N (t)
**while** $N_{actv} < N(t)$ or not All Devices **do**
  **if** $\xi_i = N_{actv}$ **then**
      $\phi \leftarrow 1$
      $N_{actv} + 1 \leftarrow N_{actv}$
  **else**
      **if** $\xi_i = \xi_j$ **then**
          **if** $I_i > I_j$ **then**
                $\phi_i \leftarrow 1, \phi_j \leftarrow 0$
            **else**
                $\phi_j \leftarrow 1, \phi_i \leftarrow 0$
        **end if**
    **else**
        **if** $\xi_i > xi_j$ **then**
            $\phi_i \leftarrow 1, \phi_j \leftarrow 0$
        **else**
            $\phi_j \leftarrow 1, \phi_i \leftarrow 0$
            **end if**
        **end if**
    $N_{actv} + 1 \leftarrow N_{actv}$
        **end if**
**end while**
**if** $N_{actv} < N(t)$ **then**
    **for** $n(t) - n_{actv} > 0$ **do**
        **if** $\phi_i = 0$ **then**
            $\phi_i \leftarrow 1$
            $N_{actv} + 1 \leftarrow N_{actv}$
        **end if**
    **end for**
**end if**
Update $\psi(t)$
```

As can be seen in Algorithm 2, the first step is listing all the devices in the network. There exist two lists, one is based on I_i parameters and the other one is based on ξ_i parameters. Both lists are created in the decreasing order so the first terms are presenting the highest valued devices. Another important parameter in self-optimization is the number of minimum active devices, N, to observe certain rate of events in the network. The N value is presented in Eq. (6). However, the ψ term changes with the locations of the active parameters. The problem in this definition is the change of $N(t)$ in each topology change. So during the run time, ψ calculation is problematic. Instead, we are using the ψ values that is calculated after the optimization process in the previous topology change. More specifically, in time t, to calculate the $N(t)$ parameter, we are using the $\psi(t-1)$ term. ψ parameter is a normalization parameter to calculate the actual covered area from the practical covered area. Using the ψ term from the previous state will cause a decrease in the precision of the calculated device number as the previously observed network topology at time $t-1$ has changed and many of the active devices in the previous topology are in passive in the network topology at time t. For a far spread IoT network topology, usage of $\psi(t-1)$ is insufficient. As the calculated covered area at time t will be smaller than the real covered area, the calculated number of devices will be inefficient and will cause a sharp decrease in the observed event rate. In this study, we are observing a dense IoT network topology, which contains many IoT devices covering the same area. Due to this, the usage of ψ parameter from a previous state will not decrease the precision of the management.

After listing all the devices then the device activation is started. The activation process primarily uses the ψ list. As can be seen in Algorithm 2, if the conflict parameter value of the device is equal to the number of devices, as this indicates that the ith device does not conflict with any other devices, the device is activated and the number of active devices is incremented by 1. If the conflict parameter is smaller than the number of devices in the network, then there exist at least one other device that this device have conflicted. In this case two possible situations are possible, equal conflict parameters between conflicted devices or different conflict parameters. If the conflict parameters are different the higher conflict device is activated while the lover one is sent to sleep state. In case of equal conflict parameters, I_i list is checked. The I_i parameters of conflicted devices are compared and the higher one is activated. The rests are sent to sleep mode. For the example network topology that is presented in Fig. 2, the decision process for TBDs are critical for two group of TBDs, 1–3 and 4–5. The 6th TBD as it does not have a conflict with others, it is directly activated. TBDs 1–3, they have conflict with each other. However, comparing the conflict parameters of TBD 1 and TBD 2, it is obvious that $\xi_1 > \xi_2$. So in decision process, TBD 1 is going to be activated while TBD 2 will be kept in sleep mode. As the TBD 2 is sent to sleep mode, TBD 3 will be activated, too. For the second group, TBDs 4 and 5, it is obvious that their conflict parameters are equal to each other. So their I_i parameters are going to be compared. It is obvious that $I_5 > I_4$ so the 5th TBD is going to be activated while the 4th TBD is sent to sleep mode.

The comparison of the I_i parameters could be seem a little confusing for the second group of TBDs, 4th and 5th. However, two pre-processes, continuing the activation process according to the queue in conflict list and the activation of all TBDs in the self-configuration, guarantee the optimality of the comparison. As the I_i comparison is only done for the conflicted ones, the devices that is going to be sent to sleep state does not create a comparison failure because the effects of this device can be ignored. Following the Algorithm 3 or Algorithm 1, the Algorithm 2 is executed at the network topology changes, i.e. device's active/passive state changes and initialization case. As the complexity of the proposed framework is another challenge, the main optimization framework needed to be investigated in terms of complexity. As can be seen in Algorithm 2, the complexity of the algorithm except for the ψ calculation is equal to O(N). However, as the calculation of ψ parameter is performed for each device couples, this part's complexity will be O(N^2). From this point of view, the complexity of the overall Algorithm 2 is O(N^2). However, as stated earlier this is because of the applied calculation method of ψ parameter, using a simpler calculation technique, the complexity of the algorithm will decrease.

3.3. Self-healing

After the self-optimization process, the network reaches its most optimal state. However, the battery lifetime is a big problem in IoT. After some time from the activation of a device, its battery power is exhausted and due to this exhaustion, device sends out a passive signal and sends itself to the passive state. At this case SoT framework re-optimizes the network. The network durability is increased by the self-healing mechanism. By durability, we indicate the total time that the area is observed. By the self-managing process, the network always exist in the most optimal topology in terms of total covered area and energy efficiency.

When a device sends out a passive signal, the network has to be optimized from the beginning as the conflict parameters and the spatial correlation parameters are going to change. Self-healing mechanism received the passive message and updates the device list by deleting the passive device from its list. After that the self-healing process passes the updated I_i parameters and the ξ_i parameters to the self-optimization process and based on these values the new network topology is decided. The algorithm of self-healing is presented in Algorithm 3.

Algorithm 3. Self-Healing Algorithm

```
Require $Passive\ signal\ from$ ith $device$
Ensure $\xi_i, I_k$
Remove $i$ from list
$N - 1 \leftarrow N$
for $n \leftarrow 1$ to $N$ do
    if $TBD$ then
        $\phi \leftarrow 1$
    else
        Send Life Signal
        $\phi \leftarrow 0$
    end if
end for
Calculate $\xi$
Calculate $I_K$
```

If there is no server request, PSDs only send a life message and then put itself into sleep state. A possible problem here is the case when both a PSD and a TBD tries to communicate at the same time. Due to interference, they have to retransmit and as stated previously, transmission is the most energy consuming process for both PSDs and TBDs. To prevent this inefficient case, we give priority to PSD communications. If the PSD needs to communicate, it sends a call to each TBD that has conflict with PSD. And then it communicates. If it sends a life signal, TBD sets a counter and until this counter reaches 0, it stays in sleep mode. After counter reaches 0, it passes to active mode again. If PSD is responding a server request then TBD waits for another call from PSD and until this call it passes to active mode. Note that PSDs check their remaining battery power before they send a signal to TBDs. If their power is not sufficient then they sent a message to the controller and inform it that they are passive.

4. Performance evaluations

The success of the proposed SoT framework is investigated in terms of total actual coverage area and the total coverage area per energy unit, δ. To observe a constant level of events the number of devices that need to be activated is previously presented in Eq. (6). In simulations, addition to the performance evaluation of SoT framework in times of durability increase, we also present the effects of different traffic rates of PSDs and TBDs on network performance and energy efficiency. By traffic rate we are implying the server requests that TBDs and PSDs receives.As the device communication causes a high increase in energy consumption, increasing communication will also decrease the lifetime and the durability of devices. As the number of devices will decrease rapidly and randomly, the total coverage area and the durability will become a problem. The proposed framework has to handle this changes in the network topology and optimize the network topology. So the performance of the proposed framework under different traffic rates is not only a valid mark of energy optimization framework but also critical to decide if the proposed framework is successful. As going to be explained more specifically, we created that traffic requests using a probabilistic approach. We defined two different probability variable, P_1 and P_2, where P_1 is used to create PSD traffic and P_2 is used to create TBD traffic. TBDs and PSDs are randomly located in the region. As the coverage area of a single device is equal to 12.57 m^2, the total observation area of the TBDs is 754.2 m^2 which is larger than the observation area so there will be 3.77 TBD in each TBD location. This guarantees conflicts. We dynamically calculated interference coefficient during the simulation. All devices are covering an area of $\pi \times R^2$ in the space. However, the devices can cover the same area so the actual covered area equals to $\psi \times \pi \times R^2$. ψ parameter is theoretically the actual covered area divided by $\pi \times R^2$. For simplicity in simulations instead of the formula presented in Eq. (5), we used an estimated equation that is presented in

$$\begin{aligned} \psi_i &= \frac{\Sigma_j 0.5 \times log_{2R}(D_{ij}) \times 1(x < 2R) + 0.5 \times 1(x > 2R) + 0.5}{N_{active}} \\ \psi &= \frac{\Sigma_i \psi_i}{N_{active}} \ \forall i,j \in N \end{aligned} \tag{13}$$

where x is the distance between devices. 13 is reached from the idea that if the distance between two nodes is greater than 2R than the interference coefficient will be equal to 1 and if the distance between this two nodes is equal to 0 then the interference coefficient will be equal to 0.5. As can be seen from this definition, interference coefficient is calculated for 2 devices case. The calculation is performed each active device pair and the general ψ parameter is calculated by taking the average of each device couple (see Table 3).

4.1. Performance verification

The success of the framework is measured in terms of total actually covered area in the region and actual covered area per energy. In the simulation, we used 60 TBD devices and 10 PSD devices with battery power of 50 W each. The change of total actual covered area is presented in Fig. 5. The minimum energy consumption, sleep mode energy consumption, is equal to 0.992 W. With a simple calculation it can be seen that the maximum life time without a energy efficient scheduling, which can only be achieved by always staying in sleep state, is equal to 50.4 h. The region is 200 m^2 and it is the maximum coverable area. The probability of observing an event is static and equal to 95%, indicating that we expect to observe the 95% of overall events that happened in the region. As it is presented in Fig. 5, with our proposed framework, for the first 125 h, the covered area is equal to 200 m^2 which covers 100% of the region. As our objective is the coverage of 95% of the overall network, which is equal to 190 m^2, we accepted this as a border for our durability calculations which happened in 133 h. After 160 h, the covered area falls to 20 m^2 which is the 10% of the coverable region. This is our lifetime border. This fall on the covered area is due to the lack of sensors. As told earlier, we used 60 TBD devices. According to the Fig. 5, the durability is increased 150% and also the lifetime is increased 220%.

Table 3
Network parameters.

Parameter	Value
A	200 m^2
R	2 m
N_{TBD}	60
N_{PSD}	10
E	50 W

Please cite this article in press as: Ö.U. Akgül, B. Canberk, Self-Organized Things (SoT): An energy efficient next generation network management, Comput. Commun. (2014), http://dx.doi.org/10.1016/j.comcom.2014.07.004

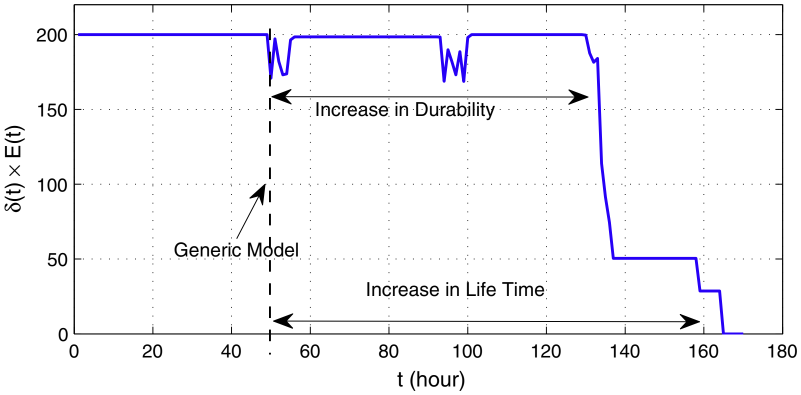

Fig. 5. Increase in durability.

In Fig. 6, the actual covered area per energy is presented. Our objection is to keep this value as high as possible as it show how energy efficient our system is. If this term is low that indicates that we are keeping some unnecessary devices in active state. However, as the time increases and as the number of devices decreases we expect a smooth fall as some devices are keeping in active mode to fulfill the observed event rate. Fig. 6 shows that the proposed system is working in a stable state around 21. When Figs. 5 and 6 is investigated together it also proves our previous statement about the decreasing coverage rate. As the coverage rate per watt does not changed the only reason of this coverage fall in Fig. 5 can only be caused by the lack of devices. This results prove that our proposed framework fulfills the expectations and increases the both coverage area and durability of the network.

4.2. Effects of connection requests

Final observations are done for the effects of the traffic load. By traffic load we mean the integration of three concept, the number of triggers for TBDs, the number of server requests for PSDs and the life signals of PSDs. So far we created these requests using a probabilistic approach. With probability p_1 we created the server request and with probability p_2 we created a trigger. So far these traffic was created statically with $p_1 = 0.5$ and $p_2 = 0.2$. However, to come up a complete conclusion we have to observe the effect of changes on the traffic load on the framework. For this reason we observed the total coverage area and the coverage area per energy for different traffic loads. To change the traffic we changed the probabilities of requests and triggers, p_1 and p_2. The results for different p_1 values are presented in Figs. 7 and 8. As can be seen in Fig. 7, the increase in PSD traffic does not caused a significant change in total coverage area during the 130 h. However, the effect of PSD traffic can be seen between 130 and 165 h. As can be seen as traffic rate of PSDs increases more TBDs are staying active state in this time interval. This is because of the privilege of PSDs. In Fig. 8, the performance of SoT structure under different traffic rates is presented. Just like Fig. 7, in Fig. 8, there exist no significant difference in terms of performance until 130 h. However, after that time, we can see that higher PSD traffic cases causes more efficient systems. From these two result it is possible to say to increase the efficiency during the lifetime, higher PSD traffics are better. However, as PSD traffic causes an increase in the event miss to observe more event traffic must be low.

In Fig. 9, the changes of total covered area with the TBD traffic is presented. As can be seen from this figure, higher traffics for TBDs causes a decrease in the durability of the network. In Fig. 10, the change of SoT performance with TBD traffic is presented. From Fig. 10, it is possible to say even though higher traffic rate decreases the durability of the SoT, it is obvious that it causes a performance increase for the time interval between 130 and 165 h.

4.3. Effects of node density

In the previous sections, it was stated that the TBD count is a strict border in the lifetime of the network. The proposed network schedules the TBD states to achieve the maximum network durability with the highest event observation rate. Based on this definition the effect of node density on the network durability is also important. In the previous simulations, the number of TBDs (nodes) is assumed to be static and equal to 60. In Fig. 11, the effect

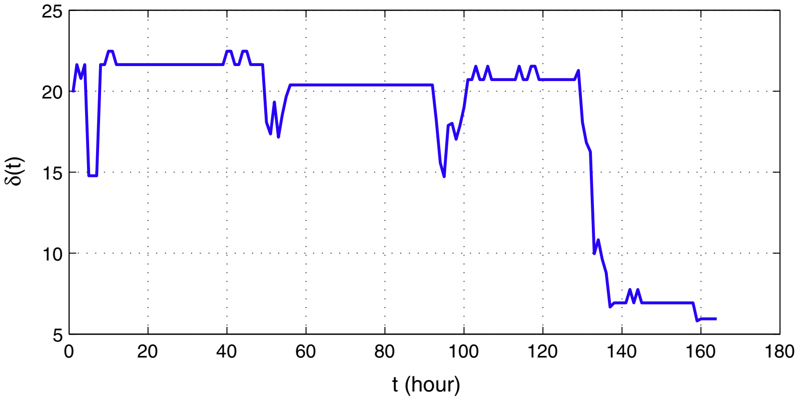

Fig. 6. Total covered area per energy.

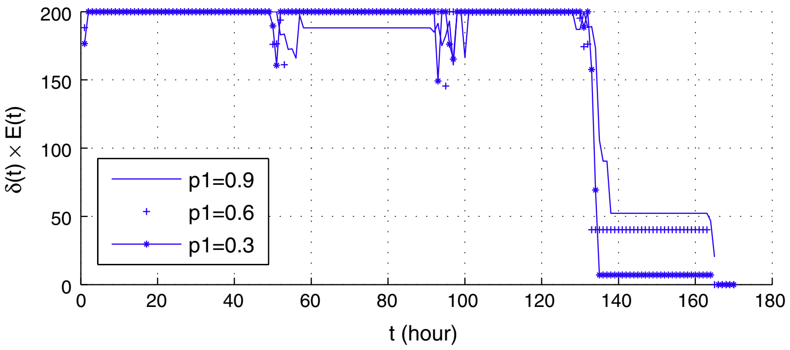

Fig. 7. Total coverage area change with PSD traffic.

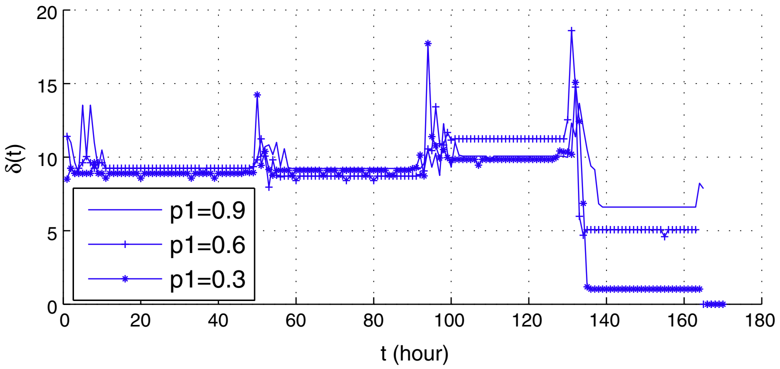

Fig. 8. Total covered area per energy change with PSD traffic.

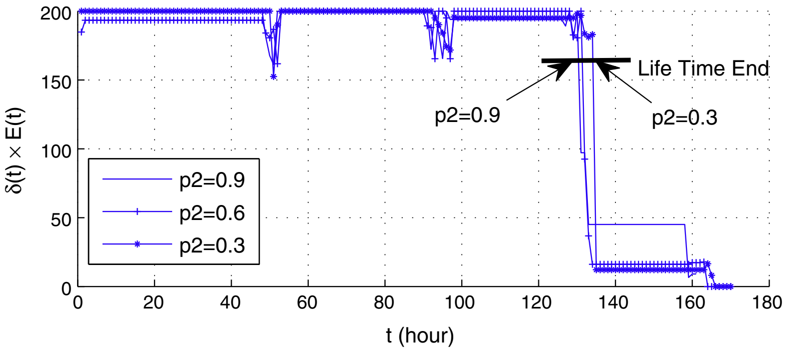

Fig. 9. Total coverage area change with TBD traffic.

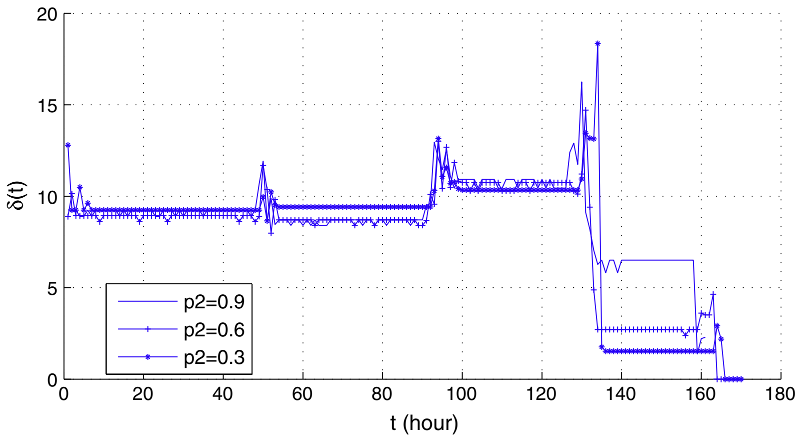

Fig. 10. Total covered area per energy change with TBD traffic.

Please cite this article in press as: Ö.U. Akgül, B. Canberk, Self-Organized Things (SoT): An energy efficient next generation network management, Comput. Commun. (2014), http://dx.doi.org/10.1016/j.comcom.2014.07.004

of node density on overall network durability is investigated. The number of nodes is changed from 60 to 160, which is a 167% increase in the number of nodes. However, this huge increase in the number of nodes resulted in 62.5% increase in durability of the network. Additionally, the increment trend of the durability with the number of nodes is more likely a logarithmic increment than a linear increment. It can be modeled as a logarithmic function with an upper bound at the infinity.

Theoretically, as each device covers an area of 12 m^2, 18 TBDs would be enough to cover the whole area and by this way the expected durability for a network containing 160 TBDs will be around 440 h. However, the simulations shows that the measured durability will be 280 h, which is smaller than the expected value. This is because of the two special attribute of the proposed framework. First one is the proposed frameworks effort to cover the whole area. To cover the whole area the number of activated devices is higher than the theoretically calculated number, which decreases the expected durability of the network. The second reason is the density of the nodes. As the number of nodes increases, their unique coverage area decreases. As the unique coverage area decreases, the ψ parameter decreases which leads to an increase in the number of active devices. Due to this reason, the active nodes per time increases massively and this decreases the durability of the IoT network.

4.4. Discussion

During the performance evaluation part, the proposed framework is investigated in terms of energy efficiency, coverage area, traffic rate and node density. Based on these evaluations and the algorithmic complexity of the proposed framework the following results can be reached.

- The proposed framework can perform 150% increase in durability and 220% increase in the overall lifetime of the network. This shows that this framework can maintain the energy efficiency demand of IoT devices.
- The proposed framework can perform the selection of remote TBDs, which increases the detection of distinct events. This algorithm depends on the idea of continuous and discrete events. As continuous events, e.g. motion, follows a sequence, the algorithm tries to increase the detection of discrete events.
- Unlike the existing frameworks, the proposed model considers the three design parameters, i.e. total coverage area, distribution of devices in 2D space and distribution of events in 2D space, to create the optimal schedule.
- The proposed framework is effective for dense IoT networks with low traffic rate.
- Increase in node density has a negative impact on the durability trend. Even though, dense network topology (5 device per m^2) is necessary for network efficiency, a huge increase in density (10 device per m^2), causes a decrease in energy efficiency due to ψ parameter.

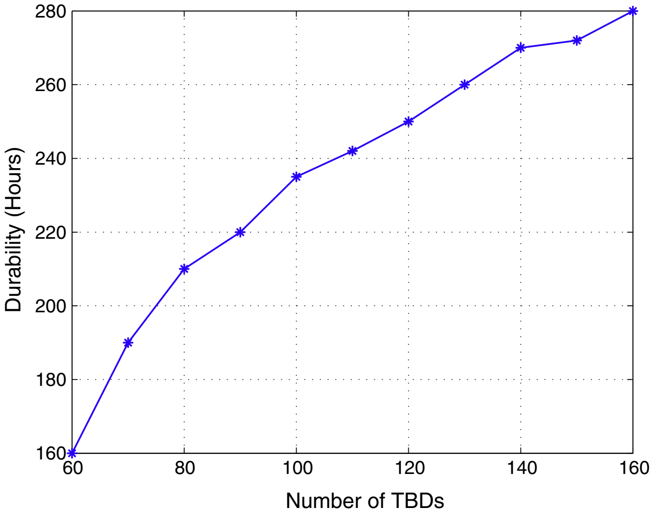

Fig. 11. Durability change with number of nodes.

5. Conclusion

In this study, A green self-organized IoT framework which we call SoT is presented that optimizes the energy consumption of the *things* and increases the life times of the *things*. In the simulations we first evaluate the success of the framework in terms of durability and coverage area per energy and we obtained 150% increase in durability with 220% increase in the overall lifetime. We also showed that our framework stabilizes the energy consumption by putting unnecessary devices into sleep mode. We used a energy efficient self-scheduling algorithm that sends unnecessary devices into sleep mode if it is possible to cover the same area with lower number of devices Finally the effects of traffic load are observed and seen that for different traffic loads, our framework guarantees the durability.

References

[1] L. Atzori, A. Iera, G. Morabito, From smart objects to social objects: the next evolutionary step of the internet of things, IEEE Commun. Mag. 52 (1) (2014) 97–105.

[2] G.F. Hurlburt, J. Voas, K.W. Miller, The internet of things: a reality check, IT Profess. 14 (3) (2012) 56–59. May–June.

[3] D. Miorandi, S. Sicari, F. De Pellegrini, I. Chlamtac, Internet of things: vision, applications and research challenges, Ad Hoc Networks 10 (7) (2012) 1497–1516.

[4] H.D. Ma, Internet of things: objectives and scientific challenges, J. Comput. Sci. Technol. 26 (6) (2011) 919–924.

[5] M. Zorzi, A. Gluhak, S. Lange, A. Bassi, From today's INTRAnet of things to a future INTERnet of things: a wireless- and mobility-related view, IEEE Wireless Commun. 17 (6) (2010) 44–51.

[6] V. Foteinos, D. Kelaidonis, G. Poulios, P. Vlacheas, V. Stavroulaki, P. Demestichas, Cognitive management for the Internet of Things: a framework for enabling autonomous applications, IEEE Veh. Technol. Mag. 8 (4) (2013) 90–99.

[7] Yongsheng Ding, Yanling Jin, Lihong Ren, Kuangrong Hao, An intelligent self-organization scheme for the Internet of Things, IEEE Comput. Intell. Mag. 8 (3) (2013) 41–53.

[8] Jia-Ming Liang, Jen-Jee Chen, Hung-Hsin Cheng, Yu-Chee Tseng, An energy-efficient sleep scheduling with QoS consideration in 3GPP LTE-advanced networks for Internet of Things, IEEE J. Emerg. Selec. Top. Circ. Syst. 3 (1) (2013) 13–22.

[9] S. Tozlu, M. Senel, Wei Mao, A. Keshavarzian, Wi-Fi enabled sensors for internet of things: a practical approach, IEEE Commun. Mag. 50 (6) (2012) 134–143.

[10] S. Balasubramaniam, J. Kangasharju, Realizing the Internet of Nano Things: challenges, solutions, and applications, IEEE Comput. 46 (2) (2013) 62–68.

[11] B. Atakan, O.B. Akan, Immune system based distributed node and rate selection in wireless sensor networks, bio-inspired models of network, Inf. Comput. Syst. 1st (11–13) (2006) 1–8.

[12] Mugen Peng, Dong Liang, Yao Wei, Jian Li, Hsiao-Hwa Chen, Self-configuration and self-optimization in LTE-advanced heterogeneous networks, IEEE Commun Mag 51 (5) (2013).

[13] M. Bala Krishna, M.N. Doja, Self-organized energy conscious clustering protocol for wireless sensor networks, in: 14th International Conference on Advanced Communication Technology (ICACT), 2012, 19–22 February 2012, pp. 521–526.

[14] O.G. Aliu, A. Imran, M.A. Imran, B. Evans, A survey of self organisation in future cellular networks, IEEE Commun. Surv. Tutorials 15 (1) (2013) 336–361.

[15] L. Angrisani, M. Bertocco, D. Fortin, A. Sona, Experimental study of coexistence issues between IEEE 802.11b and IEEE 802.15.4 wireless networks, IEEE Trans. Instrum. Measure. 57 (8) (2008) 1514–1523.

[16] H. Yan, S.A. Watterson, D.K. Lowenthal, Kang Li, R. Krishnan, L.L. Peterson, Client-centered, energy-efficient wireless communication on IEEE 802.11b networks, IEEE Trans. Mobile Comput. 5 (11) (2006) 1575–1590.

Please cite this article in press as: Ö.U. Akgül, B. Canberk, Self-Organized Things (SoT): An energy efficient next generation network management, Comput. Commun. (2014), http://dx.doi.org/10.1016/j.comcom.2014.07.004