Neutral Operation of the Minimum Energy Node in Energy-Harvesting Environments

André Riker, Marilia Curado, Edmundo Monteiro
Centre for Informatics and Systems, Department of Informatics Engineering
University of Coimbra, Coimbra, Portugal
{ariker, marilia, edmundo}@dei.uc.pt

***Abstract*—With the recent emergence of energy-harvesting technologies in wireless devices, new challenges have to be addressed by Machine-to-Machine (M2M) communication protocols. The Neutral Operation problem is a relevant problem that seeks to maintain the energy reserve of a node in a level that minimizes energy depletion and maximizes the usage of the harvested-energy. However, neutral operation in a multihop network is a more complex issue, since the nodes lack full knowledge of the network and the nodes have diverse harvesting and consumption profiles. A simplification of the Neutral Operation problem is proposed, named Neutral Operation of the Minimum Energy Node, in which the node with the lowest amount of energy determines the operation of the whole network. This paper proposes a battery-aware solution, called Routing and Aggregation for Minimum Energy (RAME), that performs data-aggregation on the traffic load according to the minimum energy reserve on the path. As part the proposed solution, a kinetic battery model has been developed to provide non-linear battery level estimation. Besides, the Routing Protocol for Low-Power and Lossy Networks (RPL) was enhanced to use the kinetic battery estimation as metric for parent node selection and to find periodically the minimum energy reserve on the available paths. The performance evaluation of the proposed mechanism using Contiki shows the benefits of RAME in comparison to the M2M standard protocols.**

I. INTRODUCTION

Machine-to-Machine (M2M) is new a communication paradigm that has driven the sensing and actuating systems towards the next generation networks, where a large amount of traffic will be generated by multimedia and traditional devices [1]. Some important M2M applications require periodic many-to-one communication, where the nodes sense the environment, communicate to the sink node [2], and the resulting traffic will be destined to cloud environments [3]. During the last years, the M2M standards have made relevant progress towards energy-efficient communication of applications that have periodic many-to-one traffic [4]. For instance, the Constrained Application Protocol (CoAP) [5] protocol enables a client to observe periodically a data-resource consuming a reduced amount of energy. Another example is the Routing Protocol for Low power and Lossy Networks (RPL) [6], which allows energy-efficient use of a tree-based network. However, energy-harvesting scenarios remain a challenge for M2M.

Neutral Operation or Energy Self-Sufficient are terms widely used to define energy-harvesting efficiency. It means that a node consumes the exact amount of harvested energy [7]. When a node is in neutral operation, there is no overuse or underuse of the energy on the battery. However, to achieve neutral operation of a multi-hop network is a very complex and even infeasible, since the nodes do not have full knowledge of the network and not all nodes have the same harvesting pattern, energy consumption, and battery capacity [8].

A simplification of the Neutral Operation problem for multi-hop energy-harvesting network can be achieved if the operation of the whole network is determined by what can be supported by the node with the lowest amount of energy. This simplification, called Neutral Operation of the Minimum Energy Node (NOMEN), guarantees that all network nodes will cooperate to turn the operation of the minimum energy node into a neutral operation.

This paper proposes a solution, named Routing and Aggregation for Minimum Energy (RAME), for the NOMEN problem. RAME is a joint battery-aware solution that encompasses routing and data-aggregation. Regarding routing, it selects the path with the maximum lowest energy reserve, which is a max-min strategy. Relatively to data-aggregation, RAME applies procedures that enable a message to be assembled with multiple payloads, and in case there is a need to reduce traffic, it aggregates the traffic load according to the minimum energy reserve on the path [9]. As part of the proposed solution, a kinetic battery model has been integrated on Contiki OS [10] to provide non-linear battery level estimation. In this work, RPL has been extended (i) to communicate control messages containing the kinetic battery estimation, (ii) to find the minimum energy reserve on the available routes, and (iii) to use the minimum energy levels as metric for route selection.

The paper is structured as follows. Section II and III introduce the related work and RAME. Section IV and V present the implementation and the performance evaluation. Section VI shows the conclusions.

II. RELATED WORK

The Neutral Operation problem has been addressed using different approaches, but the NOMEN problem, which is a simplification of the generic Neutral Operation problem, has not been addressed by the research community, according to the best of our knowledge.

Lattanzi et al. [11] propose a solution that finds a communication rate for the whole network in order to keep the nodes in Neutral Operation. However, this solution is developed considering a battery model able to store unlimited amount of energy, which is an unrealistic assumption for rechargeable nodes. In addition, this solution does not include any data-aggregation procedure to be executed.

Differently from Lattanzi et al. [11], Gao et al. [12] present a solution that uses a data-aggregation mechanisms in energy-harvesting networks. In this approach, each node decides its data-aggregation level considering only the energy in its own reserve. A node always begins its operation performing the lowest data-aggregation level, and changes it gradually if the energy in the battery increases. The following two aspects differentiate this work from RAME: (i) Gao et al. [12] do not consider the lowest energy in the path to find the data-aggregation level, and (ii) it is a centralized solution.

Zhang et al. [13] present a solution for link scheduling and data-aggregation in energy-harvesting networks. This solution computes a routing tree using the Weighted Connected Dominating Set (WCDS). The solution gives a weight to each node based on the energy harvested and finds the tree that maximizes WCDS. Regarding the data-aggregation used in this solution, a parent node aggregates all children's messages into a single message and transmits the resulting message to its parent. Thus, the amount of traffic aggregated is fixed, which means that regardless the number of received messages, a node always aggregates all data into a single payload. Following a different approach than Zhang et al. [13], RAME applies a dynamic data-aggregation on the parent's received traffic, varying it according to the lowest energy level on the path.

Jeong et al [14] propose a solution for regulating traffic load in energy-harvesting networks. In this solution, before the periodic communication is executed, a node estimates its remaining energy. In case the estimated remaining energy is expected to overflow the capacity of the energy reserve, then the node transmits aggregated data to avoid wasting of energy. If the energy reserve is expected to be depleted, the node does not transmit any data. Although this work considers data-aggregation, it does not use different data-aggregation levels to communicate different traffic loads. In addition, Jeong et al [14] rely on a linear battery model to estimate the battery level, which is a very unrealistic assumption.

The analyzed works treat the Neutral Operation problem considering a fixed data-aggregation level, assuming unrealistic battery models, and some solutions are based on centralized approaches. Thus, these solutions do not satisfy the requirements of M2M communication in energy-harvesting scenarios. The present paper fills this gap, proposing and implementing a set of components in M2M standard protocols.

III. ROUTING AND DATA-AGGREGATION SOLUTION FOR OPERATION OF THE MINIMUM ENERGY NODE

Routing and Aggregation for Minimum Energy (RAME) is a joint energy-aware solution that combines routing and data-aggregation mechanisms to solve the NOMEN problem. The main idea behind RAME is to select paths where the minimum energy node has the highest possible energy reserve and to use data-aggregation to regulate the traffic load in each path.

A. Proposal Overview

RAME contains three components, these components are depicted in Fig.1 and described as follows:

1. Battery Model: A component able to accurately estimate the remaining energy charge in the battery. It periodically increments the amount of energy harvested from the ambient and also discounts the energy spent to keep the node alive.

2. Routing: The routing component has two roles. First, to exchange messages containing the battery level, aiming to find the lowest battery level of the available paths. Second, to select the path where the minimum energy node has the highest possible energy reserve among the available paths.

3. Data Aggregation: Knowing the lowest energy in the path, a data-aggregation component regulates the traffic passing through each node. When this component aggregates data, the traffic load is reduced and the energy consumption on the path becomes smaller. Otherwise, the traffic load increases, which demands more energy from the nodes.

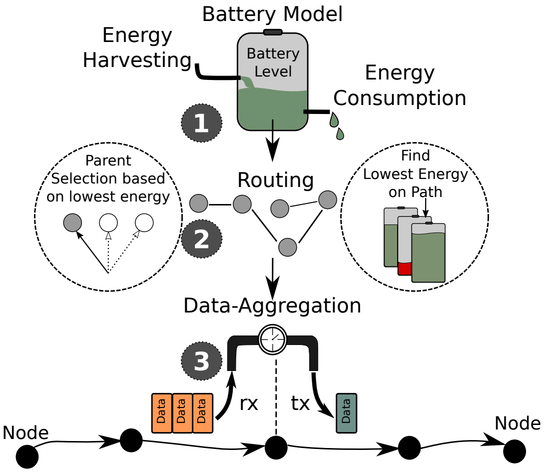


Fig. 1: RAME overview.

The details of the RAME components are presented next.

B. Kinetic Battery Model: Estimating the Node's Energy

A widely accepted solution to estimate the State-of-Charge (SoC) of a battery is the Kinetic Battery Model (KiBaM) [15]. This model considers the battery has two wells of charges. One is the available-charge well and the other is the bound-charge well, as can be seen on Fig. 2.

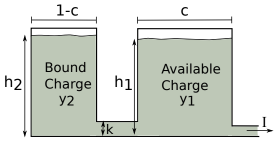


Fig. 2: Kinetic Battery Model [16].

The available-charge well supplies electrons to the output load, while the bound-charge well supplies electrons only to the available-charge well. The rate of charge that flows between the two wells is set by k, which is a fixed internal parameter, and also by the difference between h_1 and h_2. When $h_1 = 1$ the battery is fully charged and when $h_1 = 0$ the battery is fully discharged. Parameter c is a capacity ratio and corresponds to the fraction of the total charge in the battery that is available.

Equation 1 shows how to compute the amount of charge on the available and bound wells, which are denoted by y_1 and y_2, respectively [16].

$$y_1(t) = y_{1,0}e^{-k't} + \frac{(y_0k'c - I)(1 - e^{-k't})}{k'} - \frac{Ic(k't - 1 + e^{-k't})}{k'} \quad (1)$$

$$y_2(t) = y_{2,0}e^{-k't} + y_0(1 - c)(1 - e^{-k't}) - \frac{I(1 - c)(k't - 1 + e^{-k't})}{k'}$$

In this equation, k' is defined as $k' = k/c(1 - c)$, $y_{1,0}$ and $y_{2,0}$ are the amount of available and bound charges, respectively, at $t = 0$. In addition, the variable y_0 is the total charge at time $t = 0$, given by $y_0 = y_{1,0} + y_{2,0}$.

C. Routing: Finding the Lowest Energy and Selecting Parent in a Tree-Based Topology

The proposed solution is focused on tree-based topologies, since several M2M applications perform many-to-one communication. In such topologies, each node selects a parent node among a set of parent candidates to send its traffic.

To find the minimum energy node in a tree-based topology, it is necessary to exchange control messages between the nodes. As the data-traffic flows from the nodes towards the root of the tree, a particular node is interested in knowing the lowest energy level in its route to the root, which means that the energy levels of its children are not useful for RAME.

Algorithm 1 presents how RAME determines the lowest energy in the route to the root. The procedure starts at the root node, broadcasting the energy estimation of its own battery. The nodes that have the root as parent will receive the $minE_p$ and will also transmit their own *my_minE*. Therefore, the other non-root nodes receive messages indicating the $minE_p$ for each parent p, and also select as preferred parent the one that has maximum *minE*$_p$. After selecting the preferred parent, the node computes and broadcasts *my_minE*, which is the minimum between *my_Energy* and *max(minE*$_p$*)*. The value stored in *my_minE* is sent to all candidate children, enabling the process to be repeated at all network nodes.

Algorithm 1 Find Lowest Energy on Path and Parent Selection.

```
Initialize: candidate_set, my_Energy
Start
    for all parent p in candidate_set do
        Receive msg with $minE_p$
    end for
    Select p with max($minE_p$)
    my_minE ← min(my_Energy, max($minE_p$))
    Send msg with my_minE
End
```

The idea of choosing the parent candidate with the highest *minEnergy*$_p$ comes from the max-min optimization strategy.

Algorithm 1 does not address loop-avoidance or how to determine if a neighbor is a parent candidate, since these aspects are handled by the routing protocol, such as RPL.

D. Dynamic Data-Aggregation

The proposed data-aggregation solution takes advantage of the small size of the M2M messages to enable the nodes to assemble messages with many payloads. A distinction should be made between the "unitary payload" and the "concatenated payload". The former, also called small payload, refers to the smallest payload that can exist, while the latter means the payload formed by two or more unitary payloads. In this work, the unitary payload is considered to have a fixed size. Regarding data-aggregation, RAME executes the following steps periodically:

1) Extracting payloads: It extracts the unitary payloads from all application messages received from the children, discarding the headers.
2) Payload Merger and Concatenation: It processes all unitary payloads, including the extracted and the produced by the node itself. There are two aggregation procedures for the gathered payloads: (i) merge sub-sets of unitary payloads into a single unitary payload using statistical functions (e.g. Maximum, Minimum, and Average), transforming multiple unitary payloads into a single one; and (ii) concatenate all the remaining unitary payloads from the previous aggregation procedure.
3) Assembling payloads: It creates a single application layer message containing the concatenated payloads.

In step 2 is decided the number of payloads of the final message. Messages with a higher number of payloads will represent heavy traffic load for the nodes located on the route, especially for the node with lowest energy. To decide the number of unitary payloads inside the transmitted messages and consequently how many payloads must be merged, RAME uses the α parameter ($0 \leq \alpha \leq 1$), which is a function of the lowest energy on the path. Equation 2 shows how α is used to compute the number of payloads transmitted.

$$Tx_{payloads} = (Rx_{payloads} * \alpha) + 1 \quad (2)$$

As α is defined to be the parameter that determines the number of payloads that will be transmitted, the aggregation level is given by $1 - \alpha$.

IV. IMPLEMENTATION IN CONTIKI OS

This section presents how the different components of RAME have been implemented in Contiki OS. The reason to choose Contiki OS is the availability of the M2M protocols, such as CoAP, RPL, 6LowPAN, and IEEE 802.15.4.

A. Battery Model

The KiBaM model [15] was implemented in Contiki OS with the following modules: energy consumption and energy-harvesting. The energy-harvesting module of KiBaM reads a data-trace that contains the amount of harvested energy per minute. This module takes the data-trace as input, processes it and feeds the model with the equivalent charging current. The used energy-harvesting dataset contains indoor radiant light measurements collected by the Columbia University's [17].

More details about the harvested trace are given in Section V. Regarding energy consumption, the battery model is based on Powertrace [18] functionalities. Therefore, KiBaM is able to measure with satisfactory accuracy the energy consumption in the following states: Transmit, Receive, Idle Listen, Active CPU, and Low Power CPU. The part of the code that provides energy consumption and periodically computes the remaining energy in the battery is available [1].

B. Routing: Extended RPL

Regarding the routing aspects, the implementation of RAME has been developed on the IPv6 Routing Protocol for Low-Power and Lossy Networks (RPL). RPL is a tree-based protocol that creates a Directed Acyclic Graph (DAG) to route the traffic. The standard RPL determines a neighbor node as parent candidate using the rank information. Rank is similar to hop-count, because for each hop in the route the rank increases a given amount. Since the primary objective of rank is to avoid loops in the route, additional metrics can be used to select the preferred parent among the parent candidates.

RPL has a set of ICMPv6 messages defined to exchange information among the nodes. One of them, named DAG Information Object (DIO), is very relevant for the RAME implementation. DIO is a RPL message that carries information about the RPL Instance and the configuration parameters of the DAG. DIO messages travel downwards in the network, which means that they go from the root to the nodes. However, the information inside DIO is used to compute the upward paths. For that reason, RAME uses the DIO messages to determine the lowest energy level in the available paths.

Fig. 3 shows the fields of a RAME DIO message with the DAG metric container and the Node Energy Object. In the DAG metric container, the field “Routing-MC-Type” is set to 2, which corresponds to the Energy Metric Container, according to RFC 6551 [19]. Besides, in the Node Energy Object, RAME’s implementation sets the E bit as ‘1’ in order to use the E_E 8-bit field of the message.

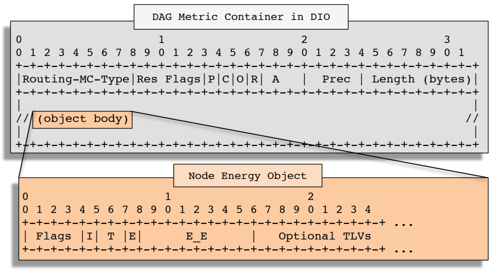


Fig. 3: DIO message and its fields.

Originally, E_E (Estimated-Energy) is an 8-bit unsigned integer field that indicates the estimated percentage of remaining energy. For the purpose of finding the lowest energy in the path, when a node running RAME sends a DIO message, the E_E field contains the numeric value stored in the variable *my_minE* (See Line 7 of Algorithm 1). In case a RAME node receives a DIO message from a parent candidate p, the E_E corresponds to the $minE_p$ (See Line 4 of Algorithm 1). Thus, using the E_E (Estimated-Energy) field of the DIO messages, it is possible to find the minimum energy level of the paths.

In addition, the RPL function that computes the cost of the parent candidates was adapted to use the minE$_p$ contained in the received DIO messages. This change enables the RPL to compare the minE$_p$ of all parent candidates, and to select the parent with highest minE$_p$.

C. Data-Aggregation Integrated in the M2M Stack Protocol

The proposed Data-Aggregation solution has been implemented to aggregate payloads of the Constrained Application Protocol (CoAP). The extraction of the CoAP payloads coming from children nodes is implemented on the IP layer. Thus, whenever a message passes through the IP layer, the extraction code verifies the fields of the message that identify the type of protocol. If this field indicates that it is a CoAP message, the code verifies if the message is self-produced or it is coming from neighbors.

By accessing the fields of the message that indicate the length of the non-CoAP header, it is possible to use the code available in Contki to parse the CoAP portion of the message. All payloads extracted are stored in a buffer. All unitary payloads have a fixed size of 2 bytes and the content of the payload is decoded as ”plain text”.

The CoAP implementation in Contiki OS allows clients to observe resources of the CoAP nodes, setting an interval in which the node will sent data periodically. The implemented aggregation takes advantage of this functionality, since the aggregation has been coded to be triggered when the node has a new data observation to send to the client. Thus, the data-aggregation procedures executed on the buffer take place when the node executes the code to send a new observation message. When the aggregation is triggered, the node determines an α value based on the minimum energy level on the path, and produces the CoAP message with one or multiple payloads.

V. PERFORMACE EVALUATION AND RESULTS

This section describes the evaluation and the obtained results of RAME. Section V-A presents the settings used for the evaluation and Section V-B shows the evaluation metrics. Section V-C presents the obtained results.

A. Configuration

Table I shows the used settings for KiBaM. The update interval is 5 minutes, which means that every 5 minutes the KiBaM updates the amount of energy charge. Regarding energy-harvesting, this evaluation considers the dataset that contains indoor light energy measurements collected by the Columbia University’s [17]. It provides a temporal-series of watt/cm^2, so to use these measurements as KiBaM input, the voltage is fixed at 5v, the area considered to harvest energy corresponds to 210 cm^2 and the conversion efficiency is 20%.

The hardware used is the following: MSP430 series 5, which has a MicroController Unit (MCU) of 16 bits with 16kB

---

[1] https://github.com/KineticBattery/Powertrace

internal RAM and 128kB Flash. The transceiver is TI CC2520 (2.4GHz), compatible with IEEE 802.15.4 and 6LoWPAN.

To measure the energy consumption, the Powertrace tool, which is available in Contiki OS, has been used. Powertrace is able to measure the energy consumption in the following consumption states: (i) Active CPU, (ii) Low Power Mode CPU, (iii) Transmit, (iv) Listen, (v) Idle Listen. In addition, Powertrace is fully integrated with the above mentioned hardware, which means that it is able to measure the energy activities related not only to the data-traffic communication but also to the data-processing and control activities demanded by the protocols. Some additional settings are shown in Table I.

TABLE I: Battery and Communication settings.

Configuration (i)	**Info**	**Configuration (ii)**	**Info**
Battery Capacity (B_c)	1000000 microAh	Max Available Charge	90% of B_c
Max Bound Charge	10% of B_c	Internal rate K	0.1
Update Periodicity	5 minutes	Unitary payload size	2 Bytes
Application Protocol	CoAP	CoAP Obs Interval	5 minutes
IP layer	6LowPAN	Routing Protocol	RPL
Low Duty Cycling	ContikiMac	Wakeup Frequency	2Hz
Mac and Physical	IEEE 802.15.4	Wireless Range	20 m

Some preliminary tests have been conducted to select the best function for the alpha parameter (see 2). Fig. 4 shows three different functions used to determine the α parameter according to the lowest energy in the path.

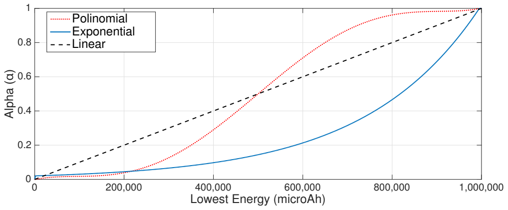


Fig. 4: Alpha (α) is a function of the lowest energy level.

However, due to the lack of space, not all α-functions will be presented. For the purpose of this evaluation, the polynomial function has been used in all experiments.

After being configured, Contiki OS is compiled and the binary image can be emulated using Cooja, which also enables the emulated nodes to be simulated as a wireless network. The obtained results are based on the Cooja simulation. Since the simulations are very time-consuming, the network was simulated with 40 nodes and the number of disjoint paths is fixed in 4. The nodes are located with 10 meters of spacing between each other. At the beginning of the simulation, the minimum energy node of each disjoint path begins the simulation with 5% less energy than the other nodes.

B. Evaluated Solutions and Metrics

Two solutions have been evaluated, namely: Routing and Aggregation for Minimum Energy (RAME) and the standard M2M protocols. The standard solution does not perform any data-aggregation and uses the Minimum Rank with Hysteresis Objective Function as metric for parent selection [20].

The following metrics have been used to measure the performance of RAME and the standard protocols.

- Energy Consumption of the Lowest Energy Node: This metric corresponds to the sum of the energy consumed by the minimum energy nodes divided by the number of disjoint paths. The consumption values have three categories: CPU, Transmission (Tx), and Reception (Rx). The reception category considers the effective reception of data and the idle listen state.
- Residual Energy of the Lowest Energy Node and Linear Trend: It measures over time the amount of charge that remains in the battery of the minimum energy node. In order to facilitate the interpretation of the tendency of the residual energy, a linear regression approach is applied on the obtained residual energy data to find a linear trend.

C. Results

Fig. 5 shows the average energy consumed by the nodes with lowest energy in the disjoint paths. As can be noticed, in comparison to the standard version of the M2M protocols, RAME reduces the energy consumption in more than 12%.

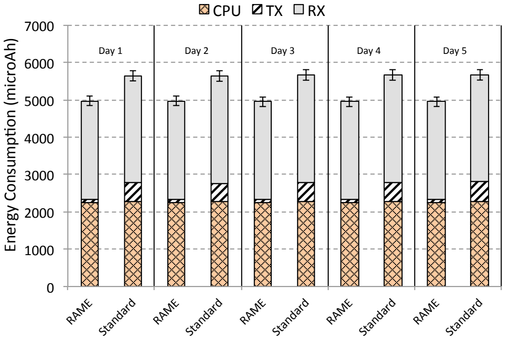
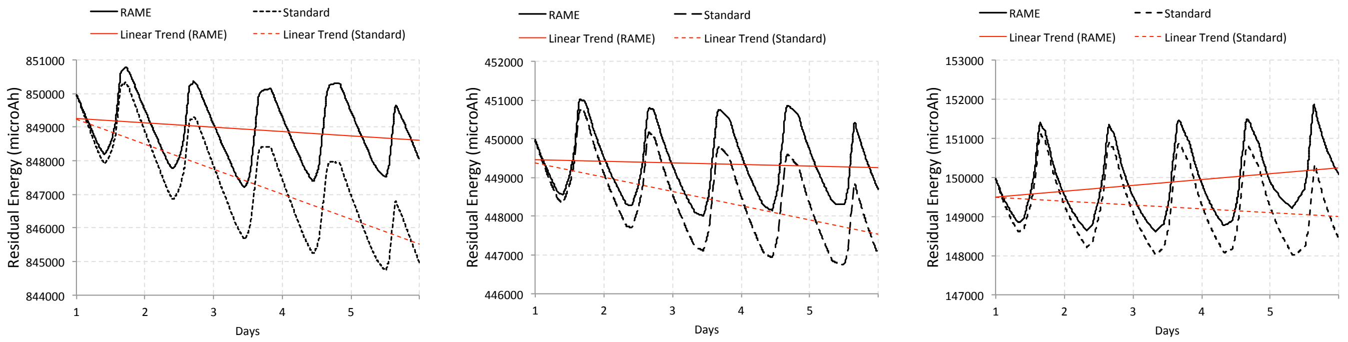


Fig. 5: Average energy consumed by the node with lowest energy, considering $\alpha = 1$.

The energy consumption gain is caused mostly by the concatenation of payloads, since for $\alpha = 1$ there is no merge of payloads. Therefore, for these results, both solutions delivered the same amount of payloads. It means that the gain in terms of energy will be even higher when RAME uses $\alpha = 0$.

Fig. 6a presents how the residual energy of the minimum energy node changes over 5 days, starting the batteries with 85% (High State-of-Charge). Both RAME and the standard solution have a decreasing linear tendency, indicating that the energy consumption is higher than the harvested energy. At this level of battery, it is desirable to have a decreasing residual tendency, since the battery only has about 15% of capacity left to store more energy. At this point, an increasing energy level could cause battery overflow, which would waste energy.

Fig. 6b shows the residual energy in a scenario where the lowest energy node has 45% of battery (Intermediate State-of-Charge). This figure also presents the linear trend of RAME, which is almost a constant line. At this level of residual energy, the level of data-aggregation is around 0.5, which means that half of the received payloads are merged before transmission. On the other hand, the standard M2M protocols maintain a higher energy consumption rate.



(a) Battery initiates with 85%.

(b) Battery begins with 45%.

(c) Battery starts at 15%.

Fig. 6: Residual battery of the node with lowest energy.

Another case is when the lowest energy node has 15% of battery (Low State-of-Charge). Fig. 6c shows that RAME has an increasing linear trend, which means that over time it is accumulating energy on the battery of the lowest energy node. However, the standard solution does not have a positive linear trend, since the energy consumed by this solution is significantly higher. A factor that contributes for the better performance of RAME is that it selects the paths where the minimum energy node has the highest possible energy reserve.

VI. CONCLUSION

This paper proposes the Routing and Aggregation for Minimum Energy (RAME) to control the energy consumption of the minimum energy node. RAME regulates the amount of traffic according to the lowest energy on the path. The proposed solution uses a kinetic battery model to provide non-linear battery level estimation. In addition, this paper also proposes a RPL extension to find the lowest energy on the path and to use this information to perform path selection. RAME has been implemented in Contiki OS, in which its components were developed to run along with the M2M protocols and Contiki OS tools. The obtained results show the benefits of RAME in comparison to the M2M standard protocols.

ACKNOWLEDGMENT

This work was partially funded by Ciencia sem Fronteiras (Brazil) Program/2013 and partially carried out in the scope of the MobiWise project: From mobile sensing to mobility advising (P2020 SAICTPAC/0011/2015), co-financed by COMPETE 2020, Portugal 2020 - POCI, European Unions ERDF, and the Portuguese Foundation for Science and Technology (FCT). This work was also co-financed by the DenseNet project PTDC/EEI-SCR/6453/2014, which is a FCT/FEDER/COMPETE 2020 project.

REFERENCES

[1] R. Immich, E. Cerqueira, and M. Curado, "Shielding video streaming against packet losses over vanets," *Wireless Networks*, vol. 22, no. 8, pp. 2563–2577, 2016.

[2] A. Riker, M. Curado, and E. Monteiro, "Group communication in machine-to-machine environments," in *Wireless Networking for Moving Objects*. Springer, 2014, pp. 219–238.

[3] B. L. Dalmazo, J. P. Vilela, and M. Curado, "Performance analysis of network traffic predictors in the cloud," *Journal of Network and Systems Management*, pp. 1–31, 2016.

[4] A. Riker, E. Cerqueira, M. Curado, and E. Monteiro, "Data aggregation for group communication in machine-to-machine environments," in *Wireless Days (WD), 2014 IFIP*. IEEE, 2014, pp. 1–7.

[5] Z. Shelby, K. Hartke, and C. Bormann, "The Constrained Application Protocol (CoAP)," RFC 7252, Internet Engineering Task Force, jun 2014. [Online]. Available: http://www.ietf.org/rfc/rfc7252.txt

[6] T. Winter, P. Thubert, A. Brandt, J. Hui, R. Kelsey, P. Levis, K. Pister, R. Struik, J. Vasseur, and R. Alexander, "RPL: IPv6 Routing Protocol for Low-Power and Lossy Networks," RFC 6550, IETF, Mar. 2012. [Online]. Available: http://www.ietf.org/rfc/rfc6550.txt

[7] O. Ozel, K. Tutuncuoglu, S. Ulukus, and A. Yener, "Fundamental limits of energy harvesting communications," *IEEE Communications Magazine*, vol. 53, no. 4, pp. 126–132, 2015.

[8] A. Riker, E. Cerqueira, M. Curado, and E. Monteiro, "Data aggregation for machine-to-machine communication with energy harvesting," in *2015 IEEE International Workshop on Measurements Networking (M N)*, Oct 2015, pp. 1–6.

[9] A. Riker, E. Cerqueira, M. Curado, and E. Monteiro., "A two-tier adaptive data aggregation approach for m2m group-communication," *IEEE Sensors Journal*, vol. 16, no. 3, pp. 823–835, Feb 2016.

[10] A. Dunkels, O. Schmidt, N. Finne, J. Eriksson, F. Österlind, N. Tsiftes, and M. Durvy, "The contiki os: The operating system for the internet of things," *Online], at http://www. contikios. org*, 2011.

[11] E. Lattanzi, E. Regini, A. Acquaviva, and A. Bogliolo, "Energetic sustainability of routing algorithms for energy-harvesting wsn," *Computer Communications*, vol. 30, no. 14, pp. 2976–2986, 2007.

[12] D. Gao, Y. Liu, F. Zhang, and J. Song, "Data aggregation routing for rechargeable wireless sensor networks in forest monitoring," *Wireless Personal Communications*, vol. 79, no. 1, pp. 773–788, 2014.

[13] J. Zhang, X. Shen, S. Tang, and G. Dai, "Energy efficient joint data aggregation and link scheduling in solar sensor networks," *Computer Communications*, vol. 34, no. 18, pp. 2217–2226, 2011.

[14] S. Jeong, H. Kim, D. K. Noh, and I. Yoon, "Energy-aware data aggregation scheme for energy-harvesting wireless sensor networks," in *Computer Communication and the Internet (ICCCI), 2016 IEEE International Conference on*. IEEE, 2016, pp. 140–143.

[15] J. Manwell and J. McGowan, "Extension of the kinetic battery model for wind/hybrid power systems," in *EWEC*, 1994, pp. 284–289.

[16] M. R. Jongerden and B. R. Haverkort, "Which battery model to use?" *IET software*, vol. 3, no. 6, pp. 445–457, 2009.

[17] M. Gorlatova, M. Zapas, E. Xu, M. Bahlke, I. J. Kymissis, and G. Zussman, "CRAWDAD dataset columbia/enhants (v. 2011-04-07)," Downloaded from http://crawdad.org/columbia/enhants/20110407, 2011.

[18] A. Dunkels, J. Eriksson, N. Finne, and N. Tsiftes, "Powertrace: Network-level power profiling for low-power wireless networks," 2011.

[19] J. Vasseur, M. Kim, K. Pister, N. Dejean, and D. Barthel, "Routing metrics used for path calculation in low-power and lossy networks," RFC 6551, March 2012. [Online]. Available: http://www.rfc-editor.org/rfc/rfc6551.txt

[20] O. Gnawali and P. Levis, "The minimum rank with hysteresis objective function," RFC 6719, September 2012. [Online]. Available: http://www.rfc-editor.org/rfc/rfc6719.txt