



HAL
open science

Fuzzy-based Objective Function for Routing Protocol in the Internet of Things

Philippe Fabian, Abderrezak Rachedi, Cedric Gueguen, Stéphane Lohier

► **To cite this version:**

Philippe Fabian, Abderrezak Rachedi, Cedric Gueguen, Stéphane Lohier. Fuzzy-based Objective Function for Routing Protocol in the Internet of Things. GLOBECOM, Dec 2018, Abu Dhabi, United Arab Emirates. 10.1109/GLOCOM.2018.8647969 . hal-02290054

HAL Id: hal-02290054

<https://hal.science/hal-02290054>

Submitted on 17 Sep 2019

HAL is a multi-disciplinary open access archive for the deposit and dissemination of scientific research documents, whether they are published or not. The documents may come from teaching and research institutions in France or abroad, or from public or private research centers.

L'archive ouverte pluridisciplinaire **HAL**, est destinée au dépôt et à la diffusion de documents scientifiques de niveau recherche, publiés ou non, émanant des établissements d'enseignement et de recherche français ou étrangers, des laboratoires publics ou privés.

Fuzzy-based Objective Function for Routing Protocol in the Internet of Things

Philippe Fabian*, Abderrezak Rachedi*, Cédric Guéguen†, Stéphane Lohier*

*University Paris-Est, LIGM (UMR8049), UPEM, F-77454, Marne-la-Vallée, France

Emails: [philippe.fabian][rachedi][lohier]@u-pem.fr

†University of Rennes 1, IRISA (UMR6074), UR1, F-35042, Rennes, France

Email: cedric.gueguen@irisa.fr

Abstract—The amount of data represented by the Internet of Things (IoT) is continually growing and is expected to reach more than one third of the total amount of Internet traffic by 2020. Wireless sensors pose a great challenge because of the varying radio conditions and the limited energy and computational capabilities they have. This is why we have developed an Objective Function (OF) that uses fuzzy logic to dynamically adapt to variable environments in wireless networks. Simulation results will show that the proposed algorithm increases performance compared to other solutions by up to 15% in terms of throughput and by up to 14% in terms of Packet Delivery Ratio (PDR) without compromising energy consumption.

Index Terms—Routing protocol, Fuzzy-based, Objective Function, Internet of Things.

I. INTRODUCTION

The Internet of Things (IoT) is composed of connected devices present in the environment, vehicles, building automation or home appliances. These sensing devices are used to improve specific environments, safety and everyday life by providing awareness and accurate measures of the surroundings and by transporting and processing relevant data. Use cases are broad and two examples include : measuring sunlight, humidity or air quality at different places in a location to monitor the environment (Wireless Sensor Network - WSN), and detecting electricity activity to control energy consumption (smart grid). Low-power and lossy networks (LLNs) are characterized by such connected wireless devices having low processing power and memory, the need to save energy consumption and a high loss ratio. Routing Protocol for Low-Power and Lossy Networks (RPL) [1] is a widely used network layer protocol in LLNs. It is a IPv6 Low-power Wireless Personal Area Network (6LoWPAN) proactive vector distance protocol, thus it is Internet-compatible. The specific case of WSNs in the IoT poses a harder challenge to routing because of dynamic radio conditions and interference in the wireless environment. Managing mobility [2] further increases the complexity of the problem and is beyond the scope of this article. Given that global traffic from IoT devices is rapidly growing [3], solutions must be found by the dawn of 2020 to adapt to this new context.

In WSNs, one or several metrics such as delay, energy consumption or reliability must be respected. For instance, in the case of a proximity sensor in a vehicle, delay is of paramount importance to avoid an accident, or a sensor inside a wall needs its energy consumption to be as low as possible.

RPL is designed to take into account up to several metrics [4] (see section II-B) to optimize one or more of them. However, it is difficult to find one common solution to different problems because optimizing one metric might cause another metric to be worse. More than that, a WSN can be a very dynamic environment in terms of radio conditions, connectivity and mobility, residual energy and specific needs of nodes. This can cause one formula to perform well in some scenarii but to perform poorly in other scenarii or under changing conditions (i.e. all batteries get very low).

Several solutions exist to address the issue. Dynamically optimizing MAC parameters has been done, like the ADAPT algorithm [5] (see section II-A). More specific to RPL, this [6] proposition uses queue utilization to load balance packets in the network in order to improve Packet Delivery Ratio (PDR). Another algorithm [7] (see section II-A) that considers all available metrics to focus on the most problematic one to improve performance has also been developed and yields good results (see section IV). However, these algorithms ultimately focus on either one metric, aim at improving one aspect of the network's performance or do not take energy into account. As a WSN can be composed of heterogeneous nodes with different needs, a dynamic solution taking into account those different needs is required to further improve network performance and to be able to better withstand worldwide traffic in the near-future [3]. It is also possible to focus on optimizing signaling [8] but the aim of this article is on improving routing.

Unlike most existing approaches, we propose a dynamic strategy to tackle with the problem at hand by introducing fuzzy logic. The general idea is to overlook all available metrics and take a decision based on all of them. The result will vary based on the value of each metric to prevent any of them to reach a critical level. In any case, all metrics will always be taken into account to avoid having one metric dominating all the others. This fuzzification process allows the algorithm to finely adapt to a dynamic environment no matter what metrics are available. In order to evaluate this new strategy, we have implemented it on the well-known RPL. Simulation results will show that this solution outperforms several algorithms. Compared to [7], our proposition yields better results in terms of PDR by up to 14% (section IV-B) and throughput by up to 15% (section IV-C).

The remaining of this paper is organized as follows : the

state of the art is presented in section II, the proposed solution is detailed in section III, performance evaluation is brought to light and discussed in section IV and section V concludes this paper.

II. RELATED WORK

As mentioned in section I, several solutions exist to approach the problem of routing in WSNs. One possible way to regroup these algorithms is according to the number of considered metrics. The algorithms can focus on :

- using one metric only;
- comparing several metrics but using only one in the end;
- combining several metrics.

Focusing on one metric is useful to reach one specific goal like energy saving or minimum delay. However, looking at only one metric might not be the most efficient way to improve performance of the whole network and different nodes might have different needs. Furthermore, if considering only one metric yields to more than one possible path it could be complicated to decide which path to choose from. To resolve these issues, one solution is look at several metrics to select the one that will be used in the end. As only one metric is ultimately used to find the best path, the optimization problem remains simple in terms of complexity and having a look at all metrics allows to take a better decision. Finally, it is possible to overlook all metrics and take all of them into account. The optimization problem becomes more complex but enables maximum control over the path to select by choosing for any metric which one is to be considered and to what extent.

A. Existing Objective Functions

In RPL (see section II-B), two objective functions are defined : the Objective Function Zero [9] (OF0) and the Minimum Rank with Hysteresis Objective Function [10] (MRHOF). OF0 simply uses the number of hops to compute the best path and MRHOF is either based on ETX or energy. As MRHOF considers only ETX or energy, it can hardly adapt to a situation where other aspects like throughput or both ETX and energy must be taken into account.

An example of an algorithm that uses several metrics to focus on one aspect of performance is the ADaptive Access Parameters Tuning (ADAPT) algorithm [5]. It is used in WSNs to improve reliability by measuring the delivery ratio of packets and modify MAC parameters in order to reach the desired PDR. In ADAPT, the delivery ratio is measured by comparing the number of sent messages against the number of received corresponding acknowledgments. The ratio is smoothed over time to avoid sudden changes of having too much impact. There are two thresholds in this algorithm : when the lower one is reached, it means the delivery ratio is not good anymore and the algorithm is run to correct the problem. The upper bound is used to avoid unnecessary runs of the algorithm which would consume too much energy. These two thresholds are set to specific values to achieve the best performance. Although ADAPT saves energy by optimizing MAC parameters, it is not designed to primarily focus on energy saving. This can be an

issue in a network where energy saving is more important than reliability.

Another example is the Non-Linear Objective Function [7] (NL-OF) which can use any number of additive metrics. Each metric value is divided by a predetermined Quality of Service (QoS) requirement used to normalize it. In order to improve the score of the objective function, all normalized values are compared to one another and the worst one is used. Two advantages to this approach are that it allows to focus on the most critical metric and the algorithm is rather simple to implement. That is, one has to only compare the normalized metrics and pick up the highest value. However, in the case where one node on a given path has a particularly bad metric value, only this metric will be considered for all nodes that will connect to this node. For instance, if one node has a very high energy consumption value, all of its children will compute their path according to energy consumption regardless of any other metric, even if these nodes have a near-full battery level.

It is important to note that any metric that is not included in computing the rank of a given node will cause this metric to simply be ignored. That is, if for instance the ETX is high enough to be the only metric considered to compute the rank, the energy will be ignored and a node with a good ETX value will have its battery drained quickly.

B. RPL overview

RPL uses the concept of rank to organize the topology. A node will connect to the neighbor with the smallest rank of the chosen instance. The rank is influenced by the given Objective Function (OF) and it is up to the current node to compute its rank given its parent's rank. An OF uses one or more metrics or constraints to compute a score which will influence the rank. The smaller the rank the better, thus a good metric/constraint will yield a low rank. A constraint has a maximum allowed value and if this value is reached, the link is considered unusable. For instance, a sensor in a time-critical application will require the delay not to be higher than a certain value. RFC 6551 [4] defines several metrics : node energy, hop count, throughput, latency and Expected Transmission Count [11] (ETX) amongst others. ETX is the number of total (re-)transmissions needed to send one packet. One can use any of these metrics or constraints to influence to rank value. When a node needs to join the network, it may broadcast a special packet called DODAG Information Solicitation (DIS) to request information from nearby nodes connected to the network (the potential parents). These potential parents will then broadcast a DODAG Information Object (DIO) to inform neighboring nodes about the current instance.

RPL uses a tree-like topology called Directed Acyclic Graph (DAG) rooted at one or several sinks. For each sink, at least one Destination Oriented Directed Acyclic Graph (DODAG) is needed. A sink might also act as a border router, connecting the topology to some wider network like the Internet (as RPL is IPv6 compatible). The exact method on how to build an instance is purposely not specified in RFC 6550 to allow the developer the freedom to adapt the OF to specific requirements.

There are two main advantages to RPL. The first is that the resulting tree-like topology allows for simple routing, as most of the time sensor nodes only need to send their data upwards to the root. The second advantage is the fact that a node needs only to probe its neighbors for their rank, so almost no computations and only a small amount of memory are needed on all nodes but the root. RPL is thus a fine choice for a many-to-one configuration. The challenge is the design of the one or many OFs to obtain the best performance and/or to reach the desired goal. As aforementioned, there are plenty of approaches that can be used but taking into account all metrics to optimize all aspects of a network is not something that has been done by many.

III. FUZZY-BASED OBJECTIVE FUNCTION

In this section, we present the proposed Fuzzy-based Objective Function (F-OF). Fuzzy logic allows a decision-making process to be more fine-grained. Instead of a binary result (i.e. TRUE or FALSE), it is possible to be somewhat in-between those two opposite values. This approach is very powerful in WSNs because it enables to take into account several metrics up to a certain degree, which dovetails well with the dynamic nature of the environment nodes are placed in. However, some metrics are such that if they have a too high value the link must be considered unusable. This is the case, for instance, if a node's energy consumption has reached the node's total energy capacity (the battery is flat). Similarly, we might as well consider the occurrence where the metric's value is very small : in the case of the battery, as long as the residual energy is above a given threshold it could be drained freely in order to improve the system performance.

A. General case

Let us consider the set N , a network composed of n nodes. Each node (vertex) $V_j \in N, j = 0, 1, \dots, n$ may use any available metric. If k is the number of metrics, we define the vector M_j containing the k metrics available to node V_j , that is :

$$M_j = [m_0, m_1, \dots, m_k] \quad (1)$$

In order to decide what metric will be considered and to what extent, we define four bounds (three intervals) such that any metric value m_i ($i = 0, 1, \dots, k$) is in one and only one of the three intervals :

- m_i^{min} : the lowest possible value of m_i ;
- m_i^{lo} : the lower bound to start considering m_i ;
- m_i^{hi} : the upper bound before considering only m_i ;
- m_i^{max} : the maximum possible value of m_i .

A metric within the first interval (between m_i^{min} and m_i^{lo}) is not considered to matter and will be ignored. A metric within the second interval (between m_i^{lo} and m_i^{hi}) will be taken into account and will influence the result. Finally, a metric within the third interval (between m_i^{hi} and m_i^{max}) will take over the rest and only this metric will be considered. This is useful if

a strong constraint is needed. Hence, the fuzzification is done through the coefficient α_i of any metric :

$$\alpha_i = \begin{cases} 0 & \text{if } m_i < m_i^{lo} \\ 1 & \text{if } m_i^{lo} \leq m_i \leq m_i^{hi} \\ +\infty & \text{if } m_i^{hi} < m_i \end{cases} \quad (2)$$

We can now define the score S_j of any node. The score is the total of all metric values such that a lower score is preferable to a higher score. A node will be selected as parent by a neighbor if it has the lowest score of all neighbors. However, we first need to normalize the metric values before adding them together. The normalization vector Γ will serve this purpose :

$$\Gamma = [\gamma_0, \gamma_1, \dots, \gamma_k] \quad (3)$$

As one specific metric has the same range of values no matter the considered node, there is only one normalization vector (i.e. not one per node). We have :

$$S_j = \sum_{i=0}^k \alpha_i \cdot \left(\frac{m_i}{\gamma_i} \right) \quad (4)$$

As aforementioned, equation 2 allows to completely ignore a metric ($\alpha_i = 0$) or to max out the score if a metric is too big ($\alpha_i = +\infty$). The coefficient α_i will not impact a metric within the second interval ($\alpha_i = 1$). It is of course possible to fine-tune the intervals' value to fit specific needs. For instance, if some node V_p needs its residual energy m_e ($m_e \in M_p$ and $0 \leq e \leq k$) to always be considered, we will write :

$$m_e^{min} = m_e^{lo} \quad (5)$$

so even on a completely full battery, the residual energy will be taken into account. Likewise, if no constraint is needed on one specific metric, it is possible to define the upper bound such as :

$$m_e^{hi} = m_e^{max} \quad (6)$$

In this case, although the score value is not explicitly set to $+\infty$, this metric might reach very high values such that the other metrics will be negligible. However, even if this is the case, the rise of this metric will be more progressive compared to setting it directly to $+\infty$. Finally, we can write the complete objective function P_ζ that will determine for any node $V_j \in N$ the best parent amongst the node's neighbors Z (Z eta) $\subset N$:

$$P_\zeta(V_j) = \min_{\zeta \in Z} (S_\zeta) \quad (7)$$

That is, P_ζ is the parent with the lowest score as defined in equation 4. Z is a strict subset of N because one node will never consider itself as a potential parent. The vector M (eq. 1) can be adjusted to dovetail to the number of available metrics to impact equation 7 accordingly.

B. ETX and energy-aware case

In this paper, three metrics amongst the ones defined in [4] are considered : hop count, ETX and energy remaining (or residual energy). If nodes were different, especially in terms of battery capacity and/or power consumption a more thorough definition of energy would be required. For instance, if two nodes are identical on all aspects except one has twice as much battery capacity as the other one, considering only the remaining power in percentage is not adapted because the node with more battery power could be more solicited due to it having more energy in storage. Also, metrics are additive so they are computed on the whole path and not only on the current node or link.

The proposed OF, Fuzzy-based Objective Function (F-OF), overlooks all metrics and uses a threshold (see section IV-A) to determine if a node has enough remaining energy. Three cases are possible (eq. 2) :

- The current node has a battery level greater than the given threshold;
- The current node's battery is below the given threshold;
- The current node has an empty battery.

In the first case, the current node will not take its battery level into account. Using only ETX will yield to better performance in terms of delay, PDR and throughput by sacrificing battery power. The second case takes into account the ETX and the remaining battery level of the current node to compute the rank. This case is very important because it will prevent a node with a very good ETX metric to be always solicited and have its battery drained rapidly. In the third case, it is not possible to use this node anymore. Normalizing the metrics can be tricky and one solution is to set the acceptable lower and upper bounds (m_i^{lo} and m_i^{hi} respectively) beforehand. The values of the bounds are discussed in section IV-A. The running of F-OF is illustrated in algorithm 1. It allows a node to search

Algorithm 1 The Fuzzy-based Objective Function rank computation.

```

1:  $etx \leftarrow 0$ 
2:  $energy < -currentNode.getResidualEnergy()$ 
3:  $rank \leftarrow 0$ 
4: if  $battery == empty$  then
5:    $rank \leftarrow +\infty$ 
6: else if  $energy > upperBound$  then
7:    $rank \leftarrow etx$ 
8: else
9:    $rank \leftarrow (etx * energy)$ 
10: end if
11: return  $(parentRank() + rank)$ 

```

for the parent with the best rank amongst all neighbors (eq. 7) by only considering ETX if its battery is almost full (lines 6-7 in algorithm 1). In the case where the battery is too low (lines 8-9), residual energy will be considered ($\alpha_e = 1$) for this node (eq. 4). The special case depicted on lines 4-5 in algorithm 1 is when the battery is completely empty, thus the node can

Parameter	Value
Topology size	150 m · 90 m
Simulation time	1 hour
Traffic model	1 packet/s
Packet count	3300 packets
Packet size	16 Bytes
Tx range	30 m
Rx rate	75 %
Batteries capacity	1080 J

TABLE I: Basic simulation parameters.

not be used at all ($\alpha_e = +\infty$). In this algorithm the rank is computed by adding the rank (cost) to the parent (variable *rank*, defined on line 3) and the (additive) rank of the parent (*parentRank()*, line 11). After going through all neighbors, the current node will “hook” to the network by becoming the child of the neighbor with the lowest returned rank (see eq. 7).

IV. PERFORMANCE EVALUATION

A. Simulation setup

The contribution has been tested using RPL on the well-known ContikiOS which is an embedded operating system that can be uploaded on sensor nodes. For this simulation, contikiOS has been run on a Linux computer using the Cooja emulator. Wismote nodes have been used for all devices. The power consumption of all nodes is based upon the datasheets of the CC2520 radio transceiver [12] and MSP430 microcontroller [13]. Nodes are also considered to be powered by 2 AAA batteries (3.0 V) which allows to determine the total energy capacity. Table I summarizes the base parameters used for all simulations.

We studied the effects of changing several of these parameters, one at a time. The nodes have been placed on a 150 m per 90 m area (see fig. 1). The studied topology is purposely not symmetric so different paths are possible from any source to the sink and choosing one or another is not obvious. For instance, one path is set to be the shortest in terms of hops for several sources but if all traffic is routed through this path congestion will occur at higher loads. Other longer paths exist so it could be better to choose one of them instead if the former path is heavily congested. The simulation time is one hour to allow the system to stabilize, yielding to more accurate results. The sources are set to send 1 packet/s and the packet count (3300 packets) is set to be less than the simulation time (3600 s) to allow sources to have time to resend packets in case of bad transmissions. The traffic load ranges from 1 packet per second per source to 6 packets per second per source. The ETX has been changed by placing nodes closer or farther away from one another. The range between two neighbor nodes goes from 10 m to 30 m, the latter being the maximum transmission range with the worst ETX value.

Given that the MSP430 and the CC2520 are designed for low power consumption, the running time of the simulations would have to be very long for the impact of battery depletion to be visible. This is why the capacity of batteries has been divided by a factor 10 to cause a one hour simulation to drain a significant (more or less 20%) amount of all nodes

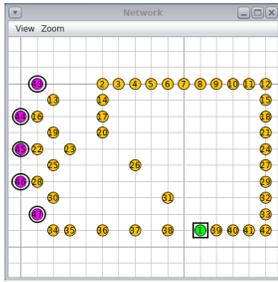


Fig. 1: The studied topology. The rectangle surrounds the sink, ellipses surround the sources and the other nodes are intermediate sensors. One grid square is 10 m · 10 m in size.

total capacity in order to study the effect of energy-saving algorithms. In view of this static energy capacity, we chose to start taking energy into account when the battery level is down by 15% ($m_{energy}^{lo} = 15$). Accordingly, m_{energy}^{hi} and m_{energy}^{max} are set to 100 (the battery is empty), at which point the rank is maximized. The ETX and energy metric values are in the \mathbb{N} set so their minimum value is 0. The ETX is always considered ($m_{etx}^{min} = m_{etx}^{lo}$) and is not constrained ($m_{etx}^{hi} = m_{etx}^{max}$).

The performance evaluation is achieved by comparing the different studied algorithms in terms of PDR, throughput and energy consumption. Given that the resulting PDR varies greatly between the different algorithms, the delay is not necessarily significant to compare them to one another because of the different sampling sizes (i.e. varying number of packets reaching the destination), leading to inaccurate delay values in some cases. Four algorithms have been evaluated :

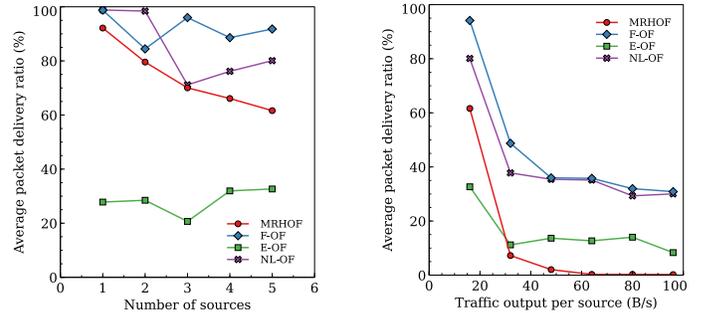
- 1) MRHOF is the OF as described in RFC 6719 [10]. It uses the squared ETX to influence the rank;
- 2) F-OF is our proposed solution;
- 3) E-OF is the Energy Objective Function that bases its rank solely on the energy metric;
- 4) NL-OF is the Non-Linear Objective Function [7] (see section II).

As results will show, E-OF has almost always the worst results¹ of all except in terms of energy where it performs the best. MRHOF mostly has average results. NL-OF and F-OF are comparable in terms of energy consumption and F-OF yields better results in terms of PDR and throughput.

B. Packet Delivery Ratio

Figure 2 plots the PDR of the four algorithms given the number of sources (fig. 2a) or the traffic density (fig. 2b). The packet delivery ratio tends to decrease as the number of sources increases because more packets are generated : when there is one source, 3300 packets are generated on the whole topology and when there are six sources, 19800 ($6 \cdot 3300$) packets are generated in total. Adding sources one by one also modifies the topology and because of this, all functions have one point

¹Note that this objective function is only intended to serve as a lower bound on energy consumption and is not studied to have the lowest possible energy consumption.



(a) The PDR against the number of sources.

(b) The PDR given different traffic densities.

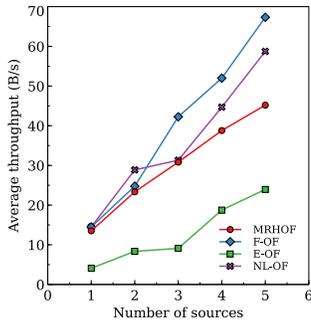
Fig. 2: The effect of changing either the number of sources or the traffic rate on PDR.

where the PDR drops significantly. The drop is seen on F-OF when there 2 sources and when there are 3 sources for the other functions. MRHOF is only slightly affected. MRHOF always only considers ETX which does not change through time, hence it always chooses similar paths when the number of sources change and the PDR drops with increasing congestion. E-OF, NL-OF and F-OF additionally rely on energy and/or the number of hops. The changing level of energy will lead certain nodes to select different parents through time. These parents will then have different additive metric values and their resulting rank will change. Thus, because of the dynamic nature of this selection process, E-OF, NL-OF and F-OF are more affected than MRHOF when adding sources. In figure 2a, E-OF has the lowest PDR with values starting at 27.8% and up to 32.7%. MRHOF drops from 92.2% (1 source) down to 61.6% when there are 5 sources. NL-OF and F-OF both start at 98.8% of PDR when there is 1 source and end up at 80.0% and 91.8% respectively. Thus, F-OF is able to perform better by approximately 30% and 10% compared to MRHOF and NL-OF respectively.

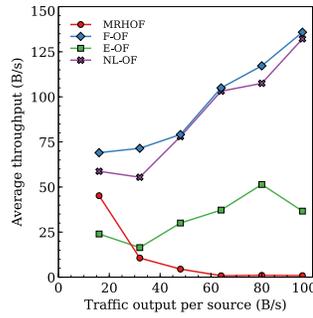
In figure 2b, the impact of the traffic density is observed. The more traffic, the harder it is for all algorithms and performance drops significantly. MRHOF tends to keep the same paths through the simulation because of the static ETX. This causes a lack of load balancing and the impact is important : MRHOF performance is very low (PDR of 2%) when traffic is heavy. E-OF performs slightly better because the energy saving will lead to some load balancing. Indeed, if one node is often chosen as a parent its battery will be drained faster, its rank will increase and it will no longer be a parent for at least some time, giving it some respite. NL-OF and F-OF have the best PDR with respectively 80.0% and 94.1% when the traffic is low and 30.0% for NL-OF and 30.1% for F-OF at the highest traffic load. When the traffic is high, the difference between NL-OF and F-OF is not significant.

C. Throughput

On figure 3a, the throughput increases rapidly as there are more sources because more packets are generated on the topology. However, the rate of throughput increase is different

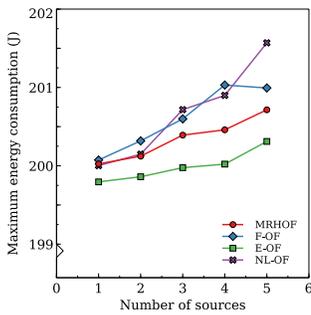


(a) The topology throughput against the number of sources.

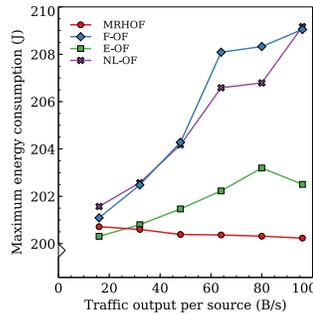


(b) The topology throughput given different traffic densities.

Fig. 3: Changing the number of sources or the traffic density affect the throughput differently.



(a) The maximum energy consumption with varying number of sources.



(b) The maximum energy consumption for different traffic loads.

Fig. 4: Study of changing either the number of sources or the traffic load on the maximum energy consumption of any node (Note the ordinates axis-cut).

for each algorithm. This rate is related to the PDR (fig. 2a) as a higher PDR means more packets make it to the destination and the throughput will be higher. E-OF has both the lowest starting throughput at 4.1 B/s and the lowest ending throughput at 24.0 B/s. The three other functions start off at approximately the same throughput, that is 13.5 B/s for MRHOF and 14.5 B/s for NL-OF and F-OF. When the number of sources reaches 5, the throughput values are 45.2 B/s, 58.7 B/s and 67.3 B/s, respectively for MRHOF, NL-OF and F-OF. Thus, F-OF performs better than MRHOF by 48.9% and better than NL-OF by 14.7%.

As with the increasing number of sources, the effect of increasing traffic density on throughput is visible on figure 3b and is related to the PDR (fig. 2b). MRHOF's throughput drops the most followed by E-OF. F-OF yields to the best throughput when the charge is low (69.0 B/s) which is better than NL-OF (58.7 B/s) by 17.6%. However, although NL-OF and F-OF are the best at heavy charges, with respective throughputs of 132.2 B/s and 135.9 B/s, the difference is less significant (F-OF is only 2.8% better).

D. Energy consumption

In terms of energy consumption, figure 4a shows that the difference between all algorithms is only marginal (no more than 1%), though E-OF still is the algorithm consuming the least power, as expected. On figure 4b the difference is more significant, with NL-OF and F-OF being similar, E-OF performing better and MRHOF consuming the least power. The reason why the consumption of E-OF is decreasing towards the end is because of the decreasing in PDR and throughput (figs. 2b and 3b). Similarly, the low PDR (fig. 2b) and resulting low throughput (fig. 3b) of MRHOF explain why its energy consumption is the lowest of all algorithms.

V. CONCLUSION

The rapidly increasing traffic in the IoT poses the problem of routing and it is difficult to design an efficient algorithm to improve PDR, throughput and energy consumption. We are convinced that fuzzy logic is the key to significantly improving routing by better adapting to the needs of nodes. Results show that a solution that encompasses all metrics through a fuzzification process yields to better PDR and throughput and that energy consumption is not higher. Future work will be twofold : focusing on using more metrics and variable bounds to allow an even better adaptation of the algorithm to different topologies, and taking into account mobility and implementing the solution on a testbed to yield to more realistic results.

REFERENCES

- [1] T. Winter, P. Thubert, A. Brandt, J. Hui, and R. Kelsey, "Rfc 6550: Rpl: Ipv6 routing protocol for low-power and lossy networks (2012)," Tech. Rep.
- [2] M. Bouaziz, A. Rachedi, and A. Belghith, "Ekf-mrpl: Advanced mobility support routing protocol for internet of mobile things: Movement prediction approach," *Future Generation Computer Systems*, 2017.
- [3] C. V. N. Index, "Global mobile data traffic forecast update, 2016–2021 white paper, accessed on may 2, 2017."
- [4] J. Vasseur, M. Kim, K. Pister, N. Dejean, and D. Barthel, "Rfc 6551: Routing metrics used for path calculation in low-power and lossy networks," Tech. Rep., 2012.
- [5] M. Di Francesco, G. Anastasi, M. Conti, S. K. Das, and V. Neri, "An adaptive algorithm for dynamic tuning of mac parameters in ieee 802.15. 4/zigbee sensor networks," in *Pervasive Computing and Communications Workshops (PERCOM Workshops)*, 2010 8th IEEE International Conference on. IEEE, 2010, pp. 400–405.
- [6] H.-S. Kim, H. Kim, J. Paek, and S. Bahk, "Load balancing under heavy traffic in rpl routing protocol for low power and lossy networks," *IEEE Transactions on Mobile Computing*, vol. 16, no. 4, pp. 964–979, 2017.
- [7] W. Khallef, M. Molnar, A. Benslimane, and S. Durand, "Multiple constrained QoS routing with RPL." IEEE, pp. 1–6. [Online]. Available: <http://ieeexplore.ieee.org/document/7997081>
- [8] D. Bendouda, A. Rachedi, and H. Haffaf, "Programmable architecture based on software defined network for internet of things: Connected dominated sets approach," *Future Generation Computer Systems*, vol. 80, pp. 188–197, 2018.
- [9] P. THUBERT, "Objective function zero for the routing protocol for low-power and lossy networks: Rfc 6552," Tech. Rep.
- [10] O. Gnawali and P. Levis, "Rfc 6719: The minimum rank with hysteresis objective function," Tech. Rep., 2012.
- [11] D. S. J. De Couto, "High-throughput routing for multi-hop wireless networks," Ph.D. dissertation, Massachusetts Institute of Technology, 2004.
- [12] T. Instruments, "Cc2520 datasheet: 2.4 ghz ieee 802.15. 4/zigbee rf transceiver," 2007.
- [13] —, "Msp430f543x, msp430f541x datasheet—mixed signal microcontroller (rev. e)," 2009.